Minimal partitions and image classification using a gradient-free perimeter approximation

Samuel Amstutz, A.A. Novotny, Nicolas van Goethem

MINIMAL PARTITIONS AND IMAGE CLASSIFICATION USING A GRADIENT-FREE PERIMETER APPROXIMATION

S. AMSTUTZ, A. NOVOTNY, AND N. VAN GOETHEM

Abstract. In this paper we propose a new optimal partition algorithm and show applications to multilabel image classification problems. Possibly noisy and blurred greyscale and color images can be processed, with or without automatic update of the labels. Regularization is performed by a non standard approximation of the total interface length, which involves a system of uncoupled linear partial differential equations and shows Γ-converge properties in the set of characteristic functions. These good mathematical properties are recovered in the numerical convergence scheme.

1. Introduction

Image processing is a huge field of research ranging from code-based algorithms to advanced mathematical tools, and where mainly two classes of problems are addressed. The first one is image *restoration* whose aim is to remove all effects responsible for an image degradation: noise, blur, missing parts, etc. Another family of problems can be referred to as image *segmentation*, where the constituents of a given image (damaged or not) are identified: it can be different colors, intensities or texture regions, but also to provide a partition in geometric vs. texture components. Image *classification* is a particular form of image segmentation, where it is emphasized that the image characteristics are sought within a prescribed number of components, called labels.

The standard greyscale image processing problem can be stated as restore and/or segment $f = A\bar{u} + \nu$ where $f : \Omega \mapsto [0, 1]$ is the observed image, $\bar{u}$ its idealized version (the undamaged image), A is a known or unknown mask operator (blur kernel or a projection operator away from the missing parts of $\bar{u}$), and ν is the noise. According to the application, one wishes to find a u which is either a continuous restoration of f or a segmented version of f.

Image restoration and image segmentation are hard problems per se, mainly due to the fact that the minimization problem $\min_{u\in H(\Omega)} J(u) := \|Au - f\|_{H(\Omega)}$ is in general ill-posed (w.r.t. f if A is known and also w.r.t. A if the mask is unknown) in the sense that small perturbations in the data may produce unbounded variations in the solution, but also because it is not necessarily a convex problem, especially when the set $H(\Omega)$ contains discrete levels. Of course, simultaneous segmentation and restoration of a blurred and noisy image is even a harder task, in particular in the case of blind deconvolution. Let us emphasize that an objective assessment of segmentation algorithms is hardly found, essentially because there is no unique ground-truth classification of an image against which the output of an algorithm may be compared.

Several mathematical models coexist in the literature to provide as output a restoration and/or a segmentation of a given image. Let us just mention the Mumford-Shah [18, 1] and the TV-L^2 (the so-called Rudin-Osher-Fatemi model) and TV-L^1 functionals [19, 12]. It is today widely recognized [21, 10, 11] that in order to obtain a solution which is smooth enough while preserving the edges, one should consider the problem $\min_{u\in H(\Omega)} J(u) + \alpha|Du|(\Omega)$ where $|Du|(\Omega)$ is the total variation of u in Ω, with Du, its distributional derivative, i.e., a measure with a diffuse part identified with ∇u outside the edges and a concentrated part on the edges, and with α a weight on the total variation of the image.

In this paper we propose a novel mathematically-founded method to classify images. More precisely, our method assigns to each pixel of a given image a label, with a prescribed upper bound on the number of labels. Typically, each label corresponds to a grey or color level. The classification can be supervised, i.e., the intensity of the label is a fixed value, or unsupervised, meaning that there is an automatic update of the label intensity. In Fig. 1 the resulting classification of a color image is shown

2010 *Mathematics Subject Classification.* 49Q10,49Q20,49M25,35J05,35J25,65K10.

Key words and phrases. Image classification, denoising, deblurring, optimal partitions, perimeter approximation.

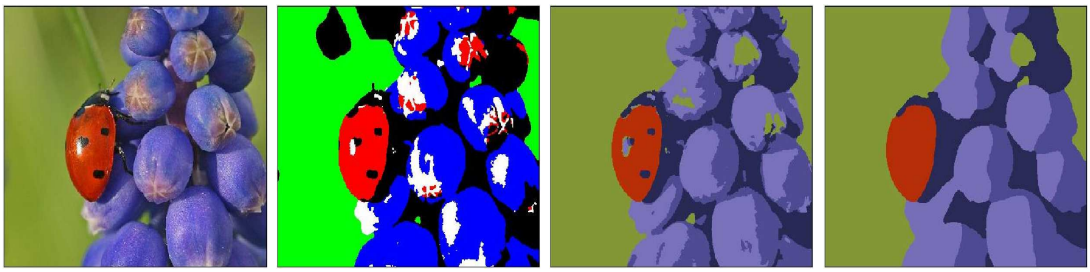


FIGURE 1. Colour classification, from left to right. First: original image. Second: supervised solution for $\alpha = 5.10^{-4}\sqrt{n}$ (n is the number of pixels). Third and fourth: unsupervised solution for $\alpha = 5.10^{-4}\sqrt{n}$ and $\alpha = 10^{-2}\sqrt{n}$, respectively. TV-model with $H = L^1$ and 5 labels (red, green, blue, white and black).

FIGURE 2. Greylevel classification, from left to right. First: original picture. Second: damaged image with salt and peper noise. Third: supervised solution for $\alpha = 4.10^{-4}\sqrt{n}$. TV-model with $H = L^1$ and 4 levels of grey.

with and without level updates, and where the effect of the parameter α is shown: a smaller value will provide a solution with a greater perimeter, that is with more details. Let us emphasize that the value of α should be specifically adapted for each kind of image to reconstruct. In particular a "large enough" value of α should be used for noisy images, in order to remove the spurious perimeter created by the noise. As an example, an unsupervised greylevel simultaneous classification and denoising of a picture with our method (and the TV-L^1 model) is shown in Fig. 2. Later, we will also show examples of black and white image deblurring with an assumed known blur kernel (as is the case in many applications, the blur being due to diffraction, motion, zoom, of a measure apparatus etc.). We will also apply our method to perform combined deblurring and denoising and to classify an image in terms of a prescribed texture.

Our method consists in a multiphase (piecewise constant) joint classification and restoration of an image, as based on

- an exact gradient-free approximation of the total variation
- an optimal partition algorithm.

Here, we mean by *exact* the fact that we approximate the continuous total variation, instead of its discretized counterpart on a chosen mesh, and without the need to a-priori set apart its diffuse and concentrated parts. It is also meant that the approximating functionals converge, as a small parameter ε tends to zero, in some *functional sense* (i.e., Γ-convergence) to the total variation. By *gradient-free* we mean that, as opposed to Modica-Mortola/Ambrosio-Tortorelli-type regularizations [17, 2], the gradient of the image does not appear in the approximating functionals. Hence, from a numerical viewpoint, our approach is better suited to the reconstruction of discontinuous images. It has the other feature to force discrete prescribed values. This is of course a major advantage as to the applications to *optimal partition* problems (see [6, 13]), which are in some sense primary to classification problems and whose basic concepts are recalled hereafter. Consider a bounded domain Ω of $\mathbb{R}^2$, a number $N \in \mathbb{N}$, functions $g_1, ..., g_N \in L^1(\Omega)$, and a parameter $\alpha > 0$. A model problem of minimal partition

reads:

$$\min_{\Omega_1,...,\Omega_N} \sum_{i=1}^{N} \left[\int_{\Omega_i} g_i(x)dx + \frac{\alpha}{2} \mathrm{Per}(\Omega_i) \right], \tag{1.1}$$

where the minimum is searched among all partitions $(\Omega_1, ..., \Omega_N)$ of Ω by subsets of finite perimeter. Here, $\mathrm{Per}(\Omega_i)$ is the relative perimeter of Ω_i in Ω, i.e. $\mathrm{Per}(\Omega_i) = |\partial\Omega_i \cap \Omega|$, hence we have

$$\frac{\alpha}{2} \sum_{i=1}^{N} \mathrm{Per}(\Omega_i) = \alpha \sum_{i<j} |\partial\Omega_i \cap \partial\Omega_j|.$$

Optimal partition problems in imaging are known to be challenging problems. Other methods as found in the litterature are based on convexification, whose major difficulty is not to depart too much from the convex envelope of the cost function. In particular these method solve the optimal partition problem exactly only for two levels (see [22, 13]). However, to address image classification, other mathematical tools, such as topological asymptotic analysis, can be efficiently applied [8].

In essence, our method is not designed for the restoration of continuous images, unless a segmented restoration is sought, two examples of which are shown in Figs. 1 and 2. It is a-priori also not well-suited to process images with texture parts that must be finely identified, since the optimal partitioning will rather identify homogeneous (i.e., texture-free) plane regions. Nevertheless it can be observed that the segmented image of Fig. 2 features some fine characteristics, as e.g., the shade on Lena's hat and some residual texture on its feather tuft. We are aware that more efficient methods exist to address these restoration problems, in particular if the image to restore shows sharp edges of connected subregions [7, 9, 14].

Nonetheless, our method is adequate for all problems whose underlying mathematical structure is related to optimal partition. In particular it is adapted to classify an image without sharp edges (a pioneer reference is [15]), a simple example of which is shown in Fig. 8(b). Another example where our method is well suited is the following. Assume that a detail of an image is occluded, i.e., is missing for some reason, and that this detail shows a structure which cannot be recovered by, e.g., harmonic expansion [9]. Then, an optimal partition formulation of the problem allows us to reconstruct the missing information, as shown in Fig. 5 for the triple and quadruple-point examples.

The paper is organized as follows. The leading ideas are introduced in Section 2 in the context of binary minimal partition. The method is then extended to the multilabel partitioning problem in Section 3. The image classification problem is addressed in Sections 4 and 5 for greyscale and color images, respectively. Section 6 deals with the classification of images based on the analysis of anisotropic textures. The deblurring problem is discussed in Section 7.

2. Binary minimal partition

2.1. **Motivation: binary image classification.** We first present a simple example of image processing which consists of binary image classification. Let us begin with some definitions and notation. We define the set

$$\mathcal{E} = L^\infty(\Omega, \{0, 1\})$$

and the function $F : \mathcal{E} \to \mathbb{R} \cup \{+\infty\}$ such that

$$F(u) = \begin{cases} \frac{1}{2}|Du|(\Omega) & \text{if } u \in BV(\Omega, \{0, 1\}) \\ +\infty & \text{otherwise.} \end{cases}$$

We recall that the total variation of $u \in L^1(\Omega)$ is defined as

$$|Du|(\Omega) = \sup\{\langle u, \operatorname{div} \xi\rangle : \ \xi \in \mathcal{C}_c^\infty(\Omega), \|\xi(x)\| \leq 1 \ \forall x \in \Omega\}, \tag{2.1}$$

and u is said of *bounded variation*, denoted $u \in BV(\Omega)$, when $|Du|(\Omega) < \infty$. Throughout we use the notation

$$\langle u, v\rangle := \int_\Omega u\, v\, dx\ ,$$

for every pair of functions u, v having suitable regularity.

The following calculus illustrates our method on a simple case. Let $f \in L^\infty(\Omega, [0,1])$ be a given image and $c_1, c_2 \in [0,1]$ be two classes representing different intensities of gray. We want to approximate f by the piecewise constant image $u\,c_1 + (1-u)\,c_2$, with $u \in \mathcal{E}$. Given $p \in [1, +\infty)$, the binary image classification problem is formulated as

$$I := \min_{u \in \mathcal{E}} \left\{ \|uc_1 + (1-u)c_2 - f\|^p_{L^p(\Omega)} + \alpha F(u) \right\} . \tag{2.2}$$

Since u is a characteristic function, we have

$$\begin{aligned} \int_\Omega |uc_1 + (1-u)c_2 - f|^p dx &= \int_\Omega |u(c_1 - f) + (1-u)(c_2 - f)|^p dx \\ &= \int_\Omega u|c_1 - f|^p dx + \int_\Omega (1-u)|c_2 - f|^p dx. \end{aligned}$$

We can then redefine I, up to an additive constant, as

$$I = \min_{u \in \mathcal{E}} \{ \mathcal{I}(u) := \langle u, g \rangle + \alpha F(u) \} , \tag{2.3}$$

with

$$g = |c_1 - f|^p - |c_2 - f|^p. \tag{2.4}$$

More generally, we shall address the numerical solution of (2.3) for an arbitrary function $g \in L^1(\Omega)$. We refer to this problem as a binary minimal partition problem.

2.2. **Approximation of the perimeter term.** Let us define the convex hull of $\mathcal{E}$

$$\tilde{\mathcal{E}} = L^\infty(\Omega, [0,1]),$$

and the function $\tilde{F} : \tilde{\mathcal{E}} \to \mathbb{R} \cup \{+\infty\}$ by

$$\tilde{F}(u) = \begin{cases} F(u) & \text{if } u \in \mathcal{E}, \\ +\infty & \text{otherwise.} \end{cases}$$

It is shown in [4] that a suitable approximation of $\tilde{F}$ is provided by the functional $\tilde{F}_\varepsilon$ defined as

$$\tilde{F}_\varepsilon(u) = \inf_{v \in H^1(\Omega)} \left\{ \varepsilon \|\nabla v\|^2_{L^2(\Omega)} + \frac{1}{\varepsilon} \left(\|v\|^2_{L^2(\Omega)} + \langle u, 1 - 2v \rangle \right) \right\}. \tag{2.5}$$

The above minimization problem is easily solved and we have the alternative expression

$$\tilde{F}_\varepsilon(u) = \frac{1}{\varepsilon} \langle 1 - L_\varepsilon u, u \rangle, \tag{2.6}$$

where $L_\varepsilon u$ is the (weak) solution of the boundary value problem with unknown $v \in H^1(\Omega)$

$$\begin{cases} -\varepsilon^2 \Delta v + v &= u \quad \text{in} \quad \Omega \\ \partial_n v &= 0 \quad \text{on} \quad \partial\Omega. \end{cases} \tag{2.7}$$

Considering $\tilde{F}_\varepsilon$ in place of $\tilde{F}$ leads to the approximate problem:

$$I_\varepsilon = \min_{u \in \tilde{\mathcal{E}}} \left\{ \mathcal{I}_\varepsilon(u) := \langle u, g \rangle + \alpha \tilde{F}_\varepsilon(u) \right\} . \tag{2.8}$$

2.3. **Mathematical properties.** We first recall some key properties of the functional $\tilde{F}_\varepsilon$ (Proposition 2.1, Theorem 2.2 and Theorem 2.3) proved in [4] using notably a result from [20]. We refer e.g. to [5, 16] for an introduction to Γ-convergence. Let us start by defining the functional

$$F_\varepsilon(u) = \inf_{v \in H^1(\Omega)} \left\{ \varepsilon \|\nabla v\|^2_{L^2(\Omega)} + \frac{1}{\varepsilon} \|v - u\|^2_{L^2(\Omega)} \right\}. \tag{2.9}$$

It is straightforward to check that, for every $u \in L^2(\Omega)$,

$$\tilde{F}_\varepsilon(u) = F_\varepsilon(u) + \frac{1}{\varepsilon} \langle u, 1 - u \rangle . \tag{2.10}$$

Proposition 2.1. *The function $\tilde{F}_\varepsilon : \tilde{\mathcal{E}} \to \mathbb{R}$ is continuous on $\tilde{\mathcal{E}}$ for the weak-$*$ topology of $L^\infty(\Omega)$, and it is the relaxation (i.e. lower semi-continuous envelope) of the function*

$$u \in \tilde{\mathcal{E}} \mapsto \begin{cases} F_\varepsilon(u) & \text{if } u \in \mathcal{E} \\ +\infty & \text{if } u \notin \mathcal{E}. \end{cases} \tag{2.11}$$

Theorem 2.2. *As $\varepsilon \to 0$, the functionals $\tilde{F}_\varepsilon$ $\Gamma-$converge to $\tilde{F}$ in $\tilde{\mathcal{E}}$, strongly in $L^1(\Omega)$.*

Theorem 2.3. *Let u^ε be an approximate minimizer of (2.8), i.e.*

$$\mathcal{I}_\varepsilon(u^\varepsilon) = \langle u^\varepsilon, g\rangle + \alpha \tilde{F}_\varepsilon(u^\varepsilon) \leq I_\varepsilon + \lambda_\varepsilon,$$

with $u^\varepsilon \in \tilde{\mathcal{E}}$ and $\lim_{\varepsilon\to 0} \lambda_\varepsilon = 0$. Then we have

$$\mathcal{I}_\varepsilon(u^\varepsilon) \to I.$$

Moreover, $(u^\varepsilon)_{\varepsilon\geq 0}$ admits cluster points, and each of these cluster points is a minimizer of (2.3).

The following pointwise convergence result will be later useful.

Proposition 2.4. *If $u \in \tilde{\mathcal{E}}$, then*

$$\lim_{\varepsilon\to 0} \tilde{F}_\varepsilon(u) = \tilde{F}(u).$$

Proof. We suppose that $u \in \mathcal{E}$, otherwise the result is straightforward in view of (2.10). We set

$$\tilde{G}_\varepsilon(u, v) = \varepsilon \|\nabla v\|^2_{L^2(\Omega)} + \frac{1}{\varepsilon}\left(\|v\|^2_{L^2(\Omega)} + \langle u, 1 - 2v\rangle\right),$$

thus $\tilde{F}_\varepsilon(u) \leq \tilde{G}_\varepsilon(u, v)$ for any $v \in H^1(\Omega)$. Let $\lambda > 0$. Arguing as in the proof of Proposition 4.2 of [20], we may find $v_\varepsilon \in H^1(\Omega)$ such that, for ε small enough, $\tilde{G}_\varepsilon(u, v_\varepsilon) \leq \tilde{F}(u) + \lambda$. We infer that $\limsup_{\varepsilon\to 0} \tilde{F}_\varepsilon(u) \leq \tilde{F}(u)$. The lim inf inequality of the Γ-convergence completes the proof. □

By the direct method of the calculus of variations, we straightforwardly prove the following existence result (see [4] for details).

Proposition 2.5. *The infima of $\mathcal{I}$ and $\mathcal{I}_\varepsilon$ in (2.3) and (2.8) are finite and attained in $BV(\Omega, \{0, 1\})$ and $\tilde{\mathcal{E}}$, respectively.*

2.4. **Algorithm.** Our algorithm is based on a continuation method, namely we construct a decreasing sequence (ε_m) of positive numbers tending to zero, and, for each $\varepsilon = \varepsilon_m$, we find an approximate minimizer of (2.8) using as initial guess the solution obtained at iteration $m - 1$. In the sequel, the subscript m will be dropped for simplicity.

2.4.1. *Description of the algorithm in the function space setting.* Plugging (2.5) into (2.8), we obtain that the subproblem at ε fixed consists is solving the following two-level minimization problem

$$I_\varepsilon = \min_{u\in\tilde{\mathcal{E}}} \inf_{v\in H^1(\Omega)} \left\{ \langle u, g\rangle + \alpha \left[\varepsilon \|\nabla v\|^2_{L^2(\Omega)} + \frac{1}{\varepsilon}\left(\|v\|^2_{L^2(\Omega)} + \langle u, 1 - 2v\rangle\right)\right]\right\}. \tag{2.12}$$

The simple structure of this problem with respect to each variable u and v leads us to use an alternating minimization algorithm. The superscript k is used to designate variables computed at iteration k. The iteration k, $k \geq 1$, consists in the two steps described below.

(1) The minimization with respect to v is straightforward. It consists in solving the boundary value problem

$$\left\{ \begin{array}{rcll} -\varepsilon^2 \Delta v^k + v^k &=& u^{k-1} & \text{in } \Omega \\ \partial_n v^k &=& 0 & \text{on } \partial\Omega. \end{array}\right. \tag{2.13}$$

(2) The minimization with respect to u is a linear programming problem in a convex set. Therefore a minimizer can always be found among the extreme points of $\tilde{\mathcal{E}}$. More precisely here, we have to minimize at each point $x \in \Omega$ the linear function

$$\phi(s) = s\, g(x) + \frac{\alpha}{\varepsilon} s(1 - 2v^k(x))$$

over $s \in [0, 1]$. Setting

$$\zeta^k(x) = g(x) + \frac{\alpha}{\varepsilon}(1 - 2v^k(x)),$$

a solution is immediately found as

$$u^k(x) = \left\{ \begin{array}{ll} 0 & \text{if } \zeta^k(x) \geq 0, \\ 1 & \text{otherwise.} \end{array}\right.$$

In other terms, u^k is the characteristic function of the level-set $\{\zeta^k < 0\}$, denoted by

$$u^k = \chi_{\{\zeta^k<0\}}.$$

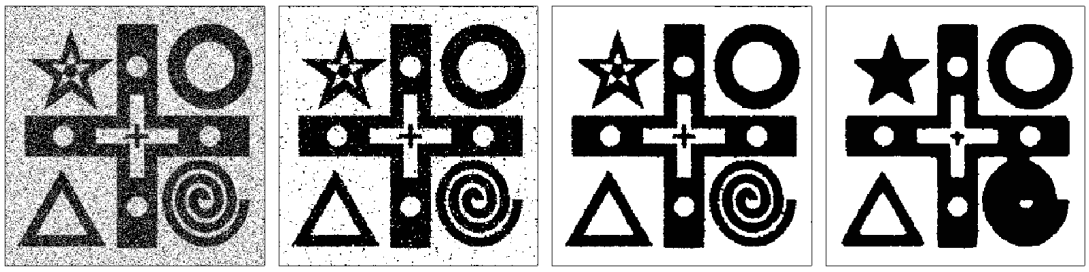


FIGURE 3. Left: original black and white noisy image. From left to right: obtained solutions for $\alpha = 2.10^{-4}\sqrt{n}$ (incomplete denoising), $\alpha = 2.10^{-3}\sqrt{n}$ (optimal perimeter weight), $\alpha = 9.10^{-3}\sqrt{n}$ (excessive perimeter weight), respectively.

This algorithm ensures a decrease of the objective function at each iteration. Moreover, each cluster point (in the weak-$*$ topology for u and the H^1 norm topology for v) is a stationary point. Of course, as the coupled problem in (u, v) is not convex, local minimizers are not excluded.

An outstanding feature of this algorithm is that u is a always a characteristic function during the iterations.

2.4.2. *Discrete version.* For solving the boundary value problem (2.7) we use finite elements on a cartesian mesh with $Q1$ shape functions. The mesh nodes coincides with the image pixels, and without any loss of generality, the mesh size is fixed to 1. The discrete counterparts of the variables u and v are therefore vectors of $\mathbb{R}^n$ where n is the number of pixels. Denoting by K and M the stiffness and mass matrices for the Laplacian, the discrete problem reads

$$I = \min_{u\in\mathbb{R}^n} \min_{v\in\mathbb{R}^n} \left\{ Mu \cdot g + \alpha \left[\varepsilon Kv \cdot v + \frac{1}{\varepsilon} \left(Mv \cdot v + Mu \cdot (\mathbb{1} - 2v) \right) \right] \right\}.$$

In the above expression, the dot stands for the standard dot product of $\mathbb{R}^n$, and $\mathbb{1} = (1, ..., 1)^T$.

In this framework the two steps of the algorithm consist in solving the linear system

$$(\varepsilon^2 K + M)v^k = u^{k-1}, \tag{2.14}$$

and setting

$$\zeta^k = M\left(g + \frac{\alpha}{\varepsilon}(\mathbb{1} - 2v^k) \right), \tag{2.15}$$

$$u^k = \chi_{\{\zeta^k < 0\}}. \tag{2.16}$$

The stopping criterion corresponds to a relative variation of the vector u (in squared ℓ^2 norm) between two successive iterations smaller than some threshold, fixed to 10^{-5}.

The numerical solution of the linear system (2.14) is performed in an efficient way with the help of the fast Fourier transform (FFT). To do so, the image u is symmetrized in both directions, leading to handle a domain $\hat{\Omega}$ with double width and height. Then periodicity conditions are assumed on the boundary of $\hat{\Omega}$, which is a convenient way of implementing Neumann boundary conditions. In this framework, the matrix products Kv and Mv represent bidimensional discrete convolutions, which are easily transferred to the Fourier domain. The Fourier transform of v is thus obtained, and v itself is retrieved by inverse FFT.

2.4.3. *Update of ε.* The parameter ε has the dimension of a length. Thus we start with a characteristic length of Ω, namely $\varepsilon_0 = \varepsilon_{max} = \sqrt{n}$. Then we divide ε by two at each iteration of an outer loop, that is, we choose $\varepsilon_m = \varepsilon_{max}/2^m$. In order to approximate (2.13) properly, ε must not be taken significantly smaller than the grid size. Thus we stop the algorithm as soon as $\varepsilon_m \leq \varepsilon_{min} = 0.1$.

2.4.4. *Initialization.* The initialization of u is performed by the expressions (2.15)-(2.16) with $\varepsilon \to +\infty$, that is, we set

$$\zeta^0 = Mg, \qquad u^0 = \chi_{\{\zeta^0 < 0\}}.$$

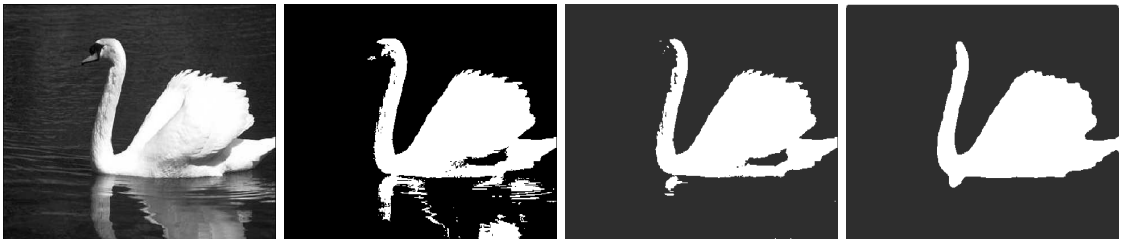


FIGURE 4. From left to right: original image, initialization ($\varepsilon = \infty$), and obtained solutions for $\varepsilon = 15.5$ and $\varepsilon = \varepsilon_{min}$, respectively (with $\alpha = 9.10^{-3}$).

2.5. **Numerical examples.** In Figure 3, we present an example of binary image classification using the above procedure. The function g is defined by (2.4) with the two levels c_1 and c_2 corresponding to black and white. Since this particular example aims at denoising, the L^1 norm is chosen.

In Figure 4, we present another example, also with black and white levels and L^1 norm. Intermediate iterations are displayed.

3. Multilabel minimal partition

In this section we extend the previous approach to the multilabel problem. We define the set

$$\mathcal{E}_N = \left\{ (u_1, ..., u_N) \in \mathcal{E}^N, \sum_{i=1}^N u_i = 1 \right\}.$$

This is the set of N-tuples of characteristic functions $(u_1, ..., u_N) = (\chi_{\Omega_1}, ..., \chi_{\Omega_N})$, where $(\Omega_1, ..., \Omega_N)$ form a partition of Ω. Given functions $g_1, ..., g_N \in L^1(\Omega)$, the minimal partition problem (1.1) reads

$$I := \min_{(u_1,...,u_N)\in\mathcal{E}_N} \left\{ \mathcal{I}(u) := \sum_{i=1}^N \left[\langle u_i, g_i \rangle + \alpha F(u_i) \right] \right\}. \tag{3.1}$$

In the same way as presented in Section 2, this problem is approximated by

$$I_\varepsilon := \min_{(u_1,...,u_N)\in\tilde{\mathcal{E}}_N} \left\{ \mathcal{I}_\varepsilon(u) := \sum_{i=1}^N \left[\langle u_i, g_i \rangle + \alpha \tilde{F}_\varepsilon(u_i) \right] \right\} \tag{3.2}$$

with

$$\tilde{\mathcal{E}}_N = \left\{ (u_1, ..., u_N) \in \tilde{\mathcal{E}}^N, \sum_{i=1}^N u_i = 1 \right\}.$$

The following result is an extension of its binary counterpart presented in Section 2.

Theorem 3.1. *Let $u^\varepsilon = (u_i^\varepsilon)_{1\le i\le N}$ be an approximate minimizer of (3.2), i.e.,*

$$\mathcal{I}_\varepsilon(u^\varepsilon) = \sum_{i=1}^N \left[\langle u_i^\varepsilon, g_i \rangle + \alpha \tilde{F}_\varepsilon(u_i^\varepsilon) \right] \le I_\varepsilon + \lambda_\varepsilon,$$

with $u_i^\varepsilon \in \tilde{\mathcal{E}}_N$ and $\lim_{\varepsilon\to 0} \lambda_\varepsilon = 0$. Then we have

$$\mathcal{I}_\varepsilon(u^\varepsilon) \to I.$$

Moreover, $(u^\varepsilon)_{\varepsilon\ge 0}$ admits cluster points, and each of these cluster points is a minimizer of (3.1).

Proof. As for Theorem 2.3, the proof relies on the Γ-convergence and equicoercivity arguments (see the details in [4]). The lim inf inequality of the Γ-convergence and the equicoercivity immediately pass to the sum in this simple case of uncoupled terms. As to the lim sup inequality, one has to construct a recovery sequence which belongs to $\tilde{\mathcal{E}}_N$, which is not automatically achieved by gathering independent recovery sequences for each variable u_i. In fact, this difficulty is overcome by simply choosing a constant recovery sequence, which is possible by virtue of Proposition 2.4. □

3.1. **Algorithm.** For ε fixed we have to solve the approximate problem

$$I_\varepsilon = \min_{(u_1,...,u_N)\in\tilde{\mathcal{E}}_N} \inf_{(v_1,...,v_N)\in H^1(\Omega)^N} \sum_{i=1}^N \left\{ \langle u_i, g_i\rangle + \alpha \left[\varepsilon \|\nabla v_i\|^2_{L^2(\Omega)} + \frac{1}{\varepsilon}\left(\|v_i\|^2_{L^2(\Omega)} + \langle u_i, 1-2v_i\rangle \right) \right] \right\}.$$

We use again an alternating minimization algorithm with respect to the variables $(u_1, ..., u_N)$ and $(v_1, ..., v_N)$. The superscript k is again used to designate these vectors at iteration k.

(1) The minimization with respect to $(v_1, ..., v_N)$ consists in solving the N boundary value problems

$$\left\{ \begin{array}{rcll} -\varepsilon^2 \Delta v_i^k + v_i^k &=& u_i^{k-1} & \text{in} \quad \Omega \\ \partial_n v_i^k &=& 0 & \text{on} \quad \partial\Omega. \end{array} \right. \tag{3.3}$$

(2) The minimization with respect to $(u_1, ..., u_N)$ is a linear programming problem. Minimizers can be found by exploring the vertices of the polyhedron $\tilde{\mathcal{E}}_N$, that is, $\mathcal{E}_N$. The practical procedure is the following. Set

$$\zeta_i^k = g_i + \frac{\alpha}{\varepsilon}(1 - 2v_i^k).$$

At each point $x \in \Omega$ we find an index $i(x)$ such that

$$\zeta^k_{i(x)}(x) = \min\{\zeta_1^k(x), ..., \zeta_N^k(x)\}. \tag{3.4}$$

We then set

$$u_i^k(x) = \left\{ \begin{array}{ll} 1 & \text{if } i = i(x) \\ 0 & \text{otherwise.} \end{array} \right.$$

The discrete counterpart is easily obtained in the same fashion as in the binary case. Again, an outstanding feature of this algorithm is that the functions $u_1^k, ..., u_N^k$ are always characteristic functions of a partition of Ω.

3.2. **Numerical validation.** The three examples of Fig. 5 are taken from and should be compared to [13]. Let $E_0, E_1, ..., E_N$ be a given partition of Ω. We define g_i, $i = 1, ..., N$, by

$$g_i = \sum_{\substack{1\le j\le N \\ j\neq i}} \chi_{E_j} = 1 - \chi_{E_i}.$$

This means that, in the set E_i, $i \geq 1$, the label i is favored, whereas in the set E_0 there is no preference. In the subsequent examples the domain Ω is the unit square with a 400×400 discretization and we choose $\alpha = 0.1\sqrt{n}$.

3.2.1. *Example with 3 labels.* The partition is as shown in Fig. 5(a) (left). The set E_0 is the black disc, while each E_i, $i = 1, 2, 3$, is assigned to a specific color, namely red, green and blue, respectively. We retrieve the triple junction, which is known to be the theoretical solution [13].

3.2.2. *Example with 4 labels.* We consider now the partitions of Fig. 5(b) and 5(c), with E_0 the black disc and 4 other subsets. In these cases, the solution is not unique, and the algorithm chooses a particular one. This choice stems from the selection of a particular minimizer in (3.4). This is in contrast with the results obtained in [13] which, due to the convexification method employed, are mixtures of minimizers.

4. Multilabel image classification

4.1. **Formulation as a minimal partition problem.** We come back to the classification problem presented in Section 2.1, but this time with N grey levels $c_1, ..., c_N \in [0, 1]$. We are given an image $f \in L^\infty(\Omega, [0, 1])$, and consider the piecewise constant image

$$w = \sum_{i=1}^N u_i c_i,$$

where u_i is the characteristic function of a subset Ω_i of Ω such that $(\Omega_1, ..., \Omega_N)$ form a partition of Ω. We have for any L^p norm on Ω

$$\|w - f\|^p_{L^p(\Omega)} = \left\| \sum_{i=1}^N u_i c_i - f \right\|^p_{L^p(\Omega)} = \left\| \sum_{i=1}^N u_i (c_i - f) \right\|^p_{L^p(\Omega)} = \sum_{i=1}^N \int_\Omega u_i |c_i - f|^p.$$

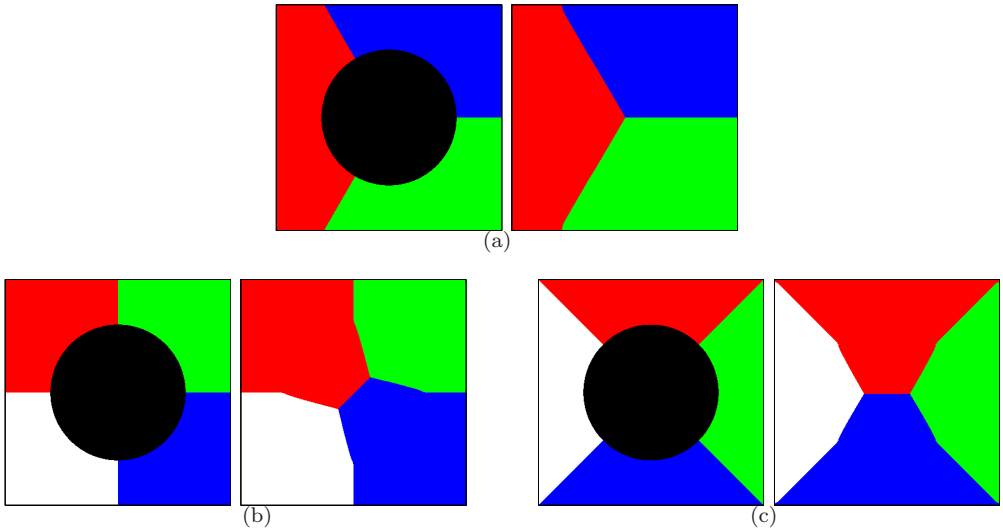


FIGURE 5. 5(a) Triple point: given partition (left) and obtained solution (right); 5(b) and 5(c) Two quadruple points: given partition (left) and obtained solution (right).

The misfit between the piecewise constant and original images is measured by

$$\|w - f\|^p_{L^p(\Omega)} = \sum_{i=1}^{N} \langle u_i, g_i \rangle, \qquad g_i = |c_i - f|^p.$$

When the levels c_i are fixed, we can directly apply the algorithm of Section 3.1.

4.2. **Update of levels.** However, it is often desirable to determine the grey levels within the classes automatically. Thus, we include a third step in the alternating minimization algorithm, consisting in solving

$$\min_{c_i \in [0,1]} \sum_{i=1}^{N} \langle u_i, g_i \rangle = \sum_{i=1}^{N} \int_\Omega u_i |c_i - f|^p.$$

This problem is separable in its variables $c_1, ..., c_N$, and each c_i must satisfy

$$c_i \in \operatorname*{argmin}_{c \in \mathbb{R}} \int_\Omega u_i |c - f|^p.$$

We distinguish between the two cases of practical interest, namely $p = 2$ and $p = 1$.

- $p = 2$**.** This is a standard problem which results in computing the arithmetic mean

$$c_i = \frac{\int_\Omega u_i f}{\int_\Omega u_i}.$$

- $p = 1$**.** Minimizing a sum of L^1 distances classically amounts to computing a median, but here, due to the weights u_i this is a little more involved. Details of the procedure in the discrete setting are given in Appendix A. Note that there may be several solutions. In this case we take the half-sum of the extreme points of the minimizing set.

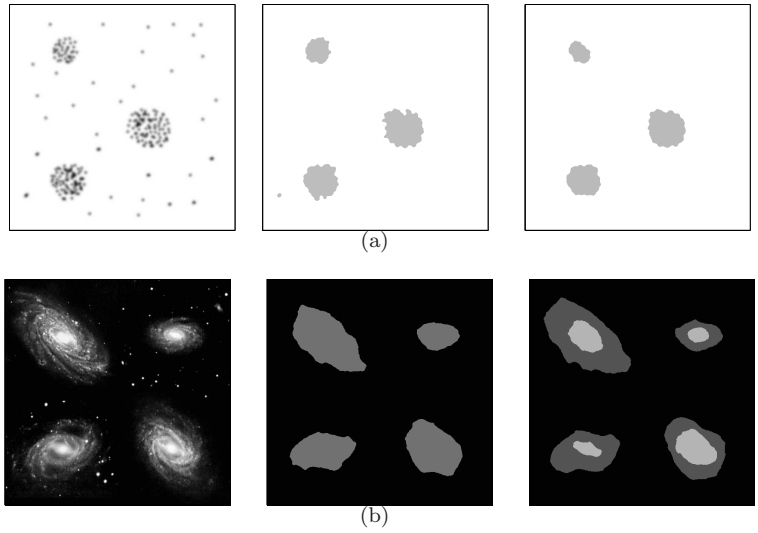
For the initialization, the levels are equidistributed in $[0, 1]$, i.e., we choose

$$c_i^0 = \frac{i-1}{N-1}.$$

4.3. **Examples.** Examples with 3 labels are shown in Figs. 6-8. In Fig. 6, an example of denoising is shown (hence we have chosen the L^1 norm). In Fig. 7, the classification of the swan picture is shown with different values of ε. This result is to be compared with its binary counterpart in Fig. 4. In Fig. 8 two examples, namely, one hand-made and one real-world images with disconnected geometric elements are featured.

FIGURE 6. Greylevel image denoising with update of levels (3 labels), L^1 norm, $\alpha = 5.10^{-3}\sqrt{n}$: original image (left), noisy image (middle) solution with 3 labels (right) and $\alpha = 5.10^{-3}$.

FIGURE 7. From left to right: original image, initialization ($\varepsilon = \infty$), and obtained solutions for $\varepsilon = 15.5$ and ε_{min}, respectively (with 3 labels, L^1 norm, $\alpha = 5.10^{-5}$).



(a)

(b)

FIGURE 8. 8(a)Image classification with 2 labels, L^1 norm and update of levels: original image (left), obtained images for $\alpha = 2.10^{-3}\sqrt{n}$ (middle) and $\alpha = 5.10^{-3}\sqrt{n}$ (right); 8(b) Image classification with update of levels, L^1 norm, $\alpha = 5.10^{-3}\sqrt{n}$: original image (left), obtained images with 2 labels (middle) and 3 labels (right).

5. Multilabel classification of color images

The original image f is represented by the three channels $(f_1, f_2, f_3) \in L^\infty(\Omega, [0,1])^3$ representing the intensity of red, blue and green respectively. Each phase Ω_i is associated to a color $(c_{i1}, c_{i2}, c_{i3}) \in [0,1]^3$ in the same RGB system. The constructed image $w = (w_1, w_2, w_3)$ is given by

$$w_j = \sum_{i=1}^{N} u_i c_{ij},$$

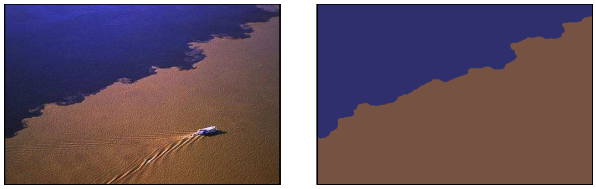


FIGURE 9. Unsupervised color image classification with 2 labels and L^1 norm, $\alpha = 5.10^{-3}\sqrt{n}$: original image (left), obtained image (right)

FIGURE 10. Unsupervised color image classification with L^1 norm, $\alpha = 5.10^{-3}\sqrt{n}$: original image (left), obtained image with 2 labels (middle), obtained image with 5 labels (right)

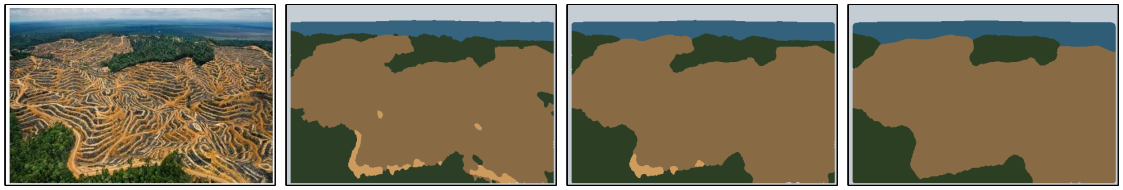


FIGURE 11. Unsupervised color image classification with L^1 norm and 5 labels, from left to right: original image and obtained images for $\alpha = 5.10^{-3}\sqrt{n}$, $\alpha = 10^{-2}\sqrt{n}$, $\alpha = 2.10^{-2}\sqrt{n}$

where u_i is the characteristic function of Ω_i. We have for each channel

$$\|w_j - f_j\|_{L^p}^p = \left\| \sum_{i=1}^N u_i c_{ij} - f_j \right\|_{L^p}^p = \left\| \sum_{i=1}^N u_i (c_{ij} - f_j) \right\|_{L^p}^p = \sum_{i=1}^N \int_\Omega u_i |c_{ij} - f_j|^p.$$

The misfit between the segmented and original images is measured by

$$\sum_{j=1}^3 \|w_j - f_j\|_{L^p}^p = \sum_{i=1}^N \langle u_i, g_i \rangle, \qquad g_i = \sum_{j=1}^3 |c_{ij} - f_j|^p.$$

We then apply the same algorithm as in Section 4. Note that the geometrical variable $u = (u_1, ..., u_N)$ as well as the auxiliary variable $v = (v_1, ..., v_N)$ remain N-dimensional vectors, and the update of levels is separable in the channels.

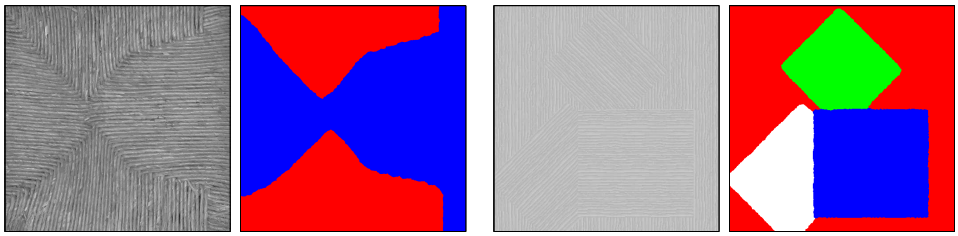
The first example (see Figure 9) is a two-label problem, initialized with a pure black phase and a pure white phase.

In the second example (see Figure 10), we first choose 2 labels, then 5 labels. In this latter case, the phases are initialized by the pure colors black, red, green, blue and white. We observe that only 4 labels remain at convergence, which is an effect of the perimeter penalization.

In the third example (see Figure 11), we choose again 5 labels. Results obtained with different values of α are depicted.

6. Anisotropy-based image classification

We come back to a greyscale image $f \in L^\infty(\Omega, [0,1])$. We are given N vectors $\xi_1, ..., \xi_N \in \mathcal{S}_2$, where $\mathcal{S}_2$ is the unit sphere of $\mathbb{R}^2$. In order to detect fluctuations oriented along the direction ξ_i, we



(a) (b)

FIGURE 12. (a) Anisotropy-based image classification with 2 labels: original image (left), obtained partition for $\alpha = 10^{-5}\sqrt{n}$ (right); (b) Anisotropy-based image classification with 4 labels: original image (left), obtained partition for $\alpha = 10^{-6}\sqrt{n}$ (right).

first regularize f by solving

$$\left\{ \begin{array}{rcl} -\Delta\phi + \phi &=& f \quad \text{in} \quad \Omega \\ \partial_n \phi &=& 0 \quad \text{on} \quad \partial\Omega. \end{array} \right. \tag{6.1}$$

Then we set

$$g_i = -(\nabla\phi.\xi_i)^2. \tag{6.2}$$

In practice, (6.1) is solved by finite elements, in the same way as described in Section 2.4.2. Then (6.2) is computed at each node using a discrete gradient based on finite differences. We point out that the function g_i can be interpreted as the topological sensitivity of the energy functional

$$\frac{1}{2}\|\nabla\phi\|^2_{L^2(\Omega)} + \frac{1}{2}\|\phi - f\|^2_{L^2(\Omega)}$$

with respect to the creation of an insulating crack normal to ξ_i, see [3] and [9] for various applications of this concept in image processing.

We then apply the multilabel minimal partition algorithm of Section 3 with the obtained functions $(g_1, ..., g_N)$.

In Figure 12(a) we show an example with 2 labels. We want to detect vertical and horizontal fluctuations, therefore we choose the vectors (ξ_i) as

$$\xi_1 = (1,0)^T,\ \xi_2 = (0,1)^T.$$

In Figure 12(b) we show an example with 4 labels, where the vectors (ξ_i) are given by

$$\xi_1 = (1,0)^T,\ \xi_2 = \frac{\sqrt{2}}{2}(1,1)^T, \xi_3 = (0,1)^T,\ \xi_4 = \frac{\sqrt{2}}{2}(-1,1)^T.$$

7. Image deblurring

In this section we apply a variant of our minimal partition model for binary image deconvolution. The blurring kernel is represented by a linear operator $A : L^2(\Omega) \to L^2(\Omega)$ such that $A1 = 1$. The given blurred (greyscale) image is $f \in L^\infty(\Omega, [0,1])$, and the reconstructed image is $w = c_1 u + c_2(1-u)$, with $u \in \mathcal{E}$ and $c_1, c_2 \in [0,1]$. We have

$$Aw = c_1 Au + c_2 A(1-u) = (c_1 - c_2)Au + c_2.$$

The deblurring problem reads

$$I := \min_{u\in\mathcal{E}} \left\{ \mathcal{I}(u) = \|(c_1 - c_2)Au + c_2 - f\|^2_{L^2(\Omega)} + \alpha F(u) \right\}. \tag{7.1}$$

We cannot write the above problem in the form (2.3), thus the alternating algorithm presented in Section 2.4 cannot be straightforwardly adapted. This is the reason why we restrict ourselves to the binary problem.

As before, we approximate $F(u)$ by $\tilde{F}_\varepsilon(u)$. Using the expression (2.6), we approximate (7.1) by

$$I_\varepsilon = \min_{u\in\tilde{\mathcal{E}}} \left\{ \mathcal{I}_\varepsilon(u) = \|(c_1 - c_2)Au + c_2 - f\|^2_{L^2(\Omega)} + \alpha\frac{1}{\varepsilon}\langle 1 - L_\varepsilon u, u\rangle \right\}, \tag{7.2}$$

FIGURE 13. Deblurring and denoising: original image (left), damaged image with blur and noise effects (middle), reconstructed image for $\alpha = 2.10^{-4}\sqrt{n}$ (right).

and use the same continuation procedure with respect to ε. An analogue to Theorem 2.3 and Proposition 2.5 can be straightforwardly derived.

For solving (7.2) at ε fixed, we use a projected gradient algorithm. The iteration k reads

$$u^{k+1} = P_{[0,1]}(u^k - t^k \nabla \mathcal{I}_\varepsilon(u^k)),$$

with

$$P_{[0,1]}(\varphi) = \max(0, \min(1, \varphi)),$$

$$\nabla \mathcal{I}_\varepsilon(u) = 2(c_1 - c_2)A^*[(c_1 - c_2)Au + c_2 - f] + \frac{\alpha}{\varepsilon}(1 - 2L_\varepsilon u)$$

and t^k determined by a line search. More precisely, t^k is initialized by $t_0^k = \|u^k\|_{L^2(\Omega)}/\|\nabla \mathcal{I}_\varepsilon(u^k)\|_{L^2(\Omega)}$, and divided by 2 until $\mathcal{I}_\varepsilon(u^{k+1}) < \mathcal{I}_\varepsilon(u^k)$.

In our experiments we take the operator $A = A_0^q$, where A_0 is the discrete convolution operator by the kernel

$$\kappa = \begin{pmatrix} 0 & a & 0 \\ a & 1-4a & a \\ 0 & a & 0 \end{pmatrix}, \qquad a = 0.15.$$

We recall that the unknown u is symmetrized and periodized, hence the convolution is performed without boundary effect. In addition, the computation of the product $Au = A_0^q u$ is efficiently performed through the FFT, without actually computing and storing the matrix of the operator A.

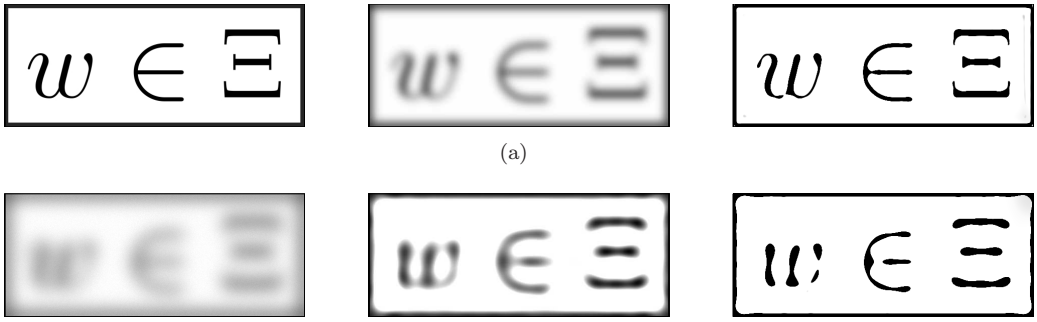
In Figure 13, f is of the form $P_{[0,1]}\left(A(u^* c_1 + (1 - u^*)c_2 + \nu)\right)$, with u^* a characteristic function and ν a random noise. The grey levels are black and white.

The example featured in Fig. 14 also shows that though the underlying minimal partition algorithm is intended for the classification of plane regions, the proposed method also permits to restore a blurred and noisy text. In order to retrieve the fine details of the text, it is necessary to choose the penalization coefficient α quite small, hence it is difficult to accommodate with a high level of noise. In Fig. 14(b), a better result is obtained when the decrease of ε is stopped before reaching the value ε_{min}, although for this value the obtained image is not binary.

8. Concluding remarks

Whereas the computer vision community is very active in developing powerful algorithms, few of these approaches are theoretically justified. In this paper we propose a mathematically sound method to perform optimal partitions and apply it to image restoration and classification of greyscale and color images. Let us emphasize that the proposed continuation approach approximates the continuous total variation and not its discretized version. This is achieved through the solving of linear partial differential equations with constant coefficients, which is performed by finite elements and fast Fourier transforms. This task is from far the dominant part of the computational effort of our algorithms. Its implementation could certainly be further improved, but code optimization was not the aim of this work.

We consider the present paper as a first step of an ongoing work where an image partition algorithm and a gradient-free approximation of the perimeter is proposed to address piecewise constant image restoration and/or classification. In particular, we have applied our algorithm to denoising, deblurring and supervised texture identification problems, as well as for inpainting. Future extensions of the methods could be multilabel deblurring and unsupervised texture identification.



(a)

(b)

FIGURE 14. Deblurring of the text "$w \in \Xi$". 14(a) Original image (left), blurred image (middle), reconstructed image for $\alpha = 10^{-5}\sqrt{n}$ (right). 14(b) blurred image with noise (left), its restoration with $\alpha = 10^{-5}\sqrt{n}$ for $\varepsilon = 0.748$ (middle) and for $\varepsilon = \varepsilon_{min}$ (right).

Appendix A. Weighted median

Let $(x_1, ..., x_n) \in \mathbb{R}^n$, $(\alpha_1, ..., \alpha_n) \in \mathbb{R}^n$ be given. We assume that the x_i are numbered in increasing order. We want to minimize

$$V(x) = \sum_{i=1}^{n} \alpha_i |x - x_i|.$$

This function is clearly convex and affine on each interval $[x_i, x_{i+1}]$. Therefore, the minimizing set is an interval of the form $[x_l, x_r]$, $1 \leq l \leq r \leq n$. The subdifferential of V at the point x_j is

$$\partial V(x_j) = \sum_{i<j} \alpha_i - \sum_{i>j} \alpha_i + \alpha_j[-1, 1].$$

Then x_j is a minimizer of V if and only if $0 \in \partial V(x_j)$, i.e.

$$-\alpha_j \leq \sum_{i<j} \alpha_i - \sum_{i>j} \alpha_i \leq \alpha_j. \tag{A.1}$$

We obtain x_l and x_r by exploring every x_j and checking if it satisfies (A.1).

References

[1] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of bounded variation and free discontinuity problems*. Oxford Mathematical Monographs. The Clarendon Press Oxford University Press, New York, 2000.

[2] L. Ambrosio and V. M. Tortorelli. Approximation of functionals depending on jumps by elliptic functionals via Γ-convergence. *Comm. Pure Appl. Math.*, 43(8):999–1036, 1990.

[3] S. Amstutz, I. Horchani, and M. Masmoudi. Crack detection by the topological gradient method. *Control Cybernet.*, 34(1):81–101, 2005.

[4] S. Amstutz and N. Van Goethem. Topology optimization methods with gradient-free perimeter approximation. *hal-00614818, version 1*, 16 Aug. 2011.

[5] H. Attouch, G. Buttazzo, and G. Michaille. *Variational analysis in Sobolev and BV spaces*, volume 6 of *MPS/SIAM Series on Optimization*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2006. Applications to PDEs and optimization.

[6] J.-F. Aujol and G. Aubert. Optimal partitions, regularized solutions, and application to image classification. *Applicable Analysis*, 84(1):15–35, 2005.

[7] J.-F. Aujol, G. Gilboa, T. Chan, and S. Osher. Structure-texture image decomposition-modeling, algorithms,and parameter selection. *Int. J. Comp. Vision*, 67(1):85–104, 2005.

[8] D. Auroux, L. Jaafar Belaid, and M. Masmoudi. A topological asymptotic analysis for the regularized grey-level image classification problem. *ESAIM, Math. Model. Numer. Anal.*, 41(3):607–625, 2007.

[9] D. Auroux and M. Masmoudi. Image processing by topological asymptotic expansion. *J. Math. Imaging Vision*, 33(2):122134, 2009.

[10] A. Chambolle. An algorithm for total variation minimization and applications. Special issue on mathematics and image analysis. *J. Math. Imaging Vision*, 20(1-2):89–97, 2004.

[11] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock. An introduction to total variation for image analysis. Fornasier, Massimo (ed.), Theoretical foundations and numerical methods for sparse recovery, Vienna, 31/8 – 4/9, 2009. Berlin: Walter de Gruyter. Radon Series on Computational and Applied Mathematics 9, 2010.

[12] A. Chambolle, V. Caselles, and M. Novaga. Total variation in imaging. In O. Scherzer, editor, *Handbook of mathematical methods in imaging*. Springer Reference. Berlin: Springer, 2011.
[13] A. Chambolle, D. Cremers, and T. Pock. A convex approach to minimal partitions. *hal-00630947, version 1*, 11 Oct. 2011.
[14] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vision*, 40(1):120–145, 2011.
[15] T. Chan and L. Vese. Active contours without edges. *IEEE Trans. Image Processing*, 10(2):266–277, 2001.
[16] G. Dal Maso. *An introduction to* Γ*-convergence*. Progress in Nonlinear Differential Equations and their Applications, 8. Birkhäuser Boston Inc., Boston, MA, 1993.
[17] L. Modica. The gradient theory of phase transitions and the minimal interface criterion. *Arch. Rational Mech. Anal.*, 98(2):123–142, 1987.
[18] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Commun. Pure Appl. Math.*, 42(5):577–685, 1989.
[19] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1-4):259–268, 1992.
[20] M. Solci and E. Vitali. Variational models for phase separation. *Interfaces Free Bound.*, 5(1):27–46, 2003.
[21] L. Vese. A study in the BV space of a denoising-deblurring variational problem. *Appl. Math. Optimization*, 44(2):131–161, 2001.
[22] C. Zach, D. Gallup, J. Frahm, and M. Niethammer. Fast global labelling for real-time stereo using multiple plabe sweeps. In *Vision, Modeling, and Visualization*. IOS press, 2008.

Laboratoire de Mathématiques d'Avignon, Faculté des Sciences, 33 rue Louis Pasteur, 84000 Avignon, France
E-mail address: samuel.amstutz@univ-avignon.fr

Laboratório Nacional de Computação Científica LNCC/MCT, Coordenação de Matemática Aplicada e Computacional, Av. Getúlio Vargas 333, 25651-075 Petrópolis - RJ, Brasil.
E-mail address: novotny@lncc.br

Universidade de Lisboa, Faculdade de Ciências, Departamento de Matemática, Centro de Matemática e Aplicações Fundamentais, Av. Prof. Gama Pinto 2, 1649-003 Lisboa, Portugal.
E-mail address: vangoeth@ptmat.fc.ul.pt