Title: Piecewise linear approximation of vector-valued images and curves via 2nd-order variational model

This paper appears in: IEEE Transactions on Image Processing

Date of Publication: 16 June 2017

Author(s): M. Zanetti, L. Bruzzone

Volume: 26, Issue: 9

Page(s): 4414 – 4429

DOI: 10.1109/TIP.2017.2716827

Piecewise linear approximation of vector-valued images and curves via 2nd-order variational model

Massimo Zanetti, and Lorenzo Bruzzone

Abstract

Variational models are known to work well for addressing image restoration/regularization problems. However, most of the methods proposed in literature are defined for scalar inputs and are used on multiband images (such as RGB or multispectral imagery) by the composition of a simple band-wise processing. This involves suboptimal results and may introduce artifacts. Only in a few cases variational models are extended to the case of vector-valued inputs. However, the known implementations are restricted to 1st-order models, while 2nd-order models are never considered. Thus, typical problems of 1st-order models such as the staircasing effect cannot be overtaken. This paper considers a 2nd-order functional model to function approximation with free discontinuities given by Blake-Zisserman (BZ) and proposes an efficient minimization algorithm in the case of vector-valued inputs. In the BZ model, the Hessian of the solution is penalized outside a set of finite length, therefore the solution is forced to be piecewise linear. Moreover, the model allows the formation of free discontinuities and free gradient discontinuities. The proposed algorithm is applied to difficult color image restoration/regularization problems and to piecewise linear approximation of curves in space.

Index Terms

Multiband image, variational methods, Blake-Zisserman, Mumford-Shah, piecewise linear approximation, block-coordinate descent method

I. Introduction

TYPICAL models for image restoration/regularization assume the image g recorded by an optical sensor to be a noisy variation of a regular signal u. Mathematical methods to image approximation aim at recovering such u by either solving an associated Partial Differential Equation (PDE) or by minimizing a specific variational energy, both depending on g. In particular, edge-driven methods recognize the portions of the image contoured by sharp variations of intensity (discontinuities) and associate them to different objects constituting the image subject. Therefore, meaningful approximations are obtained from edge-based methods when they are able to: (1) discriminate between intensity variations due to noise and those due to the presence of object edges, and, (2) return regular approximations where smoothing only reduces noise contaminations without affecting relevant edges. PDE methods are mainly based on diffusion equations. The most popular equation models are anisotropic diffusion (AD) [1] and the total variation (TV) [2] (being the latter a particular case of the former one). The main characteristic of these models is that, the smoothing process induced by diffusion is inhibited according to local features of the image in order to preserve edges. Both AD and TV are time-dependent 2nd-order PDEs. Also 4th-order PDEs are considered in literature [3] to enhance edge preservation. In most of the cases, PDEs can be seen as flows generated by variational energies. In this respect, TV and AD are considered 1st-order models as equations can be derived as flows generated by the minimization of integral energies penalizing the gradient norm. The PDE in [3] is instead associated to a minimization problem penalizing the Hessian norm, it is therefore considered a 2nd-order variational model.

In general, PDE methods do not allow the solutions to have free discontinuities and the physical meaning of the equations parameters are not fully understood [4], [5]. As a consequence, the progressive modification of PDEs to obtain more meaningful solutions moved towards variational representations. By means of variational models the class of admissible solutions can be extended to discontinuous functions. Mumford and Shah (MS) [6] proposed a flexible variational model to image approximation based on *free discontinuities*. Given an image $g : \Omega \to \mathbb{R}$

M. Zanetti and L. Bruzzone are with the Department of Information Engineering and Computer Science, University of Trento, Trento I-38123, Italy. e-mail: {massimo.zanetti,lorenzo.bruzzone}@unitn.it.

with $\Omega \subset \mathbb{R}^2$ a rectangular domain, one looks for a 1-dimensional set $K \subset \mathbb{R}^2$ and a (piecewise) smooth function $u : \Omega \to \mathbb{R}$ such that the energy functional

$$\mathcal{MS}(K, u) = \int_{\Omega \setminus K} |\nabla u|^2 + \mu |u - g|^2 \, dx + \alpha \mathcal{H}^1(K) \tag{1}$$

is minimized. Here, μ, α are positive parameters and $\mathcal{H}^1$ is the 1-dimensional Hausdorff measure. Theoretical results that establish existence and regularity of solutions have been found exploiting a weak formulation of the problem in the space of Special Functions of Bounded Variation [7]–[9]. From a numerical viewpoint, the explicit computation of a solution is a difficult problem. Among many strategies that have been proposed in literature to solve this problem (see for instance [10]–[12]), we recall the Ambrosio-Tortorelli (AT) elliptic approximation via Γ-convergence [13], which is numerically tractable [14]. Being a 1st-order model, the MS has some drawbacks. In particular, the staircasing effect [15], [16] is of major relevance, as it often limits the applicability of this model in practical situations. Briefly, this phenomenon can be explained as follows. The minimization of the gradient energy forces the solution to have a piecewise constant behavior. Therefore, steep gradients are approximated by step-wise functions with many fictitious discontinuities, as the solution is not allowed to have 1st-order variations of high magnitude. This problem can be solved by replacing the gradient term of the energy by a 2nd-order operator. Indeed, this is the solution introduced by Blake and Zisserman (BZ) [17], who proposed to penalize the Hessian (instead of the gradient) and the *size* of K_0, K_1, the discontinuity and the gradient discontinuity sets of u, respectively. The BZ approximation can be found by minimizing

$$\begin{aligned} \mathcal{BZ}(K_0, K_1, u) = & \int_{\Omega \setminus (K_0 \cup K_1)} |\mathrm{H}u|^2 + \mu |u - g|^2 \, dx \\ & + \alpha \mathcal{H}^1(K_0) + \beta \mathcal{H}^1(K_1) \end{aligned} \tag{2}$$

where μ, α, β are positive parameters. Hessian penalization allows the solution to have 1st-order variations outside $K_0 \cup K_1$, yielding to a piecewise linear approximation of the input image. A recent survey presents a summary and future perspectives about the study of the Blake-Zisserman (BZ) variational model for segmentation, including theoretical results for existence and regularity of solutions [18]. To address numerical minimization, elliptic approximations of the functional exploiting the AT technique (used in the MS case) were given by Bellettini and Coscia [19] and Ambrosio, Faina and March (AFM) [20]. The first numerical implementations are given in dimension one [21] to piecewise linear approximation of signals and in dimension two [20] to segmentation of stereo images. Recently, the problem of numerically minimizing the AFM approximation of the BZ functional on large images has been addressed in [22], where the objective functional is written in a compact matrix form and optimization is performed by means of a special version of the block-coordinate descent algorithm (BCDA) [23] that exploits the partial convexity of the functional. Other papers consider the problem of minimizing 2nd-order energies in specific spaces of functions of bounded variation, without the need of tracing the discontinuity sets [24].

The PDE approaches to image approximation mentioned before have been successfully generalized to the case of vector-valued inputs. In [25], a vector-valued version of image restoration based on TV norm has been proposed for color images. A general framework for AD to vector-valued image restoration/enhancement has been proposed in [26], which is applicable to both color images and to other vector-valued image representations (e.g., stacks of image features like texture, motion, etc.). Curvature-preserving tensor-driven PDEs have been also designed to enhance regularity of edge boundaries [27]. Regarding variational methods, large attention has been devoted to the study of 1st-order models in the vector-valued case. A gradient vector-flow approach has been studied in [28]. Fundamental theoretical results have been proved for the MS problem in the vector-valued case by generalizing the AT approach [29], [30] and allowing also for explicit computations [16], [31]. Other approaches have been also developed for the vector-valued MS by exploiting convex representation [32], by extending the active contour algorithm [33], [34] or via combinatorial optimization [35]. In [36] the local AT approximation of the MS is extended to non-local formulation accounting for texture information. Numerical experiments in the cited works confirmed for the improved capability of approximation of variational models w.r.t. to PDE approaches such as TV and AD.

On the counterpart, the current limitation of the literature is the absence of methods for addressing the approximation of vector-valued inputs based on 2nd-order models. In particular, the minimization of the BZ functional for vector-valued functions is not considered at all. In 1st-order models such as MS, the gradient penalization forces the solution to be piecewise constant. If this has useful implications to segmentation purposes, it makes image regularization/restoration unfeasible as a locally flat approximation is generally too coarse. In this paper, we propose a numerical approach to solve the image approximation problem based on the 2nd-order Blake-Zisserman functional for vector-valued functions. From a theoretical viewpoint, we prove that the discrete version of the objective functional involving tensor differential operators retains partial convexity with respect to the new variable blocks associated to co-domain dimensions. To demonstrate its effectiveness, the proposed method is applied to difficult color image denoising/restoration problems and to the recovery of polygonal boundaries from discrete noisy sampling.

The plan of the paper is as follows. In Section II, we firstly introduce the elliptical AFM approximation of the BZ functional and then we purpose two numerical approaches to address minimization in the case of vector-valued images and curves. Numerical experiments are presented in the next two sections. In Section III, the proposed algorithm to piecewise linear approximation of vector-valued images is applied to denoising/restoration of color images and compared with the well-known MS model. In Section IV, the proposed method is applied to the recovery of polygonal shapes from discrete noisy sampling. In Section V we draw the conclusions of this paper.

II. The Blake-Zisserman model for the approximation of vector-valued images and curves

The numerical handling of the 2nd-order variational model to segmentation proposed by Blake and Zisserman is unfeasible in its original strong formulation as in (2) [17]. As in the MS case, the strong formulation does not allow to prove existence of solutions: because of the set unknowns K_0, K_1, the functional lacks in lower semicontinuity. Therefore, the functional has been rewritten in the weaker space of Generalized Special Functions of Bounded Variation [37]

$$\begin{aligned}\mathcal{F}(u) = &\int_\Omega \Big(\mu|u-g|^2 + |\mathrm{H}u|^2\Big)\,dx \\ &+ (\alpha-\beta)\mathcal{H}^1(S_u) + \beta\mathcal{H}^1(S_{\nabla u}\cup S_u)\end{aligned} \tag{3}$$

where it loses its explicit dependency on the discontinuity sets, which can be regarded as geometrical properties of the only variable function u being $S_u, S_{\nabla u}$ the discontinuity and gradient discontinuity set of u, respectively. Also in the weak form the numerical minimization is still a hard issue. For this reason, elliptic approximations of the functional have been proved via Γ-convergence by exploiting the seminal idea of Ambrosio-Tortorelli for the approximation of the MS functional. This is done in dimension one by Bellettini and Coscia [19], and in dimension two (with partial results in any finite dimension) by Ambrosio, Faina and March [20].

Let $\Omega \subset \mathbb{R}^k$ be an open set, $k = 1, 2$ and $g \in L^\infty(\Omega)$ the function to be approximated (either a signal or an image). The authors of [20] and [19] have introduced two auxiliary functions $s, z : \Omega \to [0, 1]$ aimed at approximating the indicator functions of the discontinuity sets K_0, K_1 and proposed a Γ-convergence approximation of the weak functional via the family of uniformly elliptic functionals

$$\begin{aligned}\mathcal{F}_\epsilon(s,z,u) = &\,\delta\int_\Omega z^2|\mathrm{H}u|^2\,dx + \xi_\epsilon\int_\Omega (s^2+o_\epsilon)|\nabla u|^2\,dx \\ &+ (\alpha-\beta)\mathcal{AT}_\epsilon(s) + \beta\mathcal{AT}_\epsilon(z) + \mu\int_\Omega |u-g|^2\,dx,\end{aligned} \tag{4}$$

where s, z, u are in proper Sobolev spaces and $\mathcal{AT}$ is the Ambrosio-Tortorelli component

$$\mathcal{AT}_\epsilon(v) = \int_\Omega \epsilon|\nabla v|^2 + \frac{1}{4\epsilon}(v-1)^2\,dx. \tag{5}$$

Here ϵ is the convergence continuous parameter, ξ_ϵ, o_ϵ are infinitesimals and the convergence is intended for $\epsilon \to 0$. For each $\epsilon > 0$ the functional $\mathcal{F}_\epsilon$ admits a minimizing triplet $(s_\epsilon, z_\epsilon, u_\epsilon)$. Γ-convergence properties ensure that the sequence of minimizers $\{(s_\epsilon, z_\epsilon, u_\epsilon)\}_{\epsilon\to 0}$ strongly converges to a minimizer of the weak functional (3). Fixed $\epsilon > 0$, the geometrical behavior of a minimizing triplet $(s_\epsilon, z_\epsilon, u_\epsilon)$ is as follows. Due to the presence of the distance term $|u-g|^2$ the function u_ϵ is forced to be close to the input and smoothing constraints are given by the integral terms containing $|\nabla u|^2$ and $|\mathrm{H}u|^2$. For $0 < \epsilon << 1$ we have $1/4\epsilon >> 1$, thus s_ϵ and z_ϵ must be 1 almost everywhere.

Transitions from 1 to 0 are only energetically convenient to suppress high values of $|\nabla u|^2$ and $|\mathrm{H}u|^2$. Functional parameters $\delta, \mu, \alpha, \beta$ regulate the penalization of each term individually.

In the following, the AFM approximation of the BZ functional is considered in the general case of vector valued inputs, both for $k = 2$ (the case of vector-valued images) and for $k = 1$ (the case of curves in general space). The functionals are then discretized, written in matrix compact formulation and an efficient numerical algorithm to minimization is proposed.

A. *Approximation of vector-valued images*

Let $\Omega \subset \mathbb{R}^2$ be a rectangular domain and $g : \Omega \to \mathbb{R}^B$, with $B \geq 1$, a vector-valued image (such as RGB or multi-spectral image). Our aim is to derive a discretization and minimization approach to find $u : \Omega \to \mathbb{R}^B$ and $s, z : \Omega \to [0, 1]$ that minimize the functional (4). Of course, differential operators appearing in the functional must be intended as for vector valued functions. Thus, for a (sufficiently) differentiable function $v : x \mapsto (v_1(x), \dots, v_B(x))$, the symbols ∇v and $\mathrm{H}v$ refer to the vector-valued gradient (Jacobian matrix) and Hessian tensor of the function v, respectively. More specifically,

$$\begin{aligned} [\nabla v]_{bk} &= \partial_k v_b \\ [\mathrm{H}v]_{bkh} &= \partial_{kh} v_b \end{aligned} \tag{6}$$

where $k, h = 1, 2$ represent derivative order and $b = 1, \dots, B$ is the coordinate component of the variable v. The squared Euclidean norm of ∇u and $\mathrm{H}u$ is the sum of each squared tensor element

$$\begin{aligned} |\nabla v|^2 &= \sum_{b=1}^{B} \sum_{k=1}^{2} (\partial_k v_b)^2, \\ |\mathrm{H}v|^2 &= \sum_{b=1}^{B} \sum_{k,h=1}^{2} (\partial_{kh} v_b)^2. \end{aligned} \tag{7}$$

1) Discretization: The rectangular planar domain $\Omega \subset \mathbb{R}^2$ is discretized by a rectangular grid of points $\Lambda = \{(it_x, jt_y); i = 1, \dots, I,\ j = 1, \dots, J\}$ with step sizes t_x and t_y on the x and y directions respectively. The overall number of points in the grid is $p = IJ$. At each grid point (it_x, jt_y), the value of variables s, z is given in a usual gray-scale image notation as s_{ij}, z_{ij}, whereas the value of variables g, u are given band-wise, for each band b, as $(g_b)_{ij}, (u_b)_{ij}$. For ease of computation the values of variable $v \in \{g_b, u_b, s, z\}$ are rearranged by column-wise vectorization into a vector $\mathbf{v}$ of size p. The function $w(i, j) := (j - 1)I + i$ makes the bijective correspondence $[\mathbf{v}]_{w(i,j)} = v_{ij}$.

First and second order derivatives are approximated via finite difference-schemes

$$\begin{aligned} \partial_x v_{ij} &:= \frac{v_{i+1,j} - v_{i,j}}{t_x} = [\mathbf{D}_x \mathbf{v}]_{w(i,j)} \\ \partial_y v_{ij} &:= \frac{v_{i,j+1} - v_{i,j}}{t_y} = [\mathbf{D}_y \mathbf{v}]_{w(i,j)} \\ \partial_{xx} v_{ij} &:= \frac{v_{i+1,j} - 2v_{i,j} + v_{i-1,j}}{t_x^2} = [\mathbf{D}_{xx} \mathbf{v}]_{w(i,j)} \\ \partial_{yy} v_{ij} &:= \frac{v_{i,j+1} - 2v_{i,j} + v_{i,j-1}}{t_y^2} = [\mathbf{D}_{yy} \mathbf{v}]_{w(i,j)} \\ \partial_{xy} v_{ij} &:= \frac{1}{t_y} \left(\frac{v_{i+1,j+1} - v_{i,j+1}}{t_x} - \frac{v_{i+1,j} - v_{i,j}}{t_x} \right) \\ &= [\mathbf{D}_{xy} \mathbf{v}]_{w(i,j)} \end{aligned} \tag{8}$$

for $i = 1, \dots, I$ and $j = 1, \dots, J$, and matrices $\mathbf{D}_x$, $\mathbf{D}_y$, $\mathbf{D}_{xx}$, $\mathbf{D}_{yy}$ are given by

$$\begin{aligned} \mathbf{D}_x &:= \frac{1}{t_x} \mathbf{I}_M \otimes \mathbf{A}_N^1 \qquad \mathbf{D}_y := \frac{1}{t_y} \mathbf{A}_M^1 \otimes \mathbf{I}_N \\ \mathbf{D}_{xx} &:= \frac{1}{t_x^2} \mathbf{I}_M \otimes \mathbf{A}_N^2 \qquad \mathbf{D}_{yy} := \frac{1}{t_y^2} \mathbf{A}_M^2 \otimes \mathbf{I}_N \\ \mathbf{D}_{xy} &:= \mathbf{D}_y \mathbf{D}_x = \mathbf{D}_x \mathbf{D}_y \end{aligned} \tag{9}$$

where $\otimes$ is the Kronecker product. Here $\mathbf{I}_K$ denotes the identity matrix of size K and $\mathbf{A}_K^1$, $\mathbf{A}_K^2$ are square matrices of size K implementing difference schemes approximating first and second order derivatives

$$\mathbf{A}_K^1 = \begin{pmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \\ & & & & -1 \end{pmatrix} \mathbf{A}_K^2 = \begin{pmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{pmatrix} \tag{10}$$

Let $\mathbf{v}^2$ be the vector of the squared coefficients of $\mathbf{v}$, i.e., $[\mathbf{v}^2]_i = ([\mathbf{v}]_i)^2$, we can approximate differential operators over each grid point as

$$\begin{aligned} |\mathrm{H}v_{ij}|^2 &= ([\mathbf{D}_{xx}\mathbf{v}]_{w(i,j)})^2 + ([\mathbf{D}_{yy}\mathbf{v}]_{w(i,j)})^2 + 2([\mathbf{D}_{xy}\mathbf{v}]_{w(i,j)})^2, \\ |\nabla v_{ij}|^2 &= ([\mathbf{D}_x\mathbf{v}]_{w(i,j)})^2 + ([\mathbf{D}_y\mathbf{v}]_{w(i,j)})^2, \end{aligned}$$

Finally, by denoting $\mathbf{R}_\mathbf{v}$ the diagonal matrix with diagonal entries equal to the elements of $\mathbf{v}$ and $\mathbf{e} := (1, 1, \dots, 1)^T$, we can obtain a convenient discrete version of the objective functional by a 2-D composite rectangular rule. Indeed, by virtue of expressions (6), the decomposition of the vector-valued variables $\mathbf{g} = (\mathbf{g}_1, \dots, \mathbf{g}_B)$ and $\mathbf{u} = (\mathbf{u}_1, \dots, \mathbf{u}_B)$ can be conveniently split over image bands, so that we can write the discrete functional generalizing (4) to the vector-valued case as

$$\begin{aligned} &F_\epsilon(\mathbf{s}, \mathbf{z}, \mathbf{u}) = \\ &\sum_{b=1}^{B} \Big\{ \delta\, \mathbf{u}_b^T \left[\mathbf{D}_{xx}^T \mathbf{R}_{\mathbf{z}^2} \mathbf{D}_{xx} + \mathbf{D}_{yy}^T \mathbf{R}_{\mathbf{z}^2} \mathbf{D}_{yy} + 2\mathbf{D}_{xy}^T \mathbf{R}_{\mathbf{z}^2} \mathbf{D}_{xy} \right] \mathbf{u}_b \\ &+ \xi_\epsilon\, \mathbf{u}_b^T \left[\mathbf{D}_x^T \mathbf{R}_{\mathbf{s}^2} \mathbf{D}_x + \mathbf{D}_y^T \mathbf{R}_{\mathbf{s}^2} \mathbf{D}_y \right] \mathbf{u}_b + \mu\, (\mathbf{u}_b - \mathbf{g})^T (\mathbf{u}_b - \mathbf{g}) \Big\} \\ &+ (\alpha - \beta) \left[\epsilon\, \mathbf{s}^T (\mathbf{D}_x^T \mathbf{D}_x + \mathbf{D}_y^T \mathbf{D}_y) \mathbf{s} + \frac{1}{4\epsilon} (\mathbf{s} - \mathbf{e})^T (\mathbf{s} - \mathbf{e}) \right] \\ &+ \beta \left[\epsilon\, \mathbf{z}^T (\mathbf{D}_x^T \mathbf{D}_x + \mathbf{D}_y^T \mathbf{D}_y) \mathbf{z} + \frac{1}{4\epsilon} (\mathbf{z} - \mathbf{e})^T (\mathbf{z} - \mathbf{e}) \right]. \end{aligned} \tag{11}$$

The functional presents an evident partially quadratic structure as it can be written in the following way

$$\begin{aligned} &F_\epsilon(\mathbf{s}, \mathbf{z}, \mathbf{u}) \\ &\quad = \frac{1}{2} \left(\mathbf{s}^T\, \mathbf{z}^T \right) \begin{pmatrix} \mathbf{A}_s & 0 \\ 0 & \mathbf{A}_z \end{pmatrix} \begin{pmatrix} \mathbf{s} \\ \mathbf{z} \end{pmatrix} - \left(\mathbf{s}^T\, \mathbf{z}^T \right) \begin{pmatrix} \mathbf{b}_s \\ \mathbf{b}_z \end{pmatrix} \\ &\quad = \frac{1}{2} \left(\mathbf{u}_1^T, \dots, \mathbf{u}_B^T \right) \begin{pmatrix} \mathbf{A}_u & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathbf{A}_u \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_B \end{pmatrix} \\ &\qquad - \left(\mathbf{u}_1^T, \dots, \mathbf{u}_B^T \right) \begin{pmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_B \end{pmatrix} \end{aligned} \tag{12}$$

where $\mathbf{A}_s = \mathbf{A}_s(\mathbf{u})$, $\mathbf{A}_z = \mathbf{A}_z(\mathbf{u})$, $\mathbf{A}_u = \mathbf{A}_u(\mathbf{s}, \mathbf{z})$ and $\mathbf{b}_s, \mathbf{b}_z, \mathbf{b}_b$ for $b = 1, \dots, B$, are given by

$$\begin{aligned}
\mathbf{A}_s &= 2\xi_\epsilon \sum_{b=1}^{B} \mathbf{R}_{|\nabla \mathbf{u}_b|^2} + 2\epsilon(\alpha-\beta)(\mathbf{D}_x^T\mathbf{D}_x + \mathbf{D}_y^T\mathbf{D}_y) + \frac{\alpha-\beta}{2\epsilon}\mathbf{I} \\
\mathbf{b}_s &= \frac{\alpha-\beta}{2\epsilon}\mathbf{e} \\
\mathbf{A}_z &= 2\delta \sum_{b=1}^{B} \mathbf{R}_{|\mathbf{H}\mathbf{u}_b|^2} + 2\epsilon\beta(\mathbf{D}_x^T\mathbf{D}_x + \mathbf{D}_y^T\mathbf{D}_y) + \frac{\beta}{2\epsilon}\mathbf{I} \\
\mathbf{b}_z &= \frac{\beta}{2\epsilon}\mathbf{e} \\
\mathbf{A}_u &= 2\delta(\mathbf{D}_{xx}^T\mathbf{R}_{\mathbf{z}^2}\mathbf{D}_{xx} + \mathbf{D}_{yy}^T\mathbf{R}_{\mathbf{z}^2}\mathbf{D}_{yy} + 2\mathbf{D}_{xy}^T\mathbf{R}_{\mathbf{z}^2}\mathbf{D}_{xy}) + \\
&\qquad 2\xi_\epsilon(\mathbf{D}_x^T\mathbf{R}_{\mathbf{s}^2}\mathbf{D}_x + \mathbf{D}_y^T\mathbf{R}_{\mathbf{s}^2}\mathbf{D}_y) + 2\mu\mathbf{I} \\
\mathbf{b}_b &= 2\mu\mathbf{g}_b
\end{aligned} \tag{13}$$

It is worth mentioning here two facts. First, it is well recognized that, when dealing with vector-valued images the composition of separate processing on image components (i.e., bands) introduces artifacts [38]. Due to the tensor nature of the differential operators used in our model, we can avoid this phenomenon, cfr. Figure 6 (see also [39], [40]). In fact, we have that matrices $\mathbf{A}_s, \mathbf{A}_z$ incorporate information from all image bands as Hessian and gradient norms from all bands are summed up. Thus, functions $\mathbf{s}, \mathbf{z}$ are able to detect discontinuity and gradient discontinuity points gathering information from all bands of the input image. Second, the discrete functional retains quadratic structure with respect to each variable block $\mathbf{u}_b$, for $b = 1, \dots, B$. Moreover, each partially quadratic slice of the functional depends on the same matrix $\mathbf{A}_u$ and only the constant terms $\mathbf{b}_b$ vary among them. This fact has some relevance when numerical minimization is performed, as eigenvalue analysis to determine convergence parameters can be performed only once for each outer iteration.

2) Minimization strategy: The partially quadratic structure expressed in (12) allows us to address the functional minimization by following a Gauss-Seidel (GS) approach

$$\begin{cases} \mathbf{s}^{k+1} = \arg\min_{\mathbf{s}} F_\epsilon(\mathbf{s}, \mathbf{z}^k, \mathbf{u}^k) \\ \mathbf{z}^{k+1} = \arg\min_{\mathbf{z}} F_\epsilon(\mathbf{s}^{k+1}, \mathbf{z}, \mathbf{u}^k) \\ \mathbf{u}_b^{k+1} = \arg\min_{\mathbf{u}_b} F_\epsilon(\mathbf{s}^{k+1}, \mathbf{z}^{k+1}, \dots, \mathbf{u}_b, \dots) \end{cases} \tag{14}$$

where $b = 1, \dots, B$. Indeed, partial descent can be implemented along each variable block with respect to which the functional is quadratic. In order to enhance computing performance, an inexact approach based on Block-Coordinate Descent Algorithm (BCDA) [23] can be used with a modification accounting for the separable B problems involving the $\mathbf{u}$ variable. The proposed modified scheme, called VBCDA (vector-valued BCDA) is outlined in Algorithm 1.

In order to find suitable gradient related search directions $\mathbf{d}_s^k$, $\mathbf{d}_z^k$ and $\mathbf{d}_{u_b}^k$, a few iterations of a PCG solver can be applied to the linear systems $\mathbf{A}_s^k\mathbf{d}_s = \mathbf{b}_s - \mathbf{A}_s^k\mathbf{s}^k$, $\mathbf{A}_z^k\mathbf{d}_z = \mathbf{b}_z - \mathbf{A}_z^k\mathbf{z}^k$ and $\mathbf{A}_u^k\mathbf{d}_{u_b} = \mathbf{b}_b - \mathbf{A}_u^k\mathbf{u}_b^k$, where $b = 1, \dots, B$. The inexact solution of these systems can be stopped according to tolerance values that guarantee the convergence of the overall algorithm to a stationary point of the objective energy (11). The calculation of such tolerance values can be easily done by following the approach proposed in [22], which is based on bound estimates of eigenvalues of matrices $\mathbf{A}_s, \mathbf{A}_z, \mathbf{A}_u$. As previously mentioned, all B problems related to the $\mathbf{u}$ variable depend on the same matrix $\mathbf{A}_u$, thus one single tolerance value can be used to solve all B partial minimization steps based on PCG in the $\mathbf{u}$ variable. Moreover, being each one of these B sub-problems independent from the others, the computational burden of Step 3 in Algorithm 1 can be split over multiple cores (if available) in a parallel way.

Being the global energy non-convex, the initialization step is crucial as it has strong impact on the significance of the final results. Following [20], [22] energetically convenient choices for the first iterates are $\mathbf{s}^0 = \mathbf{e}$, $\mathbf{z}^0 = \mathbf{e}$ and $\mathbf{u}^0 = \mathbf{g}$. This is motivated by the fact that, the functions s, z that minimize the theoretical model are 1 almost everywhere over Ω, whereas u is to be considered an approximation of g. The method can be stopped at the iteration k such that the relative variation of the functional satisfies the following condition

$$\left| \frac{F_\epsilon(\mathbf{u}^k, \mathbf{s}^k, \mathbf{z}^k) - F_\epsilon(\mathbf{u}^{k-1}, \mathbf{s}^{k-1}, \mathbf{z}^{k-1})}{F_\epsilon(\mathbf{u}^k, \mathbf{s}^k, \mathbf{z}^k)} \right| < tol_F. \tag{15}$$

Algorithm 1 VBCDA

,

Input: $\mathbf{s}^0$, $\mathbf{z}^0$, $\mathbf{u}^0$, $\gamma_s = \gamma_z = 1$, $\gamma_u = 1.5$;

Step 1: $k = 0$;

Step 2: Inexact minimization with respect to $\mathbf{s}$ and $\mathbf{z}$:

- compute the search directions $\mathbf{d}_s^k$ and $\mathbf{d}_z^k$;
- compute $\alpha_s^k = \gamma_s \frac{-(\mathbf{A}_s^k \mathbf{s}^k - \mathbf{b}_s)^T \mathbf{d}_s^k}{\mathbf{d}_s^{k\,T} \mathbf{A}_s^k \mathbf{d}_s^k}$;
- update $\mathbf{s}^{k+1} = \mathbf{s}^k + \alpha_s^k \mathbf{d}_s^k$;
- compute $\alpha_z^k = \gamma_z \frac{-(\mathbf{A}_z^k \mathbf{z}^k - \mathbf{b}_z)^T \mathbf{d}_z^k}{\mathbf{d}_z^{k\,T} \mathbf{A}_z^k \mathbf{d}_z^k}$;
- update $\mathbf{z}^{k+1} = \mathbf{z}^k + \alpha_z^k \mathbf{d}_z^k$.

Step 3: Inexact minimization with respect to $\mathbf{u}_1, \dots, \mathbf{u}_B$. For each $b = 1, \dots, B$:

- compute the search direction $\mathbf{d}_{u_b}^k$;
- compute $\alpha_{u_b}^k = \gamma_u \frac{-(\mathbf{A}_u^k \mathbf{u}_b^k - \mathbf{b}_b)^T \mathbf{d}_{u_b}^k}{\mathbf{d}_{u_b}^{k\,T} \mathbf{A}_u^k \mathbf{d}_{u_b}^k}$
- update $\mathbf{u}_b^{k+1} = \mathbf{u}_b^k + \alpha_{u_b}^k \mathbf{d}_{u_b}^k$.

Step 4: Set $k = k + 1$ and go to Step 2, until convergence;

Several numerical experiments on different datasets allowed us to conclude that the tolerance value can be set to $tol_F = 10^{-3}$.

B. Approximation of vector-valued curves

The capability of the BZ model to recover piecewise linear approximation of data can be exploited also to approximate curves in N-space. As possible interesting applications we mention here: (1) curve (or signal) rectification, and, (2) recovering of polygonal shapes from noisy sampling. Theoretical results and a first implementation about the Blake-Zisserman model in dimension-one have only been given for scalar functions (i.e., signals) in [19], [21]. In this section we propose a more general framework for vector-valued curves and an efficient numerical algorithm that exploits the results in Section II-A. Therefore, we consider here vector fields where domain dimension is one, that is functions of the type $g : \Pi \to \mathbb{R}^N$, with $\Pi \subset \mathbb{R}$ a closed connected interval on the real line and $N > 0$ integer. Let us utilize the more usual notation for derivatives in dimension one. Given a (sufficiently) derivable function $v : \Pi \to \mathbb{R}^N$, $v : t \mapsto (v_1(t), \dots, v_N(t))$, let us denote first and second order derivatives by

$$\begin{aligned} [v'(t)]_n &= \mathrm{d}_t v_n(t) \\ [v''(t)]_n &= \mathrm{d}_{tt} v_n(t) \end{aligned} \tag{16}$$

with $n = 1, \dots, N$. The functional model that we consider in the following is a reduced version of (4), that does not include the gradient term and the s variable. This reduced model proved to be very useful in the specific task of recovering polygonal closed curves as it is not affected by the slight staircasing effect induced by the gradient term (cfr. with the discussion in Section III-B and analysis of numerical experiments in Section IV). However, for the sake of generality we remark here that all the arguments presented in the following can be easily generalized by taking into account the full version of the functional. Given a curve $g : \Pi \to \mathbb{R}^N$ in N-space, we attempt to find a piecewise linear approximation of g by looking for $u : \Pi \to \mathbb{R}^N$ and $z : \Pi \to [0, 1]$ that minimize

$$\begin{aligned} \mathcal{F}_\epsilon(z, u) = &\int_\Pi z^2 |u''|^2 \, dt + \lambda \int_\Pi |u - g|^2 \, dt \\ &+ \eta \int_\Pi \left\{ \epsilon |z'|^2 + \frac{1}{4\epsilon}(z - 1)^2 \right\} dt \end{aligned} \tag{17}$$

where λ, η are positive parameters regulating the penalization of the corresponding terms.

1) Discretization and minimization strategy: The discretization of the functional (17) follows the same principles as in Section II-A1, with the slight simplification that no Kronecker product is needed when defining the matrix differential operators.[1] In a discrete setting, the function domain is a set of points $t_1, \ldots, t_P \in \Pi$ with $t_p < t_{p+1}$ for all $p = 1, \ldots, P-1$. The discrete variables representing the curves (or signals in case of $N = 1$) are denoted by $\mathbf{v} = (\mathbf{v}_1, \ldots, \mathbf{v}_N)$ with $\mathbf{v} = \mathbf{g}, \mathbf{u}$, and their values at each coordinate n over a point t_p are notated as $[\mathbf{v}_n]_p$.

Let $\mathbf{W}$ be the diagonal matrix with diagonal entries $[\mathbf{W}]_{p,p} = t_{p+1} - t_p$, for $p = 1, \ldots, P-1$, and $[\mathbf{W}]_{P,P} = -t_P$. By considering the difference schemes of size $P \times P$ as in (10), the discrete operators implementing first and second order derivatives can be defined as $\mathbf{D}_t = \mathbf{W}^{-1}\mathbf{A}_P^1$ and $\mathbf{D}_{tt} = \mathbf{W}^{-2}\mathbf{A}_P^2$. Simple modifications to account for different types of boundary conditions are possible. Some useful examples are:

Null-Dirichlet: no modification.
Null-Neumann: $[\mathbf{D}_t]_{P,P} = 0$,
$[\mathbf{D}_{tt}]_{1,1} = [\mathbf{D}_{tt}]_{P,P} = -1$.
Periodic: $[\mathbf{D}_t]_{P,1} = 1$,
$[\mathbf{D}_{tt}]_{1,P} = [\mathbf{D}_{tt}]_{P,1} = 1$.

We are now able to write the discrete version of the BZ functional for vector-valued curves as

$$\begin{aligned} F_\epsilon(\mathbf{z}, \mathbf{u}) = & \\ & \sum_{n=1}^{N} \Big\{ \mathbf{u}_n^T \mathbf{D}_{tt}^T \mathbf{R}_{\mathbf{z}^2} \mathbf{D}_{tt} \mathbf{u}_n + \lambda (\mathbf{u}_n - \mathbf{g}_n)^T (\mathbf{u}_n - \mathbf{g}_n) \Big\} + \\ & \quad \eta \Big\{ \epsilon \big[\mathbf{z}^T \mathbf{D}_t^T \mathbf{D}_t \mathbf{z} \big] + \frac{1}{4\epsilon} (\mathbf{z} - \mathbf{e})^T (\mathbf{z} - \mathbf{e}) \Big\}. \end{aligned} \tag{18}$$

Similarly to the case of vector-valued images, this functional is quadratic with respect to the variables $\mathbf{z}$ and $\mathbf{u}_n$, in fact it can be written as

$$\begin{aligned} & F_\epsilon(\mathbf{z}, \mathbf{u}) \\ & = \frac{1}{2} \mathbf{z}^T \mathbf{A}_z \mathbf{z} - \mathbf{z}^T \mathbf{b}_z \\ & = \frac{1}{2} \left(\mathbf{u}_1^T, \ldots, \mathbf{u}_N^T \right) \begin{pmatrix} \mathbf{A}_u & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathbf{A}_u \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_N \end{pmatrix} \\ & \quad - \left(\mathbf{u}_1^T, \ldots, \mathbf{u}_N^T \right) \begin{pmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_N \end{pmatrix} \end{aligned} \tag{19}$$

where $\mathbf{A}_z = \mathbf{A}_z(\mathbf{u})$, $\mathbf{A}_u = \mathbf{A}_u(\mathbf{z})$, and $\mathbf{b}_z, \mathbf{b}_n$ for $n = 1, \ldots, N$ are given by

$$\begin{aligned} \mathbf{A}_z &= 2 \sum_{n=1}^{N} \mathbf{R}_{|\mathbf{u}_n''|^2} + 2\epsilon\eta \mathbf{D}_t^T \mathbf{D}_t + \frac{\eta}{2\epsilon} \mathbf{I} \\ \mathbf{b}_z &= \frac{\eta}{2\epsilon} \mathbf{e} \\ \mathbf{A}_u &= 2\mathbf{D}_{tt}^T \mathbf{R}_{\mathbf{z}^2} \mathbf{D}_{tt} + 2\lambda \mathbf{I} \\ \mathbf{b}_n &= 2\lambda \mathbf{g}_n \end{aligned} \tag{20}$$

and $|\mathbf{u}_n''|^2 := (\mathbf{D}_{tt}\mathbf{u}_n)^2$. To minimize this functional the same approach proposed in the previous section can be used, with obvious adaptations. For the sake of completeness, we remind here that the general minimization approach is based on a sequential partial minimization of the type

$$\begin{cases} \mathbf{z}^{k+1} = \arg\min_{\mathbf{z}} F_\epsilon(\mathbf{z}, \mathbf{u}^k) \\ \mathbf{u}_n^{k+1} = \arg\min_{\mathbf{u}_n} F_\epsilon(\mathbf{z}^{k+1}, \ldots, \mathbf{u}_n, \ldots) \end{cases} \tag{21}$$

for $n = 1, \ldots, N$, in either an exact or inexact fashion. For technicalities we refer the reader to the previous section.

[1]Kronecker product is used in Section II-A to exploit one dimensional difference schemes to work as partial derivatives on (column-wise or row-wise) vectorized images.

III. Experimental results: piecewise linear approximation of vector-valued images

We propose in this section a comparative analysis between the MS and the proposed BZ approaches to vector-valued image approximation. In particular, test cases are focused on difficult color image restoration/regularization tasks highlighting limitations of the MS model such as the staircasing effect and the crack-tip problem and demonstrating how the BZ model overcomes these issues. Computations are performed using MATLAB® R2015b, hardware is Intel® Core$^{\text{TM}}$ i5-4750 CPU @3.20 GHz, 16.00 GB Ram.

A. *Functional models considered for comparison*

Let us recall, for the sake of clarity, the functional model given by Ambrosio-Faina-March, to which we will refer in the following as **AFM**, as in (4):

$$\begin{aligned}\mathcal{F}_\epsilon(s,z,u) &= \delta\int_\Omega z^2|\mathrm{H}u|^2\,dx + \xi_\epsilon\int_\Omega (s^2+o_\epsilon)|\nabla u|^2\,dx \\ &\quad + (\alpha-\beta)\mathcal{AT}_\epsilon(s) + \beta\mathcal{AT}_\epsilon(z) + \mu\int_\Omega |u-g|^2\,dx. \end{aligned} \tag{22}$$

From a numerical point of view, the presence of the gradient term in the **AFM** functional model introduces a slight staircasing effect in the solution. Although its influence vanishes asymptotically for $\epsilon \to 0$ due to the presence of the infinitesimal ξ_ϵ, in numerical applications ϵ cannot be 0. To account for this, a *reduced* version of the **AFM** functional is considered in the numerical experiments that does not present the staircasing effect. More specifically, the reduced functional, to which we will refer in the following as **AT-BZ**, is:

$$\mathcal{F}_\epsilon(z,u) = \int_\Omega z^2|\mathrm{H}u|^2\,dx + \beta\mathcal{AT}_\epsilon(z) + \mu\int_\Omega |u-g|^2\,dx. \tag{23}$$

In this reduced version, the functional does not depend anymore on the gradient of u and the function s. Therefore, the solution is allowed to have 1st-order variations without gradient penalization. This functional formulation is also known in literature as Ambrosio-Tortorelli approximation of the BZ functional. We recall that also this functional is an approximation of the weak BZ functional as a full Γ-convergence result holds true [20]. It is worth noting that, the reduced functional **AT-BZ** can be directly compared with the well-known Ambrosio-Tortorelli approximation of the MS functional, to which we will refer in the following as **AT-MS**, given by:

$$\mathcal{F}_\epsilon(s,u) = \int_\Omega s^2|\nabla u|^2\,dx + \alpha\mathcal{AT}_\epsilon(s) + \mu\int_\Omega |u-g|^2\,dx. \tag{24}$$

Notice that both the functional models **AT-BZ** and **AT-MS** can be obtained from **AFM** by setting functional parameters to

$$\begin{aligned}\textbf{AT-BZ}: &\quad \xi_\epsilon = 0, \alpha = \beta, \delta = 1 \\ \textbf{AT-MS}: &\quad \xi_\epsilon = 1, o_\epsilon = 0, \beta = 0, \delta = 0.\end{aligned}$$

An important consequence of this, is that numerical methods to solve the minimization problems associated to the two functional models **AT-BZ** and **AT-MS** can be derived with straightforward modifications from the method proposed in Section II-A.

B. *Restoration of color images*

The study dataset is a color image representing a portion of the oil painting *Girl with a pearl earring*, by Johannes Vermeer, see Figure 1a. The image has size 600×600 pixels at 8-bits per channel. As we can see, the painting is severely affected by the *craquelure*.[2] In the numerical experiments that follow, we compare the results obtained by using the three functional models to image approximation presented in Section III-A where parameters are specifically selected to remove the craquelure effect.

In order to simplify the comparison among the three considered functional models, we will set some parameters to common values. First, the discretization parameters common to all the three models are the grid sizes t_x, t_y and the Γ-convergence parameter ϵ. They are set to $t_x = t_y = 1$ and $\epsilon = 0.01$ (see [22] Section 3.3 for details on

[2]Craquelure is the fine pattern of dense ”cracking” formed on the surface of the oil as part of the process of ageing.

(a) original image ($\mathbf{g}$)

(b) **AT-MS** ($\mathbf{u}$)

(c) **AFM** ($\mathbf{u}$)

(d) **AT-BZ** ($\mathbf{u}$)

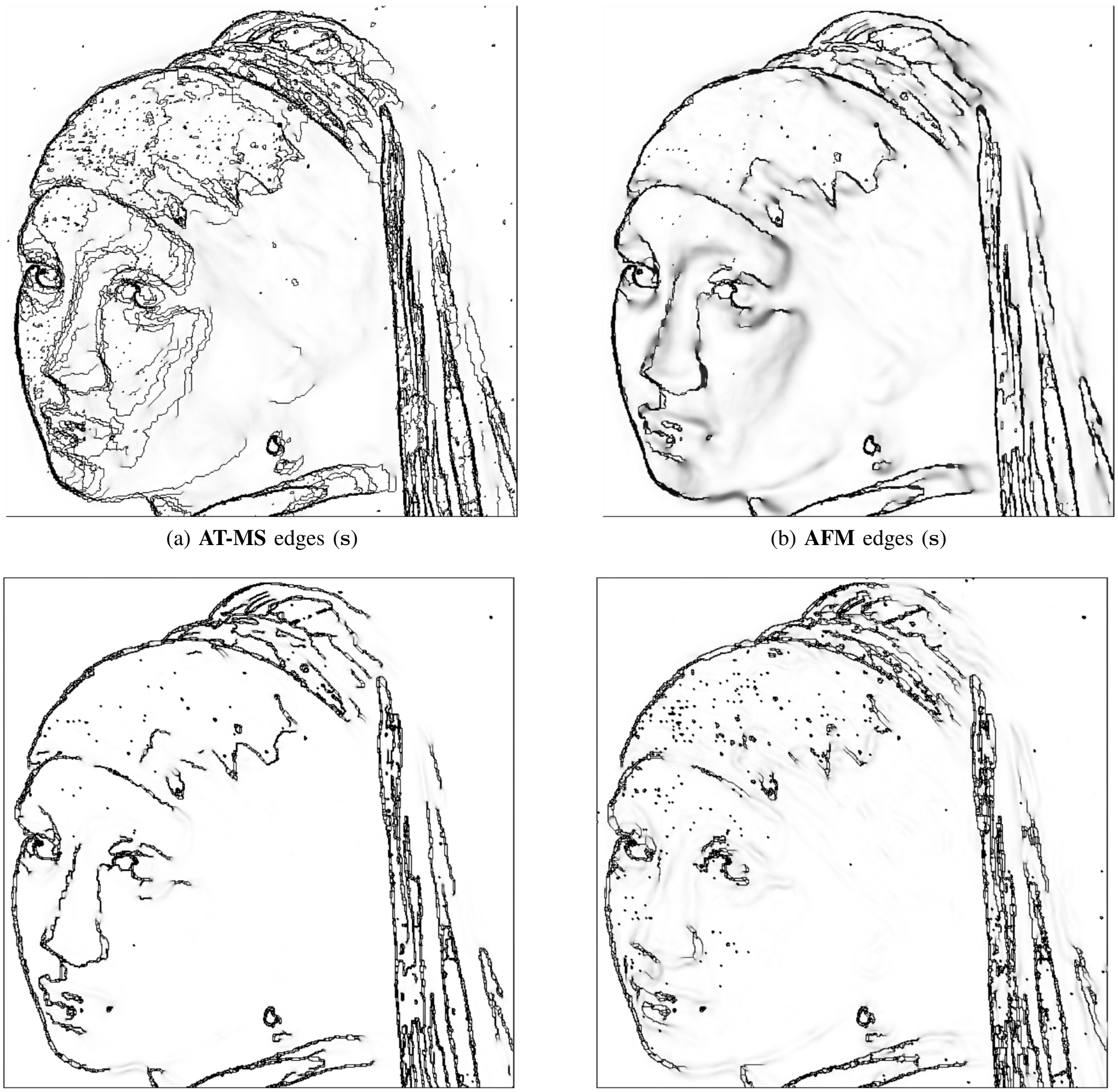
Fig. 1. Craquelure removal via MS and BZ approaches. (a) The input color image represents a portion of the oil painting *Girl with a pearl earring*. Its approximations are obtained by the three different models: (b) **AT-MS**, (c) **AFM**, (d) **AT-BZ**.

how parameter ϵ can be optimized). The remaining ones are the functional parameters: (**AFM**) $\mu, \delta, \alpha, \beta$, (**AT-BZ**) μ, β and (**AT-MS**) μ, α. To keep the maximum similarity among the three functional models, we set the remaining parameters to:

$$\begin{aligned} \textbf{AFM}: &\quad \delta = 1, \alpha = 2, \beta = 1 \\ \textbf{AT-BZ}: &\quad \beta = 1 \\ \textbf{AT-MS}: &\quad \alpha = 1 \end{aligned}$$

The *smoothing* parameter μ is set with respect to the chosen application. By tuning the parameter μ, we forced the smoothing of the image until the approximating image was not showing any craquelure feature. By decreasing the value of this parameter the smoothing effect is increased. We started with $\mu = 1$ and we decreased it by negative powers of 10, i.e., $\mu = 1, 0.1, 0.01, 0.001, \ldots$. The first value at which no craquelure was observable in all the three approximation results was $\mu = 0.01$, so results are showed according to this parameter value.

1) Comparative analysis of the approximations: The major challenge in the given image is to both remove the craquelure and still preserve the smooth color variations between shadowed and lightened regions of the girl's face.



(a) **AT-MS** edges ($\mathbf{s}$) (b) **AFM** edges ($\mathbf{s}$)

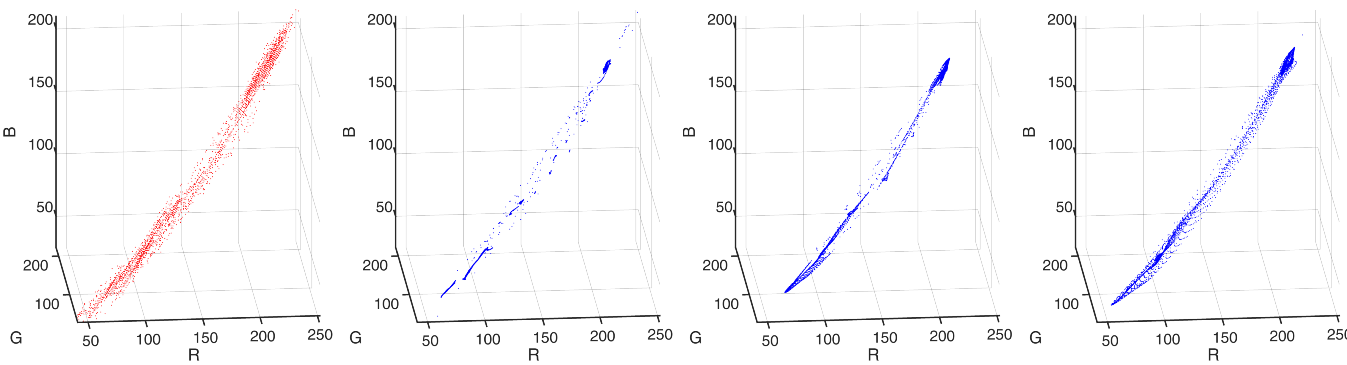
(c) **AFM** creases ($\mathbf{z}$) (d) **AT-BZ** creases ($\mathbf{z}$)

Fig. 2. Discontinuity functions computed for the three functional models. (a) **AT-MS**, (b,c) **AFM**, (d) **AT-BZ**. White corresponds to 1 and black to 0, gray values are in between.

In the **AT-MS** case the staircasing effect is very evident. Many coarse patches of constant color are clearly visible and the gradients of color are abruptly approximated by the edges of such patches, see Figure 1b. The map of these edges is given in the plot of the function $\mathbf{s}$, in Figure 2a. In the solution of **AFM** the over-segmentation is much less evident, as the gradient component is weighted by the infinitesimal ξ_ϵ. However, we can see in Figure 1c mainly two problems in the computed solution $\mathbf{u}$. First, sharp edges are still visible in some portions of the image. See for example both sides of the nose ridge and the lips boundaries. These unnatural gradients of color are mapped by the edge function $\mathbf{s}$ of **AT-BZ**, as shown in Figures 2b and 2c. Second, many complex transitions between shadow and light (that are more evident on the girl's left cheek) are lost. As we can see from the plot of the edge function $\mathbf{s}$, this unwanted effect happens in correspondence of the gray shaded regions of the $\mathbf{s}$ map. Here, the functional has penalized the gradient all over these wide regions instead of only along 1-dimensional edges. As a consequence, the image is badly approximated by a too smooth function and the complex shadow geometry is destroyed. All these problems do not show up in the solution of the **AT-BZ** model, see Figure 1d. Here, only 2nd-order information is penalized and the solution is allowed to have first order variations. As a result, the image is better approximated

(a) original image (**g**) (b) **AT-MS** (**u**) (c) **AFM** (**u**) (d) **AT-BZ** (**u**)

Fig. 3. Particular of the approximations by zooming the area in the red square in Figure 1. (a) Original image, and the results for: (b) **AT-MS**, (c) **AFM**, (d) **AT-BZ**. Notice the over-segmentation effect in (b,c).



(a) original image (**g**) (b) **AT-MS** (**u**) (c) **AFM** (**u**) (d) **AT-BZ** (**u**)

Fig. 4. Pixel scatterplots of the image portions represented in Figure 3. (a) Original image, (b) **AT-MS**, (c) **AFM**, (d) **AT-BZ**.

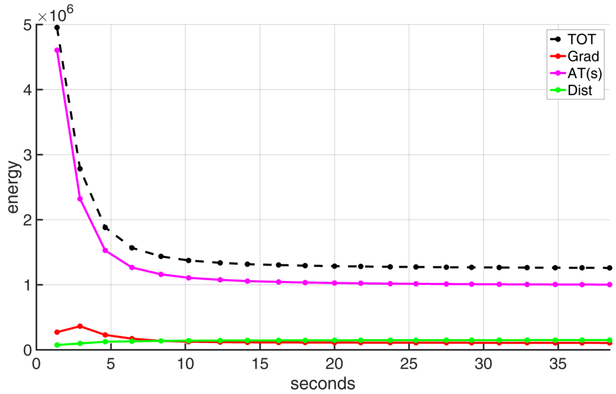
and no unwanted artifacts such as unnatural edges and too coarse shadowed areas are present. As an example, we can see from the map of detected gradient discontinuities $\mathbf{z}$ in Figure 2d that the girl's nose is not contoured by any sharp edge, neither the left cheek is over-segmented.

To better illustrate the geometrical behavior of these solutions, we will show a particular of the images (the red square of size 60×60 pixels in Figure 1a) as embedded surfaces in the RGB space. Magnifications of this part in the original image and in the three computed solutions are showed in Figure 3. The scatter plots of the embedded surfaces are illustrated in Figure 4. The high level of noise of the original image results in a scatter plot where points are almost uniformly distributed all around the RGB space's main diagonal, with two regions where points are slightly denser (corresponding to the dark brown and the light pink regions of the image). In the **AT-MS** case, as a result of the severe over-segmentation effect, pixels are clustered in different almost isolated portions of the RGB space. In the **AFM** case these clusters are slightly enlarged, however they are still distinguishable and sharply separated from background pixels. Much more regular is the scatter plot in the **AT-BZ** case. Here, the dense clusters are visible but they are more displaced in space and surrounding pixels are uniformly and regularly distributed.

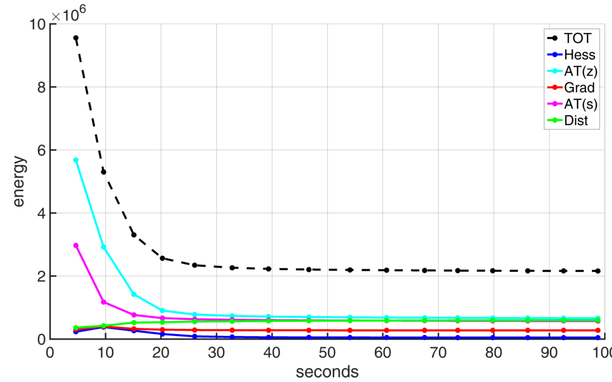
2) Numerical minimization performance: Let us analyze and compare iterations details of the minimization of the three functional models. In Table I are recorded the number of outer (k) and inner (totiter) iterations and the execution time. Inner iterations relate to the PCG solvers (triggered with diagonal preconditioner) applied to find (for each outer iteration) the gradient related search directions w.r.t. the variable blocks. As discussed in Section II-A2, the minimization w.r.t. the u variable can be separated into three convex sub-problems. Therefore, the search of a global gradient related descent direction can be split along the three sub-directions u_1, u_2, u_3 (the target image is color image, the three directions correspond to the R,G,B bands), independently. As we can see, in the case of the s and z variable blocks PCGs stopped in one iteration, whereas the u_i variable blocks required more iterations. This can be explained in terms of positive definiteness of the matrix $\mathbf{A}_u$. It follows from the numerical expression of $\mathbf{A}_u$ in (13) that the positive definiteness of this matrix decreases with the parameter μ. As a consequence, for small

TABLE I
OUTER(K)/INNER(ITER) ITERATIONS AND EXECUTION TIME OBSERVED IN THE MINIMIZATION OF THE THREE FUNCTIONAL MODELS.

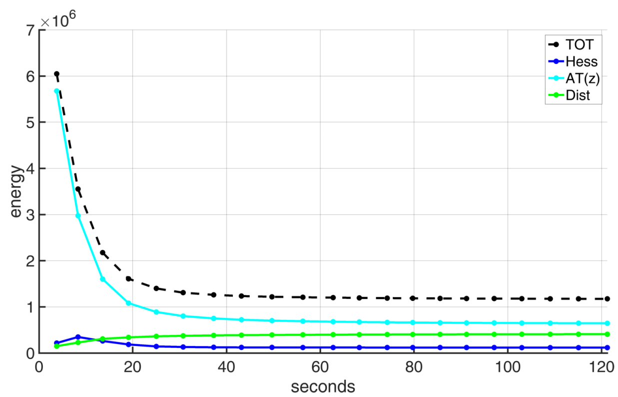
model	k	iter s	iter z	iter u_1	iter u_2	iter u_3	time (secs)
AT-MS	21	22	–	526	300	385	38.45
AFM	16	17	17	1280	742	889	98.75
AT-BZ	21	–	22	1630	877	1110	121.24



(a) **AT-MS**



(b) **AFM**

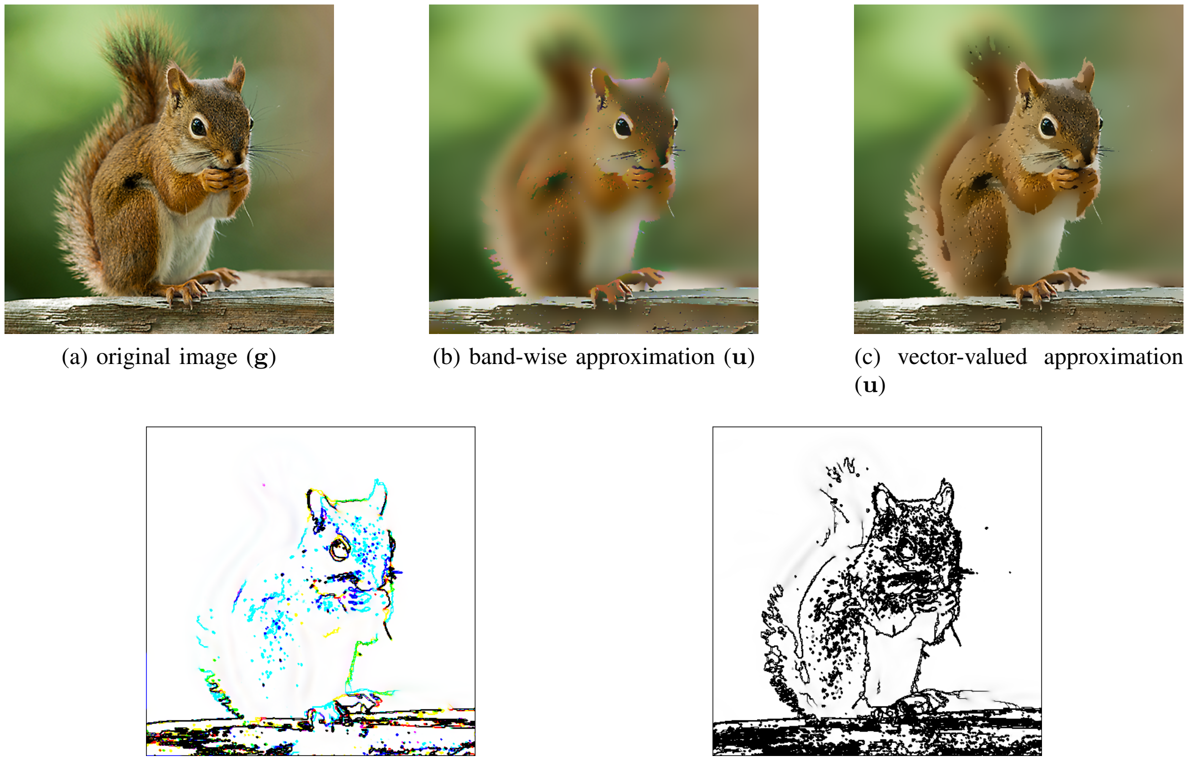


(c) **AT-BZ**

Fig. 5. Energy-versus-time at each outer iteration for the three functional minimization cases: (a) **AT-MS**, (b) **AFM**, (c) **AT-BZ**. The plots illustrate the descent of each additive term in the functional models. The black dashed line is the total energy. Blue is the Hessian component and Cyan is the AT component associated to the Hessian (present only in **AFM** and **AT-BZ**). Red is the gradient component and Magenta is the AT component associated to the gradient (present only in **AT-MS** and **AFM**). Green is the distance term.

values of μ the convexity of the quadratic form associated to $\mathbf{A}_u$ reduces and the descent requires more iterations. To mitigate this behavior, one can apply preconditioning on the PCG solvers if large images are considered (see [22] Section 3.2 for details). Another important aspect of the minimization is the competition among the functional terms induced by the parameters choice. The dynamics of this competition can be better understood by looking at the plots in Figure 5. The main fact that can be observed is that all the terms are decreasing except for the distance term. In particular, the AT components have higher decreasing rates in the first iterations, meaning that the contrast of the corresponding solutions is heavily decreasing.[3] The behavior of the distance term is as expected: due to the strong noise removal level induced by the parameters, the solution becomes more distant (in the Euclidean sense) with respect to the input image.

[3]By premature stopping of the algorithms we could notice many discontinuities that are not present in the final results.



(a) original image ($\mathbf{g}$) (b) band-wise approximation ($\mathbf{u}$) (c) vector-valued approximation ($\mathbf{u}$)

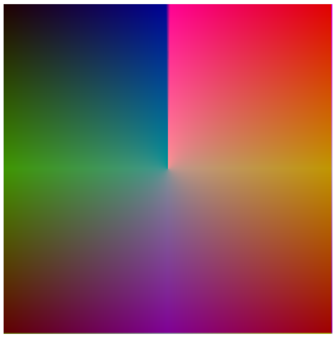
(d) band-wise crease map ($\mathbf{z}$) (e) vector-valued crease map ($\mathbf{z}$)

Fig. 6. Limitations of the band-wise approach for the BZ model. (a) Input image, (b) band-wise approximation, (c) vector-valued approximation, (d) band-wise crease map (the three crease maps obtained in the separate band-wise processing are inserted in the RGB channels), (e) vector-valued crease map. Many sharp edges are missed in the band-wise approach as they are not detectable in single channels. The vector-valued approach performs much better as discontinuities and gradient discontinuities contributions are gathered from all bands.

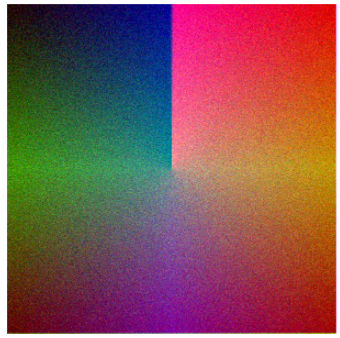
C. Relevant features of the BZ model for vector-valued inputs

1) Limitations of the band-wise approach: Typical limitations of the band-wise approach to the analysis of vector-valued images based on edge-detection models are mainly two: (1) only edges that are clearly distinguishable in single bands can be detected, and (2) obtaining meaningful results always requires separate parameter tuning for each band. Avoiding this last step typically involves artifacts such as meaningless edges or the creation of false colors. This is demonstrated on 1st-order model such as MS in [32]. As shown in Figure 6, these limitations remain valid for 2nd-order models as well. We can see that the band-wise approximation misses many edges (Figure 6b). This happens when gradient or Hessian contributions from single bands are not sufficient to characterize an edge. The RGB composition of detected edge-creases $\mathbf{z}$ in Figure 6d better illustrates this situation. Here, black traits correspond to edges that are detected in all the three bands, whereas other colors correspond to edges that are detected only in single bands or band pairs. For instance, the cyan trait represent edges that are detected in the G and B bands, but not in the R. In correspondence of these edges the band-wise approximation $\mathbf{u}$ is not sharp enough as the R band has been smoothed there. A similar argument works for other colored traits. Notice also the formation of false colors such as orange and green patches in many portions of the image, and magenta between the squirrel ears. The final result is a very poor approximation of the original image. Better results are obtained if the vector-valued approach is used, see Figure 6c. The crease function $\mathbf{z}$, see Figure 6e, gathers gradient and Hessian contributions from all bands by means of tensor differential operators (7). As a result, all relevant edges are correctly detected and the approximated image is sharp in these locations.

2) Preventing the staircasing effect: Recovering an image degraded by additive noise is a well-known inverse problem in image processing. Variational methods have been justified in this framework as the MS model can be properly derived by following a Bayesian rationale as an additive noise reduction model [41]. However, as we have seen also in Section III-B, the MS approximation can irremediably deteriorate some important features of the image because of the staircasing effect. In this experimental section, we aim at showing that the proposed **AT-BZ** model

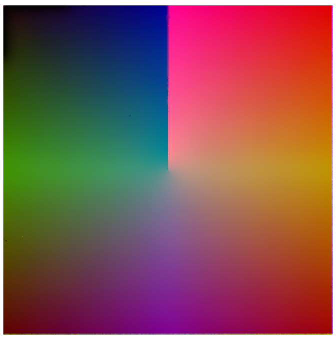


(a) clean image



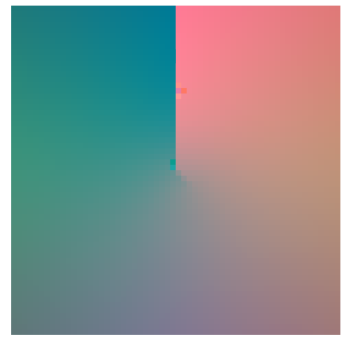
(b) noisy image ($\mathbf{g}$)

(c) **AT-MS** ($\mathbf{u}$)



(d) **AT-BZ** ($\mathbf{u}$)

(e) **AT-MS** ($\mathbf{u}$)



(f) **AT-BZ** ($\mathbf{u}$)

Fig. 7. Estimation of Gaussian additive noise in color image containing challenging geometries. (a) Synthetic noise-free generated image, (b) noisy image. Reconstructions of the noisy image are obtained by the (c) **AT-MS** model, (d) **AT-BZ** model. Particulars zoomed at the crack-tip end for the (e) **AT-MS** solution (white traits emphasize the main directions of the discontinuity edges) and the (f) **AT-BZ** solution.

actually outperforms the **AT-MS** also in terms of noise reduction being able to approximate sloped geometries without falling into step-wise solutions.

In order to do this, we consider here a synthetic color image corrupted by different levels of noise, see Figure 7. The image is reconstructed by using the **AT-MS** and the **AT-BZ** models and in both cases the variance of the removed noise is estimated from the difference image. The synthetic 8-bits (per channel) color image is 300×300 pixels and contains two challenges: (1) a crack-tip (with circular gradient) in the red band, and (2) two very smooth creases in the green and blue bands with vertical and horizontal directions, respectively. Functional parameters are the same as in the previous section. The only difference is that the smoothing parameter has been set to a smaller value $\mu = 0.001$, as the additive noise added in the experiments resulted to be more difficult to suppress. Given that the image is at 8-bits, we added additive 3-dimensional 0-mean Gaussian noise with covariance matrix given by $\Sigma = \sigma^2 \mathbf{I}_3$, where $\mathbf{I}_3$ is the identity matrix of size 3×3, and in three different trials we set $\sigma^2 = 50, 100, 200$.

The results of noise variance estimation are reported in Table II. We can easily see that in all the three cases the **AT-MS** model returned very bad approximations of the noise variance, while the reconstruction given by **AT-BZ** allowed for very precise estimates. This happened because the color geometry of the test image is highly non-constant, thus, the MS fails in approximating both the crack-tip and the orthogonal smooth creases. This fact can be clearly seen by looking at the images of the final approximations obtained via the two functional models, showed in Figure 7 (the images show the results obtained for $\sigma^2 = 200$, being this the most critical case). Notice in particular the behavior of the MS approximation at the end of the crack-tip (Figure 7e), showing the well-known phenomenon of the triple-points. In principle this phenomenon is due to the penalization of the discontinuity set, that induces the discontinuity edges to displace in optimal configurations with minimum length. This happens when they meet at $2/3\pi$ wide angles. Mumford and Shah conjectured in their seminal work [6] that the discontinuity set of a MS minimizer is the union of C^1 arcs that can only end at interior points (pure crack-tips) or meet with equal angles. It has been proved in [42] by using the calibration technique that a function with a triple-point discontinuity is a local minimizer of the homogeneous MS functional. It is worth noting that triple-points do not show up in the BZ approximation, where the solution properly follows the complex crack-tip geometry with surrounding circular

TABLE II
ESTIMATION OF ADDITIVE GAUSSIAN NOISE VARIANCE IN COLOR IMAGES USING THE MS AND THE BZ MODELS.

σ^2	model	estimated σ^2 from u_1	u_2	u_3	k	time (secs)
50	**AT-MS**	74.25	216.33	217.09	18	19.82
	AT-BZ	53.69	49.84	50.30	17	49.63
100	**AT-MS**	121.98	265.68	266.17	10	8.97
	AT-BZ	103.69	99.59	99.96	16	48.05
200	**AT-MS**	215.97	360.38	362.43	13	10.87
	AT-BZ	200.83	197.48	199.80	21	49.31

gradient (Figure 7f).

D. CSIQ dataset

The previous sections of this experimental part showed relevant features of the BZ model and focused on datasets specifically selected for emphasizing certain limitations of 1st-order models that can be overtaken by using 2nd-order models. Here it follows a more extensive analysis on the denoising performance of the BZ model which is done on a subset of images from the public CSIQ database [43].

1) Experiment setting: Images in the dataset are 512×512 pixels at 8-bits per channel. To each image g_0, we add 0-mean Gaussian noise with standard deviation corresponding to 10% of the maximum value of the image, i.e., $\sigma = 25.5$. The noisy image is notated g. The **AT-BZ** model is then applied to each noisy image g to get an approximation u for a set of parameters based on a predefined grid. Parameters values are selected so that the full range of possible behaviors of the approximation (from under- to over-fitting) is appreciable. Specifically, we let the smoothing parameter to take values $\mu = 0.1, 0.5, 1, 5$ and the parameter penalizing the size of the gradient discontinuity set to take the values $\beta = 1, 5, 10, 50, 100, 500$. To assess the performance of the noise reduction we computed quality measures based on: estimated noise variance ($\hat{\sigma}$), peak signal-to-noise ratio (PSNR), signal-to-noise ratio (SNR) and structural similarity (SSIM). In particular we have:

- $\hat{\sigma}_{ref}$: noise std. dev. estimated from $g - g_0$.
- $\hat{\sigma}_{den}$: noise std. dev. estimated from $g - u$.
- PSNR$_{nsy}$: measurement of the image degradation due to noise addition. It is computed for g using g_0 as reference.
- PSNR$_{den}$: measurement of the image denoising quality. It is computed for u using g_0 as reference.
- SNR$_{nsy}$, SNR$_{den}$, SSIM$_{nsy}$, SSIM$_{den}$: are computed similarly to PSNR$_{nsy}$ and PSNR$_{den}$.

2) Experiment results: For each image, the best approximation u^* of g is selected for the parameters values μ^* and β^* that returned the highest SSIM$_{den}$ value[4]. To get a qualitative understanding of the denoising performance, we show in Figure 9 the images of the approximations obtained for a subset of the parameters grid in the case of *lena*. As we can see, when parameter β increases (moving rightward on the image grid) the *amount* of gradient discontinuities is penalized. Therefore, solutions present less sharp edges. Edge cases are the left-most images, which are fully noisy. By increasing parameter μ (moving downward on the image grid) we get from blurred to more detailed solutions. In fact, for low values of this parameter the *distance* between the solution and the noisy image is not penalized and the minimization mainly involves the Hessian penalization. Based on the proposed grid of parameters, the optimal solution is obtained for $\mu^* = 0.5$ and $\beta^* = 50$, which corresponds to a denoising quality SSIM$_{den} = 0.97$. Quality measurements related to optimal approximations for all the considered images in the CSIQ dataset are listed in Table III. By comparing the SSIM value of the noisy images and the denoised ones we can see that image quality improvements range from 0.15 (worst case is *lake*) to 0.33 (best case is *monument*). Noisy images and corresponding approximations of these two edge cases are showed in Figure 8. A couple of remarks to conclude this analysis. First, we notice that the three quality indeces used in the experiment return different rankings of the solutions. For instance, based on the PSNR$_{den}$ index the best result would be *fisher* instead of *lena*

[4]It is well-known that PSNR and SNR likely promote blurred solutions. The SSIM has been introduced to better simulate the quality discrimination mechanism of the human brain by measuring also structural similarity instead of only L^2 distances [44].

(a) noisy image ($\mathbf{g}$) SSIM$_{nsy} = 0.65$ (b) $\mu^* = 5.0, \beta^* = 500$ SSIM$_{den} = 0.80$ (c) noisy image ($\mathbf{g}$) SSIM$_{nsy} = 0.48$ (d) $\mu^* = 1.0, \beta^* = 50$ $SSIM_{den} = 0.81$

Fig. 8. CSIQ dataset: results for the worst (a,b) and best (c,d) denoising performances. The quality of the denoising is measured in terms of the difference between SSIM$_{den}$ − SSIM$_{nsy}$, that evaluates to 0.15 for the *lake* image and to 0.33 for the *monument* image.

TABLE III
QUALITY MEASUREMENTS RELATED TO THE DENOISING TASK ON A SUBSET OF THE CSIQ DATASET. FOR EACH TEST IMAGE THE TABLE REPORTS THE QUALITY VALUES FOR THE PARAMETER VALUES THAT HAVE ACHIEVED THE HIGHEST SSIM$_{den}$ VALUE.

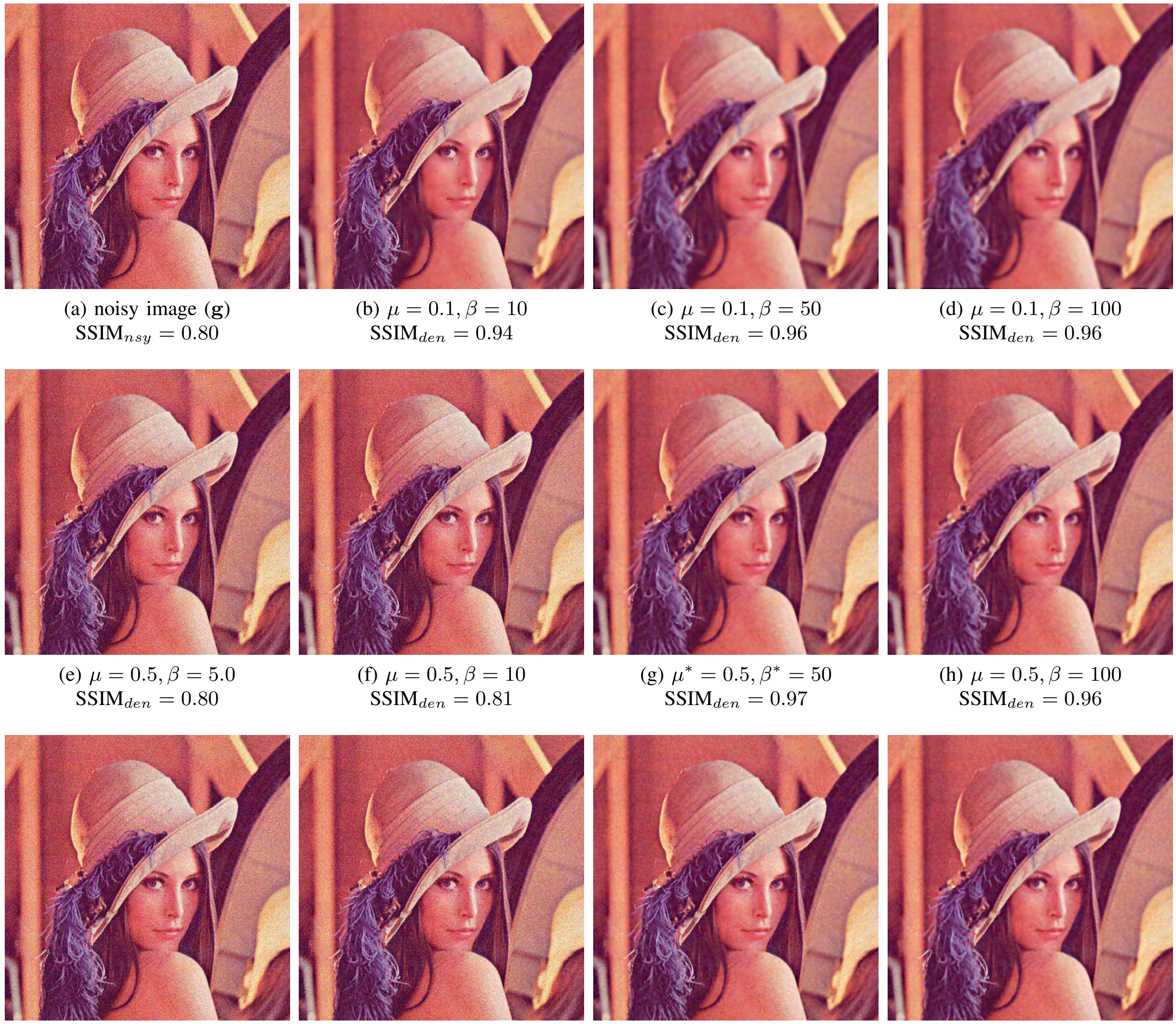
image	μ^*	β^*	$\hat{\sigma}_{ref}$	$\hat{\sigma}_{den}$	PSNR$_{nsy}$	PSNR$_{den}$	SNR$_{nsy}$	SNR$_{den}$	SSIM$_{nsy}$	SSIM$_{den}$
lena	0.5	50	25.52	23.88	20.20	29.34	15.06	24.21	0.80	0.97
1600	1.0	50	25.53	21.14	20.54	26.16	14.19	19.81	0.69	0.89
boston	1.0	50	25.48	22.13	20.14	26.71	14.68	21.25	0.47	0.76
bridge	1.0	50	25.53	21.99	20.79	27.17	16.38	22.76	0.52	0.83
child-swimming	1.0	50	25.49	20.57	20.11	25.09	14.11	19.09	0.65	0.85
fisher	0.1	10	25.52	24.93	20.19	29.63	15.87	25.31	0.63	0.94
lake	5.0	500	25.48	19.70	20.62	24.56	12.32	16.26	0.65	0.80
log-seaside	1.0	50	25.51	20.52	20.57	24.80	15.83	20.06	0.70	0.88
monument	1.0	50	25.52	22.22	20.53	27.63	15.34	22.44	0.48	0.81
native-american	0.5	50	25.51	24.69	20.41	27.86	15.69	23.13	0.57	0.87
trolley	1.0	50	25.52	20.69	20.23	25.20	14.24	19.21	0.64	0.85

(as obtained using SSIM$_{den}$). Second, we may also notice that highest values of SSIM$_{den}$ do not always correspond to the best approximations of the noise standard deviation. As an example, *log-seaside* and *native-american* have very similar SSIM$_{den}$ values (0.88 and 0.87, respectively) but their noise estimations are very different (20.52 and 24.69, respectively).

IV. EXPERIMENTAL RESULTS: POLYGONAL APPROXIMATION OF PLANAR CLOSED CURVES

In this section, we show how the feature of the BZ model that allows the formation of free gradient discontinuities is fundamental in the task of recovering the shape of closed curves from discrete noisy sampling. Like other models such as cubic smoothing splines (CSSPs), the BZ model is able to provide a smooth approximation of the curve. However, as additional feature and unlikely other methods can do, the BZ model allows also to retrieve polygonal shapes (i.e., curves with gradient discontinuities).

As a specific application, in the following we recover the polygonal shape of building footprints from discrete noisy approximations obtained in the processing of low resolution Digital Surface Models (DSMs). DSMs are 2-dimensional scalar-valued rasters and they are obtained by interpolating raw LiDAR (Light Detection and Ranging) unstructured point clouds into regular grids [45]. The value of the DSM at each grid point corresponds to the height of the object hit by the laser pulse in the location of the grid point. It is common in the remote sensing literature to extract building edges in DSMs in order to recover a discrete approximation of the building footprints. However, if the DSM resolution is low (e.g., 1m), the discrete points forming the detected edges can be far from a polygonal shape (Figure 10b) and post-processing is required to recover the building footprint. An example of edge detection in urban DSM is given in [22], [46], where the BZ model for gray-scale images is applied to the DSM and discrete approximations of the building footprints are mapped by the edge-detection function $\mathbf{s}$. We can see in Figure 10 a 3-D rendering of the DSM of an old barrack and the mapping function of its detected edges. The DSM is at spatial



(a) noisy image ($\mathbf{g}$) SSIM$_{nsy} = 0.80$

(b) $\mu = 0.1, \beta = 10$ SSIM$_{den} = 0.94$

(c) $\mu = 0.1, \beta = 50$ SSIM$_{den} = 0.96$

(d) $\mu = 0.1, \beta = 100$ SSIM$_{den} = 0.96$

(e) $\mu = 0.5, \beta = 5.0$ SSIM$_{den} = 0.80$

(f) $\mu = 0.5, \beta = 10$ SSIM$_{den} = 0.81$

(g) $\mu^* = 0.5, \beta^* = 50$ SSIM$_{den} = 0.97$

(h) $\mu = 0.5, \beta = 100$ SSIM$_{den} = 0.96$

(i) $\mu = 1.0, \beta = 5.0$ SSIM$_{den} = 0.80$

(j) $\mu = 1.0, \beta = 10$ SSIM$_{den} = 0.80$

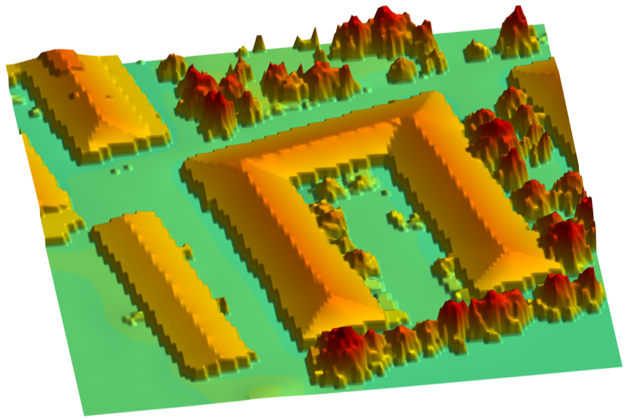
(k) $\mu = 1.0, \beta = 50$ SSIM$_{den} = 0.94$

(l) $\mu = 1.0, \beta = 100$ SSIM$_{den} = 0.96$

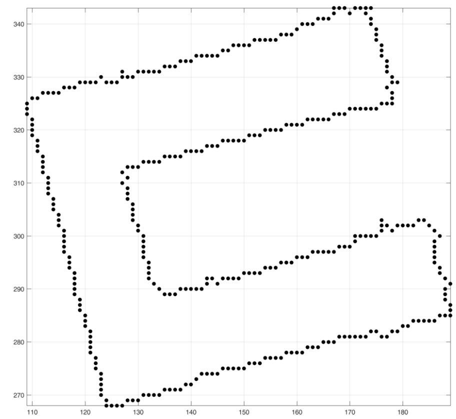
Fig. 9. Image denoise approximation for a subset of parameter values μ and β tested in the experiment ($\mu = 0.1, 0.5, 1$ and $\beta = 5, 10, 50, 100$). The highest quality value is SSIM$_{den} = 0.97$, which is obtained for $\mu^* = 0.5$ and $\beta^* = 50$.

resolution of 1m and the shape is not oriented parallel to the x, y-axis, therefore the discrete representation of the boundary is broken into many segments oriented parallel to the x, y-axis.

To recover the polygonal shape approximating these points we exploit the framework proposed in Section II-B. Input data is the set of P two-dimensional points $\{(x_p, y_p)\}_{p=1}^{P}$ representing the discrete noisy sampling of the unknown polygonal shape (the black points in Figure 10b). We set $N = 2$ and we construct the discrete variable $\mathbf{g} \in \mathbb{R}^{P \times 2}$ representing the discrete planar curve by simply assigning $[\mathbf{g}_1]_p = x_p$ and $[\mathbf{g}_2]_p = y_p$, for all $p = 1, \ldots, P$. To recover a closed curve, we minimize the functional (18) with periodic boundary conditions (cfr. Section II-B1). The BZ model (for brevity **BZ**), is tested against the typical approach to curve approximation of Cubic Smoothing Splines (for brevity **CSSP**). We recall that for **CSSP**, the solution that approximates the points $\mathbf{g}$ is the piecewise



(a) DSM $(\mathbf{g})$



(b) edge map $(\mathbf{s})$

Fig. 10. Extraction of building edges from DSM. (a) 3d rendering of a DSM representing an old barrack. (b) Edge map of the main (U-shaped) building obtained by segmenting the DSM using the BZ model for gray-scale images [22]. The points correspond to the pixels where the function $\mathbf{s}$ (the edge detection function) is 0.

cubic function $\mathbf{v}$ that minimizes the functional expression

$$G(\mathbf{v}) = q \sum_{p=1}^{P} |\mathbf{v}(t_p) - \mathbf{g}_p|^2 + (1-q) \int_{t_1}^{t_P} \mathbf{v}''(t)\, dt \tag{25}$$

where t is a parametric variable and $q \in [0,1]$ is a parameter that penalizes data fitting (q near 1) or data smoothness (q near 0). For brevity we omit details on **CSSP** implementation, we only recall that the solution can be found in a closed form. For details we refer the reader to [47], [48].

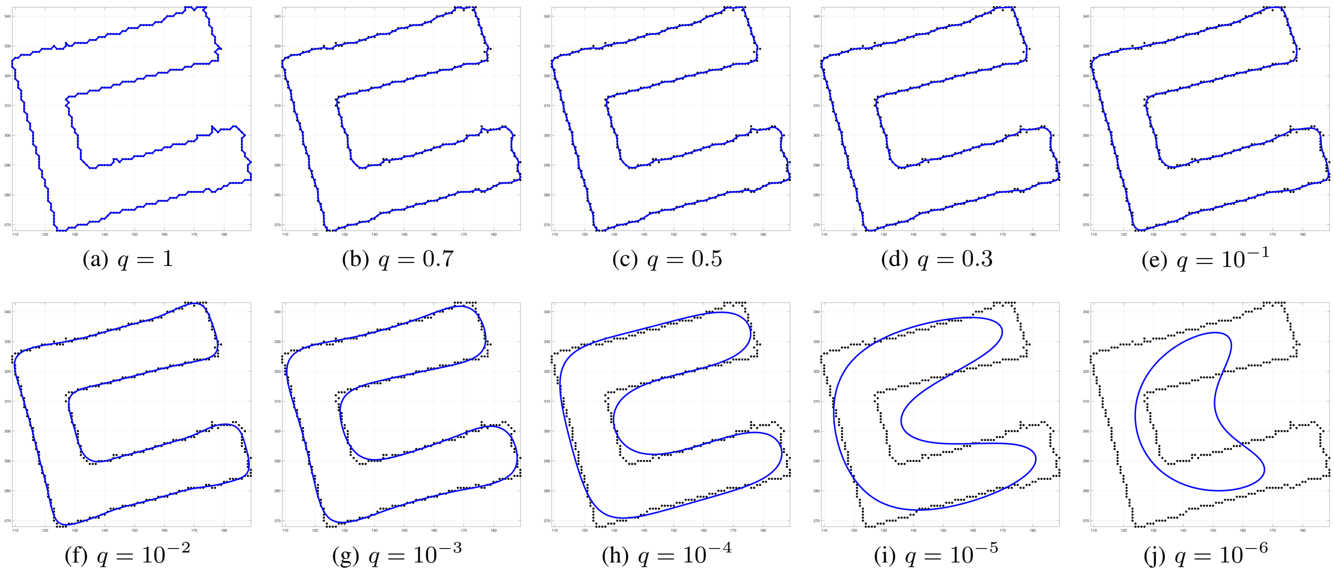
The **BZ** model depends on the two parameters λ and η, penalizing the distance of the solution to the original data $\mathbf{g}$ and the size of the gradient discontinuity set, respectively. The **CSSP** model only depends on the parameter q and the solution is not allowed to have gradient discontinuities. In order to better understand the behavior of the proposed models for a large variety of parameter selections, we defined a grid of values for λ, η and q. The whole range of possible behaviors of the resulting approximating curves (from over- to under- fitting) has been obtained for $\lambda = 10^{-4}, 10^{-5}, 10^{-6}$ and $\eta = 10^{-k}$, with $k = 1, \dots, 5$, and for several values of q between 0 and 1.

The results of curve approximation is illustrated for both **CSSP** and **BZ** models in Figures 11 and 12, respectively. Note from the results of **CSSP** that, by variation of the parameter q from 1 to 0, the behavior of the solution is from complete over-fitting to very smooth (and poor) approximation of the points. Notice that the smoothing effect of the splines does not allow to well represent the right angles of the main corners of the shape. Instead, by varying the parameters of the **BZ** we still obtain different behaviors from over- to under-fitting, but for some parameter choices we have polygonal solutions. In fact, by decreasing the contrast parameter η we allow the size of the gradient discontinuity set to be larger, thus allowing the formation of free gradient discontinuities. Polygon corners correspond to the points where the gradient of the solution is discontinuous. On the counter part, if we fix the value of η the solutions from the top row to the bottom row show increasing under-fitting. Indeed, by decreasing the parameter λ the solution is allowed to be *distant* from the original curve, thus the minimization penalizes the discontinuity set and produces very smooth curves. Among all the solutions, we can say that the best polygonal approximation is obtained for $\lambda = 10^{-5}$ and $\eta = 10^{-4}$ as the curve segments are straight segments forming right angles. It follows from the globality of the geometrical parameters in (17), that polygonal shapes at the same scale and corrupted by the same level of noise can be recovered by using identical parameter selections.

The computational burden to obtain all the approximations can be considered as negligible, as in all the cases algorithms converged in less than one second. Hardware and software used is the same as in Section III.

V. Conclusions

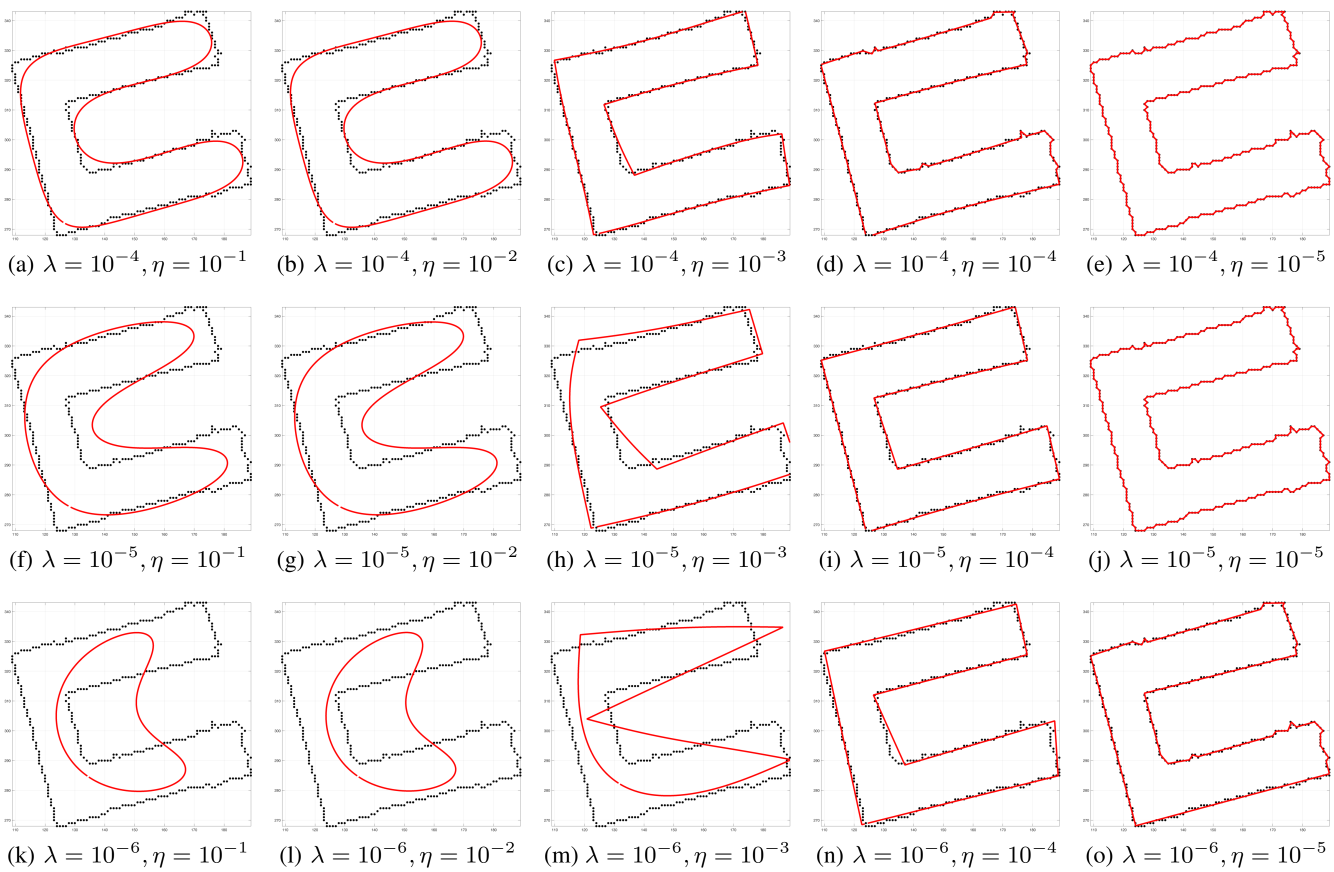
In the framework of variational methods to image approximation, the 1st-order model by Mumford-Shah is very popular. However, some intrinsic problems due to its 1st-order nature (such as the staircasing effect and



(a) $q = 1$ (b) $q = 0.7$ (c) $q = 0.5$ (d) $q = 0.3$ (e) $q = 10^{-1}$

(f) $q = 10^{-2}$ (g) $q = 10^{-3}$ (h) $q = 10^{-4}$ (i) $q = 10^{-5}$ (j) $q = 10^{-6}$

Fig. 11. Curve approximation results obtained for different parameter choices of the **CSSP** model. The range of values used in the experiments allowed us to explore the behavior of the solution from over- to under- fitting. No gradient discontinuity is allowed by the model.



(a) $\lambda = 10^{-4}, \eta = 10^{-1}$ (b) $\lambda = 10^{-4}, \eta = 10^{-2}$ (c) $\lambda = 10^{-4}, \eta = 10^{-3}$ (d) $\lambda = 10^{-4}, \eta = 10^{-4}$ (e) $\lambda = 10^{-4}, \eta = 10^{-5}$

(f) $\lambda = 10^{-5}, \eta = 10^{-1}$ (g) $\lambda = 10^{-5}, \eta = 10^{-2}$ (h) $\lambda = 10^{-5}, \eta = 10^{-3}$ (i) $\lambda = 10^{-5}, \eta = 10^{-4}$ (j) $\lambda = 10^{-5}, \eta = 10^{-5}$

(k) $\lambda = 10^{-6}, \eta = 10^{-1}$ (l) $\lambda = 10^{-6}, \eta = 10^{-2}$ (m) $\lambda = 10^{-6}, \eta = 10^{-3}$ (n) $\lambda = 10^{-6}, \eta = 10^{-4}$ (o) $\lambda = 10^{-6}, \eta = 10^{-5}$

Fig. 12. Curve approximation results obtained for different parameter choices of the **BZ** model. The range of values used in the experiments allowed us to explore the behavior of the solution from under- to over- fitting passing also through polygonal solutions. The best polygonal approximation is (i), i.e., for parameters $\lambda = 10^{-5}$ and $\eta = 10^{-4}$.

the triple point cracks) limit its applicability to solve complex problems like image denoising and restoration. To solve for this, 2nd-order methods can be used, but no attempts to implement vector-valued versions of 2nd-order models can be found in literature. This is critical when color or multispectral images need to be analyzed. In this paper we considered a 2nd-order variational model to the approximation of vector-fields in dimension two (e.g., multiband images) and one (e.g., curves in space) and we proposed efficient numerical implementations of the associated minimization problems. Specifically, we focused on the numerical minimization of the Ambrosio-Faina-March (AFM) elliptic approximation of the Blake-Zisserman (BZ) functional as it is particularly prone to numerical implementation. In the proposed numerical formulation the objective functional is written in a compact matrix form an the minimization is decomposed into quadratic sparse convex sub-problems. We proved that the minimization sup-problem associated to the vector-valued variable ($\mathbf{u}$) can be split into B further quadratic sub-problems (where B is the vector size) all depending on the same matrix. Different types of experimental studies have been done to assess the effectiveness of the proposed numerical formulation. In the first experimental part, we proposed a comparative analysis of the BZ model against the MS on difficult image restoration/denoising problems. The results show that the capability of the BZ model to approximate the input image in a piecewise linear manner produces more natural (in terms of visual interpretation) and precise (in terms of noise reduction/estimation) reconstructions of corrupted color images. In the second experimental part, we applied the BZ model to the approximation of closed curves from discrete noisy sampling. Differently from other typical curve approximation models such as Cubic Smoothing Splines, the BZ model allows the formation of free gradient discontinuities. Thus, polygonal approximation can be obtained for suitable choices of the parameters.

REFERENCES

[1] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 12, no. 7, pp. 629–639, 1990.

[2] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1, pp. 259–268, 1992.

[3] Y.-L. You and M. Kaveh, "Fourth-order partial differential equations for noise removal," *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1723–1730, 2000.

[4] J. Weickert, *Anisotropic diffusion in image processing*, ser. European Consortium for Mathematics in Industry. B. G. Teubner, Stuttgart, 1998.

[5] F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll, "Image selective smoothing and edge detection by nonlinear diffusion," *SIAM J. Numer. Anal.*, vol. 29, no. 1, pp. 182–193, 1992.

[6] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on pure and applied mathematics*, vol. 42, no. 5, pp. 577–685, 1989.

[7] E. De Giorgi, "Free discontinuity problems in calculus of variations," in *Frontiers in pure and applied mathematics*. North-Holland, Amsterdam, 1991, pp. 55–62.

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

[9] E. De Giorgi, M. Carriero, and A. Leaci, "Existence theorem for a minimum problem with free discontinuity set," *Arch. Rational Mech. Anal.*, vol. 108, no. 3, pp. 195–218, 1989.

[10] T. F. Chan and L. A. Vese, "A level set algorithm for minimizing the Mumford-Shah functional in image processing," in *Variational and Level Set Methods in Computer Vision, 2001. Proceedings. IEEE Workshop on*. IEEE, 2001, pp. 161–168.

[11] B. Bourdin and A. Chambolle, "Implementation of an adaptive finite-element approximation of the Mumford-Shah functional," *Numerische Mathematik*, vol. 85, no. 4, pp. 609–646, 2000.

[12] T. Pock, D. Cremers, H. Bischof, and A. Chambolle, "An algorithm for minimizing the Mumford-Shah functional," in *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2009, pp. 1133–1140.

[13] L. Ambrosio and V. M. Tortorelli, "Approximation of functional depending on jumps by elliptic functional via Γ-convergence," *Communications on Pure and Applied Mathematics*, vol. 43, no. 8, pp. 999–1036, 1990.

[14] P. DAmbra and G. Tartaglione, "Solution of Ambrosio–Tortorelli model for image segmentation by generalized relaxation method," *Communications in Nonlinear Science and Numerical Simulation*, vol. 20, no. 3, pp. 819–831, 2015.

[15] M. Carriero, A. Farina, and I. Sgura, "Image segmentation in the framework of free discontinuity problems," in *Calculus of variations: topics from the mathematical heritage of E. De Giorgi*, ser. Quad. Mat. Dept. Math., Seconda Univ. Napoli, Caserta, 2004, vol. 14, pp. 86–133.

[16] A. Brook, R. Kimmel, and N. A. Sochen, "Variational restoration and edge detection for color images," *Journal of Mathematical Imaging and Vision*, vol. 18, no. 3, pp. 247–268, 2003.

[17] A. Blake and A. Zisserman, *Visual reconstruction*, ser. MIT Press Series in Artificial Intelligence. MIT Press, Cambridge, MA, 1987.

[18] M. Carriero, A. Leaci, and F. Tomarelli, "A survey on the Blake–Zisserman functional," *Milan Journal of Mathematics*, vol. 83, no. 2, pp. 397–420, 2015.

[19] G. Bellettini and A. Coscia, "Approximation of a functional depending on jumps and corners," *Boll. Un. Mat. Ital. B (7)*, vol. 8, no. 1, pp. 151–181, 1994.

[20] L. Ambrosio, L. Faina, and R. March, "Variational approximation of a second order free discontinuity problem in computer vision," *SIAM J. Math. Anal.*, vol. 32, no. 6, pp. 1171–1197, 2001.

[21] G. Bellettini and A. Coscia, "Discrete approximation of a free discontinuity problem," *Numerical Functional Analysis and Optimization*, vol. 15, no. 3-4, pp. 201–224, 1994.

[22] M. Zanetti, V. Ruggiero, and M. Miranda Jr, "Numerical minimization of a second-order functional for image segmentation," *Communications in Nonlinear Science and Numerical Simulation*, vol. 36, pp. 528–548, 2016.

[23] L. Grippo and M. Sciandrone, "Globally convergent block-coordinate techniques for unconstrained optimization," *Optim. Methods Softw.*, vol. 10, no. 4, pp. 587–637, 1999.

[24] M. Bergounioux and L. Piffet, "A second-order model for image denoising," *Set-Valued and Variational Analysis*, vol. 18, no. 3, pp. 277–306, 2010.

[25] P. Blomgren and T. F. Chan, "Color TV: total variation methods for restoration of vector-valued images," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 304–309, 1998.

[26] G. Sapiro and D. L. Ringach, "Anisotropic diffusion of multivalued images with applications to color filtering," *IEEE Transactions on Image Processing*, vol. 5, no. 11, pp. 1582–1586, 1996.

[27] D. Tschumperlé, "Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE's," *International Journal of Computer Vision*, vol. 68, no. 1, pp. 65–82, 2006.

[28] V. Jaouen, P. Gonzalez, S. Stute, D. Guilloteau, S. Chalon, I. Buvat, and C. Tauber, "Variational segmentation of vector-valued images with gradient vector flow," *IEEE Transactions on Image Processing*, vol. 23, no. 11, pp. 4773–4785, 2014.

[29] M. Focardi, "On the variational approximation of free-discontinuity problems in the vectorial case," *Mathematical Models and Methods in Applied Sciences*, vol. 11, no. 04, pp. 663–684, 2001.

[30] M. Focardi and F. Iurlano, "Ambrosio-Tortorelli approximation of cohesive fracture models in linearized elasticity," 2013.

[31] J. Haehnle, "Numerical approximations of the Mumford-Shah functional for unit vector fields," *Interfaces and Free Boundaries*, vol. 13, no. 3, pp. 297–326, 2011.

[32] E. Strekalovskiy, A. Chambolle, and D. Cremers, "A convex representation for the vectorial Mumford-Shah functional," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 1712–1719.

[33] T. F. Chan, B. Y. Sandberg, and L. A. Vese, "Active contours without edges for vector-valued images," *Journal of Visual Communication and Image Representation*, vol. 11, no. 2, pp. 130–141, 2000.

[34] G. Sapiro, "Color snakes," *Computer vision and image understanding*, vol. 68, no. 2, pp. 247–253, 1997.

[35] N. El-Zehiry, P. Sahoo, and A. Elmaghraby, "Combinatorial optimization of the piecewise constant Mumford-Shah functional with application to scalar/vector valued and volumetric image segmentation," *Image and Vision Computing*, vol. 29, no. 6, pp. 365–381, 2011.

[36] M. Jung, X. Bresson, T. F. Chan, and L. A. Vese, "Nonlocal Mumford-Shah regularizers for color image restoration," *IEEE Transactions on Image Processing*, vol. 20, no. 6, pp. 1583–1598, 2011.

[37] M. Carriero, A. Leaci, and F. Tomarelli, "A second order model in image segmentation: Blake & Zisserman functional," in *Variational Methods for Discontinuous Structures*. Springer, 1996, pp. 57–72.

[38] S. Di Zenzo, "A note on the gradient of a multi-image," *Computer vision, graphics, and image processing*, vol. 33, no. 1, pp. 116–125, 1986.

[39] R. Machuca and K. Phillips, "Applications of vector fields to image processing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 3, pp. 316–329, 1983.

[40] R. Kimmel, R. Malladi, and N. Sochen, "Images as embedded maps and minimal surfaces: movies, color, texture, and volumetric medical images," *International Journal of Computer Vision*, vol. 39, no. 2, pp. 111–129, 2000.

[41] D. Mumford, "Bayesian rationale for the variational formulation," in *Geometry-Driven Diffusion in Computer Vision*. Springer, 1994, pp. 135–146.

[42] M. G. Mora, "Local calibrations for minimizers of the Mumford–Shah functional with a triple junction," *Communications in Contemporary Mathematics*, vol. 4, no. 02, pp. 297–326, 2002.

[43] E. C. Larson and D. M. Chandler, "Most apparent distortion: full-reference image quality assessment and the role of strategy," *Journal of Electronic Imaging*, vol. 19, no. 1, pp. 011 006–011 006, 2010.

[44] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[45] G. Forlani, C. Nardinocchi, M. Scaioni, and P. Zingaretti, "Complete classification of raw LIDAR data and 3d reconstruction of buildings," *Pattern Analysis and Applications*, vol. 8, no. 4, pp. 357–374, 2006.

[46] M. Zanetti and A. Vitti, "The Blake-Zisserman model for digital surface models segmentation," in *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. II-5/W2, 2013, pp. 355–360.

[47] N. Graham, "Smoothing with periodic cubic splines," *Bell System Technical Journal*, vol. 62, no. 1, pp. 101–110, 1983.

[48] Y. Wang, *Smoothing splines: methods and applications*. CRC Press, 2011.