A Variational Approach for Joint Image Recovery and Feature Extraction Based on Spatially Varying Generalised Gaussian Models

Émilie Chouzenoux[1], Marie-Caroline Corbineau[2], Jean-Christophe Pesquet[1], Gabriele Scrivanti[1*]

[1]Université Paris-Saclay, Inria, CentraleSupélec, CVN, France.
[2]Research department, Preligens,France.

*Corresponding author(s). E-mail(s): gabriele.scrivanti@centralesupelec.fr;

Abstract

The joint problem of reconstruction/feature extraction is a challenging task in image processing. It consists in performing, in a joint manner, the restoration of an image and the extraction of its features. In this work, we firstly propose a novel non-smooth and non-convex variational formulation of the problem. For this purpose, we introduce a versatile generalised Gaussian prior whose parameters, including its exponent, are space-variant. Secondly, we design an alternating proximal-based optimisation algorithm that efficiently exploits the structure of the proposed non-convex objective function. We also analyse the convergence of this algorithm. As shown in numerical experiments conducted on joint deblurring/segmentation tasks, the proposed method provides high-quality results.

Keywords: Image recovery ; Space-variant regularisation ; Alternating minimization ; Proximal algorithm ; Block coordinate descent ; Image segmentation

1 Introduction

Variational regularisation of ill-posed inverse problems in imaging relies on the idea of searching for a solution in a well-suited space. A central role in this context is played by ℓ_p spaces with $p \in (0, \infty)$, and the power p of the corresponding norms when $p \geq 1$ [1–5] or seminorms when $p \in (0, 1)$ [6–8]. For every vector $u = (u_i)_{1 \leq i \leq n} \in \mathbb{R}^n$ and $p \in (0, +\infty)$, the ℓ_p (semi-)norm is denoted by $\|u\|_p = \left(\sum_{i=1}^n |u_i|^p\right)^{1/p}$. We usually omit p when $p = 2$, so that $\|\cdot\| = \|\cdot\|_2$. The case $p \in (0, 1)$ has gained rising credit, especially in the field of sparse regularisation. An extensive literature has been focused on challenging numerical tasks raised by the non-convexity of the seminorms and the possibility of combining them with linear operators to extract salient features of the sought images [9, 10]. In [11], the more general notion of F-norm is introduced in order to establish functional analysis results on products of ℓ_{p_i}-spaces with $p_i \in (0, 2]$. For some $x = (x_i)_{1 \leq i \leq n} \in \mathbb{R}^n$, this amounts to studying the properties of penalties of the form $\sum_{i=1}^n |x_i|^{p_i}$, for some positive exponents $(p_i)_{1 \leq i \leq n}$. This approach offers a more flexible framework by considering a wider range of exponents than the standard ℓ_p-based regularisation. However, it extends the problem of choosing a suitable exponent p to a whole sequence of exponents $(p_i)_{1 \leq i \leq n}$. In [12],

the authors proposed a non-convex regulariser of the form $\sum_{i=1}^{n} |x_i|^{\varpi(|x_i|)}$, where each exponent is expressed as a function of the absolute magnitude of the data and function $\varpi(\cdot)$ is a rescaled version of the sigmoid function, taking values in the interval $[0, 1]$. In image restoration, a similar approach consists in adopting space variant regularisation models built around a Total Variation-like functional with a variable exponent $\sum_{i=1}^{n} \|(\nabla x)_i\|^{p_i}$ where ∇ is a discrete 2D gradient operator. The rationale is to select the set of parameters $(p_i)_{1\leq i\leq n}$ in order to promote either edge enhancement ($p_i = 1$) or smoothing ($p_i > 1$) depending on the spatial location encoded by index i. This model was introduced in [13] and then put into practice firstly for $p_i \in [1, 2]$ in [14] and then for $p_i \in (0, 2]$ in [15]. To conclude, in a recent work [16], the authors proposed a modular-proximal gradient algorithm to find solutions to ill-posed inverse problems in variable exponents Lebesgue spaces $L^{p(\cdot)}(\Omega)$ with $\Omega \subseteq \mathbb{R}^n$, rather than in $L^2(\Omega)$. In all of these works, the so-called space variant p-map (*i.e.* , $(p_i)_{1\leq i\leq n}$) is estimated offline in a preliminary step and then kept fixed throughout the optimisation procedure.

In this paper, we address the problem of joint image recovery and feature extraction. Image recovery amounts to retrieving an estimate of an original image from a degraded version of it. The degradation usually corresponds to the application of a linear operator (e.g., blur, projection matrix) to the image and the addition of a noise. Feature extraction problems arise when one wants to assign to an image a small set of parameters which can describe or identify the image itself. Image segmentation can be viewed as an example of feature extraction, which consists of defining a label field on the image domain so that pixels are partitioned into a predefined number of homogeneous regions according to some specific characteristics. A second example, similar to segmentation, is edge detection, where one aims at identifying the contour changes within different regions of the image. Texture retrieval is a third example. This task relies on the idea of assigning a set of parameters to each coefficient of the image – possibly in some transformed space – so that the combination of all parameters defines a "signature" that represents the content of various spatial regions. Joint image recovery and feature extraction consists in performing, in a joint manner, the image recovery and the extraction of features in the sought image.

A powerful and versatile approach for feature extraction, that we propose to adopt here, assumes that the data follow a mixture of generalised Gaussian probability distribution ($\mathcal{GGD}$) [17–19]. The $\mathcal{GGD}$ model results in a sum of weighted ℓ_{p_i}-based terms in the criterion, with general form $\sum_{i=1}^{n} \vartheta_i |x_i|^{p_i}$ with $\{\vartheta_i\}_{1\leq i\leq n} \subset [0, +\infty)$. We thus aim at jointly estimating an optimal configuration for $(\vartheta_i, p_i)_{1\leq i\leq n}$, and retrieving the image. Under an assumption of consistency within the exponents' values of a given region of the features space, we indeed obtain the desired feature extraction starting from the estimated p-map. The latter amounts to minimizing a non-smooth and non-convex cost function.

This specific structure of the proposed objective function suggests the use of an alternating minimisation procedure. In such an approach, one sequentially updates a subset of parameters through the resolution of an inner minimization problem, while the other parameters are assumed to be fixed. This approach has a standard form in the Block Coordinate Descent method (BCD) (also known as Gauss-Seidel algorithm) [20]. In the context of non-smooth and non-convex problems, the simple BCD may, however, show instabilities [21], which resulted in an extensive construction of alternative methods that efficiently exploit the characteristics of the functions, and introduce powerful tools to improve the convergence guarantees of BCD, or overcome difficulties arising in some formulations. In this respect, a central role is played by proximal methods [22, 23]: a proximally regularised BCD (PAM) for non-convex problems was studied in [24]; a proximal linearised method (PALM) and its inertial and stochastic versions were then proposed in [25] resp. [26] and [27]; in [28], the authors investigated the advantage of a hybrid semi-linearised scheme (SL-PAM) for the joint task of image restoration and edge detection based on a discrete version of the Mumford–Shah model. A structure-adapted version of PALM (ASAP) was designed in [29, 30] to exploit the block-convexity of the coupling terms and the regularity of the block-separable terms arising in some practical applications such as image colorisation and blind source separation. The extension to proximal mappings defined w.r.t. a variable

metric was firstly introduced in [31], leading to the so-called Block Coordinate Variable Metric Forward-Backward. An Inexact version and a line search based version of it were presented in [32] and [33], respectively. In [34] the authors introduced a Majorisation-Minimisation (MM) strategy within a Variable Metric Forward-Backward algorithm to tackle the challenging task of computing the proximity operator of composite functions. We refer to [35] for an in-depth analysis of how to introduce a variable metric into first-order methods. In [36], the authors introduced a family of block-coordinate majorisation-minimisation methods named TITAN. Various majorisation strategies can be encompassed by their framework, such as proximal surrogates, Lipschitz gradient surrogates, or Bregman surrogate functions. Convergence of the algorithm iterates are shown in [32, 36], under mild assumptions, that include the challenging non-convex setting. These studies emphasised the prominent role played by the Kurdyka-Łojasiewicz (KL) inequality [37].

In the proposed problem formulation, the objective function includes several non-smooth terms, as well as a quadratic term – hence Lipschitz differentiable – that is restricted to a single block of variables. This feature makes the related subproblem well-suited for a splitting procedure that involves an explicit gradient step with respect to this term, combined with implicit proximal steps on the remaining blocks of variables. Variable metrics within gradient/proximal steps would also be desirable for convergence speed purposes. As we will show, the TITAN framework from [36] allows building and analysing such an algorithm. Unfortunately, the theoretical convergence properties of TITAN assume exact proximal computations at each step, which cannot be ensured in practice in our context. To circumvent this, we thus propose and prove the convergence of an inexact version of a TITAN-based optimisation scheme. Inexact rules in the form of those studied in [32] are considered. We refer to the proposed method as to a Preconditioned Semi-Linearised Structure Adapted Proximal Alternating Minimisation (P–SASL–PAM) scheme. We investigate the convergence properties for this algorithm by relying on the KŁ property first considered in [37]. Under analytical assumptions on the objective function, we show the global convergence toward a critical point of any sequence generated by the proposed method. Then, we explicit the use of this method in our problem of image recovery and feature extraction. The performance of the approach is illustrated by means of examples in the field of image processing, in which we also show quantitative comparisons with state-of-the-art methods.

In a nutshell, the contributions of this work are (i) the proposition of an original non-convex variational model for the joint image recovery and feature extraction problem; (ii) the design of an inexact block coordinate descent algorithm to address the resulting minimisation problem; (iii) the convergence analysis of this scheme; (iv) the illustration of the performance of the proposed method through a numerical example in the field of ultrasound image processing.

The paper is organised as follows. In Section 2 we introduce the degradation model and report our derivation of the objective function for image recovery and feature extraction, starting from statistical assumptions on the data. In Section 3, we describe the proposed P-SASL-PAM method to address a general non-smooth non-convex optimization problem; secondly we show that the proposed method converges globally, in the sense that the whole generated sequence converges to a (local) minimum. The application of the P-SASL-PAM method to the joint reconstruction/segmentation problem is described in Section 4. Some illustrative numerical results are shown in Section 5. Conclusions are drawn in Section 6.

2 Model Formulation

In this section, we describe the construction of the objective function associated to the joint reconstruction/feature extraction problem. After defining the degradation model, we report the Bayesian model that is reminiscent from the one considered in [17, 19] in the context of ultrasound imaging. Then, we describe the procedure that leads us to the definition of our addressed optimization problem.

2.1 Observation Model

Let $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ be respectively the vectorised sought-for solution and the observed data, which are assumed to be related according to the following model

$$y = Kx + \omega, \tag{1}$$

where $K \in \mathbb{R}^{m \times n}$ is a linear operator, and $\omega \sim \mathcal{N}(0, \sigma^2 \mathbb{I}_m)$, *i.e.* the normal distribution with zero mean and covariance matrix $\sigma^2 \mathbb{I}_m$ with $\sigma > 0$ and $\mathbb{I}_m$ states for the $m \times m$ identity matrix. We further assume that x can be characterised by a finite set of k *features* that are defined in a suitable space, where the data are described by a simple model relying on a small number of parameters. The Generalised Gaussian Distribution ($\mathcal{GGD}$)

$$(\forall t \in \mathbb{R}) \quad \mathsf{p}(t; p, \alpha) = \frac{1}{2\alpha^{1/p}\Gamma\left(1 + \frac{1}{p}\right)} \exp\left(-\frac{|t|^p}{\alpha}\right) \tag{2}$$

with $(p, \alpha) \in (0, +\infty)^2$ has shown to be a suitable and flexible tool for this purpose [17–19]. Each feature can be identified by a pair (p_j, α_j) for $j \in \{1, \ldots, k\}$, where parameter p is proportional to the decay rate of the tail of the probability density function (PDF) and parameter α models the width of the peak of the PFD. Taking into account the role that p and α play in the definition of the PDF profile, these two parameters are generally referred to as *shape* and *scale* parameter.

Assuming that K and σ are known, the task we address in this work is to jointly retrieve x (reconstruction) and obtain a good representation of its features through an estimation of the underlying model parameters (p_j, α_j) for $j \in \{1, \ldots, k\}$ (feature extraction). Starting from a similar statistical model as the one considered in [17, 19], we infer a continuous variational framework which does not rely on the *a priori* knowledge of the exact number of features k. We derive this model by performing a *Maximum a Posteriori* estimation, which allows us to formulate the Joint Image Reconstruction and Feature Extraction task as a non-smooth and non-convex optimisation problem involving a coupling term and a block-coordinate separable one.

2.2 Bayesian Model

From (1), we derive the following likelihood

$$\mathsf{p}(y|x, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{\|y - Kx\|^2}{2\sigma^2}\right). \tag{3}$$

Assuming then that the components of x are independent conditionally to the knowledge of their feature class, the prior distribution of x is a mixture of $\mathcal{GGD}$s

$$\mathsf{p}(x|p, \alpha) = \prod_{j=1}^{k} \frac{1}{\left(2\alpha_j^{1/p_j}\Gamma\left(1 + \frac{1}{p_j}\right)\right)^{N_j}} \exp\left(-\frac{\|\overline{x}_j\|_{p_j}^{p_j}}{\alpha_j}\right). \tag{4}$$

Hereabove, for every $x \in \mathbb{R}^n$ and a feature labels set $j \in \{1, \ldots, k\}$, we define $\overline{x}_j \in \mathbb{R}^{N_j}$ as the vector containing only the N_j components of x that belong to the j-th feature. Following the discussion in [38], for the shape parameter, we choose a uniform distribution on a certain interval $[a, b] \subset [0, +\infty)$:

$$\mathsf{p}(p) = \prod_{j=1}^{k} \frac{1}{b - a} \mathsf{I}_{[a,b]}(p_j). \tag{5}$$

This choice stems from the fact that setting $a = 0$ and $b = 3$ allows covering all possible values of the shape parameter encountered in practical applications, but no additional information about this parameter is available. For the scale parameter, we adopt the Jeffreys distribution to reflect the lack of knowledge about this parameter:

$$\mathsf{p}(\alpha) = \prod_{j=1}^{k} \frac{1}{\alpha_j} \mathsf{I}_{[0,+\infty)}(\alpha_j). \tag{6}$$

Note that such kind of prior is often used for scale parameters [39]. Hereabove, I_S represents the characteristic function of some subset $S \subset \mathbb{R}$, which is equal to 1 over S, and 0 elsewhere.

2.3 Variational Model

In order to avoid to define *a priori* the number of features, we regularise the problem by considering the 2D Total Variation (TV) of the $\mathcal{GGD}$ parameters $(p, \alpha) \in (0, +\infty)^n \times (0, +\infty)^n$. The idea of using Total Variation to define a segmentation procedure is studied in [40–45] by virtue of the co-area formula: the authors propose to replace the boundary information term of the Mumford-Shah (MS) functional [46] with the TV convex integral term. This choice yields a non-tight convexification of the MS model that does not require setting the number of segments in advance. The overall segmentation procedure is then built upon two steps: the first one consists of obtaining a smooth version of the given image that is adapted to segmentation by minimising the proposed functional with convex methods; the second step consists of partitioning the obtained solution into the desired number of segments, by *e.g.* defining the thresholds with Otsu's method [47] or the k-means algorithm. The strength of our approach is that the second step (i.e. the actual segmentation step) is independent from the first one; hence it is possible to set the number of segments (*i.e.* , labels) without solving the optimisation problem again.

In the considered model, the introduction of a TV prior leads to a minimization problem that is non-convex with respect to α. To circumvent this issue, a possible strategy would involve applying the variable change $\beta = 1/\alpha$, which leads to a convex problem with regard to β. However, after performing some tests, we noticed that this choice tends to promote extreme values 0 or $+\infty$. We then opted for the following reparameterisation for the scale parameter: let $\beta = (\beta_i)_{1 \leq i \leq n} \in \mathbb{R}^n$ be such that, for every $i \in \{1, \dots, n\}$,

$$\beta_i = \frac{1}{p_i} \ln \alpha_i, \tag{7}$$

and let us choose for this new variable a non-informative Gaussian prior that is defined on the whole space of configurations with mean $\mu_\beta \geq 0$ and standard deviation $\sigma_\beta > 0$. The choice of a non-necessarily zero-mean distribution stems from the idea of having a more flexible prior to represent our reparameterised scale parameter.

Thus, replacing α with β and further introducing TV regularisation potentials (weighted by the regularisation parameters $\lambda > 0$ and $\zeta > 0$) leads to the following reformulation of distributions (4)-(6):

$$\begin{aligned} &\mathsf{p}(x|p, \beta) \\ &= \prod_{i=1}^{n} \frac{1}{2 \exp(\beta_i) \Gamma\left(1 + \frac{1}{p_i}\right)} \exp\left(-|x_i|^{p_i} \exp(-p_i \beta_i)\right) \end{aligned} \tag{8}$$

$$\mathsf{p}(p) = c_p \exp(-\lambda \mathrm{TV}(p)) \prod_{i=1}^{n} \frac{1}{b-a} \mathsf{I}_{[a,b]}(p_i) \tag{9}$$

$$\begin{aligned} &\mathsf{p}(\beta) \\ &= c_\beta \exp(-\zeta \mathrm{TV}(\beta)) \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_\beta} \exp\Big(-\frac{(\beta_i - \mu_\beta)^2}{2\sigma_\beta^2}\Big). \end{aligned} \tag{10}$$

where $(c_p, c_\beta) \in (0, +\infty)^2$ are normalisation constants. In Figure 1, we depict the probabilistic dependence graph defining the relations between variables and hyperparameters in our model.

The joint posterior distribution is determined as follows:

$$\begin{aligned} \mathsf{p}(x, p, \beta|y) &\propto \mathsf{p}(y|x, p, \beta)\mathsf{p}(x, p, \beta) \\ &\propto \mathsf{p}(y|x, p, \beta)\mathsf{p}(x|p, \beta)\mathsf{p}(p)\mathsf{p}(\beta). \end{aligned} \tag{11}$$

Let us take the negative logarithm of (11), then computing the Maximum a Posteriori estimates (*i.e.*, maximising the joint posterior distribution) is equivalent to the following optimization problem, which we refer to as the joint image reconstruction and feature extraction problem

$$\begin{aligned} &\underset{(x,p,\beta) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n}{\text{minimize}} \quad \Theta(x, p, \beta) = \frac{1}{2\sigma^2} \|y - Kx\|^2 \\ &+ \sum_{i=1}^{n} \Big(|x_i|^{p_i} e^{-p_i \beta_i} + \ln \Gamma(1 + \frac{1}{p_i}) + \iota_{[a,b]}(p_i) + \beta_i \\ &\qquad + \frac{(\beta_i - \mu_\beta)^2}{2\sigma_\beta^2} \Big) + \lambda \mathrm{TV}(p) + \zeta \mathrm{TV}(\beta). \end{aligned} \tag{12}$$

Hereabove, ι_S represents the indicator function of some subset $S \subset \mathbb{R}$, which is equal to 0 over S,

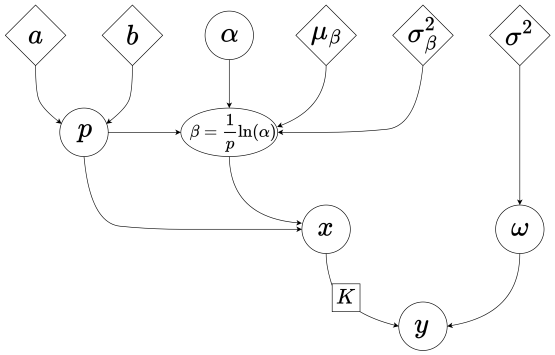


Fig. 1: Probabilistic dependence graph of our model. Hyperparameters are represented as diamonds, and variables as ellipses: a and b are the lower and the upper bound for the interval appearing in the uniform distribution of the shape parameter, p is the shape parameter, α is the original scale parameter, β is the reparameterised scale parameter with mean μ_β and standard deviation σ_β, x is the sought signal, y is the observed one, K is the linear operator, and ω is the additive Gaussian noise with standard deviation σ.

and $+\infty$ elsewhere.

In [28], the authors proposed a generalised discrete Mumford-Shah variational model that is specifically designed for the joint image reconstruction and edge detection problem. In contrast, the model we propose in (12) is well suited to encompass a wider class of problems. In Section 5, we present two applications, namely in the context of wavelet-based image restoration and in the context of joint deblurring/segmentation of ultrasound images. In particular, we notice that when restricted to variable x for a given set of parameters (p, β), the formulation (12) boils down to the flexible sparse regularisation model

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2\sigma^2}\|y - Kx\|^2 + \sum_{i=1}^{n} |x_i|^{p_i} e^{-p_i\beta_i}, \quad (13)$$

where the contribution of the ℓ_{p_i} regularisation term is itself weighted in a space varying fashion.

Function Θ in (12) is non-smooth and non-convex. It reads as the sum of a coupling term and three block-separable terms. In particular, the block-separable data-fit term relative to x is quadratic and hence has a Lipschitz continuous gradient. Our proposed algorithm aims to leverage this property, which is generally not accounted for by other BCD methods. To this aim, we exploit a hybrid scheme that involves both standard and linearised proximal steps. The details about the proposed method are presented in the next section.

3 Preconditioned Structure Adapted Semi-Linearised Proximal Alternating Minimisation (P-SASL-PAM)

In this section, we introduce a BCD-based method to address a class of sophisticated optimization problems that includes (12) as a special case. We start the section by useful preliminaries about subdifferential calculus. Then, we present the *Kurdyka-Łojasiewicz property*, which plays a prominent role in the convergence analysis of BCD methods in a non-convex setting. Finally, we define problem (20), itself generalising (12), for which we derive our proposed BCD-based algorithm and show its convergence properties. The so-called Preconditioned Structure Adapted Semi-Linearised Proximal Alternating Minimisation (P-SASL-PAM) approach mixes both standard and preconditioned linearised proximal regularisation on the different coordinate blocks of the criterion.

3.1 Subdifferential Calculus

Let us now recall some definitions and elements of subdifferential calculus that will be useful in the upcoming sections. For a proper and lower semicontinuous function $h : \mathbb{R}^n \to (-\infty, \infty]$, the domain of h is defined as

$$\operatorname{dom} h \ = \{u \in \mathbb{R}^n \mid h(u) < +\infty\}.$$

Firstly, we recall the notion of subgradients and subdifferential for convex functions.

Definition 1 (Subgradient of a convex function)**.** *Let $h : \mathbb{R}^n \to (-\infty, \infty]$ be a proper convex lower semicontinuous function. The subdifferential*

$\partial h(u^+)$ of h at $u^+ \in \mathbb{R}^n$ is the set of all vectors $r \in \mathbb{R}^n$, called subgradients of h at u^+, such that

$$\forall u \in \mathbb{R}^n \quad h(u) \geq h(u^+) + \langle r, u - u^+ \rangle.$$

If $u^+ \notin \operatorname{dom} h$, then $\partial h(u^+) = \varnothing$.

Secondly, we consider the more general notion of (limiting)-subdifferential for non-necessarily convex functions, as proposed in [48, Definition 8.3].

Definition 2 (Limiting Subdifferential)**.** *Let $h : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper and lower semicontinuous function. For a vector $u^+ \in \mathbb{R}^n$,*

- *the Fréchet subdifferential of h at u^+, written as $\hat{\partial} h(u^+)$, is the set of all vectors $r \in \mathbb{R}^n$ such that*

$$h(u) \geq h(u^+) + \langle r, u - u^+ \rangle + o(\|u - u^+\|);$$

  if $u^+ \notin \operatorname{dom} h$, then $\hat{\partial} h(u^+) = \varnothing$;
- *the limiting-subdifferential of h at u^+, denoted by $\partial h(u^+)$, is defined as*

$$\partial h(u^+) = \{r \in \mathbb{R}^n \mid \exists\, u^k \to u^+, \\ h(u^k) \to h(u^+), r^k \to r, r^k \in \hat{\partial} h(u^k)\}.$$

If h is lower semicontinuous and convex, then the three previous notions of subdifferentiality are equivalent, *i.e.* $\hat{\partial} h(u^+) = \partial h(u^+)$. If h is differentiable, then $\partial h(u^+) = \{\nabla h(u^+)\}$. Now, it is possible to formalise the notion of critical points for a general function:

Definition 3 (Critical point)**.** *Let $h : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper function. A point $u^* \in \mathbb{R}^n$ is said to be a critical (or stationary) point for h if $0 \in \partial h(u^*)$.*

Eventually, we define the notion of proximal maps relative to the norm induced by a positive definite matrix.

Definition 4. *Let $\mathcal{S}_n$ be the set of symmetric and positive definite matrices in $\mathbb{R}^{n \times n}$. For a matrix $A \in \mathcal{S}_n$, the weighted ℓ_2-norm induced by A is defined as*

$$(\forall u \in \mathbb{R}^n) \quad \|u\|_A = (u^\top A u)^{1/2}. \tag{14}$$

Definition 5. *Let $h : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper and lower semicontinuous function, let $A \in \mathcal{S}_n$ and $u^+ \in \mathbb{R}^n$. The proximity operator of function h at u^+ with respect to the norm induced by A is defined as*

$$\operatorname{prox}_h^A(u^+) = \operatorname{argmin}_{u \in \mathbb{R}^n} \left(\frac{1}{2}\|u - u^+\|_A^2 + h(u) \right). \tag{15}$$

Note that $\operatorname{prox}_h^A(u^+)$, as defined above, can be the empty set. It is nonempty for every $u^+ \in \mathbb{R}^n$, if h is lower-bounded by an affine function. In addition, it reduces to a single-valued operator when h is convex.

In order to deal with the situation when no closed-form proximal formulas are available (as it might be the case for non-trivial preconditioning metrics A), we take into account an inexact notion of proximal computation in the sense of [37, Theorems 4.2 and 5.2] and [32]:

Definition 6. *Let $h : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper and lower semicontinuous function, let $A \in \mathcal{S}_n$, $\tau > 0$ and $u^+ \in \mathbb{R}^n$. Then $u^* \in \mathbb{R}^n$ is an inexact proximal point for h at u^+ if the following* relative error *conditions are satisfied:*

(i) Sufficient Decrease Condition:

$$h(u^*) + \frac{1}{2}\|u^+ - u^*\|_A^2 \leq h(u^+) \tag{16}$$

(ii) Inexact Optimality*: there exists $r \in \partial h(u^*)$ such that*

$$\|r\| \leq \tau \|u^+ - u^*\|. \tag{17}$$

In this case we write that $u^* \in \operatorname{prox}_h^{A,\tau}(u^+)$.

Remark 1. *We highlight that when exact proximal points are considered, the optimality condition reads as*

$$0 \in \partial h(u^*) + A(u^+ - u^*) \tag{18}$$

implying that there exists $r \in \partial h(u^*)$ such that $r = A(u^* - u^+)$.

3.2 The KŁ-Property

Most of the works related to BCD-based algorithms rely on the framework developed by Attouch, Bolte, and Svaiter in their seminal paper [37] in order to prove the convergence of block alternating strategies for non-smooth and non-convex problems. A fundamental assumption in [37] is that the objective function satisfies the *Kurdyka-Łojasiewicz* (KŁ) property [49–51]. We recall the definition of this property as it was

given in [25]. Let $\eta \in (0, +\infty]$ and denote by Φ_η the class of concave continuous functions $\varphi : [0, +\infty) \to [0, +\infty)$ satisfying the following conditions:

(i) $\varphi(0) = 0$;
(ii) φ is $\mathcal{C}^1$ on $(0, \eta)$ and continuous at 0;
(iii) for every $s \in (0, \eta)$, $\varphi'(s) > 0$.

For any subset $S \subset \mathbb{R}^n$ and any point $u^+ \in \mathbb{R}^n$, the distance from u^+ to S is defined by

$$\operatorname{dist}(u^+, S) = \inf_{u \in S} \|u^+ - u\|$$

with $\operatorname{dist}(u^+, \varnothing) = +\infty$.

Definition 7 (KŁ property)**.** *Let $h : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper and lower semicontinuous function.*

(i) Function h is said to satisfy the Kurdyka-Łojasiewicz property at $u^+ \in \operatorname{dom} \partial h$ if there exist $\eta \in (0, +\infty]$, a neighbourhood U of u^+ and a function $\varphi \in \Phi_\eta$ such that, for every $u \in U$,

$$\begin{aligned} &h(u^+) < h(u) < h(u^+) + \eta \\ &\Rightarrow \varphi'(h(u) - h(u^+)) \operatorname{dist}(0, \partial h(u)) \geq 1. \end{aligned} \tag{19}$$

(ii) Function h is said to be a KŁ function if it satisfies the KŁ property at each point of $\operatorname{dom} \partial h$.

3.3 Proposed Algorithm

Let us consider that every element $\zeta \in \mathbb{R}^N$ is block-decomposed as $\zeta = (\zeta_0, \ldots, \zeta_d)$, with, for every $i \in \{0, \ldots, d\}$, $\zeta_i \in \mathbb{R}^{n_i}$, with $\sum_{i=0}^d n_i = N$. As we will show in Subsection 4.1, Problem (12) is a special instance of the general class of problems of the form

$$\underset{\zeta \in \mathbb{R}^N}{\text{minimize}} \;\; \theta(\zeta) = q(\zeta) + f(\zeta_0) + \sum_{i=1}^{d} g_i(\zeta_i), \tag{20}$$

under the following assumption:

Assumption 1.

1. *Function $q : \mathbb{R}^N \to \mathbb{R}$ is bounded from below and differentiable with Lipschitz continuous gradient on bounded subsets of $\mathbb{R}^N$.*
2. *Function $f : \mathbb{R}^{n_0} \to \mathbb{R}$ is differentiable with globally Lipschitz continuous gradient of constant $L_f > 0$, and is bounded from below.*
3. *For every $i \in \{1, \ldots, d\}$, function $g_i : \mathbb{R}^{n_i} \to (-\infty, +\infty]$ is proper, lower semicontinuous and bounded from below and the restriction to its domain is continuous.*
4. *θ is a KŁ function.*

Remark 2. *The assumption of continuity in Assumption 1.3 is standard in the context of inexact minimisation algorithm (see the assumptions in [37, Theorem 4.1, Theorem 5.2]).*

Throughout the paper we will use the following notation: for every $(\zeta_{i'})_{1 \leq i' \leq d} \in \mathbb{R}^{n_1} \times \cdots \mathbb{R}^{n_d}$ and $i \in \{0, \ldots, d\}$, $\zeta_{\neq i} = (\zeta_0, \ldots, \zeta_{i-1}, \zeta_{i+1}, \ldots, \zeta_d)$ and

$$\begin{aligned} (\forall z \in \mathbb{R}^{n_i}) \quad &(z; \zeta_{\neq i}) \\ &= (\zeta_0, \ldots, \zeta_{i-1}, z, \zeta_{i+1}, \ldots, \zeta_d). \end{aligned} \tag{21}$$

In order to proceed with the algorithm construction and analysis, let us recall the notion of partial subdifferentiation for a function $\theta : \mathbb{R}^N \longrightarrow \mathbb{R}$ as the one in (20). For every $i \in \{0, \ldots, d\}$ given a fixed $\zeta_{\neq i}$, the subdifferential of the partial function $\theta(\cdot\,; \zeta_{\neq i})$ with respect to the i-th block, is denoted as $\partial_i \theta(\cdot\,; \zeta_{\neq i})$. Given these definitions, we have the following differential calculus property (see [48, Exercises 8.8(c), Proposition 10.5]:

Proposition 1. *Let function θ be defined as in (20). Under Assumption 1, the following equality holds: for every $\zeta \in \mathbb{R}^N$,*

$$\begin{aligned} &\partial \theta(\zeta) \\ &= \{\nabla_0 q(\zeta) + \nabla f(\zeta_0)\} \times \bigtimes_{i=1}^d (\nabla_i q(\zeta) + \partial g_i(\zeta_i)) \\ &= \bigtimes_{i=0}^d \partial_i \theta(\zeta). \end{aligned} \tag{22}$$

We are now ready to introduce our block alternating algorithm P-SASL-PAM, outlined in Algorithm 1, to solve problem (20). Throughout the paper, we use the following notation: for every

$\ell \in \mathbb{N}$ and and for $i \in \{1, \dots, d\}$,

$$\begin{aligned} \zeta^{\ell+1,0} &= \zeta^{\ell}; \\ \zeta^{\ell+1,i} &= (\zeta_0^{\ell+1}, \dots, \zeta_{i-1}^{\ell+1}, \zeta_i^{\ell}, \zeta_{i+1}^{\ell}, \dots, \zeta_d^{\ell}) \\ \zeta^{\ell+1,d+1} &= \zeta^{\ell+1}. \end{aligned}$$

Algorithm 1 P-SASL-PAM to solve (20).

Initialize $\zeta_0^0 \in \operatorname{dom} f$, $\zeta_i^0 \in \operatorname{dom} g_i$ for $i \in \{1, \dots, d\}$
Set $(A_\ell)_{\ell \in \mathbb{N}} \in \mathcal{S}_{n_0}$ for every $\ell \in \mathbb{N}$
Set $\gamma_0 \in (0, 1)$ and $\gamma_i > 0$ and $\tau_i > 0$ for $i \in \{1, \dots, d\}$ and $\tau_i > 0$ for $i \in \{0, \dots, d\}$
For $\ell = 0, 1, \dots$ **until** convergence

$$\zeta_0^{\ell+1} \in \operatorname{prox}^{A_\ell, \tau_0}_{\gamma_0 q(\cdot\,; \zeta_{\neq 0}^{\ell})}\big(\zeta_0^{\ell} - \gamma_0 A_\ell^{-1} \nabla f(\zeta_0^{\ell})\big) \tag{23}$$

For $i = 1, \dots, d$

$$\zeta_i^{\ell+1} \in \operatorname{prox}^{\tau_i}_{\gamma_i \theta(\cdot\,; \zeta_{\neq i}^{\ell+1,i})}(\zeta_i^{\ell}) \tag{24}$$

end
end

The proposed method sequentially updates one of the coordinate blocks $(\zeta_0, \dots, \zeta_d)$ involved in function θ, through proximal and gradient steps. Our algorithm P-SASL-PAM, summarised in Algorithm 1, mixes both standard and linearised proximal regularisation on the coordinate blocks as in SLPAM [28], while inverting the splitting in order to gain more efficient proximal computations as in ASAP [29, 30]. On the one hand, the lack of global Lipschitz-continuity of ∇q prevents us from adopting BCVMFB [32]. On the other hand, the lack of differentiability for the whole set of block-separable functions prevents us from adopting ASAP [29, 30]. Our approach takes full advantage of the Lipschitz differentiability assumption on f to perform a linearised step for the update of variable ζ_0, while the remaining ζ_i's are updated sequentially, according to a standard proximal step. In addition, in order to accelerate the convergence, a preconditioned version of the linearised step is used, which relies on the MM-based variable metric forward-backward strategy introduced in [52]. The latter relies on the following technical assumptions:

Assumption 2. *We choose a sequence of SPD matrices $(A_\ell)_{\ell \in \mathbb{N}}$ in such a way that there exists $(\underline{\nu}, \overline{\nu}) \in (0, +\infty)^2$ such that, for every $\ell \in \mathbb{N}$,*

$$\underline{\nu} \mathbb{I}_{n_0} \preceq A_\ell \preceq \overline{\nu} \mathbb{I}_{n_0}. \tag{25}$$

Assumption 3. *The quadratic function defined, for every $\zeta_0^+ \in \mathbb{R}^{n_0}$ and every SPD matrix $A \in \mathcal{S}_n$ satisfying Assumption 2, as*

$$\begin{aligned} &(\forall \zeta_0 \in \mathbb{R}^{n_0}) \quad \phi(\zeta_0, \zeta_0^+) \\ &= f(\zeta_0^+) + \langle \zeta_0 - \zeta_0^+, \nabla f(\zeta_0^+) \rangle + \frac{1}{2} \|\zeta_0^+ - \zeta_0\|_A^2 \end{aligned} \tag{26}$$

is a majorant function of f at ζ_0^+, i.e.

$$(\forall \zeta_0 \in \mathbb{R}^{n_0}) \quad f(\zeta_0) \leq \phi(\zeta_0, \zeta_0^+). \tag{27}$$

Remark 3. *Since f satisfies Assumption 1, the* Descent Lemma *[53, Proposition A.24] applies, yielding*

$$\begin{aligned} &(\forall (\zeta_0, \zeta_0^+) \in \mathbb{R}^{n_0} \times \mathbb{R}^{n_0}) \quad f(\zeta_0) \\ &\leq f(\zeta_0^+) + \langle \zeta_0 - \zeta_0^+, \nabla f(\zeta_0^+) \rangle + \frac{L_f}{2} \|\zeta_0^+ - \zeta_0\|^2. \end{aligned}$$

This guarantees that the preconditioning matrix $A = L_f \mathbb{I}_{n_0}$ satisfies Assumption 2, with $\underline{\nu} = \overline{\nu} = L_f$. Apart from this simple choice for matrix A, more sophisticated construction strategies have been studied in the literature [52, 54, 55]. Practical choices of metrics for Problem (12) will be discussed in Section 5, which is dedicated to numerical experiments.

Remark 4. *Alternative approaches to deal with the lack of global Lipschitz continuity of ∇q could involve a backtracking strategy as in [56] or adaptive step sizes based on an estimate of the local smoothness of the function as in [57, 58].*

Remark 5. ***Esentially Cyclic update rule*** *Even though Algorithm 1 relies on a sequential update rule for the blocks of coordinates $i \in \{0, \dots, d\}$, an extension to a quasi-cyclic rule with interval $\overline{d} \geq d$ is possible. In this case, at each iteration, the index $i \in \{0, \dots, d\}$ of the updated block is randomly chosen in such a way that each of the d blocks is updated at least once*

every $\overline{d}$ steps.

P-SASL-PAM involves the computation of three proximal operators, at each iteration $\ell \in \mathbb{N}$. As we will show in Subsection 3.3.1, if these operators are exactly computed, P-SASL-PAM fits within the general algorithmic framework TITAN [36], and, as such, inherits its convergence properties. The links between the exact and the inexact form of Algorithm 1 is discussed in Subsection 3.3.2. The convergence of the inexact form of Algorithm 1 is shown in Subsection 3.4.

3.3.1 Links between P-SASL-PAM and TITAN

Let us show that the exact version of Algorithm 1 is a special instance of the TITAN algorithm from [36]. The scheme of TITAN relies on an MM strategy that, at each iteration, for each block of coordinates, minimizes a block surrogate function, *i.e.* a majorizing approximation for the restriction of the objective function to this block. Let us define formally the notion of block surrogate function in the case of Problem (20).

Definition 8 (Block surrogate function)**.** *Consider a function $h : \mathbb{R}^N \longrightarrow \mathbb{R}$. For every $i \in \{0, \dots, d\}$, function $h_i : \mathbb{R}^{n_i} \times \mathbb{R}^N \to \mathbb{R}$ is called a* block surrogate function *of h at block i if $(\zeta_i, \xi) \mapsto h_i(\zeta_i; \xi)$ is continuous in ξ, lower-semicontinuous in ζ_i and the following conditions are satisfied:*

(i) $h_i(\xi_i; \xi) = h(\xi)$ for every $\xi \in \mathbb{R}^N$

(ii) $h_i(\zeta_i; \xi) \geq h(\zeta_i; \xi_{\neq i})$ for all $\zeta_i \in \mathbb{R}^{n_i}$ and $\xi \in \mathbb{R}^N$

Function $h_i(\cdot; \xi)$ is said to be a block surrogate function of h at block i in ξ. The block approximation error *for block i at a point $(\zeta_i, \xi) \in \mathbb{R}^{n_i} \times \mathbb{R}^N$ is then defined for every $i \in \{0, \dots, d\}$ as*

$$\mathsf{e}_i(\zeta_i; \xi) := h_i(\zeta_i; \xi) - h(\zeta_i; \xi_{\neq i}).$$

Let us now show that each of the steps of Algorithm 1 is actually equivalent to minimising an objective function involving a block surrogate function of the differentiable terms in θ for block $i \in \{0, \dots, d\}$ at the current iterate.

Solving (23) in Algorithm 1 is equivalent to solving

$$\operatorname*{argmin}_{\zeta_0 \in \mathbb{R}^{n_0}} \; h_0(\zeta_0; \zeta^\ell) \tag{28}$$

where

$$(\forall \zeta_0 \in \mathbb{R}^{n_0}) \quad h_0(\zeta_0; \zeta^\ell) =$$
$$q(\zeta_0; \zeta^\ell_{\neq 0}) + f(\zeta_0^\ell) + \langle \nabla f(\zeta_0^\ell), \zeta_0 - \zeta_0^\ell \rangle + \frac{1}{2\gamma_1} \|\zeta_0^\ell - \zeta_0\|^2_{A_\ell},$$

is a surrogate function of $(q(\cdot\,; \zeta^\ell_{\neq 0}) + f$ by virtue of Assumption 3. Notice that function h_0 is continuous on the block $(\zeta_0; \zeta) \in \mathbb{R}^{n_0} \times \mathbb{R}^N$.

Solving (24) for a certain block index $i \in \{1, \dots, d\}$ in Algorithm 1 is equivalent to solving

$$\operatorname*{argmin}_{\zeta_i \in \mathbb{R}^{n_i}} \; h_i(\zeta_i; \zeta^{\ell+1,i}) + g_i(\zeta_i) \tag{29}$$

where the function

$$(\forall \zeta_i \in \mathbb{R}^{n_i}) \qquad h_i(\zeta_i; \zeta^{\ell+1,i}) =$$
$$q(\zeta_i; \zeta^{\ell+1,i}_{\neq i}) + \frac{1}{2\gamma_i} \|\zeta_i - \zeta_i^\ell\|^2$$

is a proximal surrogate of function $q(\cdot\,, \zeta^{\ell+1}_{\neq i})$ at its i-th block in $\zeta^{\ell+1,i}$. Note that function h_i is continuous on $\mathbb{R}^N$.

In a nutshell, Algorithm 1 alternates between minimization of problems involving block surrogates for the differentiable terms of function θ, and, as such, can be viewed as a special instance of TITAN [36]. This allows us to state the following convergence result for a sequence generated by Algorithm 1.

Theorem 2. *Let Assumptions 1-3 be satisfied. Assume also that the sequence $(\zeta^\ell)_{\ell \in \mathbb{N}}$ generated by Algorithm 1 is bounded. Then,*

i) $\sum_{\ell=0}^{+\infty} \|\zeta^{\ell+1} - \zeta^\ell\| < +\infty$;

ii) $(\zeta^\ell)_{\ell \in \mathbb{N}}$ converges to a critical point ζ^* for function θ in (20).

Proof. We start the proof by identifying the three block approximation errors for the block surrogate functions at an iteration $\ell \in \mathbb{N}$:

$$(\forall \zeta_0 \in \mathbb{R}^{n_0}) \quad \mathsf{e}_0(\zeta_0; \zeta^\ell)$$
$$= f(\zeta_0^\ell) + \langle \nabla f(\zeta_0^\ell), \zeta_0 - \zeta_0^\ell \rangle + \frac{1}{2\gamma_0} \|\zeta_0^\ell - \zeta_0\|^2_{A_\ell} - f(\zeta_0)$$

and for $i \in \{1, \dots, d\}$

$$(\forall \zeta_i \in \mathbb{R}^{n_i}) \quad \mathsf{e}_i(\zeta_i; \zeta^{\ell+1,i}) = \frac{1}{2\gamma_i}\|\zeta_i - \zeta_i^\ell\|^2.$$

Clearly $\mathsf{e}_0(\cdot\,; \zeta^\ell)$ (resp. $\mathsf{e}_i(\cdot\,; \zeta^{\ell+1,i})$ for $i \in \{1, \dots, d\}$) are differentiable at ζ_0^ℓ (resp. at ζ_i^ℓ for $i \in \{1, \dots, d\}$) and the following holds for every $i \in \{0, \dots, d\}$:

$$\mathsf{e}_i(\zeta_i^\ell; \zeta^{\ell+1,i}) = 0, \quad \nabla_i \mathsf{e}_i(\zeta_i^\ell; \zeta^{\ell+1,i}) = 0.$$

This shows that [36, Assumption 2] is satisfied.

From (23) and Assumptions 2-3, we deduce that

$$q(\zeta^{\ell+1,1}) + f(\zeta_0^{\ell+1}) + \frac{1}{2}\left(\frac{1}{\gamma_0} - 1\right)\underline{\nu}\|\zeta_0^{\ell+1} - \zeta_0^\ell\|^2 \leq q(\zeta^\ell) + f(\zeta_1^\ell)$$

which implies that the Nearly Sufficient Descending Property [36, (NSDP)] is satisfied for the first block of coordinate with constant $\frac{1}{2}\left(\frac{1}{\gamma_0} - 1\right)\underline{\nu}$. On the other hand, for every $i \in \{1, \dots, d\}$, function $\mathsf{e}_i(\cdot\,; \zeta^{\ell+1,i})$ satisfies [36, Condition 2], which implies that [36, (NSDP)] also holds for i-the block of coordinates with the corresponding constant $1/\gamma_i$.

Moreover, [36, Condition 4.(ii)] is satisfied by Algorithm 1 with

$$\bar{l} = \max\left\{\frac{\overline{\nu}}{2}\left(\frac{1}{\gamma_0} - 1\right), \frac{1}{\gamma_1}, \dots, \frac{1}{\gamma_d}\right\}$$

and this constant fulfill the requirements of [36, Theorem 8]. In addition, by virtue of Proposition 1, [36, Assumption 3.(i)] holds, while the requirement in [36, Assumption 3.(ii)] is guaranteed by the fact that all the block surrogate functions are continuously differentiable.

Finally, since Algorithm 1 does not include any extrapolation step, we do not need to verify [36, Condition 1], whereas [36, Condition 4.(i)] is always satisfied.

In conclusion, we proved that all the requirements of [36, Proposition 5, Theorems 6 and 8] are satisfied. [36, Proposition 5] guarantees that the sequence has the finite-length property as expressed by i),while [36, Theorems 6 and 8] state that the sequence converges to a critical point ζ^* of (20), which concludes the proof. □

3.3.2 Well-definition of Algorithm 1

Now, we show that the inexact updates involved in Algorithm 1 are well-defined. To do so, we prove that the P-SASL-PAM algorithm with exact proximal computations can be recovered as a special case of Algorithm 1.

By the variational definition of proximal operator, for every $\ell \in \mathbb{N}$, the iterates of Algorithm 1 satisfy, for every $i \in \{1, \dots, d\}$,

$$\zeta_0^{\ell+1} \in \operatorname*{argmin}_{u_0 \in \mathbb{R}^{n_0}} \Big\{ q(u_0\,; \zeta_{\neq 0}^\ell) + \langle \nabla f(\zeta_0^\ell), u_0 - \zeta_0^\ell \rangle + \frac{1}{2\gamma_0}\|u_0 - \zeta_0^\ell\|_{A_\ell}^2 \Big\} \tag{30}$$

$$\zeta_i^{\ell+1} \in \operatorname*{argmin}_{u_i \in \mathbb{R}^{n_i}} \Big\{ q(u_i\,; \zeta_{\neq i}^{\ell+1,i}) + g_i(u_i) + \frac{1}{2\gamma_i}\|u_i - \zeta_i^\ell\|^2 \Big\} \tag{31}$$

so that

$$q(\zeta^{\ell+1,1}) + \langle \nabla f(\zeta_0^\ell), \zeta_0^{\ell+1} - \zeta_0^\ell \rangle + \frac{1}{2\gamma_0}\|\zeta_0^{\ell+1} - \zeta_0^\ell\|_{A_\ell}^2 \leq q(\zeta^\ell) \tag{32}$$

$$q(\zeta^{\ell+1,i+1}) + g(\zeta_i^{\ell+1}) + \frac{1}{2\gamma_i}\|\zeta_i^{\ell+1} - \zeta_i^\ell\|^2 \leq q(\zeta^{\ell+1,i}) + g(\zeta_i^\ell), \tag{33}$$

which implies that the sufficient decrease condition (16) is satisfied for every $i \in \{0, \dots, d\}$.

The use of the Fermat's rule implies that, for every $\ell \in \mathbb{N}$, the iterates of P-SASL-PAM are such that for every $i \in \{1, \dots, d\}$ there exists $r_i \in \partial g_i(\zeta_i^{\ell+1})$ for which the following equalities are satisfied for $i \in \{1, \dots, d\}$:

$$0 = \nabla f(\zeta_0^\ell) + \nabla_0 q(\zeta^{\ell+1,1}) + \gamma_0^{-1} A_\ell(\zeta_0^{\ell+1} - \zeta_0^\ell) \tag{34}$$

$$0 = r_i + \nabla_i q(\zeta^{\ell+1,i+1}) + \gamma_i^{-1}(\zeta_i^{\ell+1} - \zeta_i^\ell) \tag{35}$$

Hence, for every $i \in \{1, \dots, d\}$

$$\|\nabla f(\zeta_0^\ell) + \nabla_0 q(\zeta^{\ell+1,1})\| \leq \gamma_0^{-1}\overline{\nu}\|\zeta_0^{\ell+1} - \zeta_0^\ell\| \tag{36}$$

$$\|r_i + \nabla_i q(\zeta^{\ell+1,i+1})\| = \gamma_i^{-1}\|\zeta_i^{\ell+1} - \zeta_i^\ell\| \tag{37}$$

which implies that the inexact optimality condition (17) is satisfied with $\tau_0 = \gamma_0^{-1}\overline{\nu}$ for the first block of coordinates and $\tau_i = \gamma_i^{-1}$ for the remaining ones. In a nutshell, Algorithm 1 is well defined, as its inexact rules hold assuming exact computation of the proximity operators, which leads to TITAN.

3.4 Convergence analysis in the inexact case

Let us now present our main result, that is the convergence analysis for Algorithm 1.

Lemma 1. *Let $(\zeta^\ell)_{\ell\in\mathbb{N}}$ be the sequence generated by Algorithm 1. Then, under Assumptions 1 and 2*

i) there exists $\mu \in (0, +\infty)$ such that for every $\ell \in \mathbb{N}$,

$$\theta(\zeta^{\ell+1}) \leq \theta(\zeta^\ell) - \frac{\mu}{2}\|\zeta^{\ell+1} - \zeta^\ell\|^2. \tag{38}$$

ii) $\sum_{\ell=0}^{+\infty} \|\zeta^{\ell+1} - \zeta^\ell\|^2 < +\infty$.

Proof. Let us start by considering the *sufficient decrease inequality* related to the first block:

$$q(\zeta^{\ell+1,1}) + \langle \nabla f(\zeta_0^\ell), \zeta_0^{\ell+1} - \zeta_0^\ell\rangle + \frac{1}{2\gamma_0}\|\zeta_0^{\ell+1} - \zeta_0^\ell\|_{A_\ell}^2 \leq q(\zeta^\ell). \tag{39}$$

Adding $f(\zeta_0^\ell) + \frac{1}{2}\|\zeta_0^{\ell+1} - \zeta_0^\ell\|_{A_\ell}^2$ on both sides of (39) yields

$$q(\zeta^{\ell+1,1}) + \langle \nabla f(\zeta_0^\ell), \zeta_0^{\ell+1} - \zeta_0^\ell\rangle + \frac{1}{2\gamma_0}\|\zeta_0^{\ell+1} - \zeta_0^\ell\|_{A_\ell}^2 + f(\zeta_0^\ell) + \frac{1}{2}\|\zeta_0^{\ell+1} - \zeta_0^\ell\|_{A_\ell}^2 \leq q(\zeta^\ell) + f(\zeta_0^\ell) + \frac{1}{2}\|\zeta_0^{\ell+1} - \zeta_0^\ell\|_{A_\ell}^2 \tag{40}$$

By applying (26) and (27) with $\zeta_0^+ = \zeta_0^\ell$ and $\zeta_0 = \zeta_0^{\ell+1}$ we obtain

$$f(\zeta_0^{\ell+1}) \leq f(\zeta_0^\ell) + \langle \zeta_0^{\ell+1} - \zeta_0^\ell, \nabla f(\zeta_0^\ell)\rangle + \frac{1}{2}\|\zeta_0^{\ell+1} - \zeta_0^\ell\|_{A_\ell}^2 \tag{41}$$

hence the LHS in (40) can be further lower bounded, yielding

$$q(\zeta^{\ell+1,1}) + f(\zeta_0^{\ell+1}) + \frac{1}{2\gamma_0}\|\zeta_0^{\ell+1} - \zeta_0^\ell\|_{A_\ell}^2 \leq q(\zeta^\ell) + f(\zeta_0^\ell) + \frac{1}{2}\|\zeta_0^{\ell+1} - \zeta_0^\ell\|_{A_\ell}^2, \tag{42}$$

hence

$$q(\zeta^{\ell+1,1}) + f(\zeta_0^{\ell+1}) + \frac{1}{2}\left(\frac{1}{\gamma_0} - 1\right)\|\zeta_0^{\ell+1} - \zeta_0^\ell\|_{A_\ell}^2 \leq q(\zeta^\ell) + f(\zeta_0^\ell). \tag{43}$$

To conclude, by Assumption 2, we get

$$q(\zeta^{\ell+1,1}) + f(\zeta_0^{\ell+1}) + \frac{1}{2}\left(\frac{1}{\gamma_0} - 1\right)\underline{\nu}\|\zeta_0^{\ell+1} - \zeta_0^\ell\|^2 \leq q(\zeta^\ell) + f(\zeta_0^\ell). \tag{44}$$

The sufficient decrease inequality for the remaining blocks of index $i \in \{1, \dots, d\}$ can be expressed as

$$q(\zeta^{\ell+1,i+1}) + g(\zeta_i^{\ell+1}) - g(\zeta_i^\ell) + \frac{1}{2\gamma_i}\|\zeta_i^{\ell+1} - \zeta_i^\ell\|^2 \leq q(\zeta^{\ell+1,i}). \tag{45}$$

The first term in the LHS of (45) for the i-th block can be similarly bounded from below with the sufficient decrease inequality for the $(i+1)$-th block, yielding

$$q(\zeta^{\ell+1,i+2}) + g(\zeta_{i+1}^{\ell+1}) - g(\zeta_{i+1}^\ell) + \frac{1}{2\gamma_{i+1}}\|\zeta_{i+1}^{\ell+1} - \zeta_{i+1}^\ell\|^2 + g(\zeta_i^{\ell+1}) - g(\zeta_i^\ell) + \frac{1}{2\gamma_i}\|\zeta_i^{\ell+1} - \zeta_i^\ell\|^2 \leq q(\zeta^{\ell+1,i}). \tag{46}$$

By applying this reasoning recursively from $i = 1$ to $i = d$, we obtain

$$q(\zeta^{\ell+1,d+1}) + \sum_{i=1}^d g(\zeta_i^{\ell+1}) - \sum_{i=1}^d g(\zeta_i^\ell)$$

$$+\sum_{i=1}^{d}\frac{1}{2\gamma_i}\|\zeta_i^{\ell+1}-\zeta_i^{\ell}\|^2 \leq q(\zeta^{\ell+1,1}) \quad (47)$$

where we recall that $q(\zeta^{\ell+1,d+1}) = q(\zeta^{\ell+1})$.

Exploiting now (47), we can lower bound the first term in the LHS of (44), which yields

$$\begin{aligned} &q(\zeta^{\ell+1})+\sum_{i=1}^{d} g(\zeta_i^{\ell+1})-\sum_{i=1}^{d} g(\zeta_i^{\ell}) \\ &\quad+\sum_{i=1}^{d}\frac{1}{2\gamma_i}\|\zeta_i^{\ell+1}-\zeta_i^{\ell}\|^2+f(\zeta_0^{\ell+1}) \\ &+\frac{1}{2}\left(\frac{1}{\gamma_0}-1\right)\underline{\nu}\|\zeta_0^{\ell+1}-\zeta_0^{\ell}\|^2 \leq q(\zeta^{\ell})+f(\zeta_0^{\ell}). \end{aligned} \quad (48)$$

By setting $\mu = \min\left\{\left(\frac{1}{\gamma_0}-1\right)\underline{\nu}, \frac{1}{\gamma_1}, \ldots, \frac{1}{\gamma_d}\right\}$, we deduce (38).

From (38), it follows that the sequence $(\theta(\zeta^{\ell}))_{\ell\in\mathbb{N}}$ is non-increasing. Since function θ is assumed to be bounded from below, this sequence converges to some real number $\underline{\theta}$. We have then, for every integer K,

$$\begin{aligned} \sum_{\kappa=0}^{K}\|\zeta^{\ell}-\zeta^{\ell+1}\|^2 &\leq \frac{1}{\mu}\sum_{\kappa=0}^{K}\left(\theta(\zeta^{\ell})-\theta(\zeta^{\ell+1})\right) \\ &= \frac{1}{\mu}(\theta(\zeta^{0})-\theta(\zeta^{K+1})) \\ &\leq \frac{1}{\mu}(\theta(\zeta^{0})-\underline{\theta}). \end{aligned} \quad (49)$$

Taking the limit as $K \to +\infty$ yields the desired summability property. □

Lemma 2. *Assume that the sequence $(\zeta^{\ell})_{\ell\in\mathbb{N}}$ generated by Algorithm 1 is bounded. Then, for every $\ell \in \mathbb{N}$, there exists $s^{\ell+1} \in \partial\theta(\zeta^{\ell+1})$ such that*

$$\|s^{\ell+1}\| \leq \rho\|\zeta^{\ell+1}-\zeta^{\ell}\|, \quad (50)$$

where $\rho \in (0, +\infty)$.

Proof. The assumed boundedness implies that there exists a bounded subset S of $\mathbb{R}^N$ such that for every $i \in \{0, \ldots, d\}$ and $\ell \in \mathbb{N}$, $\zeta^{\ell+1,i} \in S$. For every $\ell \in \mathbb{N}$, we define

$$s_0^{\ell+1} = \nabla f(\zeta_0^{\ell+1}) + \nabla_0 q(\zeta^{\ell+1}) \quad (51)$$

for which the following holds by virtue of Proposition 1

$$s_0^{\ell+1} \in \partial_0\theta(\zeta^{\ell+1}) = \{\nabla_0\theta(\zeta^{\ell+1})\}. \quad (52)$$

Then

$$\begin{aligned} \|s_0^{\ell+1}\| &\leq \|\nabla f(\zeta_0^{\ell+1})-\nabla f(\zeta_0^{\ell})\| \\ &\quad+\|\nabla f(\zeta_0^{\ell})+\nabla_0 q(\zeta^{\ell+1,1})\| \\ &\quad+\|\nabla_0 q(\zeta^{\ell+1})-\nabla_0 q(\zeta^{\ell+1,1})\|. \end{aligned}$$

From the Lipschitz continuity of ∇f and ∇q on S and the inexact optimality inequality for the first block, we conclude that

$$\|s_0^{\ell+1}\| \leq (L_f+\tau_0+L_q)\,\|\zeta^{\ell+1}-\zeta^{\ell}\|. \quad (53)$$

In the same spirit, for every $i \in \{1, \ldots, d\}$ we consider $r_i^{\ell+1} \in \partial g(\zeta_i^{\ell+1})$ satisfying the inexact optimality inequality with the corresponding τ_i. We then define

$$\begin{aligned} s_i^{\ell+1} &= \nabla_i q(\zeta^{\ell+1}) + r_i^{\ell+1} \\ &\in \nabla_i q(\zeta^{\ell+1}) + \partial g_i(\zeta_i^{\ell+1}) = \partial_i\theta(\zeta^{\ell+1}). \end{aligned} \quad (54)$$

For $i = d$, by virtue of the inexact optimality inequality,

$$\|s_d^{\ell+1}\| \leq \tau_d\|\zeta^{\ell+1}-\zeta^{\ell}\|. \quad (55)$$

On the other side, for $i = 1, \ldots, d-1$

$$\begin{aligned} \|s_i^{\ell+1}\| &= \|\nabla_i q(\zeta^{\ell+1}) + r_i^{\ell+1}\| \\ &\leq \|\nabla_i q(\zeta^{\ell+1}) - \nabla_i q(\zeta^{\ell+1,i+1})\| \\ &\qquad + \|r_i^{\ell+1} + \nabla_i q(\zeta^{\ell+1,i+1})\| \\ &\leq L_q\|\zeta^{\ell+1}-\zeta^{\ell}\| + \tau_i\|\zeta_i^{\ell+1}-\zeta_i^{\ell}\|, \end{aligned}$$

where the last estimate stems from inexact optimality inequality for the i-th block. This yields

$$\|s_i^{\ell+1}\| \leq (L_q+\tau_i)\|\zeta^{\ell+1}-\zeta^{\ell}\|. \quad (56)$$

To conclude, setting

$$s^{\ell+1} = (s_0^{\ell+1}, \ldots, s_d^{\ell+1}) \in \partial\theta(\zeta^{\ell+1}) \quad (57)$$

and $\rho = L_f + \sum_{i=0}^{d}\tau_i + dL_q$ yields (50). □

We now report a first convergence result for a sequence generated by the proposed algorithm, which is reminiscent from [24, Proposition 6]:

Proposition 3 (Properties of the cluster points set)**.** *Suppose that Assumptions 1 and 2 hold. Let $(\zeta^\ell)_{\ell\in\mathbb{N}}$ be a sequence generated by Algorithm 1. Denote by $\omega(\zeta^0)$ the (possibly empty) set of its cluster points. Then*

i) if $(\zeta^\ell)_{\ell\in\mathbb{N}}$ is bounded, then $\omega(\zeta^0)$ is a nonempty compact connected set and

$$\operatorname{dist}(\zeta^\ell, \omega(\zeta^0)) \to 0 \quad \text{as} \quad \ell \to +\infty;$$

ii) $\omega(\zeta^0) \subset \operatorname{crit}\theta$, where $\operatorname{crit}\theta$ is the set of critical points of function θ;

iii) θ is finite valued and constant on $\omega(\zeta^0)$, and it is equal to

$$\inf_{\ell\in\mathbb{N}} \theta(\zeta^\ell) = \lim_{\ell\to+\infty} \theta(\zeta^\ell).$$

Proof. The proof of the above results for the proposed algorithm is basically identical to the one for [24, Proposition 6] for PAM algorithm. In addition, we highlight that according to Assumption 1, our objective function θ is continuous on its domain. □

In conclusion, we have proved that, under Assumptions 1-3, a bounded sequence generated by the proposed method satisfies the assumptions in [37, Theorem 2.9]. Consequently, we can state the following result:

Theorem 4. *Let Assumptions 1-3 be satisfied and let $(\zeta^\ell)_{\ell\in\mathbb{N}}$ be a sequence generated by Algorithm 1 that is assumed to be bounded. Then,*

i) $\sum_{\ell=1}^{+\infty} \|\zeta^{\ell+1} - \zeta^\ell\| < +\infty$;

ii) $(\zeta^\ell)_{\ell\in\mathbb{N}}$ converges to a critical point ζ^* of θ.

We managed to show that both the exact and the inexact version of Algorithm 1 share the same convergence guarantees under Assumptions 1-3. One of the main differences between the two algorithms, as highlighted in [37], is that the former has convergence guarantees that hold for an objective function that is lower semicontinuous, whereas the latter requires its continuity on the domain. However, as it will be shown in the next section, this does not represent an obstacle to the use of Algorithm 1 in image processing applications.

4 Application of P-SASL-PAM

4.1 Smoothing of the coupling term

The application of Algorithm 1 to Problem (12) requires the involved functions to fulfil the requirements listed in Assumption 1. This section is devoted to this analysis, by first defining $d = 2$, $n_0 = n_1 = n_2 = n$, $N = 3n$ and the following functions, for every $x = (x_i)_{1\le i\le n} \in \mathbb{R}^n$, $p = (p_i)_{1\le i\le n} \in \mathbb{R}^n$, and $\beta = (\beta_i)_{1\le i\le n} \in \mathbb{R}^n$,

$$\tilde{q}(x,p,\beta) = \sum_{i=1}^{n} |x_i|^{p_i} e^{-\beta_i p_i}, \tag{58}$$

$$f(x) = \frac{1}{2\sigma^2}\|y - Kx\|_2^2, \tag{59}$$

$$g_1(p) = \sum_{i=1}^{n}\left(\ln\Gamma(1+\frac{1}{p_i}) + \iota_{[a,b]}(p_i)\right) + \lambda\,\mathrm{TV}(p), \tag{60}$$

$$g_2(\beta) = \sum_{i=1}^{n}\left(\beta_i + \frac{(\beta_i-\mu_\beta)^2}{2\sigma_\beta^2}\right) + \zeta\,\mathrm{TV}(\beta). \tag{61}$$

The first item in Assumption 1 regarding the regularity of the coupling term is not satisfied by (58). To circumvent this difficulty, we introduce the *pseudo-Huber loss function* [59] depending on a pair of parameters $\delta = (\delta_1, \delta_2) \in (0,+\infty)^2$ such that $\delta_2 < \delta_1$:

$$(\forall t \in \mathbb{R}) \quad C_\delta(t) = H_{\delta_1}(t) - \delta_2, \tag{62}$$

where H_{δ_1} is the *hyperbolic function* defined, for every $t \in \mathbb{R}$, by $H_{\delta_1}(t) = \sqrt{t^2 + \delta_1^2}$. Function (62) is used as a smooth approximation of the absolute value involved in (58). We then replace (58) with

$$q(x,p,\beta) = \sum_{i=1}^{n} (C_\delta(x_i))^{p_i}\, e^{-\beta_i p_i}. \tag{63}$$

Function C_δ is infinitely differentiable. Thus function (63) satisfies Assumption 1.

Function (59) is quadratic convex, hence it clearly satisfies Assumption 1(ii). Function (60)

is a sum of functions that are proper, lower semi-continuous and either non-negative or bounded from below. The same applies to function (61), which is also strongly convex. It results that (60) and (61) satisfy Assumption 1(iii).

Now, we must show that Θ is a KŁ function. To do so, let us consider the notion of *o-minimal structure* [60], which is a particular family $\mathcal{O} = \{\mathcal{O}_n\}_{n\in\mathbb{N}}$ where each $\mathcal{O}_n$ is a collection of subsets of $\mathbb{R}^n$, satisfying a series of axioms (we refer to [24, Definition 13], for a more complete description). We present hereafter the definition of *definable set* and *definable function* in an o-minimal structure:

Definition 9 (Definable sets and definable functions)**.** *Given an o-minimal structure $\mathcal{O}$, a set $\mathcal{A} \subset \mathbb{R}^n$ such that $\mathcal{A} \in \mathcal{O}_n$ is said to be definable in $\mathcal{O}$. A real extended valued function $f : \mathbb{R} \to (-\infty, +\infty]$ is said to be definable in $\mathcal{O}$ if its graph is a definable subset of $\mathbb{R}^n \times \mathbb{R}$.*

The importance of these concepts in mathematical optimisation is related to the following key result concerning the KŁ property [61]:

Theorem 5. *Any proper lower semicontinuous function $f : \mathbb{R}^n \to (-\infty, +\infty]$ which is definable in an o-minimal structure $\mathcal{O}$ has the KŁ property at each point of* dom ∂f.

Let us identify a structure in which all the functions involved in the definition of Θ are definable. This will be sufficient, as definability is a closed property with respect to several operations, including finite sum and composition of functions. Before that, we provide a couple of examples of o-minimal structure. The first is represented by the structure of *globally subanalytic sets* $\mathbb{R}_{\text{an}}$ [62], which contains all the sets of the form $\{(u,t) \in [-1,1]^n \times \mathbb{R} \mid f(u) = t\}$ where $f : [-1,1]^n \to \mathbb{R}$ is an analytic function that can be analytically extended on a neighbourhood of $[-1,1]^n$. The second example is the log-exp *structure* $(\mathbb{R}_{\text{an}}, \exp)$ [60, 63], which includes $\mathbb{R}_{\text{an}}$ and the graph of the exponential function. Even though this second structure is a common setting for many optimisation problems, it does not meet the requirements for ours: as shown in [64], $\Gamma^{>0}$ (i.e. , the restriction of the Gamma function to $(0,+\infty)$) is not definable on $(\mathbb{R}_{\text{an}}, \exp)$. We thus consider the larger structure $(\mathbb{R}_{\mathcal{G}}, \exp)$, where $\Gamma^{>0}$ has been proved to be definable [65]. $\mathbb{R}_{\mathcal{G}}$ is an o-minimal structure that extends $\mathbb{R}_{\text{an}}$ and is generated by the class $\mathcal{G}$ of *Gevrey* functions from [66].

We end this section with the following result, which will be useful subsequently.

Proposition 6. *The function $t \mapsto \ln\Gamma(1+\frac{1}{t})$ defined on $(0,+\infty)$ is μ-weakly convex with $\mu > \mu_0 \approx 0.1136$.*

Proof. Let us show that there exists $\mu > 0$ such that function $t \mapsto \ln\Gamma(1+\frac{1}{t}) + \mu t^2/2$ is convex on $(0,+\infty)$. The second-order derivative of this function on the positive real axis reads

$$\frac{d^2}{dt^2}\left(\ln\Gamma\left(1+\frac{1}{t}\right)+\frac{\mu}{2}t^2\right) = \frac{1}{t^3}\left(2\text{Dig}\left(1+\frac{1}{t}\right)+\frac{1}{t}\text{Dig}'\left(1+\frac{1}{t}\right)+\mu t^3\right), \tag{64}$$

where the Digamma function Dig() is the logarithmic derivative of the Gamma function. In order to show the convexity of the considered function, we need to ensure that (64) is positive for every $t \in (0,+\infty)$. By virtue of Bohr–Möllerup's theorem [67, Theorem 2.1], among all functions extending the factorial functions to the positive real numbers, only the Gamma function is log-convex. More precisely, its natural logarithm is (strictly) convex on the positive real axis. This implies that $t \mapsto \text{Dig}'(t)$ is positive. It results that the only sign-changing term in (64) is function $t \mapsto 2\,\text{Dig}\left(1+\frac{1}{t}\right)$ as $t \mapsto \text{Dig}(t)$ vanishes in a point $t_0 > 1$ ($t_0 \approx 1.46163$) which corresponds to the minimum point of the Gamma function – and therefore also of its natural logarithm [68]. As a consequence, the Digamma function is strictly positive for $t \in (t_0,+\infty)$, implying that $t \mapsto \text{Dig}\left(1+\frac{1}{t}\right)$ is strictly positive for all $t \in (0, \frac{1}{t_0-1})$. Furthermore, $t \mapsto \text{Dig}\left(1+\frac{1}{t}\right)$ is strictly decreasing and bounded from below, as shown by the

negativity of its first derivative

$$\frac{d}{dt}\mathrm{Dig}\left(1+\frac{1}{t}\right) = -\frac{1}{t^2}\mathrm{Dig}'\left(1+\frac{1}{t}\right)$$

and by the following limit

$$\lim_{t\to+\infty} \mathrm{Dig}\left(1+\frac{1}{t}\right) = \mathrm{Dig}(1) = -\mathcal{E}$$

where the last equality holds by virtue of the Gauss Digamma theorem, and $\mathcal{E}$ is Euler-Mascheroni's constant $\mathcal{E} \approx 0.57721$ [69]. In conclusion, for $t \in [\frac{1}{t_0-1}, +\infty)$, we need to ensure that the positive terms in (64) manage to balance the negative contribution of function $t \mapsto 2\mathrm{Dig}\left(1+\frac{1}{t}\right) > -2\mathcal{E}$. This leads to a condition on parameter $\mu > 0$, since we can impose that

$$0 < \mu t^3 - 2\mathcal{E},$$

where the right-hand side expression has a lower bound $\mu/(t_0-1)^3 - 2\mathcal{E}$ that is positive when

$$\mu > 2\mathcal{E}(t_0-1)^3 = \mu_0 \approx 0.1136.$$

This shows that function $t \mapsto \ln\Gamma(1+\frac{1}{t})$ is μ-weakly convex. □

4.2 Proximal computations

Let us now discuss the practical implementation of the proximal computations involved in Algorithm 1. Specifically, as we will show, none of these operators have closed-form expressions, so we need to resort to the inexact version. To ease the description, we summarise in Algorithm 2 the application of Algorithm 1 to the resolution of (12). As pointed out in [70] and in [35], the role of the *relative error conditions* (16) and (17) are more of theoretical interest than of practical use. In the following, we will illustrate optimisation procedures ensuring that condition (16) is satisfied for every block of variables at every iteration.

Proximal computation with respect to x.

Subproblem (65) in Algorithm 2 requires the computation of the proximity operator of the following

Algorithm 2 P-SASL-PAM to solve (12)

Initialize x^0, p^0 and β^0
Set $\gamma_0 \in (0,1)$, $\gamma_1 \in (0, 1/\mu_0)$, $\gamma_2 > 0$
For $\ell = 0, 1, \ldots$
Set $A_\ell \in \mathcal{S}_n$
Find

$$x^{\ell+1} \approx \mathrm{prox}^{A}_{\gamma_0 q(\cdot, p^\ell, \beta^\ell)}(x^\ell - \gamma_0 A^{-1}\nabla f(x^\ell)) \quad (65)$$
(with Algorithm 3)

$$p^{\ell+1} \approx \mathrm{prox}_{\gamma_1 \theta(x^{\ell+1}, \cdot, \beta^\ell)}(p^\ell) \quad (66)$$
(with Algorithm 5)

$$\beta^{\ell+1} \approx \mathrm{prox}_{\gamma_2 \theta(x^{\ell+1}, p^{\ell+1}, \cdot)}(\beta^\ell) \quad (67)$$
(with Algorithm 6)

separable function

$$q(\cdot, p^\ell, \beta^\ell) : x \mapsto \sum_{i=1}^{n} (C_\delta(x_i))^{p_i^\ell} e^{-\beta_i^\ell p_i^\ell},$$

within a weighted Euclidean metric induced by some matrix $A \in \mathcal{S}_n$. We notice that $x_i \mapsto (C_\delta(x_i))^{p_i^\ell}$ is non-convex whenever $p_i^\ell \in (0,1)$, for some $i \in \{1, \ldots, n\}$. In order to overcome this issue, we apply a majorisation principle [71]. Let us introduce function σ defined, for every $u \in [\delta_1, +\infty)$, as $\sigma(u) = (u-\delta_2)^p$ with $p \in (0,1]$, and vector $\delta = (\delta_1, \delta_2) \in (0, +\infty)^2$ such that $\delta_2 < \delta_1$. Since this function is concave, it can be majorised by its first-order expansion around any point $w > \delta_2$:

$$\begin{aligned}(\forall u > \delta_2) \quad (u-\delta_2)^p &\leq (w-\delta_2)^p \\ &\quad + p(w-\delta_2)^{p-1}(u-w) \\ &= (1-p)(w-\delta_2)^p + p(w-\delta_2)^{p-1}(u-\delta_2).\end{aligned} \quad (68)$$

Setting, for every $(t, t') \in \mathbb{R}^2$, $u = H_{\delta_1}(t) \geq \delta_1$, $w = H_{\delta_1}(t') \geq \delta_1$ allows us to deduce the following majorisation:

$$(C_\delta(t))^p \leq (1-p)(C_\delta(t'))^p + p(C_\delta(t'))^{p-1} C_\delta(t). \quad (69)$$

Let us now define $\mathcal{I}^\ell = \{i \in \{1, \ldots, n\} \mid p_i^\ell \geq 1\}$ and $\mathcal{J}^\ell = \{1, \ldots, n\} \setminus \mathcal{I}^\ell$. Given $v = (v_i)_{1 \leq i \leq n} \in$

$\mathbb{R}^n$, we deduce from (69) that

$$
\begin{aligned}
&(\forall x = (x_i)_{1\leq i\leq n} \in \mathbb{R}^n) \\
&\quad q(x, p^\ell, \beta^\ell) = \sum_{i\in\mathcal{I}^\ell} (C_\delta(x_i))^{p_i^\ell}\, e^{-\beta_i^\ell p_i^\ell} \qquad (70)\\
&\qquad\qquad + \sum_{i\in\mathcal{J}^\ell} (C_\delta(x_i))^{p_i^\ell}\, e^{-\beta_i^\ell p_i^\ell} \\
&\quad \leq \overline{q}(x, v, p^\ell, \beta^\ell), \qquad (71)
\end{aligned}
$$

where the resulting majorant function is separable, i.e.

$$
\overline{q}(x, v, p^\ell, \beta^\ell), = \sum_{i=1}^n \overline{q}_i(x_i, v_i, p_i^\ell, \beta_i^\ell), \qquad (72)
$$

with, for every $i \in \{1, \ldots, n\}$ and $x_i \in \mathbb{R}$,

$$
\begin{aligned}
&\overline{q}_i(x_i, v_i, p_i^\ell, \beta_i^\ell) \qquad (73)\\
&= \begin{cases} e^{-\beta_i^\ell p_i^\ell}\, (C_\delta(u_i))^{p_i^\ell}, & \text{if } p_i^\ell \geq 1 \\ e^{-\beta_i^\ell p_i^\ell}\Big((C_\delta(v_i))^{p_i^\ell}(1-p_i^\ell) & \\ \quad + p_i^\ell (C_\delta(v_i))^{p_i^\ell - 1} C_\delta(x_i)\Big) & \text{otherwise.} \end{cases}
\end{aligned}
$$

In a nutshell, each term of index $i \in \{1, \ldots, n\}$ in (72) coincides either with the i-th term of $q(\cdot, p^\ell, \beta^\ell)$ when $i \in \mathcal{I}^\ell$, or it is a convex majorant of this i-th term with respect to v_i when $i \in \mathcal{J}^\ell$. We thus propose to adopt an MM procedure by building a sequence of convex surrogate problems for the non-convex minimisation problem involved in the computation of $\mathrm{prox}^A_{\gamma_0 q(\cdot, p^\ell, \beta^\ell)}$. At the κ-th iteration of this procedure, following the MM principle, the next iterate $x^{\kappa+1}$ is determined by setting $v = x^\kappa$. We summarise the strategy in Algorithm 3.

Algorithm 3 MM algorithm to approximate $\mathrm{prox}^A_{\gamma_0 q(\cdot, p^\ell, \beta^\ell)}(x^+)$ with $x^+ \in \mathbb{R}^n$

Initialize $x^0 \in \mathbb{R}^n$
For $\kappa = 0, 1, \ldots$ **until** convergence

$$
x^{\kappa+1} = \mathrm{prox}^A_{\gamma_0 \overline{q}(\cdot, x^\kappa, p^\ell, \beta^\ell)}(x^+) \qquad (74)
$$

(with Algorithm 4)

Since function $\overline{q}(\cdot, v, p^\ell, \beta^\ell)$ is convex, proper, and lower semicontinuous, its proximity operator in the weighted Euclidean metric induced by matrix A is guaranteed to be uniquely defined. It can be computed efficiently using the Dual Forward-Backward (DFB) method [72], outlined in Algorithm 4.

Algorithm 4 DFB algorithm to compute $\mathrm{prox}^A_{\gamma_0 \overline{q}(\cdot, v; p^\ell, \beta^\ell)}(x^+)$ with $x^+ \in \mathbb{R}^n$

Initialize dual variable $w^0 \in \mathbb{R}^n$
Set $\eta \in (0, 2|||A|||^{-1})$
For $\kappa' = 0, 1, \ldots$ **until** convergence

$$
\begin{aligned}
u^{\kappa'} &= x^+ - A w^{\kappa'}, \qquad (75)\\
w^{\kappa'+1} &= w^{\kappa'} + \eta u^{\kappa'} \qquad (76)\\
&\quad - \eta \mathrm{prox}_{\eta^{-1}\gamma_0 \overline{q}(\cdot, v, p^\ell, \beta^\ell)}(\eta^{-1} w^{\kappa'} + u^{\kappa'}).
\end{aligned}
$$

Return $u^{\kappa'} \in \mathbb{R}^n$

The update in (77) can be performed componentwise since function $\overline{q}(\cdot, v, p^\ell, \beta^\ell)$ is separable. Thanks to the separability property, computing $\mathrm{prox}_{\eta^{-1}\gamma_0 \overline{q}(\cdot, v, p^\ell, \beta^\ell)}$ boils down to solving n one-dimensional optimization problems, that is

$$
\begin{aligned}
&(\forall u^+ = (u_i^+)_{1\leq i\leq n} \in \mathbb{R}^n) \\
&\quad \mathrm{prox}_{\eta^{-1}\gamma_0 \overline{q}(\cdot, v, p^\ell, \beta^\ell)}(u^+) \\
&\quad = \Big(\mathrm{prox}_{\eta^{-1}\gamma_0 \overline{q}_i(\cdot, v_i, p_i^\ell, \beta_i^\ell)}(u_i^+)\Big)_{1\leq i\leq n}. \qquad (77)
\end{aligned}
$$

More precisely,

- for every $i \in \{1, \ldots, n\}$, such that $p_i^\ell \leq 1$,

$$
\begin{aligned}
&\mathrm{prox}_{\eta^{-1}\gamma_0 \overline{q}_i(\cdot, v_i, p_i^\ell, \beta_i^\ell)}(u_i^+) \\
&= \mathrm{prox}_{\eta^{-1}\gamma_0 e^{-\beta_i^\ell p_i^\ell} p_i^\ell (C_\delta(v_i))^{p_i^\ell - 1} C_{\delta_1}}(u_i^+) \\
&= \mathrm{prox}_{\eta^{-1}\gamma_0 e^{-\beta_i^\ell p_i^\ell} p_i^\ell (C_\delta(v_i))^{p_i^\ell - 1} H_{\delta_1}}(u_i^+). \qquad (78)
\end{aligned}
$$

  The proximity operator of the so-scaled version of function H_{δ_1} can be determined by solving a quartic polynomial equation.[1]
- For every $i \in \{1, \ldots, n\}$ such that $p_i^\ell > 1$,

[1] http://proximity-operator.net/scalarfunctions.html

$$\operatorname{prox}_{\eta^{-1}\gamma_0\overline{q}_i(\cdot,v_i,p_i^\ell,\beta_i^\ell)}(u_i^+) = \operatorname{prox}_{\eta^{-1}\gamma_0 e^{-\beta_i^\ell p_i^\ell}(C_\delta)^{p_i^\ell}}(u_i^+). \quad (79)$$

The latter quantity can be evaluated through a bisection search to find the root of the derivative of the involved proximally regularised function.

Remark 6. *Due to the non-convexity of $q(\cdot, p^\ell, \beta^\ell)$, there is no guarantee that the point estimated by Algorithm 4 coincides with the exact proximity point. However, we did not notice any numerical issues in our implementation.*

Proximal computation with respect to p.

Subproblem (66) requires to compute the proximity operator of $\gamma_1\big(q(x^{\ell+1},\cdot,\beta^\ell)+g\big)$, which is equivalent to solving the following minimization problem

$$\underset{p\in[a,b]^n}{\text{minimize}} \;\; \psi^\ell(p) + \lambda\ell_{1,2}(Dp), \quad (80)$$

where, for every $p\in\mathbb{R}^n$, $\psi^\ell(p)=\sum_{i=1}^n \psi_i^\ell(p_i)$ with

$$(\forall i\in\{1,\dots,n\})(\forall p_i\in\mathbb{R})\quad \psi_i^\ell(p_i) = \begin{cases} \big(C_\delta(x_i^{\ell+1})\big)^{p_i} e^{-\beta_i^\ell p_i} + \ln\Gamma(1+\frac{1}{p_i}) \\ \quad +\frac{1}{2\gamma_1}(p_i-p_i^\ell)^2 & \text{if } p_i>0 \\ +\infty & \text{otherwise.}\end{cases} \quad (81)$$

Moreover, $D=[D_{\mathrm{h}},D_{\mathrm{v}}]$ where $(D_{\mathrm{h}},D_{\mathrm{v}})\in(\mathbb{R}^{n\times n})^2$ are the discrete horizontal and vertical 2D gradient operators, and the $\ell_{1,2}$-norm is defined as

$$(\forall p\in\mathbb{R}^n)\quad \ell_{1,2}(Dp)=\sum_{i=1}^n \|([D_{\mathrm{h}}p]_i,[D_{\mathrm{v}}p]_i)\|_2.$$

Problem (80) is equivalent to minimizing the sum of the indicator function of a hypercube, a separable component and a non-separable term involving the linear operator D. According to Proposition 6, we can ensure the convexity of each term $(\psi_i^\ell)_{1\le i\le n}$ by setting $\gamma_1<\frac{1}{\mu_0}\approx 8.805$. In order to solve (80), it is then possible to implement a Primal-Dual (PD) algorithm [73–75] as outlined in Algorithm 5.

Algorithm 5 Primal Dual Algorithm for solving (80)

Initialise the dual variables $v_1^0\in\mathbb{R}^{n\times 2}$,$v_2^0\in\mathbb{R}^n$.
Set $\tau>0$ and $\sigma>0$ such that $\tau\sigma(|||D|||^2+1)<1$.
for $\kappa=0,1,\dots$ **until** convergence

$$u^\kappa = p^\kappa - \tau(D^*v_1^\kappa + v_2^\kappa), \quad (82)$$
$$p^{\kappa+1} = \operatorname{proj}_{[a,b]^n}(u^\kappa), \quad (83)$$
$$w_1^\kappa = v_1^\kappa + \sigma D(2p^{\kappa+1}-p^\kappa), \quad (84)$$
$$v_1^{\kappa+1} = w_1^{\kappa+1} - \sigma\operatorname{prox}_{\frac{\lambda\ell_{1,2}}{\sigma}}\big(\frac{w_1^\kappa}{\sigma}\big). \quad (85)$$
$$w_2^\kappa = v_2^\kappa + \sigma(2p^{\kappa+1}-p^\kappa), \quad (86)$$
$$v_2^{\kappa+1} = w_2^{\kappa+1} - \sigma\operatorname{prox}_{\frac{\psi^\ell}{\sigma}}\big(\frac{w_2^\kappa}{\sigma}\big). \quad (87)$$

Return $p^{\kappa+1}\in[a,b]^n$

The proximity operator of the involved $\ell_{1,2}$ norm has a closed-form expression. For every $w_1=([w_1]_{i,1},[w_1]_{i,2})_{1\le i\le n}\in\mathbb{R}^{n\times 2}$ and $\lambda>0$, we have

$$\begin{aligned}&\operatorname{prox}_{\lambda\ell_{1,2}}(w_1)\\ &\quad=\Big(\operatorname{prox}_{\lambda\|\cdot\|_2}\big(([w_1]_{i,1},[w_1]_{i,2})\big)\Big)_{1\le i\le n}\\ &\quad=\left(([w_1]_{i,1},[w_1]_{i,2}) - \frac{\lambda([w_1]_{i,1},[w_1]_{i,2})}{\max\{\lambda,\|([w_1]_{i,1},[w_1]_{i,2})\|_2\}}\right)_{1\le i\le n}.\end{aligned}$$

The proximal point at $w_2^\kappa/\sigma=\big([w_2^\ell]_i/\sigma\big)_{1\le i\le n}\in\mathbb{R}^n$ of the separable term ψ^ℓ with respect to a step size $1/\sigma$ can be found by minimizing, for every $i\in\{1,\dots,n\}$, the following smooth function

$$(\forall t\in(0,+\infty))\quad \mathrm{g}_{1,i}(t)=\psi_i^\ell(t)+\frac{\sigma}{2}\Big(t-\frac{[w_2^\kappa]_i}{\sigma}\Big)^2.$$

The update in (87) then reads

$$v_2^{\kappa+1}=\big([w_2^{\kappa+1}]_i-\sigma[w_2^\kappa]_i^*\big)_{1\le i\le n}$$

where, for every $i\in\{1,\dots,n\}$, $[w_2^\kappa]_i^*$ corresponds to the unique zero of the derivative of $\mathrm{g}_{1,i}$. This zero is found by applying Newton's method

initialised with

$$\bar{w}_i = \left(\max \left\{ 10^{-3}, \frac{[w_2^\kappa]_i}{\sigma} \right\} \right)_{1 \leq i \leq n}.$$

Proximal computation with respect to β. Subproblem (67) requires the solution of the following minimisation problem:

$$\underset{\beta \in \mathbb{R}^n}{\text{minimize}} \;\; \varphi^\ell(\beta) + \zeta \ell_{1,2}(D\beta) \tag{88}$$

where D and $\ell_{1,2}$ have been defined previously and

$$(\forall \beta = (\beta_i)_{1 \leq i \leq n} \in \mathbb{R}^n) \quad \varphi^\ell(\beta) = \sum_{i=1}^n \varphi_i^\ell(\beta_i)$$

with, for every $i \in \{1, \ldots, n\}$,

$$\varphi_i^\ell(\beta_i) = \left(C_\delta(x_i^{\ell+1})\right)^{p_i^{\ell+1}} e^{-\beta_i p_i^{\ell+1}} + \beta_i + \frac{(\beta_i - \mu_\beta)^2}{2\sigma_\beta^2} + \frac{1}{2\gamma_2}(\beta_i - \beta_i^\ell)^2 \tag{89}$$

The above problem shares a structure similar to the one studied in the previous case since the objective function is the sum of the smooth convex term φ^ℓ and the non-smooth convex one $\zeta\,\mathrm{TV} = \zeta \ell_{1,2}(D\cdot)$, and it can be solved by the primal-dual procedure outlined in Algorithm 6.

Algorithm 6 Primal Dual Algorithm for minimizing (88)

Set $\tau > 0$ and $\sigma > 0$ such that $\tau\sigma |||D|||^2 \leq 1$.
Initialise the dual variable $v^0 \in \mathbb{R}^{n \times 2}$.
for $\kappa = 0, 1, \ldots$ **until** convergence

$$u^\kappa = \beta^\kappa - \tau(D^* v^\kappa), \tag{90}$$
$$\beta^{\kappa+1} = \mathrm{prox}_{\tau\varphi^\ell}(u^\kappa), \tag{91}$$
$$w^\kappa = v^\kappa + \sigma D(2\beta^{\kappa+1} - \beta^\kappa), \tag{92}$$
$$v^{\kappa+1} = w^{\kappa+1} - \sigma \mathrm{prox}_{\frac{\zeta \ell_{1,2}}{\sigma}}\left(\frac{w^\kappa}{\sigma}\right). \tag{93}$$

Return $\beta^{\kappa+1} \in \mathbb{R}^n$

At each iteration κ of Algorithm 6, the proximity operator of φ^ℓ is expressed as

$$(\forall \beta = (\beta_i)_{1 \leq i \leq n} \in \mathbb{R}^n) \quad \mathrm{prox}_{\tau\varphi^\ell}(\beta) = \left(\mathrm{prox}_{\tau\varphi_i^\ell}(\beta_i)\right)_{1 \leq i \leq n}. \tag{94}$$

For every $i \in \{1, \ldots, n\}$, $\mathrm{prox}_{\tau\varphi_i^\ell}(\beta_i)$ is the minimizer of function

$$(\forall \beta_i \in \mathbb{R}) \quad \mathrm{g}_{2,i}(\beta_i) = \varphi_i^\ell(\beta_i) + \frac{1}{2\tau}(\beta_i - u_i^\kappa)^2. \tag{95}$$

The nonlinear equation defining the unique zero of the derivative of $\mathrm{g}_{2,i}$ admits a closed-form solution that involves the Lambert W-function [76]. Indeed, let us introduce the following notation:

$$a_{1,i} = p_i^{\ell+1}\left(C_\delta(x_i^{\ell+1})\right)^{p_i^{\ell+1}}, \tag{96}$$
$$a_2 = \left(\frac{1}{\sigma_\beta^2} + \frac{1}{\gamma_2} + \frac{1}{\tau}\right)^{-1}, \tag{97}$$
$$a_{3,i} = 1 - \frac{\mu_\beta}{\sigma_\beta^2} - \frac{\beta_i^\ell}{\gamma_2} - \frac{u_i^\kappa}{\tau}. \tag{98}$$

Then

$$\begin{aligned}
&\mathrm{g}_{2,i}'(\beta_i) = 0 \\
&\iff -a_{1,i}\exp(-p_i^{\ell+1}\beta_i) + \frac{\beta_i}{a_2} + a_{3,i} = 0 \\
&\iff p_i^{\ell+1}(\beta_i + a_2 a_{3,i})\exp(p_i^{\ell+1}(\beta_i + a_2 a_{3,i})) \\
&\qquad = p_i^{\ell+1} a_{1,i} a_2 \exp(p_i^{\ell+1} a_2 a_{3,i}) \\
&\iff \beta_i \\
&= \frac{1}{p_i^{\ell+1}} W(p_i^{\ell+1} a_{1,i} a_2 \exp(p_i^{\ell+1} a_2 a_{3,i})) - a_2 a_{3,i},
\end{aligned} \tag{99}$$

where the last equivalence comes from the fact that the Lambert W-function is single-valued and satisfies the following identity for a pair $(X, Y) \in \mathbb{R} \times \left(-\frac{1}{e}, +\infty\right)$:

$$X\exp(X) = Y \iff X = W(Y). \tag{100}$$

Notice that the expression in (99) is well defined since the argument of the Lambert function is always positive.

In conclusion, the update in (91) reads as $\beta^{\kappa+1} = \left(\beta_i^{\kappa+1}\right)_{1 \leq i \leq n}$ where each component of this vector is calculated according to (99).

5 Numerical Experiments

In this section, we illustrate the performance of our approach on a problem of joint deblurring/segmentation of realistically simulated ultrasound images. We consider images with two regions (*Simu1*) and three regions (*Simu2*) extracted from [17]. Both images have dimension 256×256 pixels. The shape parameters p and the reparameterised scale parameters β are set in each region following the choices for p and α in [17], itself based on the experimental setting in [19]. This strategy allows us to have a reference configuration for β, which led us to choose a non-necessarily zero-mean Gaussian distribution as a prior for this parameter. In our experiments, we will treat μ_β as an unknown parameter, along with the regularisation parameters for the Total Variation priors. The pixel values in each region of the original image $x \in \mathbb{R}^n$ are obtained as a realisation of a random variable following a $\mathcal{GGD}$ with the corresponding shape and scale parameters p and α. We define K as the linear operator modelling the convolution with the point spread function of a 3.5 MHz linear probe obtained with the Field II ultrasound simulator [77]. To reproduce the same setting as in [17], we obtain the observed degraded images $y \in \mathbb{R}^n$ from the original images $x \in \mathbb{R}^n$ by applying the observation model (1), where we set the additive noise variance (which will be assumed to be known) to $\sigma^2 = 0.013$ for *Simu1* and $\sigma^2 = 33$ for *Simu2*. For the preconditioner, we consider a regularised version of the inverse of the Hessian of the data fidelity function in (59), given by

$$A = \sigma^2 (K^\top K + \mu \mathbb{I}_m)^{-1}$$

where $\mu = 0.1$, so that A is well defined and constant throughout the iterations. Following the procedure outlined in [17], we initialise $x^0 \in \mathbb{R}^n$ using a pre-deconvolved image obtained with a Wiener filter applied to the observed data y, $(p_i^0)_{1 \leq i \leq n}$ is drawn from an i.i.d. uniform distribution in the range $[0.5, 1.5]$, while $(\beta_i^0)_{1 \leq i \leq n}$ is drawn from an i.i.d. Gaussian distribution with mean μ_β and unit standard deviation. We set $\mu_\beta = 0$ for *Simu 1* and $\mu_\beta = 4$ for *Simu 2*, for arguments discussed in SM 2. We adopt the recovery strategy described in Section 4 and describe hereafter the setting of the model/algorithm hyperparameters.

The model parameters that need to be tuned are the $\delta_1 > 0$ and $\delta_2 > 0$ values for the pseudo-Huber function, the mean $\mu_\beta \in \mathbb{R}$ and the standard deviation $\sigma_\beta > 0$ for the reparameterised scale parameter, and finally the regularisation parameters $(\lambda, \zeta) \in (0, +\infty)^2$ for the TV terms. For parameter $\delta = (\delta_1, \delta_2)$, we applied the following choice, resulting from a rough empirical search,: $\delta_1 = 1$ while $\delta_2 = \delta_1 \times 10^{-2}$. For what concerns the Gaussian parameters of the reparameterised scale variable $(\mu_\beta, \sigma_\beta)$, the mean μ_β is the most influential on the estimated solution, so we dedicated an in-depth analysis for its choice in combination with the TV regularisation parameters (λ, ζ). More precisely, we tested different values of μ_β in the range $[-10, 10]$ in combination with a grid search for $(\lambda, \zeta) \in \{10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3\}^2$ with respect to different quality metrics and identified an optimal choice for μ_β. The standard deviation appeared less influential and is set to $\sigma_\beta = 1$ in all our experiments. The details of the analysis are illustrated in the annexed SM 2.

The algorithmic hyperparameters include the step sizes of the proximal steps, as well as the preconditioning matrix involved in the preconditioned proximal gradient step. We set $(\gamma_0, \gamma_1, \gamma_2) = (0.99, 1, 1)$ in order to meet the convergence assumptions in Algorithm 2. In particular, the choice for γ_0 approximates the highest value allowed for the step size of the preconditioned inexact FB scheme in (65), while γ_1 satisfies the condition $\gamma_1 < 8.805$ for the convexity of the function in (80).

In order to obtain the labelling of a segmented image from our estimated shape parameter (denoted by $\widehat{p}$) we use a quantisation procedure based on Matlab functions `multithresh` and `imquantize`. The former defines a desired number of quantisation levels using Otsu's method, while the latter performs a truncation of the data values according to the provided quantisation levels. We remark here that the number of labels does not need to be defined throughout the proposed optimisation procedure, but only at the final segmentation step. This step can thus be considered as a post-processing that is performed on the estimated solution.

In order to evaluate the quality of the solution, we consider the following metrics: for the estimated image, we make use of the peak signal-to-noise ratio (PSNR) defined as follows, x being the original signal and $\widehat{x}$ the estimated one:

$$\mathrm{PSNR} = 10\log_{10}\left(n \max_{i\in\{1,\dots,n\}} \{x_i, \hat{x}_i\}^2 / \|x_i - \hat{x}_i\|^2\right),$$

and of the structure similarity measure (SSIM) [78]. For the segmentation task we compute the percentage OA of correctly predicted labels.

The stopping criteria for Algorithm 2 outer and inner loops are set by defining a threshold level on the relative change between two consecutive iterates of the involved variables, the relative change of the objective values of two consecutive iterates and a maximum number of iterations. The outer loop in Algorithm 2 stops whenever $\ell = 10000$ or when both $\|\zeta^{\ell+1} - \zeta^{\ell}\|/\|\zeta^{\ell}\| < 10^{-4}$ and $|\theta(\zeta^{\ell+1}) - \theta(\zeta^{\ell})|/|\theta(\zeta^{\ell})| < 10^{-4}$. The MM procedure to compute $x^{\ell+1}$ in Algorithm 3 is stopped after 300 iterations or when $\|x^{\kappa+1} - x^{\kappa}\|/\|x^{\kappa}\| < 10^{-3}$. The DFB procedure in Algorithm 4 to compute $u^{\kappa+1}$ is stopped after 300 iterations or when $\|u^{\kappa+1} - u^{\kappa}\|/\|u^{\kappa}\| < 10^{-3}$. The PD procedure in Algorithm 5, and Algorithm 6 computing $p^{\ell+1}$ (resp. $\beta^{\ell+1}$) terminates after 200 iteration or when $\|p^{\kappa+1} - p^{\kappa}\|/\|p^{\kappa}\| < 10^{-3}$ (resp. $\|\beta^{\kappa+1} - \beta^{\kappa}\|/\|\beta^{\kappa}\| < 10^{-3}$).

Figure 2 illustrates in the first and second line the B–mode image of the original x, of the degraded y, and of the reconstructed image $\hat{x}$ on both examples. The B–mode image is the most common representation of an ultrasound image, displaying the acoustic impedance of a 2-dimensional cross section of the considered tissue. The reconstructed results in Figure 2 (right) show clearly reduced blur and sharper region contours. We then report in the third and fourth lines of Figure 2 the estimated shape parameter and the segmentation obtained via the aforementioned quantisation procedure, which confirms its good performance. We notice that our estimated $\hat{p}_i$ values are consistent with the original ones and the fact that the results for *Simu2* are slightly less accurate than the ones for *Simu1* is in line with the results presented in [17, Table III] for P-ULA, HMC and PP-ULA, suggesting that the configuration of the parameters for *Simu2* is quite challenging.

Table 1 proposes a quantitative comparison of our results against those of the methods considered in [17]: a combination of Wiener deconvolution and Otsu's segmentation [47], a combination of LASSO deconvolution and SLaT segmentation [40], the adjusted Hamiltonian Monte Carlo (HMC) method [79], the Proximal Unadjusted Langevin algorithm (P-ULA) [80] and its preconditioned version (PP-ULA) [17] for joint deconvolution and segmentation. From this table, we can conclude that the proposed variational method is able to compete with state-of-the-art Monte Carlo Markov Chain techniques in terms of both segmentation and deconvolution performance. For what concerns the computational time, the average time (over 10 runs of the algorithm) required by P-SASL-PAM to meet the stopping criteria $\|\zeta^{\ell+1} - \zeta^{\ell}\|/\|\zeta^{\ell}\| < 10^{-4}$ and $|\theta(\zeta^{\ell+1}) - \theta(\zeta^{\ell})|/|\theta(\zeta^{\ell})| < 10^{-4}$ corresponds to 493.2 seconds (approximately $8'13''$) for *Simu1* and 536.4 seconds (approximately $8'56''$) for *Simu2*. Simulations were run on Matlab 2021b on an Intel Xeon Gold 6230 CPU 2.10GHz. In Table 1, we report the computational times for PULA, HMC and PP-ULA from [17, TABLE II], which were obtained on Matlab 2018b on an Intel Xeon CPU E5-1650 3.20GHz.

Eventually, Figure 3 (a)-(b) show the evolution of the mean value of the cost function for both *Simu1* (a) and *Simu2* (b) along 500 iterations for ten different sampling of p^0 and β^0, while Figure 3 (c)-(d)illustrate on a logarithmic scale the relative distance from the iterates to the solution $\|\zeta^{\ell} - \zeta^{\infty}\|/\|\zeta^{\infty}\|$ for *Simu1* (c) and *Simu2* (d), showing the convergence of our algorithm.

Additional experiments can be found in Supplementary Material: SM1, showing that for standard wavelet-based image restoration problems the proposed regularisation outperforms other sparsity measures.

6 Conclusions

We investigated a novel approach for the joint reconstruction/feature extraction problem. The novelty in this work lies both in the problem formulation and in the resolution procedure. Firstly,

	Simu1				*Simu2*			
METHOD	PSNR	SSIM	OA	TIME	PSNR	SSIM	OA	TIME
Wiener-Otsu	37.1	0.57	99.5	–	35.4	0.63	96.0	–
LASSO-SLaT	39.2	0.60	99.6	–	37.8	0.70	98.3	–
P-ULA	38.9	0.45	98.7	2 h 27 min	37.1	0.57	94.9	3 h 06 min
HMC	40.0	0.62	99.7	1 h 08 min	36.4	0.64	98.5	4 h 14 min
PP-ULA	40.3	0.62	99.7	12 min	38.6	0.71	98.7	39 min
OURS	40.2	0.61	99.9	8 min	38.1	0.70	97.7	9 min

Table 1: PSNR, SSIM, OA scores and Computational time for *Simu1* and *Simu2* from [17]. The symbol "–" means the result was not available in the reference paper.

we proposed a new variational model in which we introduced a flexible sparse regularisation term for the reconstruction task. Secondly, we designed an inexact version of a TITAN-based block alternating optimisation scheme, whose aim is to exploit the structure of the problem and the properties of the functions involved in it. We established convergence results for the proposed algorithm whose scope goes beyond the image processing problems considered in our work. We illustrated the validity of the approach on numerical examples in the case of a joint deconvolution-segmentation problem. We also included comparisons with state-of-the-art methods with respect to which our proposal registers a similar qualitative and quantitative performance. An attractive aspect of the proposed work is that the space variant parameters defining the flexible sparse regularisation do not need to be defined in advance, but are inherently estimated by the iterative optimisation procedure. For what concerns the tuning of the hyperparamters of the model, the design of an automatic strategy could be an interesting development of the work, for instance through supervised learning.

Acknowledgments. This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 861137. The authors thank Ségolène Martin for her careful reading of the initial version of this manuscript.

References

[1] Daubechies, I., Defrise, M., De Mol, C.: An iterative thresholding algorithm for linear inverse problems with a sparsity constrains. Communications on Pure and Applied Mathematics **57** (2004)

[2] Grasmair, M., Haltmeier, M., Scherzer, O.: Sparse regularization with lq penalty term. Inverse Problems **24**, 055020 (2008)

[3] Lorenz, D.: Convergence rates and source conditions for Tikhonov regularization with sparsity constraints. Journal of Inverse and Ill-posed Problems **16**(5), 463–478 (2008)

[4] Ramlau, R., Resmerita, E.: Convergence rates for regularization with sparsity constraints. Electronic transactions on numerical analysis ETNA **37**, 87–104 (2010)

[5] Tibshirani, R.: Regression shrinkage and selection via the Lasso. Journal of the Royal Statistical Society: Series B (Methodological) **58**, 267–288 (1996)

[6] Chartrand, R.: Exact reconstruction of sparse signals via nonconvex minimization. IEEE Signal Processing Letters **14**, 707–710 (2007)

[7] Grasmair, M.: Well-posedness and convergence rates for sparse regularization with sublinear ℓ^q penalty term. Inverse Problems & Imaging **3**(3), 383–387 (2009)

[8] Zarzer, C.: On Tikhonov regularization with non-convex sparsity constraints. Inverse Problems **25**, 025006 (2009)

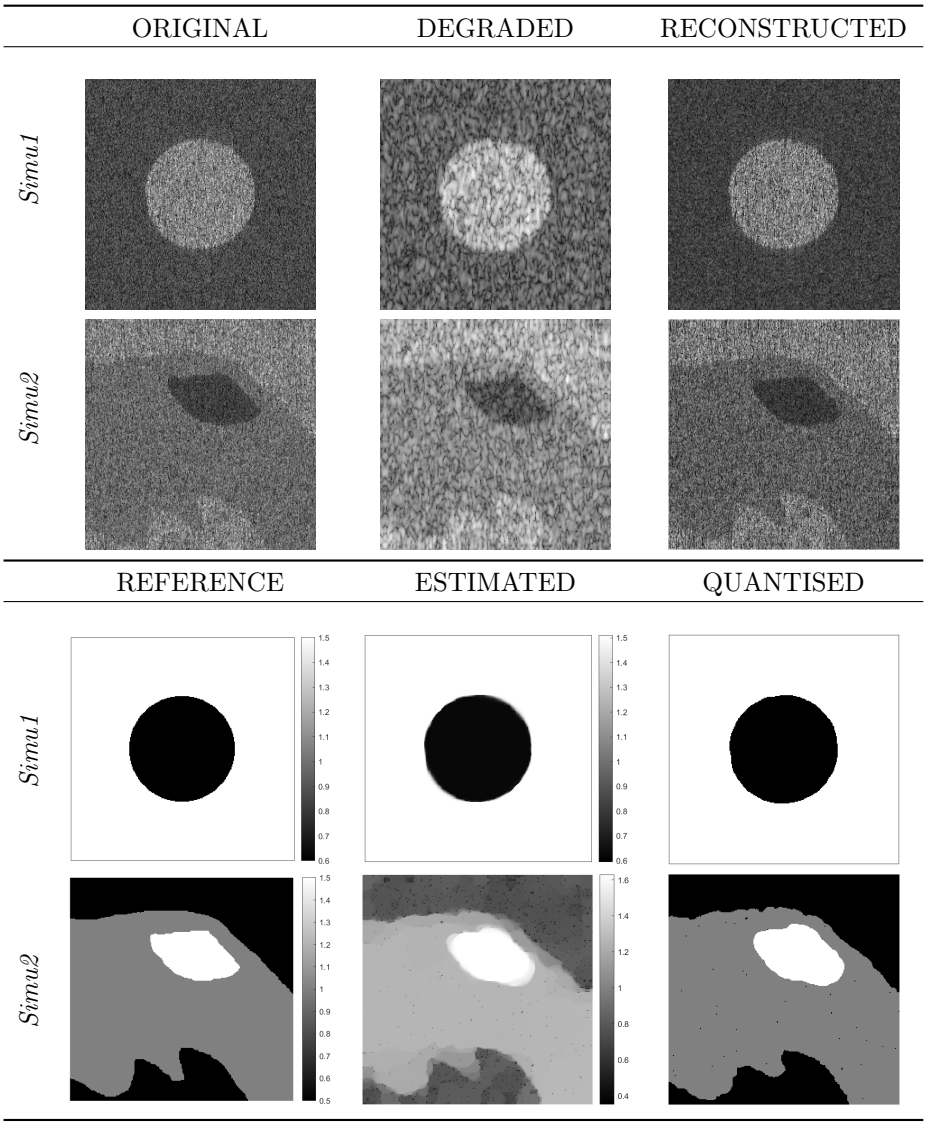


Fig. 2: First and Second lines: B–mode of Simu1 and Simu2. The B–mode image is the most common type of ultrasound image, displaying the acoustic impedance of a 2-dimensional cross section of the considered tissue. All images are presented in the same scale [0,1]. Third and Fourth lines: Segmentation of the shape parameter for *Simu1* and *Simu2*: reference p, estimated $\hat{p}$ and quantised $\bar{p}$.

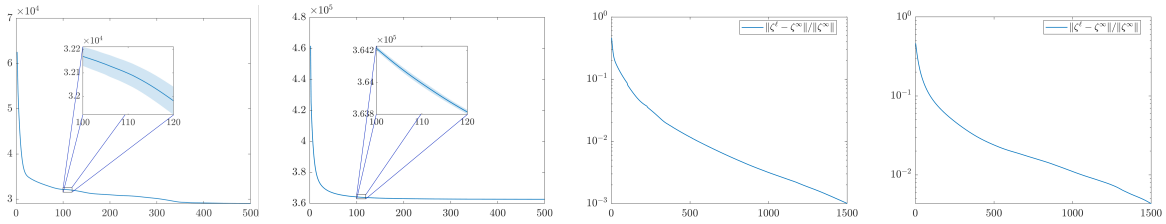
[9] Ghilli, D., Kunisch, K.: On monotone and primal-dual active set schemes for ℓ_p-type problems, $p \in (0, 1]$. Computational Optimization and Applications **72**, 45–85 (2019)

[10] Hintermüller, M., Wu, T.: Nonconvex TVq-models in image restoration: analysis and a Trust-Region regularization-based superlinearly convergent solver. SIAM Journal on Imaging Sciences **6**, 1385–1415 (2013)

[11] Lorenz, D., Resmerita, E.: Flexible sparse regularization. Inverse Problems **33** (2016)

[12] Afonso, M., Sanches, J.M.: Adaptive order non-convex lp-norm regularization in image restoration. Journal of Physics: Conference Series **904**(1), 012016 (2017)

[13] Blomgren, P., Chan, T.F., Mulet, P., Wong, C.K.: Total variation image restoration:



(a) (b) (c) (d)

Fig. 3: Decay of the objective value along 500 iterations for *Simu1* (a) and *Simu2* (b). We considered ten random sampling for p^0 and β^0. The continuous line in the plot represents the mean objective value at each iteration, and the shaded area, highlighted in the zoomed region at the centre spanning over 20 iterations, corresponds to the confidence interval related to the standard deviation. Logarithmic plot of the relative distance from the iterates ζ^ℓ to the solution ζ^∞ over 1500 iterations for *Simu1* (c) and *Simu2* (d).

Numerical methods and extensions. IEEE International Conference on Image Processing (1997)

[14] Chen, Y., Levine, S., Rao, M.: Variable exponent, linear growth functionals in image restoration. SIAM Journal on Applied Mathematics **66**, 1383–1406 (2006)

[15] Lanza, A., Morigi, S., Pragliola, M., Sgallari, F.: Space-variant generalised Gaussian regularisation for image restoration. Computer Methods in Biomechanics and Biomedical Engineering: Imaging and Visualization **7**, 1–14 (2018)

[16] Lazzaretti, M., Calatroni, L., Estatico, C.: Modular-proximal gradient algorithms in variable exponent lebesgue spaces. SIAM Journal on Scientific Computing **44**(6), 3463–3489 (2022)

[17] Corbineau, M.-C., Kouamé, D., Chouzenoux, E., Tourneret, J.-Y., Pesquet, J.-C.: Preconditioned P-ULA for joint deconvolution-segmentation of ultrasound images. IEEE Signal Processing Letters **26**(10), 1456–1460 (2019)

[18] Do, M.N., Vetterli, M.: Wavelet-based texture retrieval using generalized Gaussian density and Kullback-Leibler distance. IEEE Transactions on Image Processing **11**(2), 146–158 (2002)

[19] Zhao, N., Basarab, A., Kouamé, D., Tourneret, J.-Y.: Joint segmentation and deconvolution of ultrasound images using a hierarchical Bayesian model based on generalized Gaussian priors. IEEE Transactions on Image Processing **25**(8), 3736–3750 (2016)

[20] Hildreth, C.: A quadratic programming procedure. Naval Research Logistics Quarterly **4**(1), 79–85 (1957)

[21] Tseng, P.: Convergence of a block coordinate descent method for nondifferentiable minimization. Journal of Optimization Theory and Applications **109**, 475–494 (2001)

[22] Combettes, P.L., Pesquet, J.-C.: Proximal splitting methods in signal processing, pp. 185–212. Springer, New York, NY (2011)

[23] Combettes, P.L., Pesquet, J.-C.: Fixed point strategies in data science. IEEE Transactions on Signal Processing **69**, 3878–3905 (2021)

[24] Attouch, H., Bolte, J., Redont, P., Soubeyran, A.: Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the Kurdyka-Łojasiewicz inequality. Mathematics of Operations Research **35**(2), 438–457 (2010)

[25] Bolte, J., Sabach, S., Teboulle, M.: Proximal alternating linearized minimization for nonconvex and nonsmooth problems. Mathematical Programming **146**(1-2), 459–494 (2014)

[26] Pock, T., Sabach, S.: Inertial proximal alternating linearized minimization (iPALM) for nonconvex and nonsmooth problems. SIAM Journal on Imaging Sciences **9**(4), 1756–1787 (2016)

[27] Hertrich, J., Steidl, G.: Inertial stochastic palm and applications in machine learning. Sampling Theory, Signal Processing, and Data Analysis **20**(1), 4 (2022)

[28] Foare, M., Pustelnik, N., Condat, L.: Semi-Linearized proximal alternating minimization for a discrete Mumford–Shah model. IEEE Transactions on Image Processing **29**, 2176–2189 (2020)

[29] Nikolova, M., Tan, P.: Alternating proximal gradient descent for nonconvex regularised problems with multiconvex coupling terms (2017). https://hal.archives-ouvertes.fr/hal-01492846

[30] Tan, P., Pierre, F., Nikolova, M.: Inertial alternating generalized forward–backward splitting for image colorization. Journal of Mathematical Imaging and Vision **61**, 672–690 (2019)

[31] Censor, Y., Lent, A.: Optimization of "log x" entropy over linear equality constraints. Siam Journal on Control and Optimization **25**, 921–933 (1987)

[32] Chouzenoux, E., Pesquet, J.-C., Repetti, A.: A block coordinate variable metric forward–backward algorithm. Journal of Global Optimization, 1–29 (2016)

[33] Bonettini, S., Prato, M., Rebegoldi, S.: A block coordinate variable metric linesearch based proximal gradient method. Computational Optimization and Applications (2018)

[34] Repetti, A., Wiaux, Y.: Variable metric forward-backward algorithm for composite minimization problems. SIAM Journal on Optimization **31**(2), 1215–1241 (2021)

[35] Bonettini, S., Porta, F., Prato, M., Rebegoldi, S., Ruggiero, V., Zanni, L.: Recent Advances in Variable Metric First-Order Methods, pp. 1–31. Springer, Cham (2019)

[36] Hien, L.T.K., Phan, D.N., Gillis, N.: An inertial block majorization minimization framework for nonsmooth nonconvex optimization. J. Mach. Learn. Res. **24**, 18–11841 (2020)

[37] Attouch, H., Bolte, J., Svaiter, B.F.: Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods. Mathematical Programming, Series A **137**(1), 91–124 (2011)

[38] Chaâri, L., Pesquet, J.-C., Tourneret, J.-Y., Ciuciu, P., Benazza-Benyahia, A.: A hierarchical bayesian model for frame representation. In: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 4086–4089 (2010). https://doi.org/10.1109/ICASSP.2010.5495737

[39] Jeffreys, H.: An invariant form for the prior probability in estimation problems. Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences **186**(1007), 453–461 (1946). Accessed 2024-02-07

[40] Cai, X., Chan, R., Nikolova, M., Zeng, T.: A three-stage approach for segmenting degraded color images: Smoothing, Lifting and Thresholding (SLaT). Journal of Scientific Computing **72**, 1313–1332 (2017)

[41] Cai, X., Chan, R., Schönlieb, C.-B., Steidl, G., Zeng, T.: Linkage between piecewise constant Mumford–Shah model and Rudin–Osher–Fatemi model and its virtue in image segmentation. SIAM Journal on Scientific Computing **41**(6), 1310–1340 (2019)

[42] Cai, X., Chan, R., Zeng, T.: A two-stage image segmentation method using a convex variant of the Mumford–Shah model and thresholding. SIAM Journal on Imaging Sciences **6**(1), 368–390 (2013)

[43] Chambolle, A., Cremers, D., Pock, T.: A convex approach to minimal partitions. SIAM Journal on Imaging Sciences **5**, 1113–1158 (2012)

[44] Chan, R., Yang, H., Zeng, T.: A two-stage image segmentation method for blurry images with Poisson or multiplicative Gamma noise. SIAM Journal on Imaging Sciences **7**(1), 98–127 (2014)

[45] Pascal, B., Vaiter, S., Pustelnik, N., Abry, P.: Automated data-driven selection of the hyperparameters for total-variation based texture segmentation. Journal of Mathematical Imaging and Vision **63**, 923–952 (2021)

[46] Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. Communications on Pure and Applied Mathematics **42**, 577–685 (1989)

[47] Otsu, N.: A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics **9**(1), 62–66 (1979)

[48] Rockafellar, R.T., Wets, M., Wets, R.J.B.: Variational Analysis. Grundlehren der mathematischen Wissenschaften. Springer, Heidelberg (2009)

[49] Kurdyka, K.: On gradients of functions definable in o-minimal structures. Annales de l'Institut Fourier **48**(3), 769–783 (1998)

[50] Łojasiewicz, S.: Une propriété topologique des sous-ensembles analytiques réels. Equ. Derivees partielles, Paris 1962, Colloques internat. Centre nat. Rech. sci. 117, 87-89 (1963). (1963)

[51] Łojasiewicz, S.: Sur la géométrie semi- et sous- analytique. Annales de l'Institut Fourier **43**(5), 1575–1595 (1993)

[52] Chouzenoux, E., Pesquet, J.-C., Repetti, A.: Variable metric Forward-Backward algorithm for minimizing the sum of a differentiable function and a convex function. Journal of Optimization Theory and Applications **162**, 107–132 (2014)

[53] Bertsekas, D.P.: Nonlinear Programming, 2nd edn. Athena Scientific, Nashua (1999)

[54] Erdogan, H., Fessler, J.A.: Monotonic algorithms for transmission tomography. IEEE Transactions on Medical Imaging **18**(9), 801–814 (1999)

[55] Hunter, D., Lange, K.: A Tutorial on MM Algorithms. The American Statistician **58**, 30–37 (2004)

[56] Salzo, S.: The variable metric forward-backward splitting algorithm under mild differentiability assumptions. SIAM Journal on Optimization **27**(4), 2153–2181 (2017)

[57] Malitsky, Y., Mishchenko, K.: Adaptive gradient descent without descent. In: III, H.D., Singh, A. (eds.) Proceedings of the 37th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 119, pp. 6702–6712 (2020)

[58] Latafat, P., Themelis, A., Stella, L., Patrinos, P.: Adaptive proximal algorithms for convex optimization under local Lipschitz continuity of the gradient (2023)

[59] Charbonnier, P., Blanc-Féraud, L., Aubert, G., Barlaud, M.: Deterministic edge-preserving regularization in computed imaging. IEEE Transactions on Image Processing **6 2**, 298–311 (1997)

[60] Van Den Dries, L.: Tame Topology and O-minimal Structures. London Mathematical Society Lecture Note Series. Cambridge University Press, Cambridge (1998)

[61] Bolte, J., Daniilidis, A., Lewis, A., Shiota, M.: Clarke subgradients of stratifiable functions. SIAM Journal on Optimization **18**(2), 556–572 (2007)

[62] Gabrielov, A.: Complements of subanalytic sets and existential formulas for analytic functions. Inventiones mathematicae **125**, 1–12 (1996)

[63] Wilkie, A.: Model completeness results for expansions of the ordered field of real numbers by restricted Pfaffian functions and the exponential function. Journal of the American Mathematical Society **9**, 1051–1094

(1996)

[64] Van Den Dries, L., Macintyre, A., Marker, D.: Logarithmic-exponential power series. Journal of the London Mathematical Society **56**(3), 417–434 (1997)

[65] Van Den Dries, L., Speissegger, P.: The field of reals with multisummable series and the exponential function. Proceedings of The London Mathematical Society **81**, 513–565 (2000)

[66] Tougeron, J.: Sur les ensembles semi-analytiques avec conditions gevrey au bord. Annales Scientifiques De L Ecole Normale Superieure **27**, 173–208 (1994)

[67] Artin, E.: The Gamma Function. Courier Dover Publications, New York (2015)

[68] Wrench, J.W.: Concerning two series for the Gamma function. Mathematics of Computation **22**(103), 617–626 (1968)

[69] Andrews, G.E., Askey, R., Roy, R.: Special Functions. Encyclopedia of Mathematics and its Applications. Cambridge University Press, Cambridge (1999)

[70] Repetti, A., Wiaux, Y.: Variable metric forward-backward algorithm for composite minimization problems. SIAM Journal on Optimization **31**(2), 1215–1241 (2021)

[71] Schifano, E.D., Strawderman, R.L., Wells, M.T.: Majorization-Minimization algorithms for nonsmoothly penalized objective functions. Electronic Journal of Statistics **4**(none), 1258–1299 (2010)

[72] Combettes, P.L., Dũng, Bằng Công Vũ: Proximity for sums of composite functions. Journal of Mathematical Analysis and Applications **380**(2), 680–688 (2011)

[73] Condat, L.: A Primal–Dual splitting method for convex optimization involving lipschitzian, proximable and linear composite terms. Journal of Optimization Theory and Applications **158** (2013)

[74] Komodakis, N., Pesquet, J.-C.: Playing with duality: An overview of recent primal-dual approaches for solving large-scale optimization problems. IEEE Signal Processing Magazine **32**(6), 31–54 (2015)

[75] Vũ, B.C.: A splitting algorithm for dual monotone inclusions involving cocoercive operators. Advances in Computational Mathematics **38**, 667–681 (2013)

[76] Corless, R., Gonnet, G., Hare, D., Jeffrey, D., Knuth, D.: On the Lambert W function. Advances in Computational Mathematics **5**, 329–359 (1996)

[77] Jensen, J.A.: Simulation of advanced ultrasound systems using field ii. In: 2004 2nd IEEE International Symposium on Biomedical Imaging: Nano to Macro, pp. 636–6391 (2004)

[78] Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004)

[79] Robert, C., Elvira, V., Tawn, N., Wu, C.: Accelerating MCMC algorithms. Wiley Interdisciplinary Reviews: Computational Statistics **10** (2018)

[80] Pereyra, M.: Proximal Markov chain Monte Carlo algorithms. Statistics and Computing **26** (2013)

Supplementary Material: A Variational Approach for Joint Image Recovery and Feature Extraction Based on Spatially Varying Generalised Gaussian Models

Émilie Chouzenoux[1], Marie-Caroline Corbineau[2], Jean-Christophe Pesquet[1], Gabriele Scrivanti[1*]

[1]Université Paris-Saclay, Inria, CentraleSupélec, CVN, France.
[2]Research department, Preligens,France.

*Corresponding author(s). E-mail(s): gabriele.scrivanti@centralesupelec.fr;

1 Wavelet-based restoration

We illustrate the performance of our approach in the context of wavelet-based image restoration. The sought solution $x \in \mathbb{R}^n$ corresponds to the wavelet coefficients of a source image $\overline{\mathrm{x}}$, *i.e.* $\overline{\mathrm{x}} = Wx$ where $W : \mathbb{R}^n \to \mathbb{R}^n$ models the Db8 wavelet orthogonal transform with 4 decomposition levels [1]. We choose $\overline{\mathrm{x}} \in [0, 255]^n$ to be the image *Westconcord* illustrated in Figure 1(a) and set $H \in \mathbb{R}^{n\times n}$ as the blur operator modelling the convolution, using zero padding, with a 11×11 Gaussian Kernel with zero-mean and standard deviation $\sigma_{\mathrm{b}} \in \{1.5, 2\}$, which generates the blurred images $H\overline{x} \in \mathbb{R}^n$ illustrated in Figure 1(b) and Figure 1(e). Hence, in this example, the linear operator in (1) corresponds to $K = HW$. Finally, we generate four noisy observation $y \in \mathbb{R}^{n\times n}$ by applying model (1) using zero-mean Gaussian noise with standard deviation $\sigma = 5$ (Figure 1(c) and Figure 1(f)) and $\sigma = 10$ (Figure 1(d) and Figure 1(g)).

In this context we set $[a, b] = [10^{-3}, 10]$. We initialise the algorithm with $x^0 = W^\top y$ that is the wavelet decomposition of the degraded image, while $(p_i^0)_{1\leq i\leq n}$ is drawn from an i.i.d. uniform distribution in the range $[0.5, 1.5]$ and $(\beta_i^0)_{1\leq i\leq n}$ is drawn from an i.i.d. Gaussian distribution with mean μ_β and unit standard deviation, and the choice $\mu_\beta = 5$ will be soon discussed. No preconditioning is considered in this example, that is $A_\ell = \mathbb{I}_m$ for every $\ell = 1, 2, \ldots$ until convergence.

For comparisons, we consider the following models.

- The LASSO regularisation model (LASSO)

$$\underset{x\in\mathbb{R}^n}{\text{minimise}} \quad \frac{1}{2\sigma^2}\|Kx - y\|^2 + \xi\|x\|_1, \qquad (1)$$

  where $\xi > 0$ is a regularisation parameter that is chosen through a grid search for PSNR and SSIM. We address (1) by means of a Forward-Backward algorithm.
- The Adaptive Order Non-Convex ℓ_p-norm regularization model (AONCLP) from [2]

$$\underset{x\in\mathbb{R}^n}{\text{minimise}} \quad \frac{1}{2\sigma^2}\|Kx - y\|^2 + \xi \sum_{i=1}^{n} |x_i|^{\varpi(x_i)} \qquad (2)$$

  with $\varpi : \mathbb{R} \longrightarrow \mathbb{R}$ defined as $t \mapsto 1/\left(1 + \exp(-0.51(\log(|t| - 1))\right)$ and $\xi > 0$ is a regularisation parameter that is chosen through a grid search for PSNR and SSIM. We address (2) by means of a Split Augmented Lagrangian Shrinkage Algorithm as described in [2].

- The Generalised Multivariate Exponential Power (GMEP) regularisation model from [3]:

$$\underset{x\in\mathbb{R}^n}{\text{minimise}} \quad \frac{1}{2\sigma^2}\|Kx-y\|^2 + \xi\sum_{m=1}^{M}\sum_{c=1}^{C_m}(\rho_m^{1/\chi_m}x_{m,c}^2+\delta)^{\chi_m}, \quad (3)$$

where M is the number of wavelet subbands (in our case $M = 13$), C_m the number of elements of the m-th subband ($\sum_{m=1}^{M} C_m = n$), $\delta = 10^{-6}$, and the parameters $\{\rho_1,\dots,\rho_M\}$ and $\{\chi_1,\dots,\chi_M\}$ are obtained with a maximum likelihood estimator, run on each subband, of a Wiener-filtered version of the observed image y. Finally, $\xi > 0$ is a regularisation parameter that is chosen through a grid search for PSNR and SSIM. Problem (3) is addressed by means of the Majorize-Minimize Memory Gradient (3MG) algorithm [4, 5].

In Figure 2 we illustrate the restored images obtained by the four regularisation models on the four degraded versions of the *Westconcord* image. The proposed model generates solutions which are competitive with the ones obtained by GMEP, with less artefacts and less saturated values than the ones obtained with LASSO and AONCLP. The good performance of our model is also supported by the quantitative evaluation illustrated in Table 1 in terms of PSNR and SSIM.

For what concerns the convergence properties of the algorithm, Figure 3 illustrates the decay of the objective values over 500 iterations for the different combinations of $(\sigma_{\mathtt{b}}, \sigma)$. To study the stability with respect to the initialisations, we considered 10 random sampling for p^0 and β^0: the continuous lines in the plots represent the mean objective values at each iteration and the shaded area, highlighted in the zoomed region at the centre of the plot, corresponds to the confidence interval related to the standard deviation of the objective value at each iteration. Figure 4 illustrates on a logarithmic scale the relative distance from the iterates to the solution $\|\zeta^\ell - \zeta^\infty\|/\|\zeta^\infty\|$, showing the convergence of our algorithm.

Figure 5, Figure 6 and Figure 7 illustrate the procedure that we adopted to identify an optimal value of parameter μ_β in combination with λ and ζ, for the reconstruction of the *Westconcord* image: in these figures we show in different forms the PSNR and SSIM scores for all the combinations of $(\lambda,\zeta) \in \{0.01, 0.1, 1, 10, 100, 1000\}^2$ and $\mu_\beta \in \{0, 2, 5, 7, 10\}$. In particular, Figure 5 and Figure 6 show the PSNR and SSIM scores respectively in the forms of matrices, where each row corresponds to a value of λ and each column to a value of ζ. For every combination $(\sigma_{\mathtt{b}}, \sigma)$, we used the same colour scale for all the values of μ_β, in order to highlight the variability of the results with respect to the whole triplet $(\lambda, \zeta, \mu_\beta)$: it can be seen that the parameter having the highest impact on the quality of the solution is μ_β, while the impact of λ and β is weaker, especially on the SSIM, in spite of the fact that the choice for those values spans over 6 orders of magnitude. This is further highlighted in Figure 7, where the data are illustrated in the form of boxplots (PSNR scores in the left column and SSIM scores in the right column): from these plots we also infer that $\mu_\beta = 5$ is a stable and optimal choice.

To conclude, we illustrate the performance of our method on a second group of test images: *Galaxy* which is a test image included in Matlab, *Buildings* from [6], and *Skyscrapers* from [7]. In this case, we set the noise level $\sigma = 10$ and blur level $\sigma_{\mathtt{b}} \in \{1.5, 2\}$. Figure 8 displays the degraded images, along with the reconstructions obtained with LASSO, AONCLP, GMEP, and our model, while Tables 2 and 3 report the PSNR and SSIM scores, which again advocates for the use of the proposed model. We also report for each method and for each image, the average computational time over 10 simulations required to meet the stopping criteria $\|\zeta^{\ell+1} - \zeta^\ell\|/\|\zeta^\ell\| < 10^{-4}$ and $|\theta(\zeta^{\ell+1}) - \theta(\zeta^\ell)|/|\theta(\zeta^\ell)| < 10^{-4}$. Simulations were run on Matlab 2021b on an Intel Xeon Gold 6230 CPU 2.10GHz. A Matlab profiling of the simulations showed that the most time-consuming step is the update in (87), which is based on Newton's method. A possible solution to accelerate this step could consist in pre-computing a sufficiently large sample of solutions and replacing the computation with an indexing operation with respect to a lookup table.

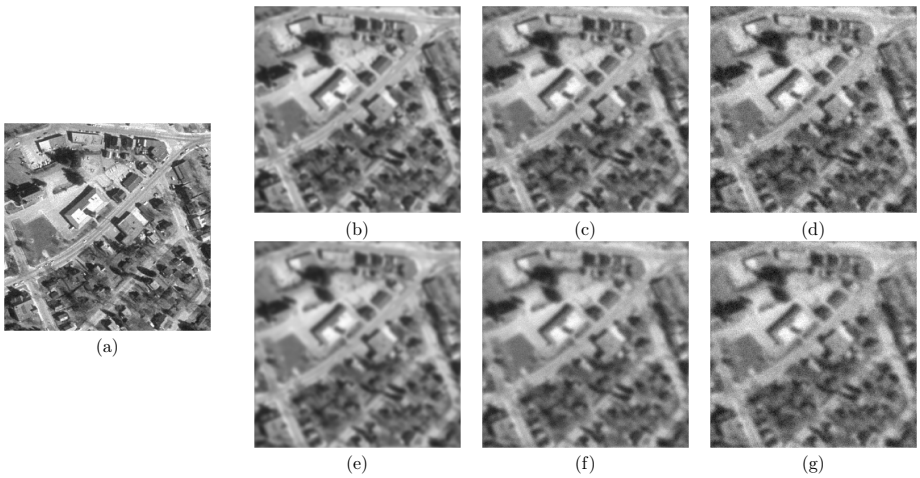


Fig. 1: *Westconcord* image: (a) original image, (b) blurred image with $\sigma_{\mathtt{b}} = 1.5$, (c) degraded image with $(\sigma_{\mathtt{b}}, \sigma) = (1.5, 5)$ (PSNR = 22.49, SSIM = 0.55), (d) degraded image with $(\sigma_{\mathtt{b}}, \sigma) = (1.5, 10)$ (PSNR = 22.18, SSIM = 0.49); (e) blurred image with $\sigma_{\mathtt{b}} = 2$, (f) degraded image with $(\sigma_{\mathtt{b}}, \sigma) = (2, 5)$ (PSNR = 21.07, SSIM = 0.43), (g) degraded image with $(\sigma_{\mathtt{b}}, \sigma) = (2, 10)$ (PSNR = 21.01, SSIM = 0.38).

	$\sigma_{\mathtt{b}} = 1.5$				$\sigma_{\mathtt{b}} = 2$			
	$\sigma = 5$		$\sigma = 10$		$\sigma = 5$		$\sigma = 10$	
METHOD	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
LASSO	24.48	0.66	23.33	0.58	23.65	0.59	22.62	0.50
AONCLP	24.32	0.64	23.29	0.56	22.87	0.54	22.56	0.49
GMEP	24.61	**0.68**	23.91	**0.63**	22.94	0.57	22.70	**0.54**
OURS	**24.83**	**0.68**	**24.27**	0.62	**24.41**	**0.60**	**24.42**	0.53

Table 1: PSNR and SSIM scores for *Westconcord* example.

2 Parameter Tuning for joint deblurring/segmentation

Figures 9-12 illustrate the procedure that we adopted to identify an optimal value of parameter μ_β in combination with λ and ζ, for the reconstruction of *Simu1* and *Simu2*. In these figures, for all the combinations of $(\lambda, \zeta) \in \{0.01, 0.1, 1, 10, 100, 1000\}^2$ and $\mu_\beta \in \{-5, -2, 0, 2, 5\}$ for *Simu1* and $\mu_\beta \in \{0, 2, 3, 4, 5\}$ for *Simu2*, we show in different forms the PSNR and SSIM scores for the reconstruction of the images, while for the shape parameter we rely on the RMSE the OA of the segmentation. In particular, Figures 9 and 10 illustrate the results in the form of matrices, where each row corresponds to a value of λ and each column to a value of ζ. Figures11 and 12 illustrate the results in the form of boxplots. The values of the PSNR and SSIM are mostly affected by the value of μ_β, while less variability is related to the choice of λ and ζ. On the other side, RMSE and OA depend on a combination of the three parameters (in particular on a combination of μ_β and λ). In conclusion, for the choice of μ_β, we relied on the value that could provide us with the best trade-off between high PSNR/SSIM, high OA and low RMSE. For *Simu1*, this choice is clearly represented by $\mu_\beta = 0$, as with this value we manage to optimise all the metrics. For *Simu2*, the best trade-off is represented by $\mu_\beta = 4$, as $\mu_\beta = 5$ yields high PSNR/SSIM levels on one side, but unsatisfactory OA and RMSE on the other.

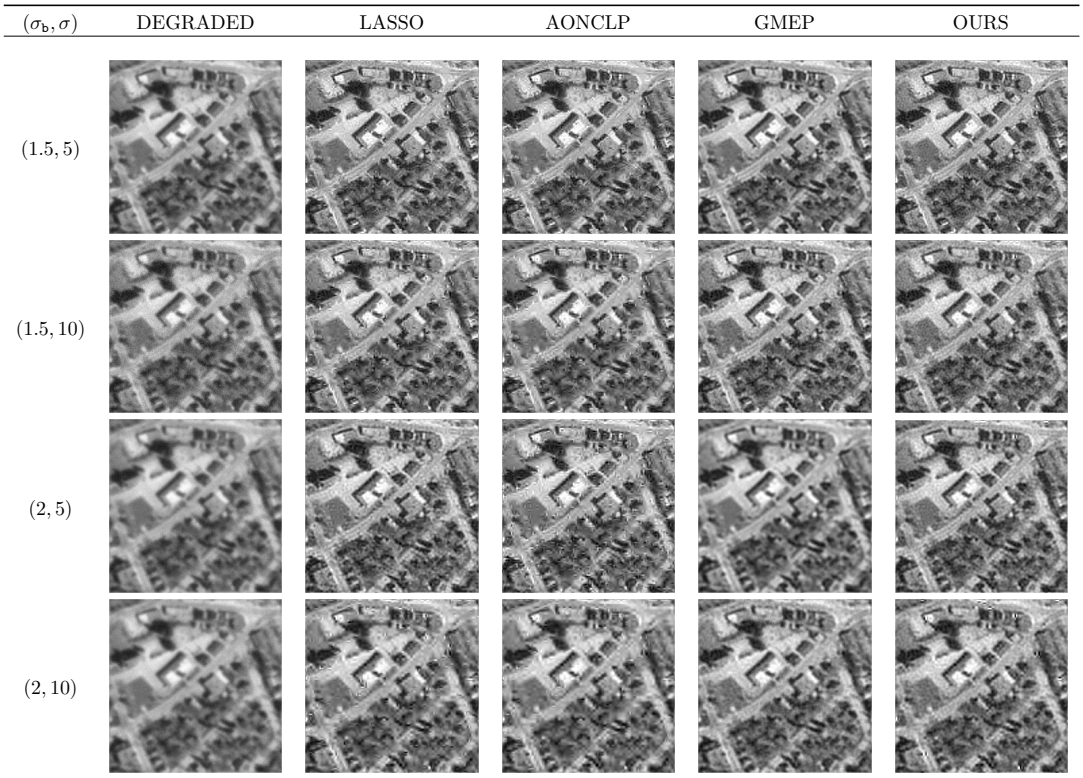


Fig. 2: Estimated solutions for *Westconcord* image. First column: Degraded image. Second column: LASSO regularisation. Third column: AONCLP regularisation. Fourth column: GMEP regularisation. Fifth column: Proposed regularisation.

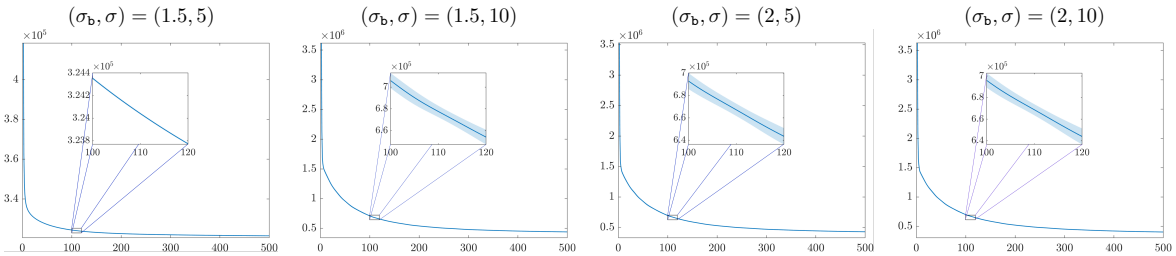


Fig. 3: Decay of the objective value along 500 iterations for *Westconcord* example, using different combinations of (σ_b, σ). We considered 10 random sampling for p^0 and β^0. The continuous line in the plot represents the mean objective value at each iteration and the shaded area, highlighted in the zoomed region at the centre of the plot spanning over 20 iterations, corresponds to the confidence interval related to the standard deviation.

References

[1] Daubechies, I., Sweldens, W.: Factoring wavelet transforms into lifting steps. Journal of Fourier Analysis and Applications **4**, 247–269 (1998)

[2] Afonso, M., Sanches, J.M.: Adaptive order non-convex lp-norm regularization in image restoration. Journal of Physics: Conference Series **904**(1), 012016 (2017)

[3] Marnissi, Y., Benazza-Benyahia, A.,

	Galaxy			*Buildings*			*Skyscrapers*		
METHOD	PSNR	SSIM	TIME	PSNR	SSIM	TIME	PSNR	SSIM	TIME
LASSO	25.25	0.34	42.8	24.35	0.38	34.2	21.57	0.32	30.8
AONCLP	25.15	0.32	21.4	24.08	0.37	24.5	21.70	0.31	42.4
GMEP	**25.75**	0.37	12.7	24.45	0.38	14.1	21.43	0.33	13.78
OURS	25.12	**0.37**	219.7	**25.23**	**0.40**	441.5	**21.88**	**0.33**	405.9

Table 2: PSNR scores, SSIM scores, and computational times (in seconds) for $(\sigma_{\mathtt{b}}, \sigma) = (1.5, 10)$.

	Galaxy			*Buildings*			*Skyscrapers*		
METHOD	PSNR	SSIM	TIME	PSNR	SSIM	TIME	PSNR	SSIM	TIME
LASSO	24.72	0.28	43.7	23.12	0.29	35.1	20.91	0.24	35.1
AONCLP	24.66	0.28	15.0	23.20	0.29	27.4	20.79	0.24	27.4
GMEP	24.98	0.30	12.2	23.30	0.30	10.9	20.35	0.25	10.9
OURS	**25.08**	**0.32**	329.1	**24.92**	**0.30**	766.4	**21.53**	**0.25**	379.1

Table 3: PSNR scores, SSIM scores and computational times (in seconds) for $(\sigma_{\mathtt{b}}, \sigma) = (2, 10)$.

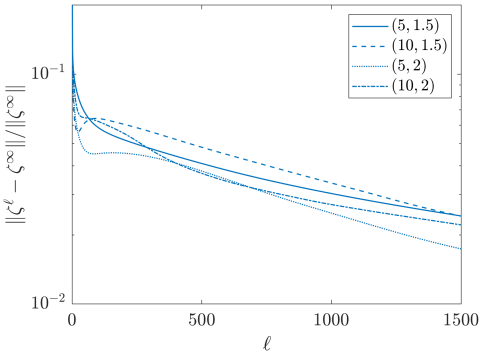


Fig. 4: Convergence plots for four settings of (σ_b, σ), in terms of distance from the iterates ζ^ℓ to the solution ζ^∞, for *Westconcord* example.

Chouzenoux, E., Pesquet, J.-C.: Generalized multivariate exponential power prior for wavelet-based multichannel image restoration. In: 2013 IEEE International Conference on Image Processing, pp. 2402–2406 (2013). https://doi.org/10.1109/ICIP.2013.6738495

[4] Chouzenoux, E., Idier, J., Moussaoui, S.: A majorize–minimize strategy for subspace optimization applied to image restoration. IEEE Transactions on Image Processing **20**(6), 1517–1528 (2011)

[5] Chouzenoux, E., Jezierska, A., Pesquet, J.-C., Talbot, H.: A majorize-minimize subspace approach for $\ell_2 - \ell_0$ image regularization. SIAM Journal on Imaging Sciences **6**(1), 563–591 (2013)

[6] Caesar, H., Uijlings, J., Ferrari, V.: Joint calibration for semantic segmentation. In: British Machine Vision Conference 2015, p. 29 (2015). BMVA Press

[7] Lanza, A., Morigi, S., Pragliola, M., Sgallari, F.: Space-variant generalised Gaussian regularisation for image restoration. Computer Methods in Biomechanics and Biomedical Engineering: Imaging and Visualization **7**, 1–14 (2018)

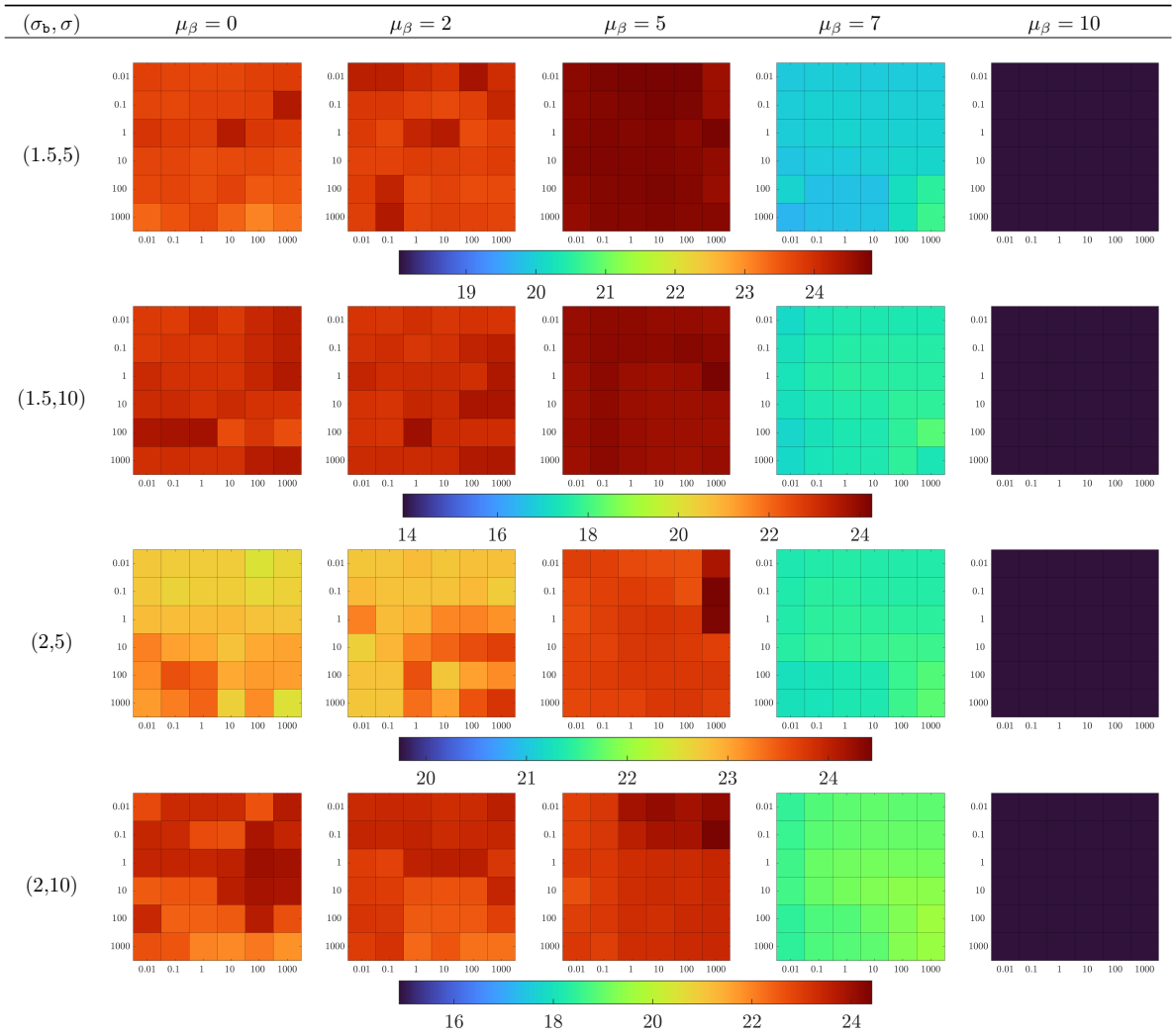


Fig. 5: Illustration of the PSNR scores obtained with a grid search on parameters (λ, ζ) for *Westconcord* for 5 different choices for parameter μ_β. In each matrix, rows correspond to the values for λ and columns to the values for ζ. For each parameter we chose 6 values in a logarithmic scale between 10^{-2} and 10^3.

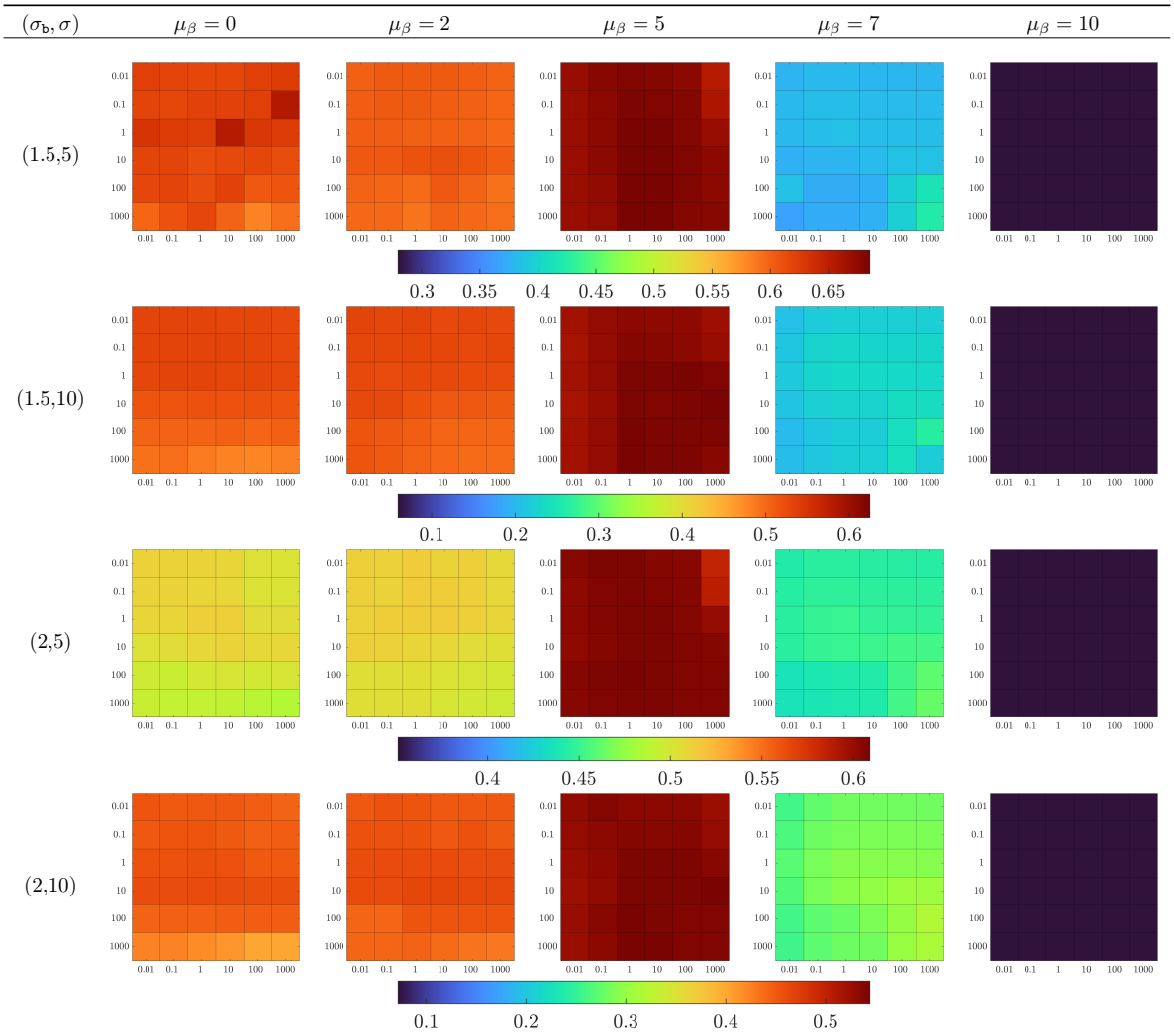


Fig. 6: Illustration of the SSIM scores obtained with a grid search on parameters (λ, ζ) for *Westconcord* for 5 different choices for parameter μ_β. In each matrix, rows correspond to the values for λ and columns to the values for ζ. For each parameter we chose 6 values in a logarithmic scale between 10^{-2} and 10^3.

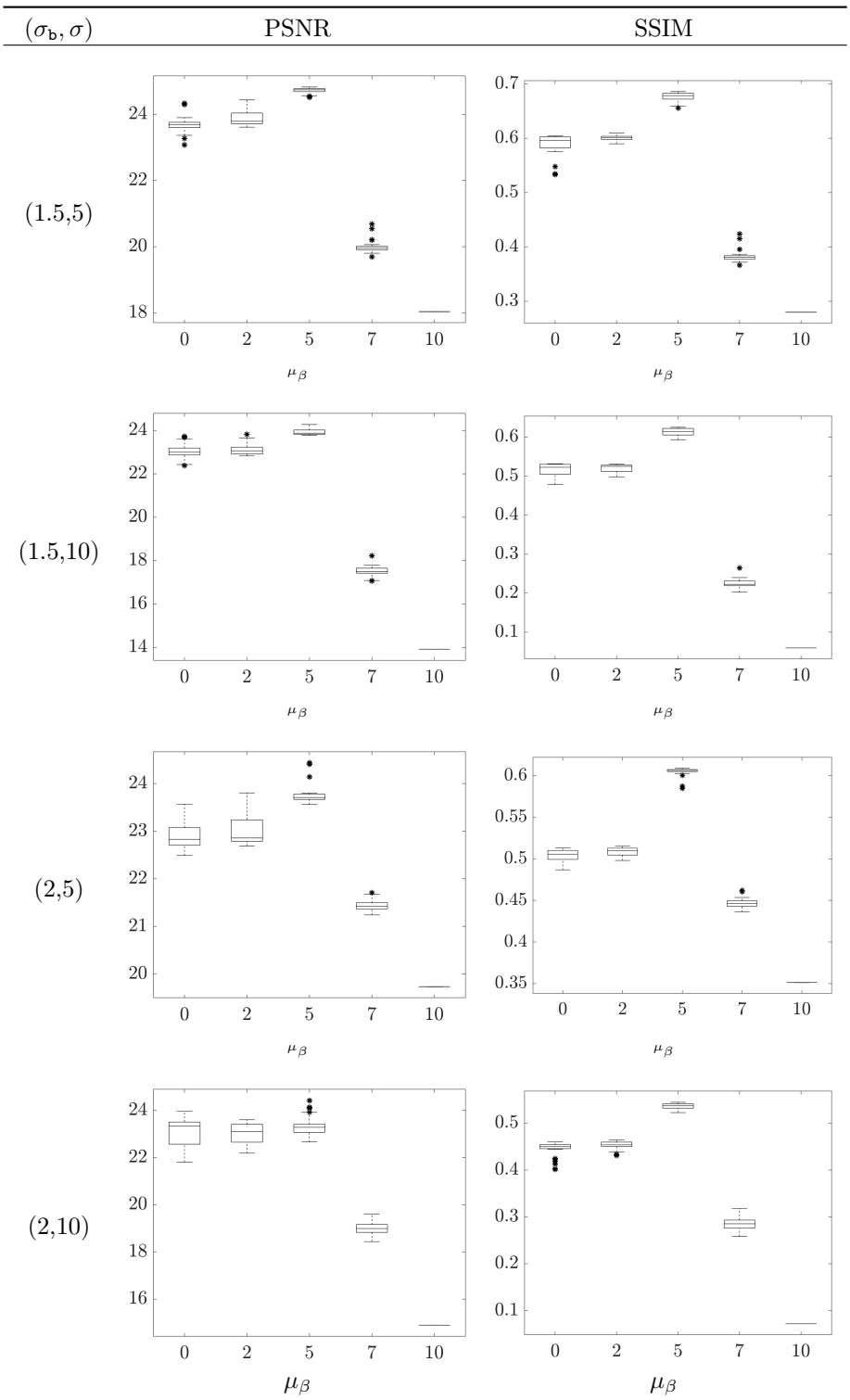


Fig. 7: Boxplots of the PSNR scores (left column) and SSIM scores (right column) obtained with a grid search on parameters (λ, ζ) for *Westconcord* for 5 different choices for parameter μ_β.The rows corresponds to different combinations of $(\sigma_{\mathtt{b}}, \sigma)$ with $\sigma_{\mathtt{b}} \in \{1.5, 2\}$ and $\sigma \in \{5, 10\}$.

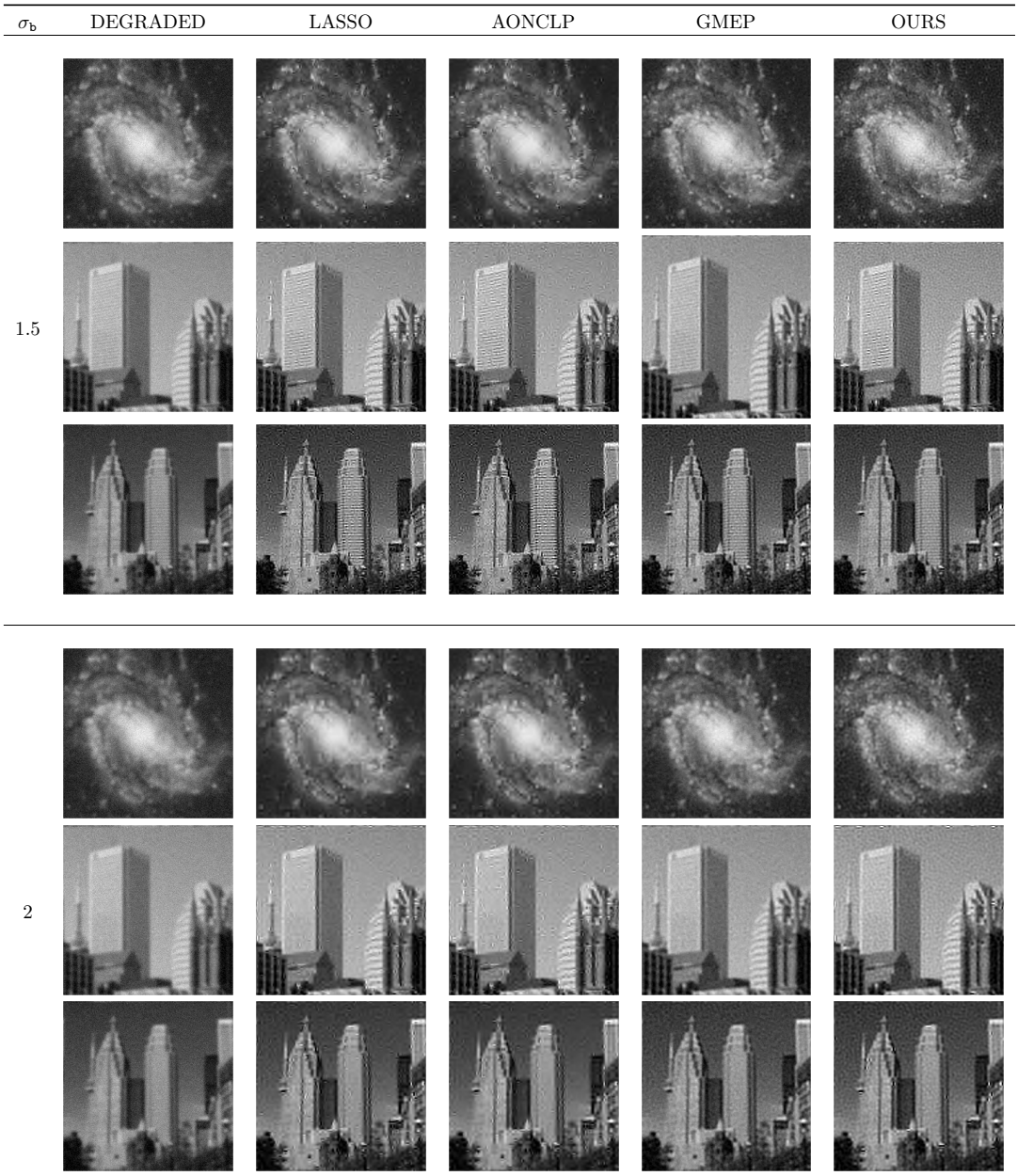


Fig. 8: Estimated solutions for *Galaxy*, *Buildings* and *Skyscrapers* and $\sigma = 10$. First column: Degraded image. Second column: LASSO regularisation. Third column: AONCLP regularisation. Fourth column: GMEP regularisation. Fifth column: Proposed regularisation. First line: $\sigma_b = 1.5$, Second line: $\sigma_b = 2$

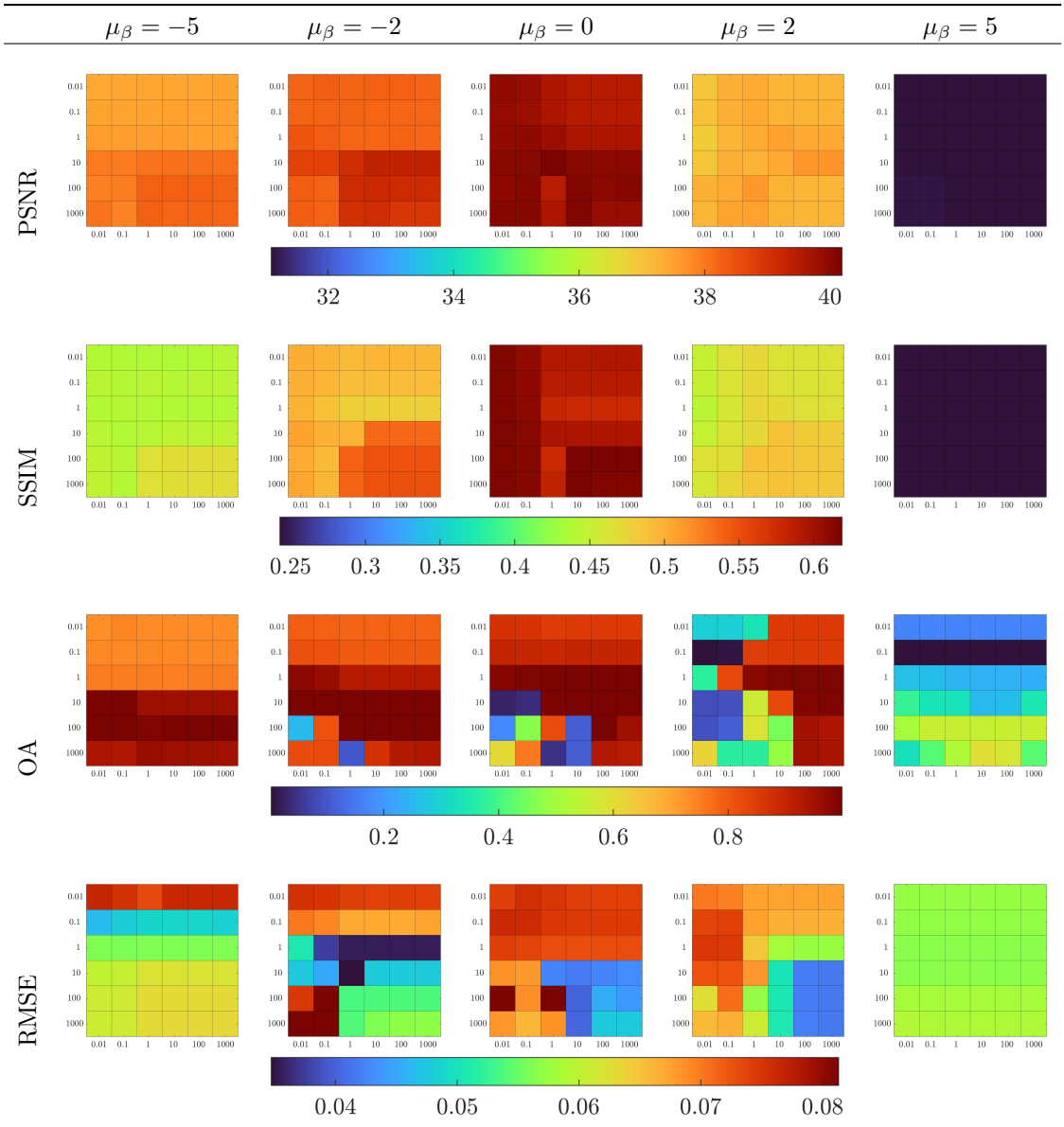


Fig. 9: Illustration of the PSNR, SSIM, OA and RMSE values obtained with a grid search on parameters (λ, ζ) for *Simu1* and 5 different choices for parameter μ_β. In each matrix, rows correspond to the values for λ and columns to the values for ζ. For each parameter we chose 6 values in a logarithmic scale between 10^{-2} and 10^3.

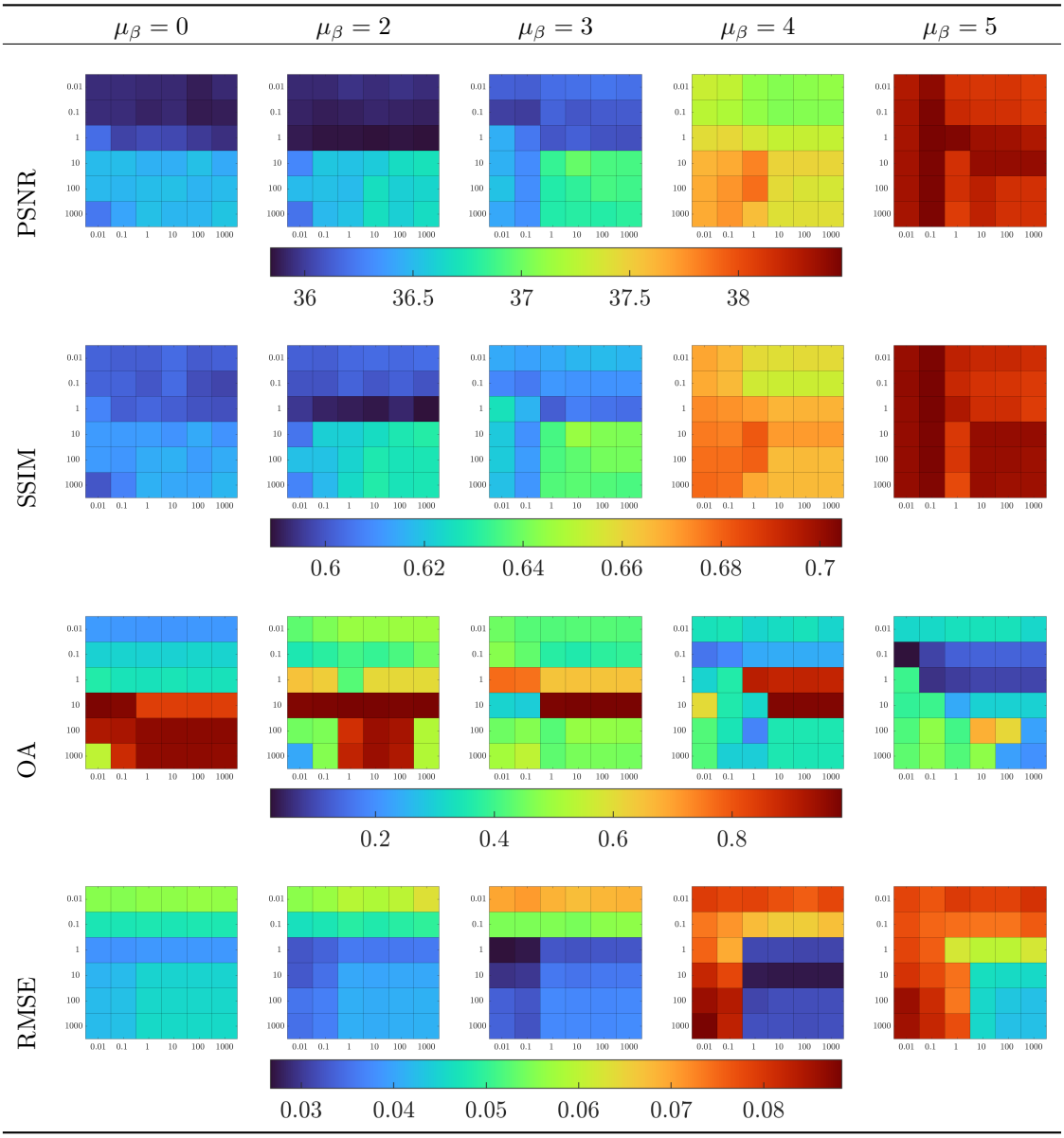


Fig. 10: Illustration of the PSNR, SSIM, OA and RMSE values obtained with a grid search on parameters (λ, ζ) for *Simu2* and 5 different choices for parameter μ_β. In each matrix, rows correspond to the values for λ and columns to the values for ζ. For each parameter we chose 6 values in a logarithmic scale between 10^{-2} and 10^3.

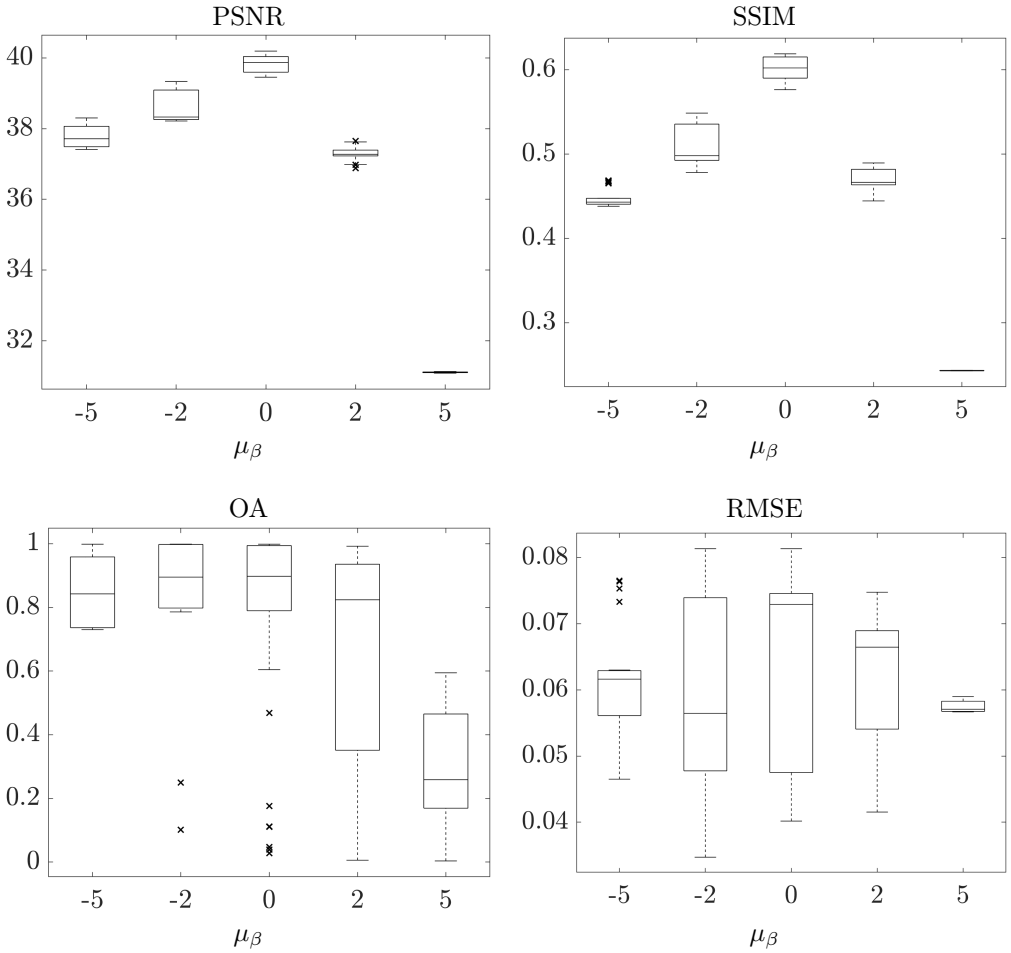


Fig. 11: Boxplots of the PSNR, SSIM, OA and RMSE values obtained with a grid search on parameters (λ, ζ) for *Simu1* and 5 different choices for parameter μ_β.

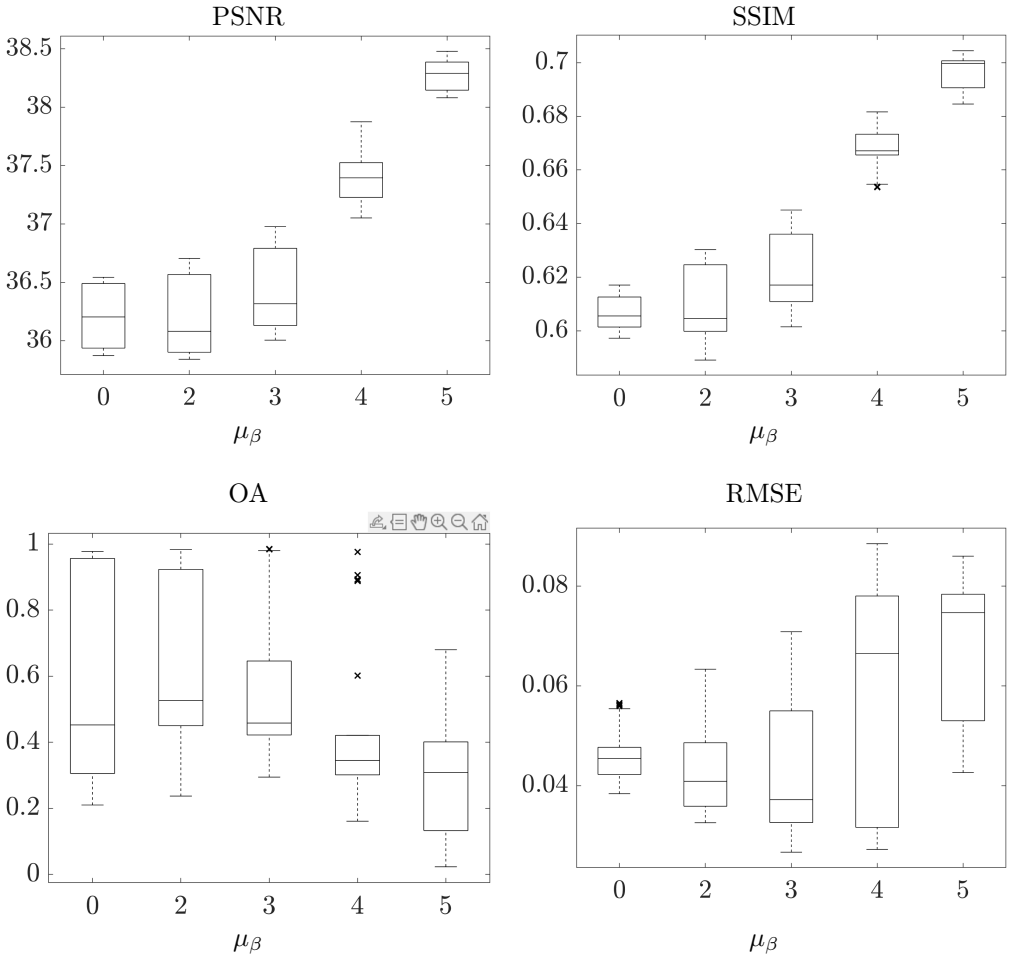


Fig. 12: Boxplots of the PSNR, SSIM, OA and RMSE values obtained with a grid search on parameters (λ, ζ) for *Simu2* and 5 different choices for parameter μ_β.