IML FISTA: Inexact MuLtilevel FISTA for Image Restoration

Guillaume Lauga, Elisa Riccietti, Nelly Pustelnik, Paulo Gonçalves

IML FISTA: Inexact MuLtilevel FISTA for Image Restoration *

Guillaume Lauga†, Elisa Riccietti†, Nelly Pustelnik‡, and Paulo Gonçalves†

Abstract. This paper presents IML FISTA, a multilevel inertial and inexact forward-backward algorithm, based on the use of the Moreau envelope to build efficient and useful coarse corrections. Such construction is provided for a broad class of composite optimization problems with proximable functions. This approach is supported by strong theoretical guarantees: we prove both the rate of convergence and the convergence of the iterates to a minimum in the convex case, an important result for ill-posed problems. We evaluate our approach on several image reconstruction problems and we show that it considerably accelerates the convergence of classical methods such as FISTA, for large-scale images.

Key words. multilevel optimization, inertial methods, image restoration, inexact proximal methods.

MSC codes. 68U10, 65K10, 46N10

1. Introduction.

Many problems in signal and image processing involve minimising a sum of a data fidelity and a regularization term. In this article we put emphasis on problems that are generally written as:

$$\widehat{x} \in \underset{x\in\mathbb{R}^N}{\operatorname{Argmin}}\, F(x) := f(\mathrm{A}x) + g(\mathrm{D}x) \tag{1.1}$$

where $f : \mathbb{R}^M \to (-\infty, +\infty]$ and $g : \mathbb{R}^K \to (-\infty, +\infty]$ belong to the class of convex, lower semi-continuous (l.s.c), and proper functions on $\mathbb{R}^M$ (resp. $\mathbb{R}^K$) and $A : \mathbb{R}^N \to \mathbb{R}^M$, $D : \mathbb{R}^N \to \mathbb{R}^K$ are linear operators. Moreover, f is assumed to be differentiable with L_f-Lipschitz gradient, while g is usually non-smooth. F is supposed to be coercive.

In the context of image restoration, we aim to recover a good quality image $\widehat{x}$ from a corrupted version $z = \mathrm{A}\bar{x} + \epsilon$ of an original image $\bar{x}$, where A models a linear degradation operator and ϵ stands for additive noise. This problem is known to be ill-posed, and is generally tackled by solving a regularized least squares problem of the form (1.1), where the regularization function g allows us to choose the properties one wishes to impose on the solution.

Many iterative algorithms have been proposed in the literature to estimate $\widehat{x}$ (cf. for instance [15,21,23,46] and references therein). Most of them are based on the use of proximal methods, as g is non differentiable, and they all share the same weakness: the required computational time for the reconstruction explodes with the size of the problem. This is particularly critical when the proximity operator of $g \circ \mathrm{D}$ cannot be computed explicitly, as it

*Submitted to the editors on April 19, 2023.

Funding: The authors would like to thank the GdR ISIS for funding the MOMIGS project and the ANR-19-CE48-0009 Multisc'In project. We also gratefully acknowledge the support of the Centre Blaise Pascal's IT test platform at ENS de Lyon (Lyon, France) for the computing facilities. The platform operates the SIDUS [49] solution developed by Emmanuel Quemener. .

†Univ Lyon, Inria, EnsL, UCBL, CNRS, LIP, UMR 5668, F-69342, Lyon Cedex 07, France (guillaume.lauga@ens-lyon.fr, elisa.riccietti@ens-lyon.fr, paulo.goncalves@ens-lyon.fr).

‡Ens de Lyon, CNRS, Laboratoire de Physique, F-69342, Lyon, France (nelly.pustelnik@ens-lyon.fr).

is the case when $g \circ \mathrm{D}$ encodes a total variation [11] or a non-local total variation [17]. Indeed, for these two state-of-the-art regularizations, the proximity operator of $g \circ \mathrm{D}$ can be estimated by an iterative procedure in the dual (cf. [4]), which considerably increases the cost of the optimization. Many methods circumvent this dual optimization by directly introducing dual steps paired with primal steps to reach a minimizer [7, 14, 22], but their cost for large-scale problems remains high. The already challenging task of designing algorithms that can handle large-scale problems turns even harder when inexact proximity operators are to be dealt with.

Various attempts have been made to accelerate the resolution of standard convex optimization problems, i.e., to decrease the number of necessary iterations to reach convergence. Among the most efficient methods, the *fast iterative soft thresholding algorithm* (FISTA) [5, 12], is based on an inertial/Nesterov principle that we propose to recall in a more general context, writing the optimization problem as follows:

$$\underset{x\in\mathbb{R}^N}{\text{minimize}}\ L(x) + R(x), \tag{1.2}$$

where L (resp. R) shares the same properties as f (resp. g). Problem (1.2) naturally encompasses Problem (1.1), and the $k-$th iteration of FISTA reads for every $k = 0, 1, \ldots$ as:

$$x_{k+1} = \mathrm{prox}_{\tau_k R}(y_k - \tau_k \nabla L(y_k)) \tag{1.3}$$

$$y_{k+1} = x_{k+1} + \alpha_k (x_{k+1} - x_k) \tag{1.4}$$

where $\tau_k > 0$ and $\alpha_k = \frac{t_k - 1}{t_{k+1}}$. In particular, $t_k = 1$ for all $k \in \mathbb{N}$ corresponds to the classical forward-backward (FB) algorithm for which $y_{k+1} = x_{k+1}$. More generally, the sequence $\{t_k\}_{k\in\mathbb{N}}$ can be chosen in different ways, yielding different variants of FISTA, but it must verify the general condition $t_k - t_{k+1}^2 + t_{k+1} > 0$ to guarantee ergodic convergence (i.e., convergence of the objective function to the optimal value). A common practice is to choose $t_0 = 1$ and $t_k = \left(\frac{k+a-1}{a}\right)^d$ for all $k \in \mathbb{N}^*$, with $d \in (0, 1]$ and $a > \max\{1, (2d)^{\frac{1}{d}}\}$ [2, Definition 3.1]. This choice ensures convergence of the objective function with rate $o(1/k^{(2d)})$ and, under mild conditions [2], weak convergence of the iterates to a minimizer. Here the parameter d defines a continuous way to go from a standard FB to FISTA by steadily adding inertia. We will restrict ourselves to this specific choice in the following.

To go further with acceleration techniques, we aim to use the structure of these optimization problems to reduce both the number of iterations needed to converge and the computation time through some dimensionality reduction techniques. This is a recurring idea in a lot of algorithms proposed in the literature considering either stochastic block selection [13, 27, 30, 50] or subspace methods [18, 19]. Specifically here, we seek to combine inertial techniques with *multilevel* approaches that exploit different resolutions of the same problem. In such methods the objective function is approximated by a sequence of functions defined on reduced dimensional spaces (coarse scales) and descent steps are calculated at coarse levels with smaller cost before being transfered back to the fine level. Our goal is to embed such coarse correction into the descent step computed at fine level in (1.3) before computing the approximation of the proximity operator to benefit from both types of acceleration: inertial and multilevel.

Multilevel approaches have been mainly studied for the resolution of partial differential equations (PDEs), in which f and g are both supposed to be differentiable [9, 32, 45]. Indeed,

most of the multilevel algorithms are based on the seminal work of Nash [45] and are applied to smooth objective functions minimized by first order methods. They have been employed in many applications, such as photoacoustic tomography [38], discrete tomography [48] and phase retrieval [31]. They have been also extended to second (or higher) order optimization in [10, 32, 36].

Only recently this idea has been extended in [37, 47] to define multilevel forward-backward algorithms applicable to problem (1.2) in the case where R is non differentiable and its proximity operator is known in closed form expression. In the experiments of [47], the framework is restricted to $R = \|W \cdot\|_1$ with W an orthogonal wavelet transform in the context of image restoration and restricted to $R = \|\cdot\|_1$ in the case of face recognition [37]. These works were the first attempts to introduce multilevel methods in non-smooth optimization and they introduced key concepts such as the smoothing of R to obtain first order coherence between levels. Similar ideas have been proposed in [1] with adaptive restriction operators. This method requires strong convexity assumption on f to benefit of additional convergence properties.

In our previous works, based on similar concepts, we proposed a multilevel forward-backward algorithm [41] and a multilevel FISTA [40], both with stronger convergence guarantees than the one proposed in [1, 37, 47], e.g., the convergence to a minimizer of the objective function. Our results do not require strong convexity assumption.

Here, we extend our algorithmic procedure and its associated convergence guarantees to the more general case where the proximity operator of R is not necessarily known in explicit form. To take into account the inexact computation of the proximity operator, we replace the forward-backward step in Equation (1.3) by:

$$(\forall x \in \mathbb{R}^N) \qquad \mathrm{T}_i^{\epsilon}(x) \approx_{i,\epsilon} \operatorname{prox}_{\tau R}(x - \tau \nabla L(x)) \tag{1.5}$$

for some step-size $\tau > 0$. In this expression, the index $i = \{0, 1, 2\}$ will refer to one of the three types of approximation that we will consider hereafter and ϵ corresponds to the induced approximation error [2]. Accordingly, the inexact and inertial forward-backward iterate reads:

$$x_{k+1} = \mathrm{T}_i^{\epsilon_k}(y_k), \tag{1.6}$$

$$y_{k+1} = x_{k+1} + \alpha_k(x_{k+1} - x_k). \tag{1.7}$$

By injecting coarse corrections into the iterative scheme (1.6)-(1.7), we propose a family of multilevel inertial methods that we call IML FISTA for *Inexact MultiLevel FISTA*. It provides a multilevel extension of inertial strategies such as FISTA [2, 5], fully adaptable to solve all problems of the form (1.2): whether the proximity operator of R is known in an explicit form, or whether its approximation requires the solution of a subproblem at each iteration. Our approach relies on the Moreau envelope, which in many cases can be easily derived to define smooth coarse approximations of R. Furthermore, we show that under mild assumptions, the convergence guarantees of inertial forward-backward algorithms [2] hold also for IML FISTA. In particular, this is true for the convergence of the iterates, an important result for ill-posed problems. To the best of our knowledge, this result was never established for multilevel, inertial, and inexact proximal methods.

In addition, we propose a detailed version of the algorithm to solve Problem (1.1), specifically designed for image restoration. Notably, we discuss the construction of coarse models

and of information transfer operators that have good properties for image deblurring and image inpainting problems.

Contributions and organization of the article.

- In Section 2, we develop the first multilevel framework for inertial and inexact forward-backward for the resolution of Problem (1.2), which encompasses other multilevel methods previously proposed in the literature. We carry out the associated convergence analysis of the iterates and of the objective function.
- In Section 3, the proposed algorithm is specifically adapted to image restoration problems of the form (1.1), when the proximity operator of $g \circ \mathrm{D}$ is not necessarily known in closed form. In addition, we focus on the design of transfer operators for image reconstruction problems, considering wavelet based architectures.
- Extensive numerical experiments are performed in Section 4, to compare the performances of IML FISTA versus FISTA on image reconstruction problems.

2. Multilevel, Inexact and Inertial algorithm.

This first section focuses on the minimization of Problem (1.2) to present the proposed IML FISTA in the most general context. As in classical multilevel schemes for smooth optimization, our framework exploits a hierarchy of objective functions, representative of F at different levels (scales or resolutions), and alternates minimization among these objective functions. The basic idea is to compute cheaper refinements at coarse resolution, which after prolongation to the fine levels, are used to update the current iterate.

2.1. IML FISTA Algorithm.

Without loss of generality and for the sake of clarity, we consider the two-level case: we index by h (resp. H) all quantities defined at the fine (resp. coarse) level. We thus define $F_h := F : \mathbb{R}^{N_h} \to (-\infty, +\infty]$ the objective function at the fine level where $N_h = N$, such that $F_h = L_h + R_h$ with $L_h := L$ and $R_h := R$. We associate this objective function at fine level with its coarse level approximation which we denote $F_H : \mathbb{R}^{N_H} \to (-\infty, +\infty]$, with $N_H < N_h$, and in which L_H, R_H are lower dimensional approximations of L and R.

One standard step of our algorithm can be summarized by the following three instructions:

$$\bar{y}_{h,k} = \mathrm{ML}(y_{h,k}), \tag{2.1}$$

$$x_{h,k+1} = \mathrm{T}_i^{\epsilon_{h,k}}(\bar{y}_{h,k}), \tag{2.2}$$

$$y_{h,k+1} = x_{h,k+1} + \alpha_{h,k}(x_{h,k+1} - x_{h,k}) \tag{2.3}$$

which are developped in detail in Algorithm 2.1, and where ML encompasses Steps 3 to 11. Given the current iterate $y_{h,k}$ at fine level, we can decide to update it either by a standard fine step, combining Steps 10 and 12-14 of the algorithm or by performing iterations at the coarse level (cf. steps 5-8) followed by a standard fine step (cf. 12-14). A particular attention needs to be paid to steps 5-8, which produce a coarse correction that is used to define an intermediate fine iterate $\bar{y}_{h,k}$. The coarse correction is used to update the auxiliary variable $y_{h,k}$ and not $x_{h,k}$ directly (see Equations (2.2) and (2.3)). Thus, to obtain this coarse correction, the current iterate $y_{h,k}$ is projected to the coarse level thanks to a projection operator I_h^H, and it is used as the initialisation for the minimization of the coarse approximation F_H, which generates a

Algorithm 2.1 IML FISTA

Set $x_{h,0}, y_{h,0} \in \mathbb{R}^N$, $t_{h,0} = 1$
while Stopping criterion is not met **do**
  if Descent condition and $r < p$ **then**
    $r = r + 1$,
    $s_{H,k,0} = I_h^H y_{h,k}$ Projection
    $s_{H,k,m} = \Phi_{H,m-1} \circ .. \circ \Phi_{H,0}(s_{H,k,0})$ Coarse minimization
    Set $\bar{\tau}_{h,k} > 0$,
    $\bar{y}_{h,k} = y_{h,k} + \bar{\tau}_{h,k} I_H^h (s_{H,k,m} - s_{H,k,0})$ Coarse step update whose size is set by $\bar{\tau}_{h,k}$
  else
    $\bar{y}_{h,k} = y_{h,k}$
  end if
  $x_{h,k+1} = \mathrm{T}_i^{\epsilon_{h,k}}(\bar{y}_{h,k})$ forward-backward step
  $t_{h,k+1} = \left(\frac{k+a}{a}\right)^d$, $\alpha_{h,k} = \frac{t_{h,k}-1}{t_{h,k+1}}$
  $y_{h,k+1} = x_{h,k+1} + \alpha_{h,k}(x_{h,k+1} - x_{h,k})$. Inertial step
end while

sequence $(s_{H,k,\ell})_{\ell \in \mathbb{N}}$, where k represents the current iteration at the fine level and ℓ indexes the iterations at the coarse level. This sequence is defined by $s_{H,k,\ell+1} = \Phi_{H,\ell}(s_{H,k,\ell})$, with $\Phi_{H,\ell}$ any operator such that $F_H(s_{H,k,m}) \leq F_H(s_{H,k,0})$ for some $m > 0$. While this operator has to implicitly adapt to the current step k, its general construction does not depend on k. After m iterations at the coarse level we obtain a coarse direction $s_{H,k,m} - s_{H,k,0}$, prolongated at the fine level to update $y_{h,k}$.

A multilevel scheme requires transferring information from one level to an other. To do so, we define two transfer information operators: a linear operator $I_h^H : \mathbb{R}^{N_h} \to \mathbb{R}^{N_H}$ referred to as the *restriction operator* that sends information from the fine level to the coarse level, and reciprocally $I_H^h : \mathbb{R}^{N_H} \to \mathbb{R}^{N_h}$, the *prolongation operator* that sends information from the coarse level back to the fine level.

The central point of multilevel approaches is to ensure that the correction term $s_{H,k,m} - s_{H,k,0}$, after prolongation from the coarse to the fine level, leads to a decrease of F_h. For this, particular care must be taken in the selection of the following elements:

(i) the coarse model F_H,
(ii) the minimization scheme $\Phi_{H,\cdot}$,
(iii) the information transfer operators I_h^H and I_H^h.

We detail these choices in the following subsections.

2.1.1. Coarse model F_H. In our algorithm the construction of coarse functions relies on smoothing the non differentiable R_h [6] to maintain fidelity with the fine model, and at the same time impose desirable properties to the coarse model.

As demonstrated in [40, 41], smoothing is a natural choice to extend ideas coming from the classical smooth case [33] to multilevel proximal gradient methods. We take the ideas originally proposed in [37, 47], and develop them further in the present contribution.

Definition 2.1. (Smoothed convex function [6, Definition 2.1]) *Let R be a convex, l.s.c.,*

and proper function on $\mathbb{R}^N$. *For every* $\gamma > 0$ R_γ *is a smoothed convex function if there exist scalars* η_1, η_2 *satisfying* $\eta_1 + \eta_2 > 0$ *such that the following holds:*

$$(\forall y \in \mathbb{R}^N) \qquad R_\gamma(y) - \eta_1\gamma \leq R(y) \leq R_\gamma(y) + \eta_2\gamma. \tag{2.4}$$

Such smoothed convex functions exist if the smoothing is done according to the principles developed in [6] where the sum $\eta_1 + \eta_2$ depends on R and on the type of smoothing.

Definition 2.2. (Coarse model F_H for non-smooth functions.) *The coarse model* F_H *is defined for the point* $y_h \in \mathbb{R}^{N_h}$ *as:*

$$F_H = L_H + R_{H,\gamma_H} + \langle v_H, \cdot \rangle, \tag{2.5}$$

where

$$v_H = I_h^H \left(\nabla L_h(y_h) + \nabla R_{h,\gamma_h}(y_h)\right) - \left(\nabla L_H(I_h^H y_h) + \nabla R_{H,\gamma_H}(I_h^H y_h)\right). \tag{2.6}$$

R_{h,γ_h} *and* R_{H,γ_H} *are smoothed versions of* R_h *and* R_H *respectively, and they verify Definition 2.1 with smoothing parameters* $\gamma_h > 0$ *and* $\gamma_H > 0$.

Adding the linear term $\langle v_H, \cdot \rangle$ to $L_H + R_{H,\gamma_H}$ allows to impose the so-called *first order coherence* recalled in Definition 2.4 below.

Remark 2.3. Note that if R_h and R_H are smooth by design, one can simply replace R_{H,γ_H} and R_{h,γ_h} by R_H and R_h. The construction stays otherwise the same. Thus one can still use our multilevel proximal algorithm with smooth functions.

Definition 2.4. (First order coherence [37, 45, 47]). *The first order coherence between the smoothed version of the objective function* F_h *at the fine level and the coarse level objective function* F_H *is verified in a neighbourhood of* y_h *if the following equality holds:*

$$\nabla F_H(I_h^H y_h) = I_h^H \nabla \left(L_h + R_{h,\gamma_h}\right)(y_h). \tag{2.7}$$

Lemma 2.5. *If* F_H *is given by Definition 2.2, it necessarily verifies the first order coherence (Definition 2.4).*

Proof. Considering the gradient of the coarse model F_H and combining it with the definition of v_H in Equation (2.6), yields

$$\begin{aligned} \nabla F_H(I_h^H y_h) &= \nabla L_H(I_h^H y_h) + \nabla R_{H,\gamma_H}(I_h^H y_h) + v_H, \\ &= I_h^H \left(\nabla L_h(y_h) + \nabla R_{h,\gamma_h}(y_h)\right). \end{aligned} \tag{2.8}$$

∎

This condition ensures that, in the neighbourhood of the current iterates $y_h = y_{h,k}$ and $I_h^H y_{h,k} = s_{H,k,0}$, smoothed version of the fine and of the coarse level objective functions are coherent up to order one [47].

Figure 1 illustrates the effect of the first order coherence on the alignment of the gradients of smooth objective functions at fine and coarse levels.

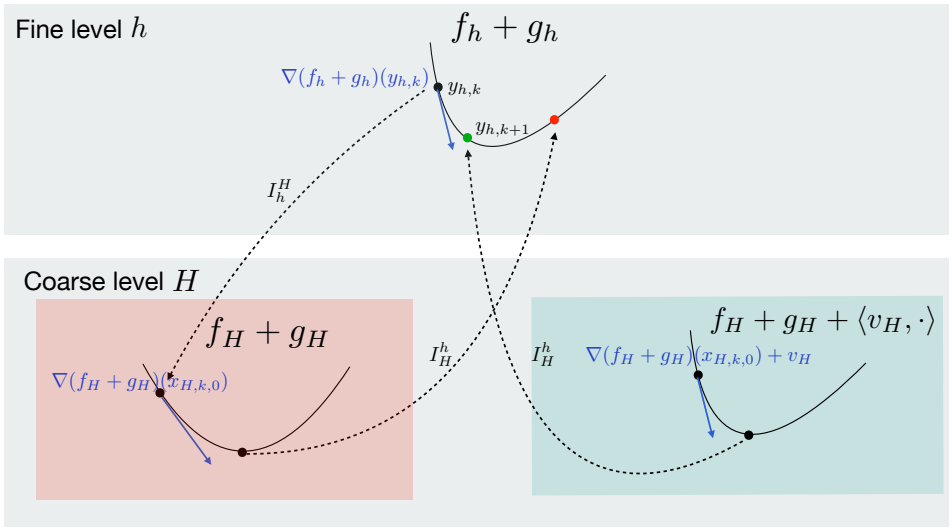

Figure 1: Illustration of the first order coherence between the two smooth functions $f_h + g_h$ and $f_H + g_H$. Left lower part: Without first order coherence, points minimizing $f_H + g_H$ do not necessarily minimize $f_h + g_h$. Right lower part: First order coherence rotates the graph of $f_H + g_H$ around $s_{H,k,0}$ so that minimizing $f_H + g_H$ also entails minimizing $f_h + g_h$.

2.1.2. Choice of coarse iterations. The operators $\Phi_{H,\bullet}$ aim to build a sequence producing a sufficient decrease of F_H after m iterations.

Definition 2.6. *(Coarse model decrease) Let $(\Phi_{H,\ell})_{\ell\in\mathbb{N}}$ be a sequence of operators such that there exists an integer $m > 0$ that guarantees that if $s_{H,m} = \Phi_{H,m-1} \circ \ldots \circ \Phi_{H,0}(s_{H,0})$ then $F_H(s_{H,m}) \leq F_H(s_{H,0})$. Moreover, $s_{H,m} - s_{H,0}$ is bounded.*

Some typical choices for $\Phi_{H,\ell}$ are the gradient descent step, inertial gradient descent step, forward-backward step or inertial forward-backward step (see [40] for a comparison of these operators in a multilevel context - the choice depends on the intensity of degradation for image reconstruction problems). These operators guarantee that $s_{H,m} - s_{H,0}$ is a bounded (through convergence of the sequence [2]) descent direction for F_H.

2.1.3. Construction of information transfer operators. Going from one level to the other requires several information transfers. For this purpose we use the following classical definition.

Definition 2.7. *The two operators $I_h^H : \mathbb{R}^{N_h} \to \mathbb{R}^{N_H}$ and $I_H^h : \mathbb{R}^{N_H} \to \mathbb{R}^{N_h}$ are coherent information transfer (CIT) operators, if there exists $\nu > 0$ such that:*

$$I_H^h = \nu (I_h^H)^T. \tag{2.9}$$

There exists numerous ways to construct CIT operators. The most standard one for multilevel methods is the dyadic decimated weighted operator [8], which, in the particular case of squared grids of size $\sqrt{N_h} \times \sqrt{N_h}$ and $\sqrt{N_H} \times \sqrt{N_H}$ at fine and coarse level respectively, and for

$N_H = N_h/4$ corresponding to a decimation factor of 2 along rows and columns, reads:

(2.10)
$$I_h^H = \frac{1}{16} \underbrace{\begin{pmatrix} 2 & 1 & 0 & \dots & & & 0 \\ 0 & 1 & 2 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & & & & 0 \\ 0 & \dots & & 0 & 1 & 2 & 1 \end{pmatrix}}_{\sqrt{N_h}/2 \times \sqrt{N_h}} \otimes \underbrace{\begin{pmatrix} 2 & 1 & 0 & \dots & & & 0 \\ 0 & 1 & 2 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & & & & 0 \\ 0 & \dots & & 0 & 1 & 2 & 1 \end{pmatrix}}_{\sqrt{N_h}/2 \times \sqrt{N_h}} \in \mathbb{R}^{N_H \times N_h}.$$

The pair (I_h^H, I_H^h) provides a simple and intuitive way to transfer information back and forth between fine and coarse scales, by means of linear B-spline interpolation. Other operators of the form of (2.10) corresponding to higher order interpolation have been proposed in [25] and are commonly used in multigrid methods for solving PDEs [26]. The literature on transfer operators being much more developed in the context of PDEs, it gives a rich starting point for multilevel optimization algorithms. In particular, the authors of [34] introduced a learning framework to optimize multigrid PDEs solvers that pay great attention to the properties of the information transfer operators.

2.1.4. Fine model minimization with multilevel steps. With the previous definitions of F_H, $\Phi_{H,\bullet}$ and I_h^H, the following lemmas prove that minimization at the coarse level also induces a descent direction at the fine level.

Lemma 2.8. (Descent direction for the fine level smoothed function). *Let us assume that I_h^H and I_H^h are CIT operators and that F_H satisfies Definition 2.2. and $\Phi_{H,\bullet}$ verifies Definition 2.6. Then, $I_H^h(s_{H,m} - s_{H,0})$ is a descent direction for $L_h + R_{h,\gamma_h}$.*

Proof. Set $y_h \in \mathbb{R}^{N_h}$ and let us define $p_H := s_{H,m} - s_{H,0}$. Recall that $s_{H,0} = I_h^H y_h$. From the definition of descent direction we have that:

$$\langle p_H, \nabla F_H(s_{H,0}) \rangle \leq 0.$$

By the first order coherence and imposing $I_h^H = \nu \left(I_H^h\right)^T$ we obtain

$$\langle p_H, \nabla F_H(s_{H,0}) \rangle = \langle p_H, I_h^H \nabla (L_h + R_{h,\gamma_h})(y_h) \rangle = \nu \langle I_H^h(p_H), \nabla (L_h + R_{h,\gamma_h})(y_h) \rangle \leq 0. \quad \blacksquare$$

We can now go a step further and derive a bound on the decrease of the (non-smooth) objective function at the fine level $F_h := L_h + R_h$. Following [37, 47], we search a proper step size $\bar{\tau}_h$ that avoids "too" big corrections from the coarse level by guaranteeing that:

$$(L_h + R_{h,\gamma_h})(y_h + \bar{\tau}_h I_H^h(s_{H,m} - s_{H,0})) \leq (L_h + R_{h,\gamma_h})(y_h). \tag{2.11}$$

Lemma 2.9. (Fine level decrease). *If the assumptions of Lemma 2.8 hold, the iterations of Algorithm 2.1 ensure:*

$$F_h(y_h + \bar{\tau} I_H^h(s_{H,m} - s_{H,0})) \leq F_h(y_h) + (\eta_1 + \eta_2)\gamma_h. \tag{2.12}$$

Proof. This directly comes from the definition of a smoothed convex function (Definition 2.1). If there exists a value of $\bar{\tau}_h$ satisfying Equation (2.11), we have:

$$\begin{aligned} F_h(y_h + \bar{\tau}_h I_H^h(s_{H,m} - s_{H,0})) &\leq (L_h + R_{h,\gamma_h})(y_h + \bar{\tau}_h I_H^h(s_{H,m} - s_{H,0})) + \eta_2\gamma_h \\ &\leq (L_h + R_{h,\gamma_h})(y_h) + \eta_2\gamma_h \\ &\leq F_h(y_h) + (\eta_1 + \eta_2)\gamma_h. \end{aligned} \tag{2.13}$$

■

This result shows that a coarse level minimization step, leads to a decrease of F_h, up to a constant $(\eta_1 + \eta_2)\gamma_h$ that can be made arbitrarily small by driving γ_h to zero.

This type of result is commonly found in the literature of multilevel algorithms [37, 40, 41, 47] but is not sufficient to guarantee the convergence of the generated sequence. In the next section we derive stronger convergence guarantees.

2.2. Convergence of the iterates. In order to obtain the convergence of the iterates to a minimizer of $F = F_h$ and the rate of convergence of the objective function, we need to take into account two types of inexactness in the computation of an iterate: one on the proximity operator of R_h and one on the gradient of L_h. The error on the gradient will allow us to compute coarse corrections with our multilevel framework, while the error on the proximity operator will allow us to consider approximation of proximity operators whose closed form is unknown.

The goal of this section is to show that an iteration of our algorithm (Steps 12-14 in Algorithm 2.1) can be reformulated as:

$$\begin{aligned} x_{h,k+1} &\simeq_i^{\epsilon_{h,k}} \operatorname{prox}_{\tau_h R_h}\left(y_{h,k} - \tau_h \nabla L_h\left(y_{h,k}\right) + e_{h,k}\right), \\ y_{h,k+1} &= x_{h,k+1} + \alpha_{h,k}(x_{h,k+1} - x_{h,k}), \end{aligned} \tag{2.14}$$

where we introduce $e_{h,k}$ to model uncertainties on the gradient step due to the multilevel corrections and the pair $(i, \epsilon_{h,k})$ introduced in (1.5), to designate the type and the accuracy of the proximity operator approximation. Such rewriting allows us to fit in the framework described by the authors of [2] to define an over-relaxed (inexact and inertial) forward-backward algorithm.

Inexactness due to coarse corrections. As presented in the algorithm, a coarse correction is inserted before a typical fine level step. We can see this coarse correction as some kind of error on the gradient of L_h. In a typical multilevel step, at the fine level (cf. Steps 12 and 8 of Algorithm 2.1), the update would take the form:

$$x_{h,k+1} \simeq_i^{\epsilon_{h,k}} \operatorname{prox}_{\tau_h R_h}\left(\bar{y}_{h,k} - \tau_h \nabla L_h\left(\bar{y}_{h,k}\right)\right), \tag{2.15}$$

with

$$\bar{y}_{h,k} = y_{h,k} + \bar{\tau}_{h,k} I_H^h(s_{H,k,m} - s_{H,k,0}). \tag{2.16}$$

We now present the following lemma that provides a bound on this update term.

Lemma 2.10. (Coarse corrections are finite) *Let β_h and β_H be the Lipschitz constants of L_h and L_H, respectively. Assume that we compute at most p coarse corrections. Let $\tau_h, \tau_H \in (0, +\infty)$ be the step sizes taken at fine and coarse levels, respectively. Assume that $\tau_H < \beta_H^{-1}$ and that $\tau_h < \beta_h^{-1}$ and denote $\bar{\tau}_h = \sup_k \bar{\tau}_{h,k}$. Then the sequence $(e_{h,k})_{k\in\mathbb{N}}$ in $\mathbb{R}^{N_h}$ generated by Algorithm 2.1 is defined as:*

$$e_{h,k} = \tau_h \left(\nabla L_h(y_{h,k}) - \nabla L_h(\bar{y}_{h,k}) + (\tau_h)^{-1} \bar{\tau}_{h,k} I_H^h(s_{H,k,m} - s_{H,k,0}) \right) \tag{2.17}$$

if a coarse correction has been computed, and $e_{h,k} = 0$ otherwise. This sequence is such that $\sum_{k\in\mathbb{N}} k\|e_{h,k}\| < +\infty$.

Proof. We are not concerned with the proximity operator (backward step) in Equation (2.15) so we focus on the forward step. Considering $\nabla L_h(\bar{y}_{h,k}) = \nabla L_h(\bar{y}_{h,k}) - \nabla L_h(y_{h,k}) + \nabla L_h(y_{h,k})$ and $\bar{y}_{h,k} = \bar{y}_{h,k} + y_{h,k} - y_{h,k}$, the forward step can be rewritten as:

$$\begin{aligned} \bar{y}_{h,k} - \tau_h \nabla L_h(\bar{y}_{h,k}) &= y_{h,k} - \tau_h \nabla L_h(y_{h,k}) \\ &\quad + \tau_h \left(\nabla L_h(y_{h,k}) - \nabla L_h(\bar{y}_{h,k}) + \frac{1}{\tau_h}(\bar{y}_{h,k} - y_{h,k}) \right). \end{aligned} \tag{2.18}$$

And so, each time a multilevel step is performed, it induces at iteration k, an error that reads:

$$e_{h,k} = \tau_h \left(\nabla L_h(y_{h,k}) - \nabla L_h(\bar{y}_{h,k}) + (\tau_h)^{-1} \bar{\tau}_{h,k} I_H^h(s_{H,k,m} - s_{H,k,0}) \right).$$

Now, assuming that we use inertial inexact proximal gradient steps at the coarse level, the corresponding minimization verifies Definition 2.6 on the decrease of F_H and produces bounded sequences if constructed according to the rules of [2, Definition 3.1, Theorem 4.1] as the sequences $(s_{H,k,\ell})_{k\in\mathbb{N},\ell\in\mathbb{N}^*}$ converge. The sequence $(e_{h,k})_{k\in\mathbb{N}}$ has at most p non zero terms, which we show are bounded:

$$\tau_h^{-1}\|e_{h,k}\| = \|\nabla L_h(y_{h,k}) - \nabla L_h(\bar{y}_{h,k}) + (\tau_h)^{-1}\bar{\tau}_h I_H^h(s_{H,k,m} - s_{H,k,0})\| \tag{2.19}$$

$$\leq \beta_h \bar{\tau}_h \|I_H^h(s_{H,k,m} - s_{H,k,0})\| + (\tau_h)^{-1}\bar{\tau}_h \|I_H^h(s_{H,k,m} - s_{H,k,0})\| \tag{2.20}$$

$$\leq \bar{\tau}_h \left(\beta_h + \frac{1}{\tau_h} \right) \|I_H^h(s_{H,k,m} - s_{H,k,0})\|. \tag{2.21}$$

The second inequality is deduced from the fact that L_h has a β_h-Lipschitz gradient and that $\bar{y}_{h,k} - y_{h,k} = \bar{\tau}_{h,k} I_H^h(s_{H,k,m} - s_{H,k,0})$. Finally using that $(\|s_{H,k,0} - s_{H,k,m}\|)_{k\in\mathbb{N}}$ is bounded, we have:

$$\tau_h^{-1}\|e_{h,k}\| \leq \bar{\tau}_h \left(\beta_h + \frac{1}{\tau_h} \right) \sup_{k\in\mathbb{N}} \|I_H^h(s_{H,k,m} - s_{H,k,0})\| < +\infty \tag{2.22}$$

which concludes the proof. ∎

Inexactness due to approximation of the proximity operator. To account for inexactness in the proximity operator computation, one needs to enlarge the notion of subdifferential through the following definition [2]:

Definition 2.11. (ϵ-subdifferential) *The ϵ-subdifferential of R at $z \in$ dom R is defined as:*

$$\partial_\epsilon R(z) = \{y \in \mathbb{R}^N \mid R(x) \geq R(z) + \langle x - z, y\rangle - \epsilon, \forall x \in \mathbb{R}^N\}. \tag{2.23}$$

Based on this definition, three different types of approximations of proximity operators were proposed.

Definition 2.12. (Type 0 approximation [20]). *We say that $z \in \mathbb{R}^N$ is a type 0 approximation of* $\text{prox}_{\gamma R}(y)$ *with precision ϵ, and we write $z \simeq_0 prox_{\gamma R}(y)$, if and only if:*

$$\|z - \text{prox}_{\gamma R}(y)\| \leq \sqrt{2\gamma\epsilon}. \tag{2.24}$$

Definition 2.13. (Type 1 approximation [52]). *We say that $z \in \mathbb{R}^N$ is a type 1 approximation of* $\text{prox}_{\gamma R}(y)$ *with precision ϵ, and we write $z \simeq_1 prox_{\gamma R}(y)$, if and only if:*

$$0 \in \partial_\epsilon \left(R(z) + \frac{1}{2\gamma}\|z - y\|^2 \right). \tag{2.25}$$

Definition 2.14. *(*Type 2 approximation [52]). *We say that $z \in \mathbb{R}^N$ is a type 2 approximation of* $\text{prox}_{\gamma R}(y)$ *with precision ϵ, and we write $z \simeq_2 prox_{\gamma R}(y)$, if and only if:*

$$\frac{y - z}{\gamma} \in \partial_\epsilon R(z). \tag{2.26}$$

Approximation of type 2 implies approximation of type 1 [2, 52] and under some conditions discussed in [52], approximation of type 0 implies approximation of type 2.

When these approximations are used in forward-backward-based algorithms, convergence guarantees are known from the literature: approximations of type 1 and 2 are covered by [2] for inertial versions of the forward-backward algorithm, while the type 0 approximation is treated in [20] only for the forward-backward algorithm. Typical cases of image restoration, where dual optimization is used, are based on approximations of type 2 (see Section 3).

The type of chosen approximation defines how the sequence $(\epsilon_{h,k})_{k\in\mathbb{N}}$ will be summable against k^{2d} and thus, it does not depend on the multilevel framework.

Convergence of Algorithm 2.1. We now discuss the convergence of our algorithm for the three types of approximation of the proximity operator.

We first consider a standard inexact forward-backward with a finite number of multilevel coarse corrections.

Theorem 2.15 (Approximation of Type 0). *Let us suppose in Algorithm 2.1 that $\forall k \in \mathbb{N}^*$, $\alpha_{h,k} = 0$ at step 14, that the assumptions of Lemma 2.10 hold, and that the sequence $(\epsilon_{h,k})_{k\in\mathbb{N}}$ is such that $\sum_{k\in\mathbb{N}} \sqrt{\|\epsilon_{h,k}\|} < +\infty$. Set $x_{h,0} \in \mathbb{R}^{N_h}$ and choosing approximation of Type 0, the sequence $(x_{h,k})_{k\in\mathbb{N}}$ converges to a minimizer of F_h.*

Proof. The proof stems from Theorem 3.4 in [20] applied to the defined sequence. ∎

Theorem 2.16 (Approximations of Type 1 and Type 2). *Let us suppose in Algorithm 2.1, that $\forall k \in \mathbb{N}^*$, $t_{h,k+1} = \left(\frac{k+a}{a}\right)^d$, with (a, d) satisfying the conditions in [2, Definition 3.1], and that the assumptions of Lemma 2.10 hold. Moreover, if we assume that:*

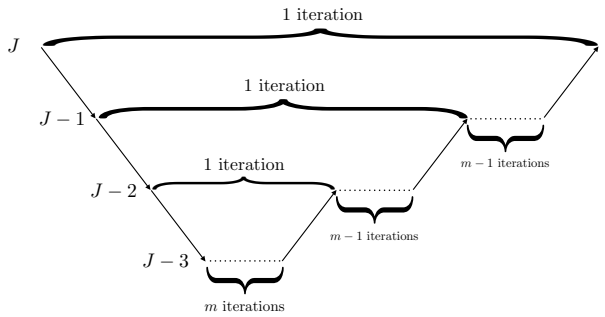

Figure 2: Scheme of a typical V-cycle for a multilevel algorithm with 4 levels and 3 coarse levels. When we use p times the coarse model, we repeat p times this V-cycle scheme.

$\sum_{k=1}^{+\infty} k^d \sqrt{\epsilon_{h,k}} < +\infty$ *in the case of Type 1 approximation* ($i = 1$),
$\sum_{k=1}^{+\infty} k^{2d} \epsilon_{h,k} < +\infty$ *in the case of Type 2 approximation* ($i = 2$),
then, we have that:

- *The sequence* $(k^{2d}(F_h(x_{h,k}) - F_h(x^*)))_{k\in\mathbb{N}}$ *belongs to* $\ell_\infty(\mathbb{N})$.
- *The sequence* $(x_{h,k})_{k\in\mathbb{N}}$ *converges to a minimizer of* F_h.

Proof. We combine [2, Theorem 3.5, 4.1, and Corollary 3.8] with Lemma 2.10 to prove the desired result. ∎

Theorem 2.15 and 2.16 generalize convergence results previously obtained in [40, Theorem 1]. When $\epsilon_{h,k} = 0$ for all k, we recover the convergence result obtained in [41, Theorem 1] for exact proximity operators.

2.3. Extension to the multilevel case. Extending these convergence results to more than two levels is straightforward. If the algorithm is used on J levels, we just have to apply the analysis derived above to each pair of consecutive levels. Then, recursively, showing that the coarsest level produces a bounded coarse correction will ensure that the upper finer level will converge to one of its minimizers, producing in turn a bounded coarse correction for the next upper finer level, and so on.

Defining the coarse cycles. We use the following notation for the multilevel schemes. If the dimension of the problem at fine level is $N_h = (2^J)^2$, following the classical wavelet nomenclature, we index with J the finest level. So, for an image of size 1024×1024, $J = 10$. The coarse levels are then associated to $J-1$, $J-2$, $J-3$, etc. We use V-cycles [8], as depicted in Figure 2.

3. IML FISTA for image reconstruction. In this section we adapt our Inexact MultiLevel FISTA to image reconstruction problems such as Problem (1.1). We present our problem in a multilevel context, then we propose CIT operators designed for image reconstruction problems and we derive the construction of a good coarse model through a specific choice of smoothing. Finally, we detail the computation of the proximity operator of $g_h \circ \mathrm{D}_h$.

3.1. Definition of the problem at fine level. Let us express Problem (1.1) in multilevel notations:

$$\widehat{x} \in \underset{x_h \in \mathbb{R}^{N_h}}{\operatorname{Argmin}} F_h(x) := f_h(\mathrm{A}_h x_h) + g_h(\mathrm{D}_h x_h) \tag{3.1}$$

with $A_h \in \mathbb{R}^{M_h \times N_h}$ and $\mathrm{D}_h \in \mathbb{R}^{(N_h \times \widetilde{K}) \times N_h}$ $(\widetilde{K}, M_h > 0)$. The parameter $\tilde{K}$ expresses the fact that operator D_h can map x_h to a higher dimensional space, e.g. $\widetilde{K} = 2$ for Total Variation penalization. In this expression, $x_h = \left(x_h^i\right)_{1 \leq i \leq N_h}$ is the vectorized version of an image X_h of $N_{h,r}$ rows and $N_{h,c}$ columns, and where each pixel corresponds to a vector of $C \geq 1$ components (e.g. $C = 3$ for the RGB bands of a color image), which thus gives $N_h = N_{h,r} \times N_{h,c} \times C$. In the following, as the operators we present apply separately to each channel, for the sake of clarity and without loss of generality, we present their construction for grayscale images corresponding to $C = 1$.

3.1.1. Examples of data fidelity term $f_h \circ \mathrm{A}_h$.

Deblurring problem. When the degradation of the image corresponds to a blurring effect, the operator A_h is a convolution matrix built from a two dimensional Point Spread Function (PSF). As it is the case for Gaussian blurs, the PSF function takes often the form of a separable kernel (horizontally and vertically) and A_h can be decomposed into a Kronecker product:

$$\mathrm{A}_h = \mathrm{A}_{h,r} \otimes \mathrm{A}_{h,c} \tag{3.2}$$

with $\mathrm{A}_{h,r} \in \mathbb{R}^{N_{h,c} \times N_{h,c}}$ and $\mathrm{A}_{h,c} \in \mathbb{R}^{N_{h,r} \times N_{h,r}}$. From the numerical viewpoint, this Kronecker decomposition is particularly efficient for processing large images, and can be easily implemented with the HNO package [35]. Finally, as it is common in image restoration, the data-fidelity term is a least square regression:

$$(\forall x_h \in \mathbb{R}^{N_h}) \qquad f_h(\mathrm{A}_h x_h) = \frac{1}{2}\|\mathrm{A}_h x_h - z_h\|_2^2 = \frac{1}{2}\sum_{i=1}^{N_h}((\mathrm{A}_h x_h)^i - (z_h)^i)^2. \tag{3.3}$$

Inpainting problem. When the degraded image coincides with the original image with missing pixels, the reconstruction task is called inpainting and A_h is a measurement operator that keeps a random subset $I \subseteq \{1, \ldots, N_h\}$ of pixels of the image and deletes the others. Formally A_h takes the form of a diagonal matrix with a Bernoulli random variable (zeros and ones) on its entries, and it plays the role of a mask applied to the image x_h:

$$(\mathrm{A}_h x_h)^i = \begin{cases} x_h^i & \text{if } i \in I \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

In this case too, the data-fidelity term is a least square regression as in Equation (3.3).

Remark 3.1. Conveniently, such data-fidelity terms are separable and may be generally expressed with a proper l.s.c. function $\Psi : \mathbb{R} \to]-\infty, +\infty]$ such that:

$$(\forall u \in \mathbb{R}^{M_h}) \qquad f_h(u) := \sum_{i=1}^{M_h} \Psi(u^i). \tag{3.5}$$

3.1.2. Examples of regularization term $g_h \circ \mathrm{D}_h$.

Total Variation. The operator D_h associated with the Total Variation (TV) computes the first order differences between the component i of x_h and its horizontal/vertical nearest neighbours $(x_h^{i_c}, x_h^{i_r})$ (lower/right in the image case). It is defined such that for all $x_h \in \mathbb{R}^{N_h}$, and for each component $i \in \{1, \dots, N_h\}$,

$$(\mathrm{D}_h x_h)^i = \left[\; x_h^i - x_h^{i_r} \, , \, x_h^i - x_h^{i_c} \; \right], \tag{3.6}$$

with special attention to the management of border effects. Here $\mathrm{D}_h x_h$ belongs to $\mathbb{R}^{N_h \times 2}$ ($\widetilde{K} = 2$). With this definition, the classical isotropic Total Variation semi-norm [4] reads:

$$g_h(\mathrm{D}_h x_h) = \lambda_h \sum_{i=1}^{N_h} \| \, (\mathrm{D}_h x_h)^i \, \|_2 = \lambda_h \sum_{i=1}^{N_h} \sqrt{|x_h^i - x_h^{i_1}|^2 + |x_h^i - x_h^{i_2}|^2} = \lambda_h \|\mathrm{D}_h x_h\|_{2,1} \tag{3.7}$$

with $\lambda_h > 0$.

Non-Local Total Variation. The operator D_h associated with the Non-Local Total Variation (NLTV) extends TV to a non local neighbourhood of the current pixel i. In words, it is the operator that computes the weighted differences between the current pixel i of an image x_h and a subset $\mathcal{N}_i$ of pixels localized near i.

For every $x_h \in \mathbb{R}^{N_h}$, and at each pixel $i \in \{1, \dots, N_h\}$, for some given weights $\omega^{i,j} > 0$,

$$(\mathrm{D}_h x_h)^i = \left[\; \omega^{i,j} \left(x_h^i - x_h^j \right) \; \right]_{j \in \mathcal{N}_i}. \tag{3.8}$$

Here $\mathrm{D}_h x_h$ belongs to $\mathbb{R}^{N_h \times \widetilde{K}}$ and $\widetilde{K}$ is the cardinality of the subset $\mathcal{N}_i$. For every $i \in \{1, \dots, N_h\}$ and $j \in \mathcal{N}_i$, the weights $\omega^{i,j} > 0$ depend on the similarity (e.g., ℓ_2 norm) between patches that are centered around components i and j of the image [17].

As for the isotropic Total Variation semi-norm, a ℓ_p ($p \geq 1$) based NLTV semi-norm takes on the form:

$$g_h(\mathrm{D}_h x_h) = \lambda_h \sum_{i=1}^{N_h} \| \, (\mathrm{D}_h x_h)^i \, \|_p, \tag{3.9}$$

with $\lambda_h > 0$.

Remark 3.2. (*General formulation of the regularization for image reconstruction problems*) As for the data fidelity term, the sums in the expressions (3.7) and (3.9) may be expressed as separable functions. To this end, we define a proper l.s.c. function $\varphi : \mathbb{R}^{\widetilde{K}} \to]-\infty, +\infty]$ such that, for any $v = (v^i)_{1 \leq i \leq N_h}$ and $v^i \in \mathbb{R}^{\widetilde{K}}$, we have:

$$g_h(v) := \lambda_h \sum_{i=1}^{N_h} \varphi(v^i). \tag{3.10}$$

In the specific case of isotropic TV, v^i is equal to $\left[\; x_h^i - x_h^{i_1} \quad x_h^i - x_h^{i_2} \; \right]$ and $\varphi(\cdot) = \|\cdot\|_2$ (cf. (3.7)). Whereas in the case of NLTV, $v^i = \left[\omega^{i,j}(x_h^i - x_h^j)\right]_{j \in \mathcal{N}_i}$ and $\varphi(\cdot) = \|\cdot\|_p$ (cf. (3.9)).

Although the formulations (3.5) and (3.10) are particularly suitable for computing the proximity operator of g_h and also for approximating the proximity operator of $g_h \circ \mathrm{D}_h$, the existence of separable forms is not necessary to define our multilevel algorithm applied to image reconstruction problems (see Section 3.4).

3.2. Information transfer for image reconstruction problems. In the context of image reconstruction problems, we consider CIT operators that relies on wavelet bases (referred to as wavelet CIT in the following). The idea of constructing such information transfer operators traces back to works dedicated to image deblurring problems either based on biorthogonal wavelets [16] or Haar and Symmlet wavelets [24, 28, 29]. Our objective is to obtain a computationally efficient coarse approximation of a vector lying in a higher resolution space, from the approximation coefficients of its discrete wavelet transform (DWT). We impose in this context that $N_h = (N_{h,r} \times N_{h,c}) = 4 \times N_H = (2\,N_{H,r} \times 2\,N_{H,c})$. For a generic quadrature mirror filter $\mathbf{q} = (q_1, \ldots, q_m)$ (or any filter one could think of):

$$I_h^H := (\mathbf{R}_{\mathbf{q},r} \otimes \mathbf{R}_{\mathbf{q},c})\,, \tag{3.11}$$

where $\mathbf{R}_{\mathbf{q},c}$ is the decimated $N_{H,r}$-by-$N_{h,r}$ matrix (every other line is kept) of the $N_{h,r}$-by-$N_{h,r}$ Toeplitz matrix engendered by $\mathbf{q}$ as :

$$\begin{pmatrix} q_1 & q_2 & \ldots & q_m & 0 & \ldots & 0 \\ 0 & 0 & q_1 & q_2 & \ldots & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ 0 & \ldots & 0 & 0 & 0 & q_1 & q_2 \end{pmatrix}.$$

Similarly $\mathbf{R}_{\mathbf{q},r}$ is the decimated $N_{H,c}$-by-$N_{h,c}$ matrix (every other line is kept) of the $N_{h,c}$-by-$N_{h,c}$ Toeplitz matrix engendered by $\mathbf{q}$. For both matrices the vector $\mathbf{q}$ is completed with the right number of 0s to reach the size of $N_{h,r}$ or $N_{h,c}$.

3.3. Fast coarse models for image restoration problems. A challenging numerical problem is to keep the efficiency of matrix-vector product computation at coarse level if it exists at fine level. For instance, when considering convolutions, if the convolution matrix is expressed with a Kronecker product, such structure can be preserved with the right definition of operators at coarse levels.

A_H in the deblurring problem. Thanks to the Kronecker factorization of both A_h and I_h^H, the coarsened operator A_H can be written as:

$$\mathrm{A}_H = \left(\mathbf{R}_{\mathbf{q},c} A_{h,r} \mathbf{R}_{\mathbf{q},c}^T\right) \otimes \left(\mathbf{R}_{\mathbf{q},r} A_{h,c} \mathbf{R}_{\mathbf{q},r}^T\right)$$

preserving the same computational efficiency. Thus in image restoration problems where a separable blur is used, it is straightforward to design coarse operators (which can be computed beforehand) that are fast for matrix-vector products while keeping fidelity to the fine level.

A_H in the inpainting problem. Due to the specific diagonal form of A_h, the coarsened inpainting operator A_H simply stems from decimating the rows and the columns of A_h by a factor 2. $\mathrm{A}_H \in \mathbb{R}^{N_H \times N_H}$ remains a diagonal indicator matrix of a pixel subset $J \subseteq \{1, \ldots, N_H\}$

acting as a mask on the coarse image:

$$(\mathrm{A}_H x_H)^j = \begin{cases} x_H^j & \text{if } j \in J \\ 0 & \text{otherwise} \end{cases}$$

Examples of operators D_H. For the regularization operators, the construction is simpler. For both TV and NLTV, we use the same hyper-parameters (maximum number of patches, size of patches, computation of similarity between patches, etc.) for D_H as for D_h. D_H is thus playing the same role as D_h but for images of size N_H. For every $x_H \in \mathbb{R}^{N_H}$, and at each pixel $i \in \{1, \dots, N_H\}$, given some weights $\omega^{i,j} > 0$ we define

$$(\mathrm{D}_H x_H)^i = \left[\ \omega^{i,j} \left(x_H^i - x_H^j \right) \ \right]_{j \in \mathcal{N}_i}. \tag{3.12}$$

Here $\mathrm{D}_H x_H$ belongs to $\mathbb{R}^{N_H \times \widetilde{K}}$.

3.4. Coarse model construction. For the coarse model it is natural in this context to choose L_H and R_H as

$$L_H = f_H \circ \mathrm{A}_H, \quad R_H = g_H \circ \mathrm{D}_H,$$

where A_H, D_H are defined as described above and f_H, g_H are the restrictions of f_h and g_h to a subspace of reduced dimension. We then have:

$$(\forall x_H \in \mathbb{R}^{N_H}), \ \ f_H(\mathrm{A}_H x_H) = \frac{1}{2} \sum_{i=1}^{N_H} ((\mathrm{A}_H x_H)^i - (z_H)^i)^2,$$
$$g_H(\mathrm{D}_H x_H) = \lambda_H \sum_{i=1}^{N_H} \|(\mathrm{D}_H x_H)^i\|_p.$$

Ideally, in order to speed up the computations, one would like to choose an approximation R_H whose proximity operator is known under closed form, even when R_h does not possess this desirable property. However, we have seen in our experiments that choosing an R_H not faithful to R_h deteriorates the performance of the multilevel algorithm (for instance, when R_h is the TV based norm, choosing a Haar wavelet based norm for R_H is sub-optimal, even though there is a link between Haar wavelet and total variation thresholdings. [39, 51]).

This motivates the construction presented in Section 2 that we adapt here to our problem: we replace R_h and R_H by their corresponding smooth Moreau envelopes, which present several interesting properties.

Definition 3.3. (Moreau envelope). *Let $\gamma > 0$ and $R \colon \mathbb{R}^N \to (-\infty, +\infty]$ a convex, lower semi-continuous, and proper function. The Moreau envelope of R, denoted by ${}^{\gamma}R$, is the convex, continuous, real-valued function defined by*

$${}^{\gamma}R = \inf_{y \in \mathbb{R}^N} R(y) + \frac{1}{2\gamma} \|\cdot - y\|^2. \tag{3.13}$$

${}^{\gamma}R can be expressed explicitly with $\mathrm{prox}_{\gamma R}$ *[3, Remark 12.24] as follows:*

$$ {}^{\gamma}R(x) = R(\mathrm{prox}_{\gamma R}(x)) + \frac{1}{2\gamma}\|x - \mathrm{prox}_{\gamma R}(x)\|^2. $$

Moreover, ${}^{\gamma}R$ *is Fréchet differentiable on* $\mathbb{R}^N$*, and its gradient is* γ^{-1}*-Lipschitz and such that [3, Prop. 12.30]*

$$ \nabla({}^{\gamma}R) = \gamma^{-1}(\mathrm{Id} - \mathrm{prox}_{\gamma R}). \tag{3.14} $$

However, the last equation is not directly applicable because we assumed that the proximity operator of $g \circ \mathrm{D}$ had no explicit form. Therefore, instead of directly using the Moreau envelope of R, we first compute the Moreau envelope of g and then we compose it with D. This smoothing respects Definition 2.1 :

Lemma 3.4. *${}^{\gamma}g \circ \mathrm{D}$ is a smoothed convex function in the sense of Definition 2.1.*

Proof. Remark that ${}^{\gamma}g$ is a smooth convex function in the sense of Definition 2.1 [6]. By [6, Lemma 2.2], the fact that ${}^{\gamma}g \circ \mathrm{D}$ is a smooth function applied to a linear transformation concludes the proof. ∎

This smooth approximation has the following interesting property:

Lemma 3.5. [43, Lemma 3.2] *For any $x \in \mathbb{R}^N$, $\mathrm{D} : \mathbb{R}^N \to \mathbb{R}^K$ and $g : \mathbb{R}^K \to \mathbb{R}$ a convex, l.s.c., and proper function, we have that:*

$$ \nabla\left({}^{\gamma}g \circ \mathrm{D}\right)(x) = \gamma^{-1}\mathrm{D}^*\left(\mathrm{D}x - \mathrm{prox}_{\gamma g}(\mathrm{D}x)\right). \tag{3.15} $$

This means that an explicit form of $\mathrm{prox}_{\gamma_h g_h}$ is sufficient to express the gradient of ${}^{\gamma_h}g_h(\mathrm{D}_h\cdot)$. Accordingly, we define the following coarse model, where the first order coherence is enforced between the two objective functions, similarly smoothed at fine and coarse levels:

Definition 3.6. *A coarse model for the image restoration problem (1.1) is defined at iteration k of a multilevel algorithm as:*

$$ F_H(s_H) = (f_H \circ \mathrm{A}_H)(s_H) + ({}^{\gamma_H}g_H \circ \mathrm{D}_H)(s_H) + \langle v_{H,k}, s_H\rangle, \tag{3.16} $$

where $v_{H,k}$ will be set to:

$$ \begin{aligned} v_{H,k} =& I_h^H\left[(\nabla(f_h \circ \mathrm{A}_h) + \nabla({}^{\gamma_h}g_h \circ \mathrm{D}_h)(y_{h,k})\right] \\ &- (\nabla(f_H \circ \mathrm{A}_H) + \nabla({}^{\gamma_H}g_H \circ \mathrm{D}_H))(s_{H,k,0}). \end{aligned} \tag{3.17} $$

3.5. Computation of the proximity operator of $g_h \circ \mathrm{D}_h$. If D_h is the projection on a tight frame (e.g., a union of wavelets), meaning that $\mathrm{D}_h\mathrm{D}_h^* = \mu\mathrm{Id}$ for a constant $\mu > 0$ and D_h^* the adjoint of D_h, the proximity operator of $g_h \circ \mathrm{D}$ is expressed explicitly through the proximity operator of g_h, which is known in a large number of cases.

Otherwise, a common way of estimating the proximity operator is through the *dual problem*. Denoting $R_h = g_h \circ \mathrm{D}_h$, we have that (see for instance [42]):

$$ (\forall x \in \mathbb{R}^{N_h}) \qquad \mathrm{prox}_{\gamma R_h}(x) := \mathrm{prox}_{\gamma g_h \circ \mathrm{D}_h}(x) = x - \mathrm{D}_h^*\widehat{u} \tag{3.18} $$

with:

$$\widehat{u} \in \underset{u\in\mathbb{R}^K}{\arg\min}\ \frac{1}{2}\|\mathrm{D}_h^* u - x\|^2 + \gamma g_h^*(u), \tag{3.19}$$

where g_h^* is the convex conjugate of g_h. This problem is known as the dual problem. An approximation of $\widehat{u}$ may be obtained by applying any convenient optimization method to (3.19). For instance FISTA would yield the following sequence (choosing $u_0 = v_0$):

$$u_{k+1} = \left(\mathrm{Id} - \gamma \mathrm{prox}_{g_h/\gamma}\left(\cdot/\gamma\right)\right)\left((\mathrm{Id} - \mathrm{D}_h\mathrm{D}_h^*)v_k + \gamma_h \mathrm{D}_h x\right) \tag{3.20}$$

$$v_{k+1} = (1+\alpha_k)u_{k+1} - \alpha_k u_k. \tag{3.21}$$

where the first step is deduced from the Moreau decomposition [3]. Dual optimization is a simple way to estimate the proximity operator while offering guarantees on the computed approximation, as stated in the following lemma.

Proposition 3.7. (Dual optimization yields approximation of type 2) *Assume that* $(u_k)_{k\in\mathbb{N}}$ *is a minimizing sequence for the dual function in* (3.19)*. This yields:*

- *A convergent sequence* $(x - \mathrm{D}_h^* u_k)_{k\in\mathbb{N}}$ *to the proximity operator* (3.18)*.*
- *This sequence provides a type 2 approximation of the proximity operator.*

Proof. The first point comes from [52, Theorem 5.1]. Then the approximation of type 2 comes from [52, Proposition 2.2, and 2.3]. ∎

Remark 3.8. The smoothing obtained with the Moreau envelope gives the possibility to perform exact gradient steps on F_H at the coarse level, as follows:

$$\begin{aligned} s_{H,k,\ell+1} &= \Phi_{H,\ell}(s_{H,k,\ell}) \\ &= s_{H,k,\ell} - \tau_H\left(\nabla(f_H \circ \mathrm{A}_H)(s_{H,k,\ell}) + \nabla({}^{\gamma_H}g_H \circ \mathrm{D}_H)(s_{H,k,\ell}) + v_{H,k}\right) \\ &= s_{H,k,\ell} - \tau_H\left(\nabla(f_H \circ \mathrm{A}_H)(s_{H,k,\ell}) + \tau_H^{-1}\mathrm{D}_H^*\left(\cdot - \mathrm{prox}_{\tau_H g_H}(\cdot)\right)\left(\mathrm{D}_H s_{H,k,\ell}\right) + v_{H,k}\right). \end{aligned}$$

From this formulation, one can see that for each sub-iteration performed at fine level on (3.19) by (3.20)-(3.21) to compute the proximity operator of $g_h \circ \mathrm{D}_h$, one may use for the same computational cost at least N_h/N_H (factor of dimensionality reduction from fine to coarse level) iterations at coarse level. This is a great improvement from exact proximal methods where the computational cost of a fine level iteration could account for roughly N_h/N_H iterations at coarse level instead of N_h/N_H times the number of sub-iterations here.

3.6. IML FISTA for image restoration. We can now present in details, the version of our IML FISTA algorithm, specifically designed to address image restoration problems.

Our method is sketched in Algorithm 3.1. The step sizes for both levels can be set to a value below the thresholds that warrant convergence (Lemma 2.10). Those thresholds depend on the Lipschitz constants of the considered functions, which are either known explicitly or bounded below by a linear search. As for the step size $\bar{\tau}_{h,k}$, it is chosen by a linear search so that Lemma 2.9 is verified.

To ensure the convergence of the iterates, we impose at most p calls to the F_H model (each one corresponding to a full V-cycle).We choose not to use any downscaling condition (as they may be time-consuming to check) and we fix beforehand, the multilevel pattern in all our experiments in Section 4.

Algorithm 3.1 IML FISTA for image restoration

Input: $x_{h,0}$, $\tau_h, \tau_H, \epsilon_h, \gamma_h, \gamma_H, m, p > 0$, $r = 0$, $t_{h,0} = 1$, $a > 2$, $d \in [0, 1]$
while $k \leq k_{\max}$ **do**
  if $r < p$ **then**
    $r = r + 1$
    $s_{H,k,0} = I_h^H y_{h,k}$
    $v_{H,k} = I_h^H \nabla \left(f_h \circ A_h + {}^{\gamma_h} g_h \circ \mathrm{D}_h\right)(y_{h,k}) - \nabla \left(f_H \circ A_H + {}^{\gamma_H} g_H \circ \mathrm{D}_H\right)(s_{H,k,0})$
    for $\ell = 0 \ldots m - 1$ **do**
      $s_{H,k,\ell+1} = s_{H,k,\ell} - \tau_H \left(\nabla f_H \circ \mathrm{A}_H + {}^{\gamma_H} g_H \circ \mathrm{D}_H\right)(s_{H,k,\ell}) + v_{H,k}$
    end for
    $\bar{y}_{h,k} = y_{h,k} + \bar{\tau}_{h,k} I_H^h (s_{H,k,m} - s_{H,k,0})$,
  else
    $\bar{y}_{h,k} = y_{h,k}$
  end if
  $x_{h,k+1} = \mathrm{T}_i^{\epsilon_{h,k}}(\bar{y}_{h,k})$
  $t_{h,k+1} = \left(\frac{k+a}{a}\right)^d$, $\alpha_{h,k} = \frac{t_{h,k}-1}{t_{h,k+1}}$
  $y_{h,k+1} = x_{h,k+1} + \alpha_{h,k}(x_{h,k+1} - x_{h,k})$
end while

4. Experimental results. The objective of this section is to illustrate the benefits of the proposed IML FISTA in various image reconstruction tasks, especially involving large-scale images. We show that FISTA and IML FISTA both converge to the same solution but IML FISTA always converges faster, ensuring a good reconstruction in few iterations and thus providing a method of tremendous interest for large-scale imaging applications. Code and examples are available here.

4.1. Experimental setting.

Degradation types. We consider two types of image reconstruction problems: a restoration problem where the linear operator A is a Gaussian blur, and an inpainting problem where A models the action of random pixel deletion. In both cases, we consider an additive white Gaussian noise of standard deviation σ.

Minimization problem. At fine level, we consider the state-of-the-art optimization problem in this context, the minimization of the sum of a quadratic data-fidelity term and a sparsity prior based on a total variation $\ell_{1,2}$-norm:

$$(\forall x \in \mathbb{R}^{N_h}), \qquad F_h(x) = \frac{1}{2}\|\mathrm{A}_h x - z_h\|_2^2 + \lambda_h \|\mathrm{D}_h x\|_{1,2}, \tag{4.1}$$

with $\lambda_h > 0$. In all the experiments, the regularisation parameter λ_h was chosen by a grid search, in order to maximize the SNR of $\widehat{x}$ computed by FISTA at convergence. Finally, we choose as initialization $x_{h,0}$ the Wiener filtering of z.

Experiment datasets. We consider two color images of different sizes to evaluate the impact of the problem's dimension: "ImageNet Car" the picture of a yellow car of size $512 \times 512 \times 3$, taken from the ImageNet dataset, and a picture taken by the James Webb Space Telescope


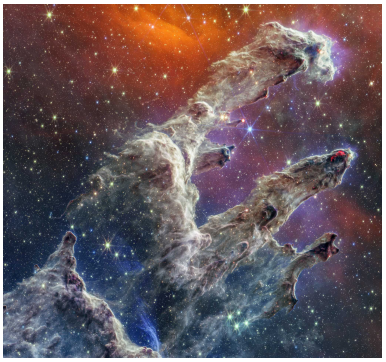

Figure 3: ImageNet Car "ILSVRC2012_test_00000164". Source. Pillars of Creation. Credits: SCIENCE: NASA, ESA, CSA, STScI (Image processing): Joseph DePasquale (STScI), Alyssa Pagan (STScI), Anton M. Koekemoer (STScI). Source.

with its Near-Infrared Camera and its Mid-Infrared Instrument of the structure called "Pillars of Creation" of size $2048 \times 2048 \times 3$ (Figure 3). Pixels values are normalized so that the maximum value across all channels is 1.

Multilevel structure. For all our experiments we use a 5-levels hierarchy. For "Pillars of Creation" the first level corresponds to an image of size $2048 \times 2048 \times 3$, and the fifth level to an image of size $128 \times 128 \times 3$. Similarly for "ImageNet Car" the first level corresponds to an image of size $512 \times 512 \times 3$ and the fifth level to an image of size $32 \times 32 \times 3$.

The coarse model associated to (4.1) is written as:

$$(\forall x \in \mathbb{R}^{N_H}), \qquad F_H(x) = \frac{1}{2}\|\mathrm{A}_H x - z_H\|_2^2 + \lambda_H \left({}^{\gamma_H} g_H \left(\mathrm{D}_H x\right)\right) + \langle v_H, x\rangle, \tag{4.2}$$

with $\lambda_H > 0$, $z_H = I_h^H z_h$ and g_H the $\ell_{1,2}$-norm applied on the N_H components of $\mathrm{D}_H x$, as for the fine level. As the dimension of the problem is reduced by a factor 4 each time we pass from a level to a coarser one, we choose the value of the regularization parameter λ_H at a coarse level as a quarter of the value of the regularization parameter of the immediate upper level. In practice, this ratio gives the best performance in terms of decrease of the fine level objective function. The CIT operators were built for every pair of levels with "Symmlet 10" wavelets corresponding to a filter size of 20 coefficients.

Based on our previous study in [40], we always impose $p = 2$ coarse corrections (V-cycles) with $m = 5$ iterations per level, and always carried out at the beginning of the optimization process, as this configuration showed to perform well for different levels of degradation. This appears to be a common setting in the multilevel literature [31,36–38,40,41,47,48]. Increasing the number of coarse corrections may be occasionally beneficial, while sometimes this makes the potential gain decrease. Being difficult to know this without solving several times the same optimization problem, we deem more important to display a configuration that is satisfactory regardless of the specific problem parameters.

Accuracy of the computation of the proximity operator. Convergence guarantees of the algorithm are directly linked to the decrease of the error introduced by estimating the proximity

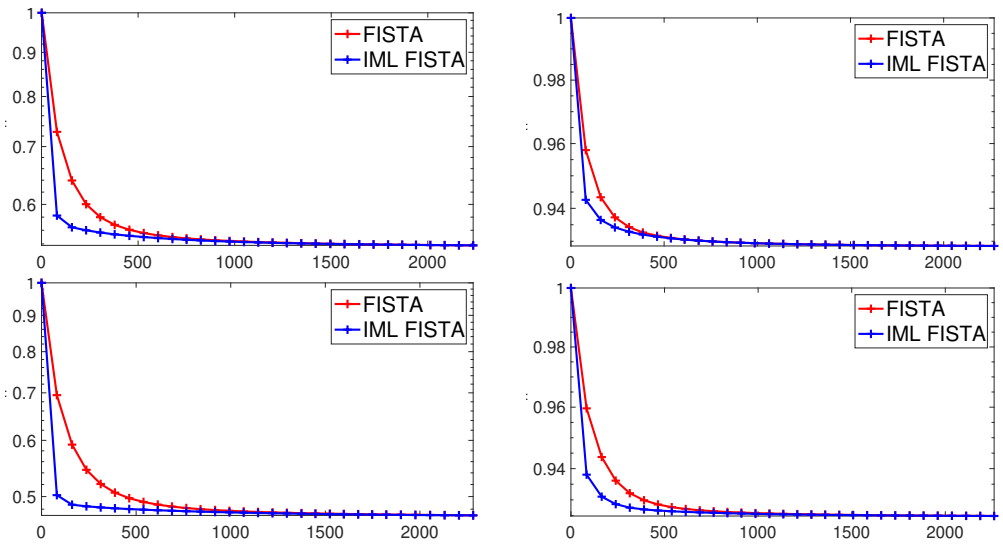

Figure 4: Deblurring $\ell_{1,2}$-TV for the Pillars of Creation image. Objective function (normalized w.r.t. the initial value) vs CPU time (sec). First column: σ(noise) = 0.01; second column: σ(noise) = 0.05. First row: dim(PSF) = 20, σ(PSF) = 3.6; second row: dim(PSF) = 40, σ(PSF) = 7.3.

operator at each iteration. The necessary speed decrease depends on the choice of d (Step 15 in Algorithm 3.1) and on the type of approximation we are using. Indeed, based on the convergence result derived earlier (Theorem 2.16), going from $d = 1$ to $d = 0$ relaxes the speed decrease. In all cases, a lower error is correlated with a higher computational cost, which is why some strategies rather use a fixed budget of sub-iterations to compute the proximity operator [4]. This fixed budget comes at the cost of a limited precision one can obtain on the estimated solution and may lead to divergence after a large number of iterations.

This problem was notably addressed in [44], where the authors introduced the Speedy Inexact Proximal-Gradient Strategy (SIP). It is based on a dynamic strategy to update the number of sub-iterations, increasing it each time two following iterates yield a difference of objective functions' values under a fixed tolerance. Since the coarse correction of our algorithm tends to accentuate the decrease of the objective function between two consecutive iterates, we fix this tolerance to 0. Each time the criterion is verified, instead of increasing the number of sub-iterations, we change the stopping criterion in the proximity operator estimation by reducing the tolerance *tol* on the relative distance between two consecutive sub-iterates by a factor 10. This minimization is carried out by FISTA coupled with a warm start strategy as in [4]. We set the initialization value of *tol* based on the reconstruction quality of images in a Total Variation based denoising problem (that is equivalent to one computation of the associated proximity operator). Thus $tol = 10^{-8}$ at the start of the optimization process unless stated otherwise.

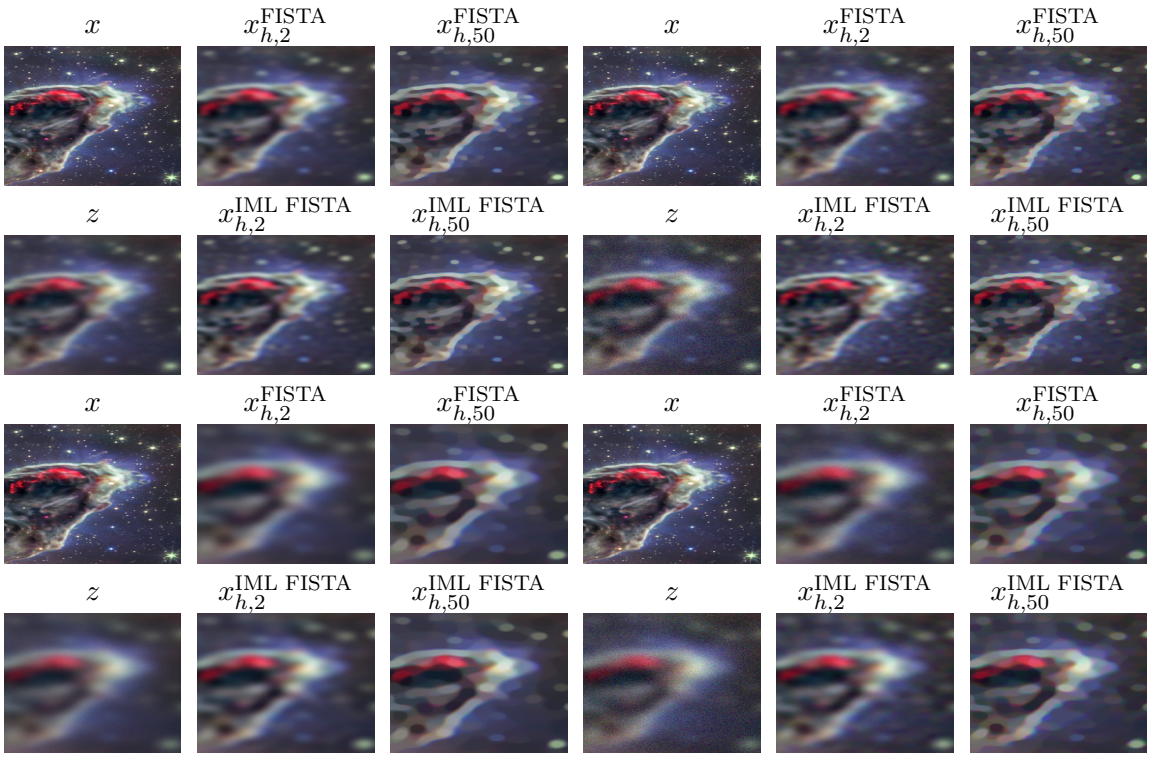

Figure 5: Deblurring $\ell_{1,2}$-TV for the Pillars of Creation image. Small crop of the image after 2 iterations and after 50 iterations for FISTA (top row) and IML FISTA (bottom row) compared to the original (x) and degraded (z) images.
First column: $\sigma(\text{noise}) = 0.01$; second column: $\sigma(\text{noise}) = 0.05$. First row: $\dim(\text{PSF}) = 20$, $\sigma(\text{PSF}) = 3.6$; second row: $\dim(\text{PSF}) = 40$, $\sigma(\text{PSF}) = 7.3$.

4.2. IML FISTA results on image restoration: deblurring. To get a full picture of the performance of IML FISTA, we compare it to standard FISTA in four scenarios, corresponding to four different combinations of the size of the Gaussian blur PSF and of the value of the standard deviation $\sigma(\text{noise})$ of the Gaussian noise. These four scenarios are described in Table 1. In each of the following figures, the order of the four plots reflects the scenarios reported in this table. For each of these four scenarios we tested two regularizations: $\ell_{1,2}$-TV and $\ell_{1,2}$-NLTV. Because the relative behaviour of IML FISTA with respect to FISTA is similar for the two regularizations, for the sake of conciseness, we only report here the results obtained with the $\ell_{1,2}$-TV prior.

Figure 4 and Figure 5 provide a first set of results for the 2048×2048 Pillars of Creation image. We can see that in all cases, the decreasing of the objective function of IML FISTA is faster than that of FISTA. Given the cost of estimating proximity operators for TV and NLTV based regularizations, the computational overhead of a multilevel step is almost negligible, as we expected (cf. Figure 4 and Remark 3.8). Thus, the two low cost coarse corrections are sufficient for our algorithm to gain an advantage that FISTA cannot recover without decreasing

Blur \ Noise	$\sigma(\text{noise}) = 0.01$	$\sigma(\text{noise}) = 0.05$
$\dim(\text{PSF}) = 20$, $\sigma(\text{PSF}) = 3.6$	low blur, low noise	low blur, high noise
$\dim(\text{PSF}) = 40$, $\sigma(\text{PSF}) = 7.3$	high blur, low noise	high blur, high noise

Table 1: Four scenarios of Gaussian blur degradation with additive Gaussian noise.

the tolerance on the computation of the proximity operator. As a result, this would entail higher computation time at each iteration as the error must decrease with the number of iterations to converge. Most interestingly, if we compare the methods at the very early stages of the optimization process, after the same number of iterations, IML FISTA reaches a much lower value for the objective function, leading to a much better reconstruction. The difference is particularly striking after 2 iterations (Figure 5).

One can also notice that increasing the noise (and thus increasing the value of regularization term λ) degrades the relative performance of our algorithm compared to FISTA. This behaviour was observed in the exact proximal case (with wavelet based regularization [40]) albeit it is far less pronounced here. Similarly, increasing the blur size improves the relative performances of IML FISTA, just like in the exact case.

We stress that the potential of multilevel strategies, especially for high levels of degradations (i.e., blurring and noise), is particularly evident for large scale images: on smaller problems the overhead introduced by the method may overcome the gain obtained in passing to lower resolutions. This is evident when looking at the results obtained in the same degradation context for the Yellow Car image of size $512 \times 512 \times 3$. We reproduce in Figure 6 the evolution of the objective function when the regularization is the $\ell_{1,2}$-TV norm. As one can see, the relative performances of IML FISTA compared to those of FISTA are less impressive than in the case of the Pillars of Creation image.

4.3. IML FISTA results on image inpainting. Here again, we consider four scenarios based on two variables: the percentage of missing pixels and the standard deviation of the Gaussian noise $\sigma(\text{noise})$. These four scenarios are specified in Table 2. For each of these four

Inpainting \ Noise	$\sigma(\text{noise}) = 0.01$	$\sigma(\text{noise}) = 0.05$
missing pixels 50%	low inpainting, low noise	low inpainting, high noise
missing pixels 90%	high inpainting, low noise	high inpainting, high noise

Table 2: Four scenarios of inpainting degradation with additive Gaussian noise.

scenarios we tested two regularizations: $\ell_{1,2}$-TV and $\ell_{1,2}$-NLTV. In this case we only report the results obtained with the $\ell_{1,2}$-NLTV prior.

Again, in all cases, the objective function decreases faster with IML FISTA than with FISTA, proving that the computational cost of multilevel steps is almost negligible. The two performed coarse corrections bring a considerable advantage to the minimization achieved with IML FISTA (Figure 7). Comparing the two methods after only two iterations of IML FISTA and of FISTA, is particularly convincing as we can observe it in Figure 8. Moreover,

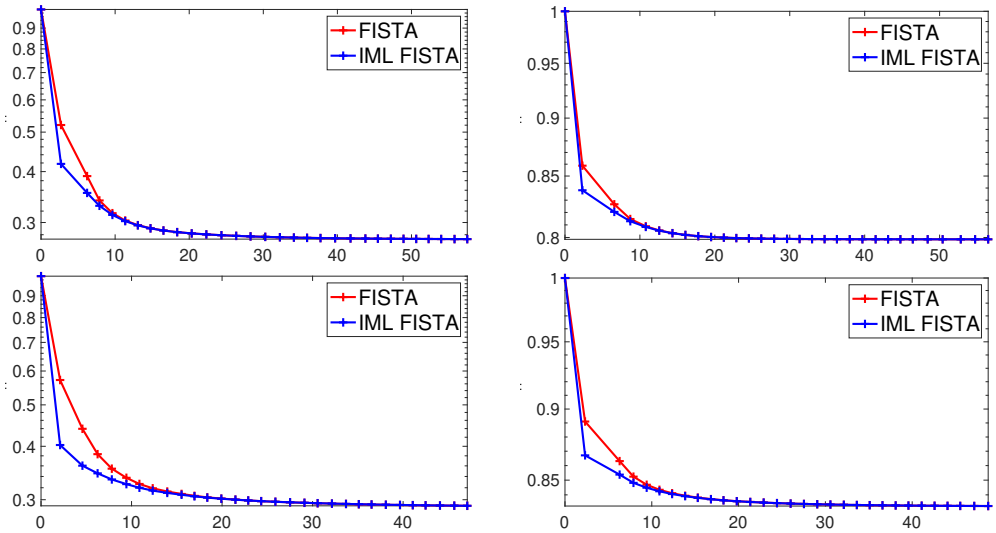

Figure 6: Deblurring $\ell_{1,2}$-TV for the Yellow Car image. Objective function (normalized with initialization value) vs CPU time (sec). First column: $\sigma(\text{noise}) = 0.01$; second column: $\sigma(\text{noise}) = 0.05$. First row: $\dim(\text{PSF}) = 20$, $\sigma(\text{PSF}) = 3.6$; second row: $\dim(\text{PSF}) = 40$, $\sigma(\text{PSF}) = 7.3$.

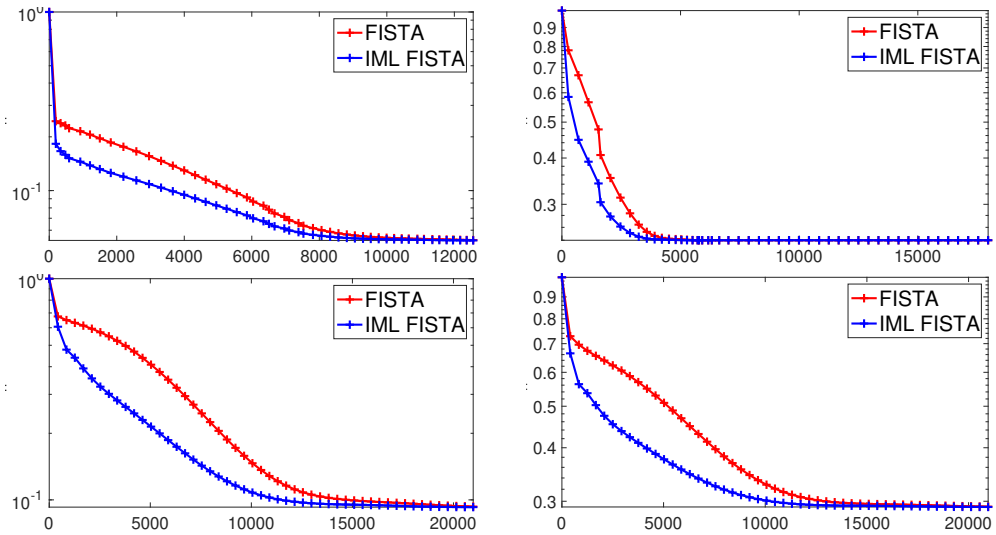

Figure 7: Inpainting $\ell_{1,2}$-NLTV for the Pillars of Creation image. Objective function (normalized with initialization value) vs CPU time (sec). First column: $\sigma(\text{noise}) = 0.01$; second column: $\sigma(\text{noise}) = 0.05$. First row: missing pixels 50%; second row: missing pixels 90%.

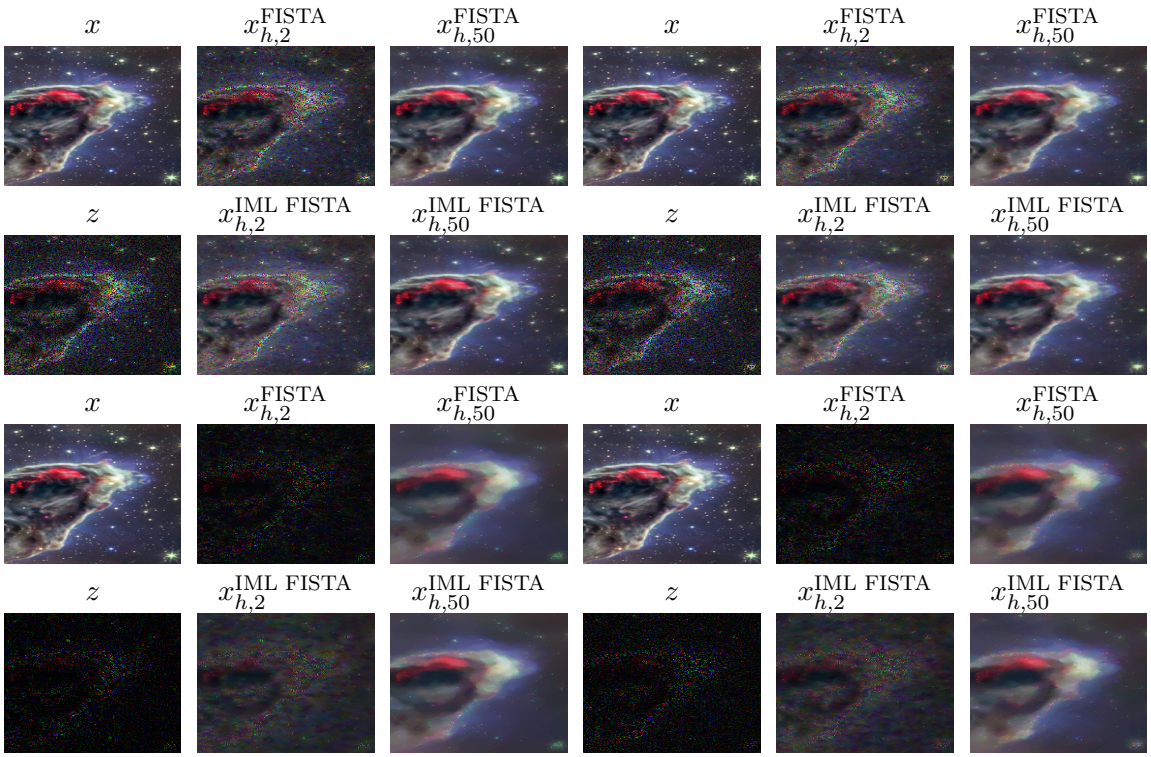

Figure 8: Inpainting $\ell_{1,2}$-NLTV for the Pillars of Creation image. Small crop of the image at 2 iterations and after 50 iterations for FISTA (top row) and IML FISTA (bottom row) compared to the original (X) and degraded (Z) images.
First column: $\sigma(\text{noise}) = 0.01$; second column: $\sigma(\text{noise}) = 0.05$. First row: missing pixels 50%; second row: missing pixels 90%.

as it was already the case for the deblurring task, IML FISTA outperforms FISTA in terms of convergence speed, specifically when the original image is heavily corrupted.

As for the deblurring task, we display the results under the same degradation contexts (i.e., inpainting and noise) for the Yellow Car image. We reproduce in Figure 9 the evolution of the objective function when the regularization is the $\ell_{1,2}$-NLTV norm. In contrast to the deblurring case, IML FISTA still performs better than FISTA for an inpainting task on a relatively small image size. This suggests that the dependency of IML FISTA's performances to the problem dimension is clearly linked to the degradation context.

4.4. Impact of the information transfer operators. In this section we investigate the influence of the choice of the CIT on the performance of our multilevel algorithm.

From the experiments of the previous section, we claim that the performances result from the combination of a faithful coarse minimization and a good design of the information transfer operators. This latters should allow to capture information over wider regions than what is typically done in multilevel optimization [37, 45, 47] where the filter used to generate

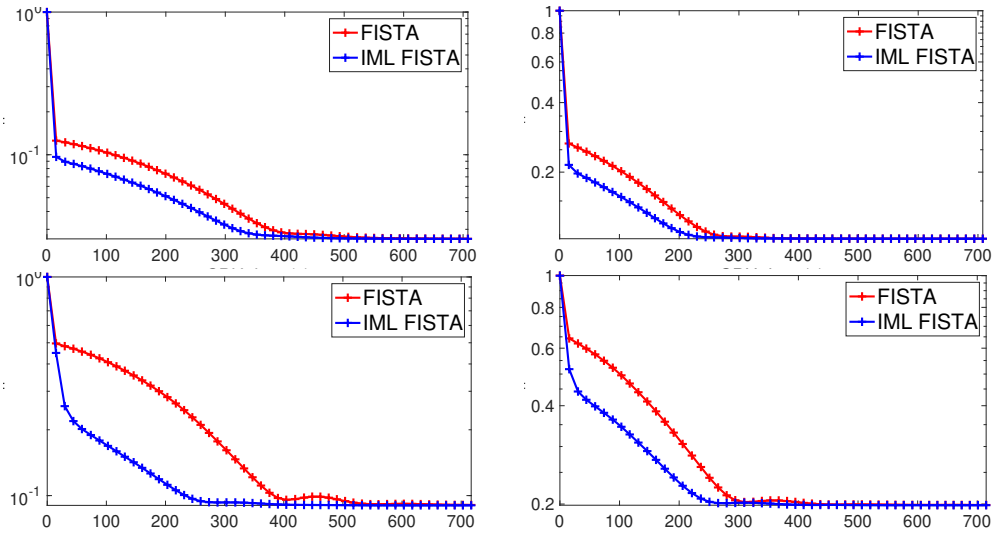

Figure 9: Inpainting $\ell_{1,2}$-NLTV for the Yellow Car image. Objective function (normalized with initialization value) vs CPU time (sec). First column: $\sigma(\text{noise}) = 0.01$; second column: $\sigma(\text{noise}) = 0.05$. First row: missing pixels 50%; second row: missing pixels 90%.

the information transfer operators is of a rather small size.

To test this hypothesis we compare the CIT built using the "Haar" wavelet (filter size equal to 2), the "Daubechies 20" wavelet (filter size equal to 20), and the "Symmlet 10" wavelet. The last two have the same quadrature mirror filter length. For deblurring problems, no significant difference was observed between these three CITs. The influence was more noticeable for inpainting problems, and the results are displayed in Figure 10. The principal factor seems to be the length of the filter, this is not surprising since it determines the domain extension over which pixels are aggregated. Nonetheless, even with the Haar wavelet, IML FISTA reaches lower objective function values faster than FISTA, meaning that the use of coarse models is beneficial to the optimization regardless of the wavelet-based CIT chosen.

5. Conclusion. We have proposed a convergent inexact multilevel FISTA method for image restoration able to handle state-of-the-art regularizations in this context. The proposed algorithm has theoretical convergence guarantees that are comparable to those obtained with leading algorithms in the field. From a practical point of view, our method shows very good performance on a wide range of degradation configurations and reaches good approximations of the optimal solution in a much smaller CPU time than FISTA, on large size problems. It also allows to deal with non differentiable functions whose proximity operator is not available under closed form, making the procedure applicable to a broad set of problems.

Among its many advantages, IML FISTA provides good quality reconstructions faster than FISTA. This opens up a great opportunity to deal with problems of large dimension, especially when limited computational resources prevent convergence from being achieved in a systematic way.In addition, this accelerated coarse approximation could play an important role in applications where image reconstruction is only a pre-processing task for a more

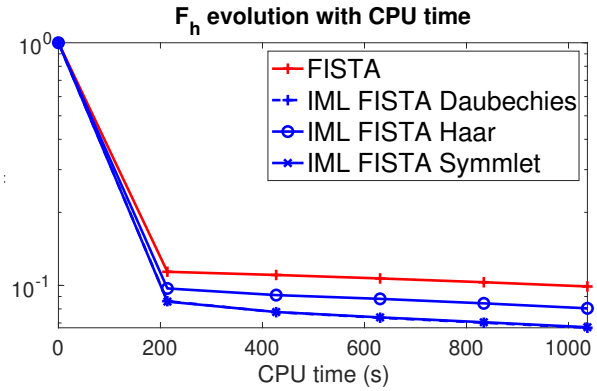

Figure 10: Inpainting $\ell_{1,2}$-NLTV for the Pillars of Creation image. σ(noise) = 0.01, missing pixels 50%. Comparison of objective function for three information transfer operators: "Haar", "Daubechies 20" and "Symmlet 10".

comprehensive treatment.

The main challenge for future work is to provide a better understanding of the link between the coarse and the fine level optimization. If the effect of coarse iterations and their impact on the different frequency components of the error is well studied for partial differential equations, much remains to be understood for proximal multilevel methods and specifically in the context of image restoration. For instance, it is yet not clear which is the best way of constructing lower level models, which deeply influences the performance of the method but depends on the problem at hand [40], or what are the conditions that make the coarse optimization useful for the general problem. One of the factors identified in this article is the nature and the intensity of the degradation : more degradation means better performance for IML FISTA compared to FISTA, while lower signal-to-noise ratio may worsen the results.

Acknowledgments. The authors would like to thank the GdR ISIS for the funding of the MOMIGS project and the ANR-19-CE48-0009 Multisc'In project. We also gratefully acknowledge the support of the Centre Blaise Pascal's IT test platform at ENS de Lyon (Lyon, France) for the computing facilities. The platform operates the SIDUS [1] solution developed by Emmanuel Quemener.

REFERENCES

[1] A. Ang, H. De Sterck, and S. Vavasis, *Mgprox: A nonsmooth multigrid proximal gradient method with adaptive restriction for strongly convex optimization*, arXiv preprint arXiv:2302.04077, (2023).

[2] J.-F. Aujol and C. Dossal, *Stability of Over-Relaxations for the Forward-Backward Algorithm, Application to FISTA*, SIAM Journal on Optimization, (2015), pp. 2408–2433, https://doi.org/10.1137/140994964.

[3] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, CMS Books in Mathematics, Springer International Publishing, New York, 2017, https://doi.org/10.1007/978-3-319-48311-5.

[4] A. Beck and M. Teboulle, *Fast Gradient-Based Algorithms for Constrained Total Variation Image*

Denoising and Deblurring Problems, IEEE Transactions on Image Processing, 18 (2009), pp. 2419–2434, https://doi.org/10.1109/TIP.2009.2028250.

[5] A. Beck and M. Teboulle, *A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems*, SIAM Journal on Imaging Sciences, (2009), pp. 183–202, https://doi.org/10.1137/080716542.

[6] A. Beck and M. Teboulle, *Smoothing and First Order Methods: A Unified Framework*, SIAM Journal on Optimization, 22 (2012), pp. 557–580, https://doi.org/10.1137/100818327.

[7] S. Boyd, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*, Foundations and Trends® in Machine Learning, 3 (2010), pp. 1–122, https://doi.org/10.1561/2200000016.

[8] W. L. Briggs, V. E. Henson, and S. F. McCormick, *A Multigrid Tutorial, Second Edition*, Society for Industrial and Applied Mathematics, second ed., Jan. 2000, https://doi.org/10.1137/1.9780898719505.

[9] H. Calandra, S. Gratton, E. Riccietti, and X. Vasseur, *On High-Order Multilevel Optimization Strategies*, SIAM Journal on Optimization, 31 (2021), pp. 307–330, https://doi.org/10.1137/19M1255355.

[10] H. Calandra, S. Gratton, E. Riccietti, and X. Vasseur, *On high-order multilevel optimization strategies*, SIAM Journal on Optimization, 31 (2021), pp. 307–330, https://doi.org/10.1137/19M1255355.

[11] A. Chambolle, V. Caselles, M. Novaga, D. Cremers, and T. Pock, *An introduction to Total Variation for Image Analysis.* preprint, Nov. 2009.

[12] A. Chambolle and C. H. Dossal, *On the convergence of the iterates of" fista"*, Journal of Optimization Theory and Applications, 166 (2015), p. 25.

[13] A. Chambolle, M. J. Ehrhardt, P. Richtárik, and C.-B. Schönlieb, *Stochastic primal-dual hybrid gradient algorithm with arbitrary sampling and imaging applications*, SIAM Journal on Optimization, (2018), pp. 2783–2808, https://doi.org/10.1137/17M1134834.

[14] A. Chambolle and T. Pock, *A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging*, Journal of Mathematical Imaging and Vision, 40 (2011), pp. 120–145, https://doi.org/10.1007/s10851-010-0251-1.

[15] A. Chambolle and T. Pock, *An Introduction to Continuous Optimization for Imaging*, Acta Numerica, 25 (2016), pp. 161–319.

[16] L. Cheng, H. Wang, and Z. Zhang, *The solution of ill-conditioned symmetric toeplitz systems via two-grid and wavelet methods*, Computers & Mathematics with Applications, 46 (2003), pp. 793–804, https://doi.org/10.1016/S0898-1221(03)90142-8.

[17] G. Chierchia, N. Pustelnik, B. Pesquet-Popescu, and J.-C. Pesquet, *A Non-Local Structure Tensor Based Approach for Multicomponent Image Recovery Problems*, IEEE Transactions on Image Processing, 23 (2014), pp. 5531–5544, https://doi.org/10.1109/TIP.2014.2364141. arXiv:1403.5403 [cs, math].

[18] E. Chouzenoux, J. Idier, and S. Moussaoui, *A majorize–minimize strategy for subspace optimization applied to image restoration*, IEEE Transactions on Image Processing, 20 (2010), pp. 1517–1528.

[19] E. Chouzenoux, S. Martin, and J.-C. Pesquet, *A local mm subspace method for solving constrained variational problems in image recovery*, preprint, (2022).

[20] P. Combettes and V. Wajs, *Signal Recovery by Proximal Forward-Backward Splitting*, SIAM Multiscale Model. Simul., 4 (2005), pp. 1168–1200.

[21] P. L. Combettes and J.-C. Pesquet, *Proximal Splitting Methods in Signal Processing*, Springer New York, New York, NY, 2011, pp. 185–212, https://doi.org/10.1007/978-1-4419-9569-8_10.

[22] L. Condat, *A Primal–Dual Splitting Method for Convex Optimization Involving Lipschitzian, Proximable and Linear Composite Terms*, Journal of Optimization Theory and Applications, 158 (2013), pp. 460–479, https://doi.org/10.1007/s10957-012-0245-9.

[23] L. Condat, D. Kitahara, A. Contreras, and A. Hirabayashi, *Proximal Splitting Algorithms for Convex Optimization: A Tour of Recent Advances, with New Twists*, Dec. arXiv:1912.00137 [math].

[24] L.-J. Deng, T.-Z. Huang, and X.-L. Zhao, *Wavelet-based two-level methods for image restoration*, Communications in Nonlinear Science and Numerical Simulation, 17 (2012), pp. 5079–5087, https://doi.org/10.1016/j.cnsns.2012.04.001.

[25] M. Donatelli, *An algebraic generalization of local Fourier analysis for grid transfer operators in multi-*

grid based on Toeplitz matrices: AN ALGEBRAIC GENERALIZATION OF LFA, Numerical Linear Algebra with Applications, 17 (2010), pp. 179–197, https://doi.org/10.1002/nla.704.

[26] M. Donatelli, *An Iterative Multigrid Regularization Method for Toeplitz Discrete Ill-Posed Problems*, Numerical Mathematics: Theory, Methods and Applications, 5 (2012), pp. 43–61, https://doi.org/10.4208/nmtma.2011.m12si03.

[27] M. J. Ehrhardt, E. S. Riis, T. Ringholm, and C.-B. Schönlieb, *A geometric integration approach to smooth optimisation: Foundations of the discrete gradient method*, arXiv preprint arXiv:1805.06444, (2018).

[28] M. I. Espanol, *Multilevel methods for discrete ill-posed problems: Application to deblurring*, PhD thesis, 2009.

[29] M. I. Español and M. E. Kilmer, *Multilevel Approach For Signal Restoration Problems With Toeplitz Matrices*, SIAM Journal on Scientific Computing, 32 (2010), pp. 299–319, https://doi.org/10.1137/080715780.

[30] O. Fercoq and P. Richtárik, *Accelerated, parallel, and proximal coordinate descent*, SIAM Journal on Optimization, 25 (2015), pp. 1997–2023.

[31] S. W. Fung and Z. Wendy, *Multigrid Optimization for Large-Scale Ptychographic Phase Retrieval*, SIAM Journal on Imaging Sciences, 13 (2020), pp. 214–233, https://doi.org/10.1137/18M1223915.

[32] S. Gratton, A. Sartenaer, and P. L. Toint, *Recursive trust-region methods for multiscale nonlinear optimization*, SIAM Journal on Optimization, 19 (2008), pp. 414–444, https://doi.org/10.1137/050623012.

[33] S. Gratton, A. Sartenaer, and P. L. Toint, *Recursive Trust-Region Methods for Multiscale Nonlinear Optimization*, SIAM Journal on Optimization, 19 (2008), pp. 414–444, https://doi.org/10.1137/050623012.

[34] D. Greenfeld, M. Galun, R. Basri, I. Yavneh, and R. Kimmel, *Learning to Optimize Multigrid PDE Solvers*, in Proceedings of the 36th International Conference on Machine Learning, K. Chaudhuri and R. Salakhutdinov, eds., vol. 97 of Proceedings of Machine Learning Research, PMLR, June 2019, pp. 2415–2423.

[35] P. C. Hansen, J. G. Nagy, and D. P. O'Leary, *Deblurring Images*, Society for Industrial and Applied Mathematics, 2006, https://doi.org/10.1137/1.9780898718874.

[36] C. P. Ho, M. Kočvara, and P. Parpas, *Newton-type multilevel optimization method*, Optimization Methods and Software, (2019), pp. 1–34, https://doi.org/10.1080/10556788.2019.1700256.

[37] V. Hovhannisyan, P. Parpas, and S. Zafeiriou, *MAGMA: Multilevel Accelerated Gradient Mirror Descent Algorithm for Large-Scale Convex Composite Minimization*, SIAM Journal on Imaging Sciences, 9 (2016), pp. 1829–1857, https://doi.org/10.1137/15M104013X.

[38] A. Javaherian and S. Holman, *A Multi-Grid Iterative Method for Photoacoustic Tomography*, IEEE Transactions on Medical Imaging, (2017), pp. 696–706, https://doi.org/10.1109/TMI.2016.2625272.

[39] U. Kamilov, E. Bostan, and M. Unser, *Wavelet Shrinkage With Consistent Cycle Spinning Generalizes Total Variation Denoising*, IEEE Signal Processing Letters, 19 (2012), pp. 187–190, https://doi.org/10.1109/LSP.2012.2185929.

[40] G. Lauga, E. Riccietti, N. Pustelnik, and P. Gonçalves, *Multilevel Fista For Image Restoration*, IEEE ICASSP, Rhodes, Greece, (4-10 June 2023).

[41] G. Lauga, E. Riccietti, N. Pustelnik, and P. Gonçalves, *Méthodes proximales multi-niveaux pour la restauration d'images*, Nancy, France, Sept. 2022.

[42] H. T. V. Le, N. Pustelnik, and M. Foare, *The faster proximal algorithm, the better unfolded deep learning architecture? The study case of image denoising*, p. 6.

[43] T. D. Luu, J. Fadili, and C. Chesneau, *Sampling from non-smooth distribution through Langevin diffusion*, (2017), p. 27.

[44] P. Machart, S. Anthoine, and L. Baldassarre, *Optimal Computational Trade-Off of Inexact Proximal Methods*, Oct. arXiv:1210.5034 [cs].

[45] S. G. Nash, *A Multigrid Approach to Discretized Optimization Problems*, Optimization Methods and Software, 14 (2000), pp. 99–116, https://doi.org/10.1080/10556780008805795, https://arxiv.org/abs/https://doi.org/10.1080/10556780008805795.

[46] N. Parikh and S. Boyd, *Proximal Algorithms*, Foundations and Trends in Optimization, 1 (2014), p. 123–231.

[47] P. Parpas, *A Multilevel Proximal Gradient Algorithm for a Class of Composite Optimization Problems*, SIAM Journal on Scientific Computing, 39 (2017), pp. S681–S701, https://doi.org/10.1137/16M1082299.

[48] J. Plier, F. Savarino, M. Kočvara, and S. Petra, *First-Order Geometric Multilevel Optimization for Discrete Tomography*, in Scale Space and Variational Methods in Computer Vision, A. Elmoataz, J. Fadili, Y. Quéau, J. Rabin, and L. Simon, eds., vol. 12679, Springer International Publishing, Cham, 2021, pp. 191–203, https://doi.org/10.1007/978-3-030-75549-2_16. Series Title: Lecture Notes in Computer Science.

[49] E. Quemener and M. Corvellec, *SIDUS—the Solution for Extreme Deduplication of an Operating System*, Linux J., 2013 (2013).

[50] A. Salim, L. Condat, K. Mishchenko, and P. Richtárik, *Dualize, split, randomize: Fast nonsmooth optimization algorithms*, (2020).

[51] G. Steidl, J. Weickert, T. Brox, P. Mrázek, and M. Welk, *On the Equivalence of Soft Wavelet Shrinkage, Total Variation Diffusion, Total Variation Regularization, and SIDEs*, SIAM Journal on Numerical Analysis, 42 (2004), pp. 686–713, https://doi.org/10.1137/S0036142903422429.

[52] S. Villa, S. Salzo, L. Baldassarre, and A. Verri, *Accelerated and Inexact Forward-Backward Algorithms*, SIAM Journal on Optimization, 23 (2013), pp. 1607–1633, https://doi.org/10.1137/110844805.