



**HAL**  
open science

# Nonlinear Multilayered Representation of Graph-Signals

Moncef Hidane, Olivier Lezoray, Abderrahim Elmoataz

► **To cite this version:**

Moncef Hidane, Olivier Lezoray, Abderrahim Elmoataz. Nonlinear Multilayered Representation of Graph-Signals. *Journal of Mathematical Imaging and Vision*, 2013, 45 (2), pp.114-137. 10.1007/s10851-012-0348-9 . hal-00801756

**HAL Id: hal-00801756**

**<https://hal.science/hal-00801756v1>**

Submitted on 18 Mar 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Nonlinear Multilayered Representation of Graph-Signals

Moncef Hidane · Olivier Lézoray ·  
Abderrahim Elmoataz

© Springer Science+Business Media, LLC 2012

**Abstract** We propose a nonlinear multiscale decomposition of signals defined on the vertex set of a general weighted graph. This decomposition is inspired by the hierarchical multiscale ( $BV, L^2$ ) decomposition of Tadmor, Nezzar, and Vese (Multiscale Model. Simul. 2(4):554–579, 2004). We find the decomposition by iterative regularization using a graph variant of the classical total variation regularization (Rudin et al, Physica D 60(1–4):259–268, 1992). Using tools from convex analysis, and in particular Moreau’s identity, we carry out the mathematical study of the proposed method, proving the convergence of the representation and providing an energy decomposition result. The choice of the sequence of scales is also addressed. Our study shows that the initial scale can be related to a discrete version of Meyer’s norm (Meyer, Oscillating Patterns in Image Processing and Nonlinear Evolution Equations, 2001) which we introduce in the present paper. We propose to use the recent primal-dual algorithm of Chambolle and Pock (J. Math. Imaging Vis. 40:120–145, 2011) in order to compute both the minimizer of the graph total variation and the corresponding dual norm. By applying the graph model to digital images, we investigate the use of nonlocal methods to the multiscale decomposition task. Since the only assumption needed to apply our method is that the input data is living on a graph, we are also able to tackle the task of adaptive multi-

scale decomposition of irregularly sampled data sets within the same framework. We provide in particular examples of 3-D irregular meshes and point clouds decompositions.

**Keywords** Signal decomposition · Multiscale representations · Nonlocal total variation · Weighted graphs · Hierarchical models

## 1 Introduction

Our main concern in this paper is to introduce a multiscale representation for data sets defined on the vertex set of a general undirected graph. Before entering into technical details, we introduce the motivation and background for our work.

### 1.1 Multiscale Representations in Image Analysis

The importance of multiscale representations in the field of image analysis stems from the fact that, generally, natural images contain features of different scales. Beside edges which are important for image understanding, these features range from homogeneous regions to oscillatory patterns of texture and noise. As a consequence, a mathematical tool for providing adaptive representations of different image features is a major step towards higher level vision tasks.

A unified framework for the multiscale representation of digital images is the scale-space paradigm (see [42] and references therein). A scale-space representation of an image  $f : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$  embeds  $f$  in a family  $\{T_t f : \Omega \rightarrow \mathbb{R}, t \geq 0\}$ . As parameter  $t$  increases, images  $T_t f$  get smoother, starting from  $T_0 f = f$ . Such scale-scale representations are obtained by imposing a set of axioms on the operators  $T_i$  [1].

---

M. Hidane (✉) · O. Lézoray · A. Elmoataz  
ENSICAEN, CNRS, GREYC Image Team, Université de Caen  
Basse-Normandie, 6 Boulevard Maréchal Juin,  
14050 Caen Cedex, France  
e-mail: [moncef.hidane@unicaen.fr](mailto:moncef.hidane@unicaen.fr)

O. Lézoray  
e-mail: [olivier.lezoray@unicaen.fr](mailto:olivier.lezoray@unicaen.fr)

A. Elmoataz  
e-mail: [abderrahim.elmoataz-billah@unicaen.fr](mailto:abderrahim.elmoataz-billah@unicaen.fr)

Recently, inverse scale-space representations have been introduced in [32, 37]. These representations embed an original image  $f : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$  in a family  $\{T_t f : \Omega \rightarrow \mathbb{R}, t \geq 0\}$  such that  $T_0 f = 0$  and  $\lim_{t \rightarrow \infty} T_t f = f$ . In comparison with the standard scale-space theory, the order is reversed: the representation starts with a constant image and then moves toward the original data. However, the difference between the two approaches is not limited to the order in which the original data is recovered. A deeper difference comes from the formulation of the problem. While the scale-space paradigm is formulated directly in the language of partial differential equations, the inverse scale-space idea comes from a variational point of view.

A related but different representation was proposed in [39, 40] based on total variation (TV) regularization [36]. This time, the representation embeds  $f : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$  in a sequence  $\{T_n f : \Omega \rightarrow \mathbb{R}, n \geq 0\}$  such that  $T_0 f = 0$  and  $\lim_{n \rightarrow \infty} T_n f = f$ . In this setting, one minimizes the Rudin-Osher-Fatemi (ROF) functional

$$E(u; f) = \lambda \int_{\Omega} |\nabla u| + \frac{1}{2} \|f - u\|^2, \tag{1}$$

where  $f : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$  is a noisy image,  $\int_{\Omega} |\nabla u|$  denotes the TV seminorm, and the minimization is carried over the space of functions of bounded variations [12]. As pointed out by Meyer in [28], image denoising leads to image decomposition: the original image  $f$  is decomposed into a part  $u$  that extracts edges and a part  $v = f - u$  that contains texture and/or noise. It yields to the *one scale* decomposition

$$f = u + v. \tag{2}$$

The scalar  $\lambda$  in (1) is then viewed as a parameter dictating the separation of scales. In [39], Tadmor, Nezzar, and Vese extend the one scale decomposition (2) to a multiscale hierarchical representation by iteratively decomposing the successive residuals. This idea is the starting point of our present work.

### 1.2 Graph Representation of Discrete Data

We are interested in this paper in data sets having an underlying graph structure. Such a graph structure can be naturally present in the data under consideration (e.g., with square lattices), or it can be explicitly constructed (e.g., with proximity graphs). In both cases, the resulting graph encodes pairwise similarities between data instances. Those similarities are in turn used to generate more global descriptions of the underlying data. The pair represented by the data and the associated graph is called a graph-signal.

Adopting a graph-signal model for digital images leads to the so-called nonlocal methods (see [7, 20, 23] and references therein).

We also use in this paper the graph-signal model to handle data defined on irregularly shaped domains as well as

scattered data points. For such data sets, the adaption of classical multiscale methods such as those based on wavelets is not straightforward. We show that as soon as a graph structure can be associated with the data, an efficient multiscale representation can be obtained. In particular, we apply our method to generate multiscale representations of 3-D triangular meshes and point clouds.

The model for the rest of the paper is a weighted graph  $G = (V, E, w)$  consisting of a vertex set  $V = \{1, \dots, n\}$ , an edge set  $E \subseteq V \times V$  and a nonnegative symmetric weight function  $w : E \rightarrow \mathbb{R}^+ : w(i, j) = w(j, i)$  for all  $(i, j) \in E$ . The quantity  $w(i, j)$  represents a similarity or proximity measure between the two vertices  $i$  and  $j$ . This measure is usually computed as a decreasing transformation of a prior distance measure. The graph structure can be encoded by its (weighted) adjacency matrix  $W \in \mathbb{R}^{n \times n}$  defined by

$$W_{i,j} = \begin{cases} w(i, j) & \text{if } (i, j) \in E, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

In the sequel, the symbol  $W_{i,\cdot}$  is used to denote the  $i^{th}$  row, while  $W_{\cdot,j}$  denotes the  $j^{th}$  column of a matrix  $W$ . The transpose of a vector  $x$  is denoted by  $x^T$ .

### 1.3 Contributions

In this work, we provide a general tool to generate adaptive multiscale representations of data defined on the vertex set of a general weighted graph. The local similarities encoded by the graph structure are turned into a multiscale description by adopting a variational point of view. More precisely, we formulate a graph version of the classical TV regularization and use it to extract successive layers<sup>1</sup> in the spirit of [39]. The mathematical study of the proposed representation is carried out. In particular, the convergence towards the input data is proved, and an energy decomposition result is stated.

The issue concerning the choice of the first scale parameter is fully addressed. Our study shows that this quantity is related to a discrete version of Meyer’s norm [28], that we extend to weighted graphs. We show that the recent primal-dual algorithm of Chambolle and Pock can be used in order to compute this norm.

We finally give advice regarding the choice of the graph structure and the weighting function and apply our approach to generate multiscale representations of digital images, 3-D triangular meshes and point clouds.

### 1.4 Outline

The paper is organized as follows. In Sect. 2, we recall facts from convex analysis needed for the mathematical study of

<sup>1</sup>A very preliminary version of this work was published in [24].

the proposed method. Section 3 formulates the graph variant of TV regularization. Different perspectives concerning this approach are outlined and the minimization of the functional is discussed. The multilayered representation is presented in Sect. 4, where the principle is given, the mathematical study carried out, and the link between the computation of the first scale and a discrete weighted version of Meyer’s norm is drawn. Section 5 addresses the computation of the latter norm using the recent primal-dual algorithm of [13]. We provide numerical experiments in Sect. 6 and conclude in Sect. 7.

## 2 Notations and Facts from Convex Analysis

We recall in this section some results that will help us carry out the mathematical study of our method. In particular, we review some concepts from convex analysis including the notion of a proximity operator introduced in [30] and popularized in signal processing since [18]. A detailed account about proximity operators can be found in [18]. For the use of proximity operators in splitting methods we refer the reader to [17] and references therein. Further details about convex analysis in the finite-dimensional setting can be found e.g. in [5, 25].

Let  $H$  be a finite-dimensional vector space equipped with a scalar product  $\langle \cdot, \cdot \rangle$ . The associated norm is denoted by  $\|\cdot\| = \sqrt{\langle \cdot, \cdot \rangle}$ . A function  $f : H \rightarrow ]-\infty, +\infty]$  is said to be proper if its domain,  $\text{dom}(f) := \{x \in H : f(x) < +\infty\}$ , is nonempty. The function  $f$  is said to be convex if its epigraph,  $\text{epi}(f) := \{(x, a) \in H \times \mathbb{R} : f(x) \leq a\}$ , is convex in  $H \times \mathbb{R}$ , and lower semicontinuous if  $\text{epi}(f)$  is closed in  $H \times \mathbb{R}$ . The set of all proper convex and lower semicontinuous functions from  $H$  to  $]-\infty, +\infty]$  is denoted by  $\Gamma_0(H)$ .

Let  $C$  denote a nonempty closed convex subset of  $H$ . The indicator function of  $C$ ,  $\iota_C$ , is defined for all  $x \in H$  by

$$\iota_C(x) = \begin{cases} 0 & \text{if } x \in C, \\ +\infty & \text{otherwise.} \end{cases} \tag{4}$$

Notice that  $\iota_C \in \Gamma_0(H)$ .

For  $f \in \Gamma_0(H)$  and  $z \in H$ , Moreau showed in [30] that the function  $x \in H \mapsto \frac{1}{2}\|x - z\|^2 + f(x)$  achieves its infimum at a unique point called proximity operator of  $f$  at point  $z$  and denoted by  $\text{prox}_f z$ :

$$\text{prox}_f z = \underset{x \in H}{\text{argmin}} \frac{1}{2}\|x - z\|^2 + f(x). \tag{5}$$

If  $f = \iota_C$ , then one recovers the definition of the convex projection operator denoted by  $\text{proj}$ :

$$\text{prox}_{\iota_C} z = \underset{C}{\text{proj}} z. \tag{6}$$

The notion of proximity operator thus generalizes the concept of projection onto nonempty closed convex sets. It leads

to Moreau’s identity (14) which in turn generalizes the representation of a Hilbert space as the direct sum of a closed subspace and its orthogonal complement [31].

The conjugate of  $f \in \Gamma_0(H)$  is the function  $f^* \in \Gamma_0(H)$  defined for all  $y \in H$  by

$$f^*(y) = \sup_{x \in H} \{\langle x, y \rangle - f(x)\}. \tag{7}$$

The conjugate of  $\iota_C$  is the support function of  $C$ , denoted by  $\sigma_C$ :

$$\iota_C^*(y) = \sup_{x \in C} \langle x, y \rangle = \sigma_C(y). \tag{8}$$

The conjugate of a norm  $\|\cdot\|$  is the indicator of its dual norm  $\|\cdot\|_*$  unit ball:<sup>2</sup>

$$\|\cdot\|_*^*(y) = \iota_{\{\|\cdot\|_* \leq 1\}}(y), \tag{9}$$

where the dual norm  $\|\cdot\|_*$  is defined for all  $x \in H$  by  $\|x\|_* = \sup_{\|y\| \leq 1} \langle x, y \rangle$ . The following holds for all  $f \in \Gamma_0(H)$ :

$$f^{**} = f, \tag{10}$$

$$f(x) + f^*(y) \geq \langle x, y \rangle, \quad \text{for all } x, y \in H. \tag{11}$$

The subdifferential of  $f$  at point  $x$ ,  $\partial f(x)$ , is the set of points  $y$  such that equality holds in (11):

$$\partial f(x) = \{y \in H : f(x) + f^*(y) = \langle x, y \rangle\}. \tag{12}$$

We have

$$\begin{cases} \inf_{x \in H} f(x) = -f^*(0), \\ \text{and} \\ \inf_{x \in H} f(x) = f(x_0) \iff 0 \in \partial f(x_0). \end{cases} \tag{13}$$

Moreau’s identity, introduced in [30] and generalized in [18], states that for all  $f \in \Gamma_0(H)$  and all  $z \in H$  the following equality holds

$$z = \text{prox}_{\lambda f} z + \lambda \text{prox}_{f^*/\lambda}(z/\lambda), \quad \text{for all } \lambda > 0. \tag{14}$$

Equation (14) leads to a whole family of decomposition schemes parametrized by the function set  $\Gamma_0(H)$  and the parameter  $\lambda$ .

## 3 Total Variation Based Regularization on Graphs

### 3.1 Norms and Operators

Let  $N \geq 1$  be an integer. We denote by  $X$  the set  $\mathbb{R}^N$  and by  $Y$  the set  $\mathbb{R}^{N \times N}$ . Elements of  $X$  represents signals defined on the vertex set of a general graph. Both  $X$  and  $Y$  are endowed with the usual inner products:  $\langle u, v \rangle_X = \sum_{i=1}^N u_i v_i$

<sup>2</sup>Star as a superscript denotes the convex conjugate function while star as a subscript denotes the dual norm.

for  $u, v \in X$  and  $\langle p, q \rangle_Y = \sum_{i,j=1}^N p_{i,j}q_{i,j}$  for  $p, q \in Y$ . The associated norms are denoted by  $\|\cdot\|_X$  and  $\|\cdot\|_Y$ . For simplicity of notation, we will sometimes drop the subscripts  $X$  and  $Y$  when there is no possible ambiguity. We also consider on  $Y$  the  $\|\cdot\|_{1,2}$  norm and its dual norm  $\|\cdot\|_{\infty,2}$  defined for all  $p \in Y$  by:

$$\|p\|_{1,2} = \sum_{i=1}^N \|p_{i,\cdot}^\top\|_X = \sum_{i=1}^N \left( \sum_{j=1}^N p_{i,j}^2 \right)^{1/2}, \tag{15}$$

$$\|p\|_{\infty,2} = \max_{1 \leq i \leq N} \|p_{i,\cdot}^\top\|_X = \max_{1 \leq i \leq N} \left( \sum_{j=1}^N p_{i,j}^2 \right)^{1/2}. \tag{16}$$

Consider a symmetric matrix  $W \in Y$  with nonnegative entries representing the weighted adjacency matrix of a given graph, as explained in Sect. 1.2. We associate with the matrix  $W$  the first order weighted difference operator  $\nabla_w : X \rightarrow Y$  defined for all  $u \in X$  and  $i, j \in \{1, \dots, N\}$  by

$$(\nabla_w u)_{i,j} = \sqrt{w_{i,j}}(u_j - u_i). \tag{17}$$

The quantity  $\|(\nabla_w u)_{i,\cdot}\|_X$  then represents a measure of the local variation of the signal  $u$  at the node  $i$ .

The operator  $\nabla_w$  has been introduced in [6, 20, 23]. A closely related operator has been considered in the context of machine learning in [44].

The adjoint of  $\nabla_w$ , denoted by  $\nabla_w^*$ , is the unique operator from  $Y$  to  $X$  verifying the following identity

$$\langle \nabla_w u, p \rangle_Y = \langle u, \nabla_w^* p \rangle_X. \tag{18}$$

Its expression is given by:

$$(\nabla_w^* p)_i = \sum_{j=1}^N \sqrt{w_{i,j}}(p_{j,i} - p_{i,j}). \tag{19}$$

The divergence operator  $\text{div}_w : Y \rightarrow X$  is defined as  $\text{div}_w = -\nabla_w^*$ , namely

$$(\text{div}_w p)_i = \sum_{j=1}^N \sqrt{w_{i,j}}(p_{i,j} - p_{j,i}). \tag{20}$$

The composition of the two latter operators leads to the unnormalized graph Laplacian  $\Delta_w : X \rightarrow X$  defined as  $\Delta_w = -\frac{1}{2} \text{div}_w \circ \nabla_w$ . Its expression is given by:

$$(\Delta_w u)_i = \sum_{j=1}^N w_{i,j}(u_i - u_j). \tag{21}$$

The properties of the unnormalized graph Laplacian have been studied in different contexts. We recall here one specific property that will be useful for the development of our study. For the proof, we refer the interested reader to [29] where an overview of the properties of the unnormalized graph Laplacian can also be found.

**Proposition 1** *Let  $\Delta_w$  be the Laplacian operator associated with a given weighted graph. The smallest eigenvalue of  $\Delta_w$  is  $\lambda_1 = 0$ . Furthermore, the multiplicity of 0 as an eigenvalue is equal to the number of connected components of the graph.*

### 3.2 $G_w$ Norm

In [28], Y. Meyer addressed some limitations of the ROF model [36]. He then proposed a modified model where the  $L^2$  norm quantifying the dissimilarity between the input and the sought functions is replaced by the  $G$  norm. The latter norm is defined on the  $G$  space that models signals with large oscillations. The precise definitions and justification for such a choice are given in [28]. An intuitive interpretation of the  $G$  norm can be gained when considering its relation to ROF model. In our finite-dimensional setting, this relation will be further given in Proposition 3. Equipped with the divergence operator in (20), we now introduce the corresponding concepts in our finite-dimensional setting over graphs.

The space  $G_w$  is defined as follows

$$G_w = \{u \in X : \exists p \in Y, u = \text{div}_w p\}. \tag{22}$$

The  $G_w$  norm is then given by

$$(\forall u \in G_w) \quad \|u\|_{G_w} = \inf \{ \|p\|_{\infty,2} : p \in Y, \text{div}_w p = u \}. \tag{23}$$

For  $\mu > 0$ ,  $G_w^\mu$  denotes the closed ball

$$G_w^\mu = \{u \in G_w : \|u\|_{G_w} \leq \mu\}. \tag{24}$$

We note that a discrete version of Meyer’s space and norm were already introduced in [3] based on the discretization of the continuous differential operators involved in the ROF model. An analogue of Proposition 3 was given there. Our version has several advantages. It allows to take account of nonlocal interactions when modeling digital images. It also makes it possible to consider the  $G$  norm of irregularly sampled signals, provided a graph structure can be associated with the spatial locations of the samples.

The next result identifies the space  $G_w$  with the set of zero-mean graph-signals. It is the analogous of Proposition 2.2 in [3].

**Proposition 2** *Consider a weighted graph with adjacency matrix  $W$ . If the graph is connected, then*

$$G_w = X_0 := \left\{ u \in X : \sum_{i=1}^N u_i = 0 \right\}. \tag{25}$$

*Proof* From Eq. (20) we see that  $G_w \subseteq X_0$ . The range of the operator  $\Delta_w$  is included in  $X_0$ . If the graph is connected, then using the result of Proposition 1, we see that

the dimension of the null space of  $\Delta_w$  is equal to 1. Hence, the dimension of its range is equal to  $N - 1$ , which is exactly the dimension of the space  $X_0$ , so that both spaces are equal. Hence, for all  $u \in X_0$ , there exists  $v \in X$  such that  $u = \Delta_w v$ . Defining  $v' = -\frac{1}{2} \nabla_w v$ , we have  $u = \text{div}_w v'$ , which concludes the proof.  $\square$

All the graphs considered in the sequel are assumed connected, unless otherwise stated.

### 3.3 Total Variation over Graphs

The isotropic total variation associated with the adjacency matrix  $W$  and the difference operator  $\nabla_w$  is given for all  $u \in X$  by

$$J_w(u) := \|\nabla_w u\|_{1,2} = \sum_{i=1}^N \left( \sum_{j=1}^N w_{i,j} (u_j - u_i)^2 \right)^{1/2}. \quad (26)$$

The functional in (26) can be regarded as a measure of “smoothness” of data living on general graphs. It is the sum of the local variations around all the nodes of the graph. The choice of the  $\ell_1$  norm is motivated by the analogy to the classical total variation over Euclidean domains and its well known ability to preserve sharp discontinuities. Alternatively, this choice can be seen as a regularizing prior promoting the sparsity of the local variations vector.

Notice that  $J_w$  defines a seminorm over  $X$ . Using Eqs. (20)–(23) and the fact that the norms  $\|\cdot\|_{1,2}$  and  $\|\cdot\|_{\infty,2}$  are dual, we see that the TV seminorm is the support function of the  $G_w$  unit norm ball:

$$J_w(u) = \sup\{\langle u, v \rangle_X : v \in G_w^1\} = \sigma_{G_w^1}(u). \quad (27)$$

The result in (27) it is a general fact, namely, each closed sublinear function is the support function of a closed convex set ([25], Chap. C, Theorem 3.1.1).

### 3.4 Total Variation Based Regularization on Graphs

We now consider the classical ROF model in the context we presented above. Given an initial observation  $f \in X$  living on a graph and a scale parameter  $\lambda > 0$ , the graph version of the ROF model consists in solving the following convex optimization problem:

$$\underset{u \in X}{\text{minimize}} E(u; f, \lambda), \quad (28)$$

where

$$E(u; f, \lambda) = \lambda J_w(u) + \frac{1}{2} \|u - f\|_X^2. \quad (29)$$

Functional  $E$  corresponds to the particular case of  $p = 1$  in the family of functionals introduced in [20]. It has been studied in [23] and has found numerous applications in image

and mesh processing [20], data filtering [26], image zooming [22]. In [2, 35], the authors proposed to adapt the penalization to the topology of the underlying function. The same approach has been taken in [43] for motion deblurring.

The model (28)–(29) can be seen under different perspectives. In the subsequent sections, we give three different point of view justifying the introduction of (28)–(29).

#### 3.4.1 Bayesian Interpretation

The Bayesian derivation of (28)–(29) can be obtained when considering the standard denoising problem. Let us assume that a true graph-signal  $\bar{u} \in X$  is degraded by an additive centered Gaussian white noise  $n$  of standard deviation  $\sigma$ , yielding the following model:  $u = \bar{u} + n$ . Adopting the MAP estimate, we see that the functional (29) is obtained when considering the following prior density probability function:

$$(\forall u \in X) \quad p(u) \propto e^{-\|\nabla_w u\|_{1,2}}. \quad (30)$$

#### 3.4.2 An Iterative Neighborhood Filter

Previous approaches to solve problem (28)–(29) dealt with the non-differentiability of the functional  $E$  in (29) by smoothing the penalty term  $J_w$ . In our context, this strategy amounts to replacing  $J_w$  defined in Eq. (26) by the smoothed penalty  $J_w^\epsilon$  defined for all  $u \in X$  by

$$J_w^\epsilon(u) := \sum_{i=1}^N \left( \sum_{j=1}^N w_{i,j} (u_j - u_i)^2 + \epsilon^2 \right)^{1/2}, \quad (31)$$

for a small fixed  $\epsilon$  (e.g.  $\epsilon = 10^{-4}$ ). Now, the modified energy  $E^\epsilon(u; f, \lambda) = \lambda J_w^\epsilon(u) + \frac{1}{2} \|u - f\|_X^2$  is smooth and still strictly convex. Differentiating and applying a fixed point iteration to the resulting system of nonlinear equations (see [20] for further details, and in particular for the convergence of the proposed scheme) leads to the following iterative scheme

$$\begin{cases} u^{(0)} = f, \\ u_i^{(n+1)} = \frac{f_i + \lambda \sum_{j=1}^N \gamma_{i,j}^{(n)} u_j^{(n)}}{1 + \lambda \sum_{j=1}^N \gamma_{i,j}^{(n)}}, \end{cases} \quad (32)$$

where

$$\gamma_{i,j}^{(n)} = W_{i,j} \left( \frac{1}{\sqrt{\|(\nabla_w u^{(n)})_{i,\cdot}^T\|_X^2 + \epsilon^2}} + \frac{1}{\sqrt{\|(\nabla_w u^{(n)})_{j,\cdot}^T\|_X^2 + \epsilon^2}} \right). \quad (33)$$

Notice that the coefficients  $\gamma_{i,j}^{(n)}$  depend on the current iterate  $u^{(n)}$ . The convergence of algorithm (32) is very slow in practice. However, the presented scheme allows to interpret the TV-regularized solution as a nonlinear data-dependent

iterative filter [15]. At each iteration, the raw data  $f$  at a given node  $i$  is averaged with the current iterate values at the neighboring nodes to get the updated value. The coefficients of the filter depend on the parameter  $\lambda$  and the amplitude of the current (smoothed) local variations. When the latter local variation is dominant, the raw data is preserved, thus explaining the well known discontinuity preserving property of TV regularization. Notice however that the local variation should be compared with the size of the parameter  $\lambda$ . The latter remark gives an interpretation of the parameter  $\lambda$  as a scale parameter.

### 3.4.3 $u + v$ Decomposition

Following [28], the ROF model can be casted into the more general setting of  $u + v$  models where an original signal is decomposed into a geometric component  $u$  and oscillating (and maybe noisy) part  $v$ .  $u + v$  models include the ones proposed in [3, 4, 33, 41].

Using the tools recalled in Sect. 2, we see that solving the graph ROF model (28)–(29) amounts to computing the proximity operator of  $\lambda J_w$ . Applying Moreau’s identity leads to the following decomposition

$$f = \text{prox}_{\lambda J_w}(f) + \lambda \text{prox}_{J_w^*/\lambda}(f/\lambda). \tag{34}$$

Applying equations (27), (8) and (10) we see that  $J_w^*/\lambda$  is equal to  $\iota_{G_w^1}$ . Finally, using Eq. (6), we get the important characterization

$$f = \text{prox}_{\lambda J_w}(f) + \text{proj}_{G_w^\lambda}(f). \tag{35}$$

Equation (35) allows to precise the behavior of (28)–(29) as a decomposition model. Namely, the regularized solution is obtained by subtracting the projection of  $f$  onto the space  $G_w$  from  $f$ . The relative importance of the projection regarding the parameter  $\lambda$  is precisely quantified by its  $G_w$  norm.

### 3.5 Minimization

We now consider the issues concerning the minimization of (28)–(29). As already stated in Sect. 3.4.2, the convergence of the fixed-point algorithm (32) is too slow in practice. A further and probably more serious drawback concerning (32) is the introduction of the smoothing parameter  $\epsilon$  which modifies the formulation of the problem. For these reasons, we do not consider algorithm (32) for the numerical simulation, but we keep in mind the interpretation given in Sect. 3.4.2.

Algorithms that solve the exact TV problem can be divided into two categories. The first category solves the dual problem associated with (28)–(29). The forerunner algorithms in this category are the fixed-point and projected

gradient algorithms proposed by Chambolle in [9, 10]. In fact, the projected gradient of [10] corresponds to a forward-backward algorithm on the dual problem [17].

The second category consists in the so-called primal-dual algorithms, the first of which was proposed by Chan, Golub, and Mulet [16]. We refer the reader to [14] for a review of numerical methods in total variation restoration.

Recently, Chambolle and Pock [13] proposed a very flexible primal-dual algorithm that exhibits very good numerical performance. We detail the adaptation of the fixed-point algorithm of [9] and the primal-dual algorithm of [13] to our setting and perform a performance comparison on digital images.

#### 3.5.1 Chambolle’s Fixed Point Algorithm

The dual formulation of problem (28)–(29) corresponding to the decomposition (35) is the following:

$$\begin{aligned} &\text{minimize } \|f - \lambda \text{div}_w p\|_X^2, \\ &\text{subject to } \|p\|_{\infty,2} \leq 1. \end{aligned} \tag{36}$$

The projection algorithm proposed in [9] to solve (36) yields in our setting the following iterative scheme:

$$\begin{cases} p^0 = 0, \\ p_{i,j}^{n+1} = \frac{p_{i,j}^n + \tau(\nabla_w(\text{div}_w p^n - f/\lambda))_{i,j}}{1 + \tau\|(\nabla_w(\text{div}_w p^n - f/\lambda))_{i,j}\|_X}. \end{cases} \tag{37}$$

The iterations in (37) converge under the following restriction on the time step  $\tau$ :

$$0 < \tau \leq \frac{1}{\|\text{div}_w\|^2}, \tag{38}$$

where  $\|\text{div}_w\|$  is the norm of the divergence operator induced by the norms  $\|\cdot\|_X$  and  $\|\cdot\|_Y$ :

$$\|\text{div}_w\| = \sup\{\|\text{div}_w p\|_X : p \in Y, \|p\|_Y \leq 1\}.$$

The following inequality was proved in [23]

$$\forall p \in Y, \quad \|\text{div}_w p\|_X^2 \leq 4m\|p\|_Y^2, \tag{39}$$

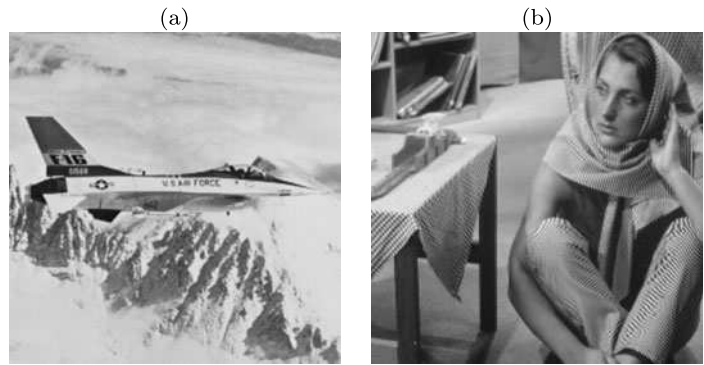
where  $m := \max_{1 \leq i \leq n} d_i$ , and  $d_i = \sum_{j=1}^n w_{i,j}$  is the degree of vertex  $i$ . Consequently, if  $0 < \tau \leq \frac{1}{4m}$ , the convergence of the sequence  $(p^n)_{n \geq 0}$  is guaranteed. Finally, the unique solution  $\hat{u}$  of problem (28)–(29), is given by  $\hat{u} = f - \lambda \text{div}_w p^\infty$ , where  $p^\infty$  is the limit of  $(p^n)_{n \geq 0}$  defined in (37).

#### 3.5.2 Chambolle-Pock Primal-Dual Algorithm

The convex optimization problem addressed in [13] is the following:

$$\text{minimize}_{x \in X} F(Kx) + G(x). \tag{40}$$

**Fig. 1** Test images used for the comparison between algorithms (37) and (42)



In (40),  $X$  and  $Y$  denote two general finite-dimensional vector spaces,  $F \in \Gamma_0(Y)$ ,  $G \in \Gamma_0(X)$  and  $K : X \rightarrow Y$  a linear operator. The saddle-point problem associated with (40) is

$$\min_{x \in X} \max_{y \in Y} \langle Kx, y \rangle + G(x) - F^*(y), \tag{41}$$

where  $F^*$  is the conjugate of  $F$ . The authors propose the following iterative resolution:

$$\begin{cases} y^{n+1} = \text{prox}_{\sigma F^*}(y^n + \sigma K\bar{x}^n), \\ x^{n+1} = \text{prox}_{\tau G}(x^n - \tau K^*y^{n+1}), \\ \bar{x}^{n+1} = x^{n+1} + \theta(x^{n+1} - x^n), \end{cases} \tag{42}$$

where  $K^*$  denotes the adjoint operator of  $K$ . The convergence of algorithm (42) is guaranteed if  $\theta = 1$  and  $0 < \tau\sigma L^2 < 1$  where  $L = \|K\| = \max_{\|x\| \leq 1} \|Kx\|$  denotes the induced norm of  $K$ . We refer the reader to [13] for detailed study of the speed of convergence of (42). Notice that the authors of [13] also proposed an accelerated version of (42). A second remark concerning (42) is that by taking the dual problem of (40), one can replace the iterations in (42) by the equivalent iterations:

$$\begin{cases} x^{n+1} = \text{prox}_{\tau G}(x^n - \tau K^*\bar{y}^n), \\ y^{n+1} = \text{prox}_{\sigma F^*}(y^n + \sigma Kx^{n+1}), \\ \bar{y}^{n+1} = y^{n+1} + \theta(y^{n+1} - y^n), \end{cases} \tag{43}$$

We now apply algorithm (42) to problem (28)–(29). The sets  $X$  and  $Y$  are the ones defined in Sect. 3.1. The identification of the operators is the following:  $K \equiv \nabla_w$ ,  $F \equiv \|\cdot\|_{1,2}$ ,  $G \equiv \frac{1}{2\lambda} \|\cdot - f\|_X^2$ . For  $y \in Y$  we have

$$\text{prox}_{\sigma F^*}(y) \stackrel{(9)}{=} \text{prox}_{\sigma \iota_{B_{\infty,2}(1)}}(y) \stackrel{(6)}{=} \text{proj}_{B_{\infty,2}(1)}(y) =: \tilde{y}, \tag{44}$$

where  $\tilde{y}_{i,j} = \frac{y_{i,j}}{\max(1, \|y_{i,\cdot}\|_X)}$  and  $B_{\infty,2}(\tau)$  is the  $\|\cdot\|_{\infty,2}$  ball of radius  $\tau$ . The computation of the second proximity operator in (42) is straightforward because the involved function is quadratic. The expression is given by

$$\text{prox}_{\tau G}(x) = \frac{\lambda x + \tau f}{\tau + \lambda}. \tag{45}$$

### 3.5.3 Performance Comparison on Digital Images

In this section, we compare the performance of the algorithms presented in the two latter sections. We restrict our setting to digital images where each pixel represents a vertex, and consider the square lattice graph with binary weights. In order to perform the comparison, we have chosen different values for the parameter  $\lambda$ , namely,  $\lambda = 5, 10, 100, 250$ . As each iteration of the two algorithms has the same algorithmic complexity, we have decided to let both algorithms run for a fixed common number of iterations  $M$ . We have taken  $M = 500$ . The original images on which we have performed the experiment are shown in Fig. 1. Figures 2 and 3 show the images obtained when applying both algorithms with different values of  $\lambda$ . Figures 4 and 5 show the corresponding energy profiles in logarithmic scale.

While the fixed-point algorithm (37) performs well for reasonable values of  $\lambda$ , it suffers from very slow convergence when  $\lambda$  is too important. The situation where  $\lambda$  is set to a high value is rarely encountered for denoising/deblurring applications in image processing. However, in the multiscale representation we seek in this paper, the initial value for  $\lambda$  will be very high in order to extract a first geometrical layer from the input data.

## 4 Multilayered Representation of Graph-Signals

Throughout this section,  $W$  denotes a weighted adjacency matrix encoding a weighted graph  $G$  and  $f \in X$  is a real-valued graph-signal on  $G$ . The multilayered representation we propose is based on iterative regularization as explained in the following section.

### 4.1 Principle

Applying the minimization (28)–(29) to  $f$  with a fixed  $\lambda > 0$  yields the *one scale* decomposition  $f = u + v$  as explained in Sect. 3.4.3. This one scale decomposition can be turned into a *multiscale* representation by iteratively applying the



**Fig. 2** Results of applying algorithms (37) and (42) with different values of  $\lambda$  to the standard Barbara image



minimization process to the successive residuals as initially proposed in [39, 40].

Let  $(\lambda_n)_{n \geq 0}$  denote a sequence of decreasing positive scales. We consider the following iterative regularization:

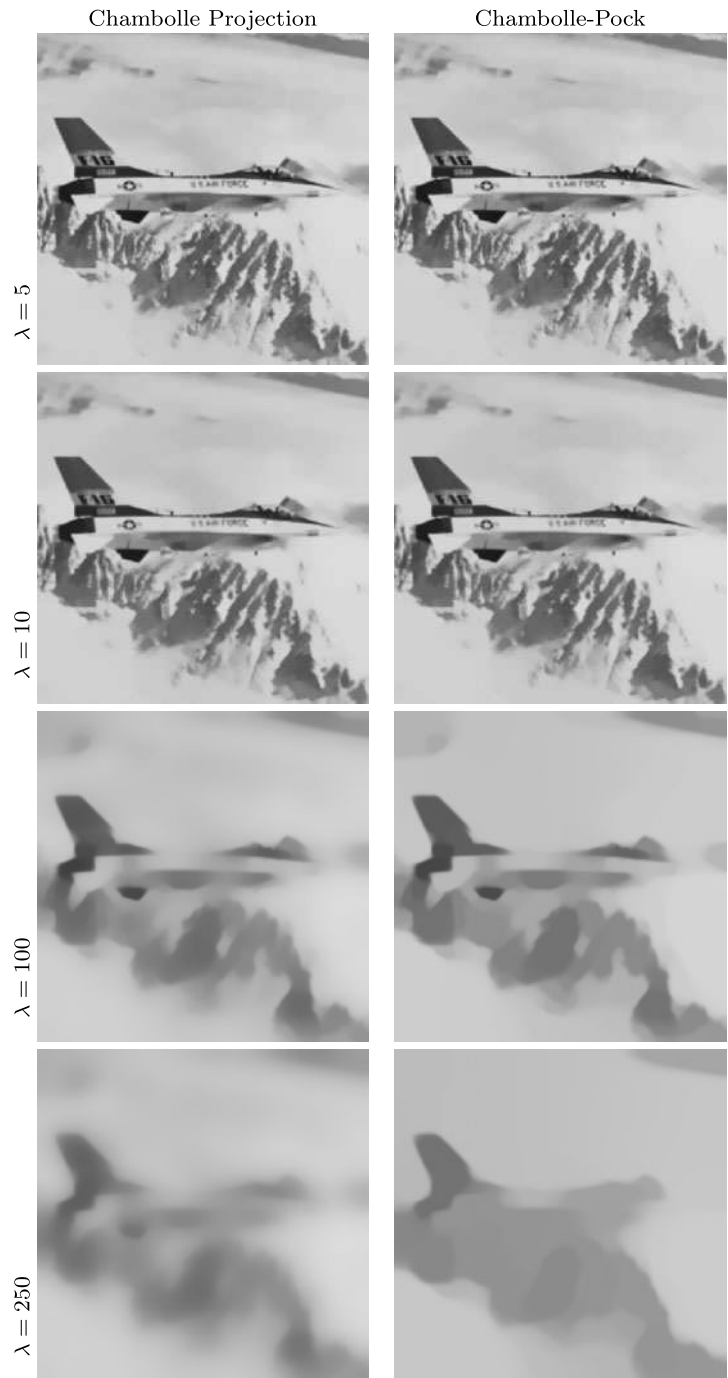
$$\begin{cases} v_{-1} = f, \\ u_i = \operatorname{argmin}_{u \in X} E(u; v_{i-1}; \lambda_i), \quad i \geq 0, \\ v_i = v_{i-1} - u_i, \quad i \geq 0, \end{cases} \quad (46)$$

where each residual  $v_i$  is further decomposed using (28)–(29). Algorithm (46) yields the following decomposition:

$$(\forall n \geq 0) \quad f = \sum_{i=0}^n u_i + v_n. \quad (47)$$

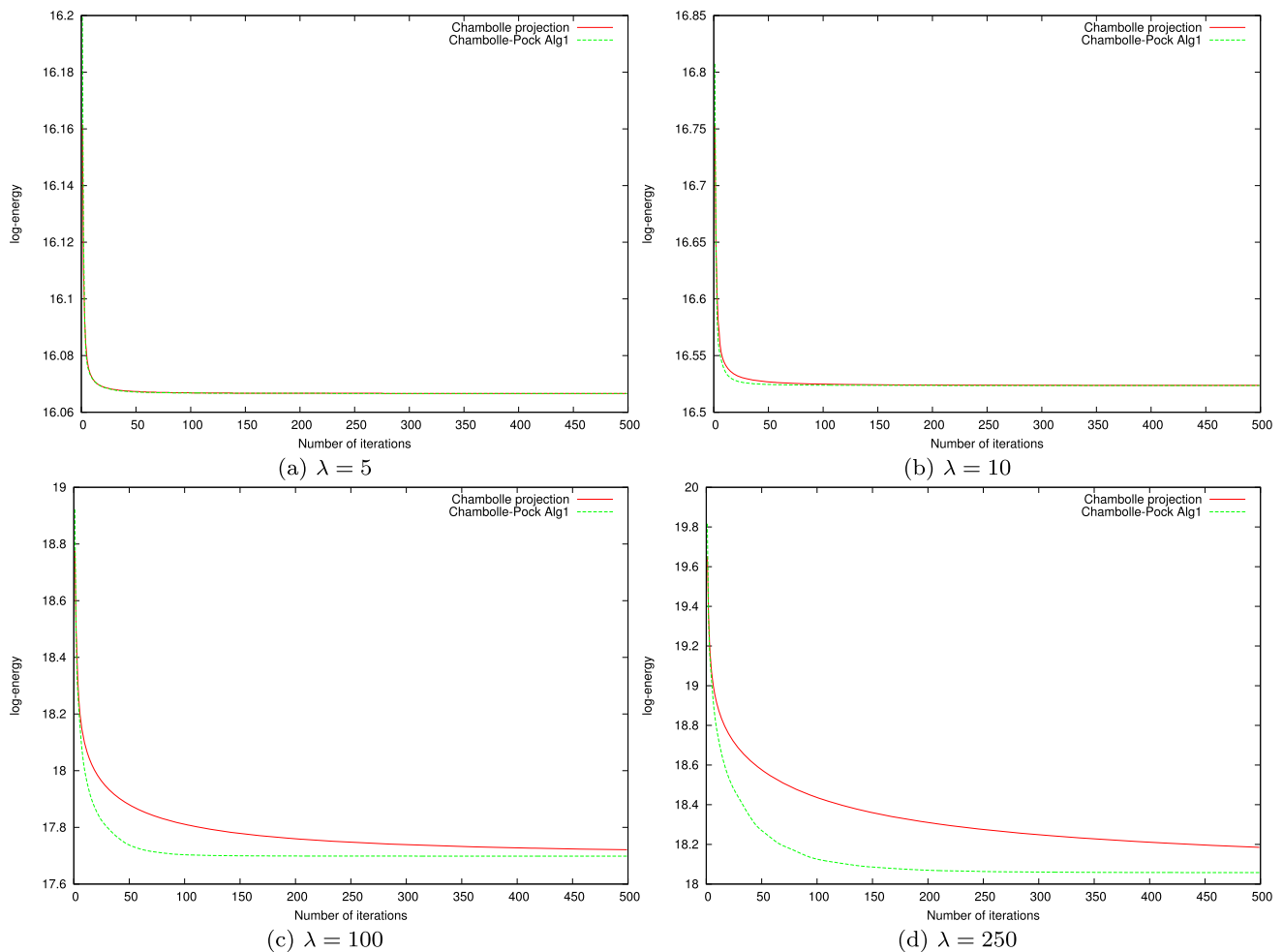
Starting with an initial parameter  $\lambda_0$ , a first decomposition of  $f$  is obtained by applying (28)–(29) with  $\lambda = \lambda_0$ , yielding  $f = u_0 + v_0$ . The layer  $u_0$  should be interpreted as a

**Fig. 3** Results of applying algorithms (37) and (42) with different values of  $\lambda$  to the standard Jetplane image



first sketch of the graph-signal  $f$ , while the residual  $v_0$  is to be understood as a *detail* coefficient. Modifying the scale parameter from  $\lambda_0$  to  $\lambda_1$  and applying (28)–(29) to  $v_0$  with  $\lambda = \lambda_1$  yields the subsequent decomposition  $v_0 = u_1 + v_1$ . Now the term  $u_1$  can be interpreted as a second layer extracted from  $f$  by means of  $v_0$ . Iterating the same process  $n$  times leads to identity (47). The  $u_i$ 's thus represent different layers of  $f$  captured at different scales. These layers are parametrized by three variables: the graph topology through

the adjacency matrix  $W$ , the energy function  $E$ , and the sequence  $\lambda_0, \dots, \lambda_n$  involved in the successive minimizations. It is clear that in order to extract the successive layers in a coherent manner, the sequence of scales  $(\lambda_i)_{i \geq 0}$  should be decreasing. In terms of image decomposition, this assumption has the following simple interpretation: as the process (46) evolves, the successive minimizers extract more texture from the original image [39]. The asymptotic behavior of (47) is studied in the following section.



**Fig. 4** Energy profiles in log-scale obtained when applying algorithms (37) and (42) with different values of  $\lambda$  to the standard Barbara image

### 4.2 Mathematical Study

It turns out that the characterization (35) is a powerful tool to study the convergence of our representation. We have the following result.

**Proposition 3** Consider a positive sequence  $(\lambda_n)_{n \geq 0}$ . Let  $f \in X$  and  $(u_i)_{i \geq 0}$  obtained by applying algorithm (46). We have

$$\left\| f - \sum_{i=0}^n u_i \right\|_{G_w} = \lambda_n. \tag{48}$$

If the sequence  $(\lambda_n)_{n \geq 0}$  is decreasing, then the series  $\sum_{i=0}^{\infty} u_i$  converges to  $f$

$$\sum_{i=0}^{\infty} u_i = f. \tag{49}$$

*Proof* The first equation follows immediately from the characterization (35). Indeed, we have  $f - \sum_{i=0}^n u_i = v_n = \text{proj}_{G_w^{\lambda_n}}(v_{n-1})$ .

If the sequence  $(\lambda_n)_{n \geq 0}$  is decreasing, then we get convergence for the  $G_w$  norm:

$$\lim_{n \rightarrow \infty} \left\| f - \sum_{i=0}^n u_i \right\|_{G_w} = 0, \tag{50}$$

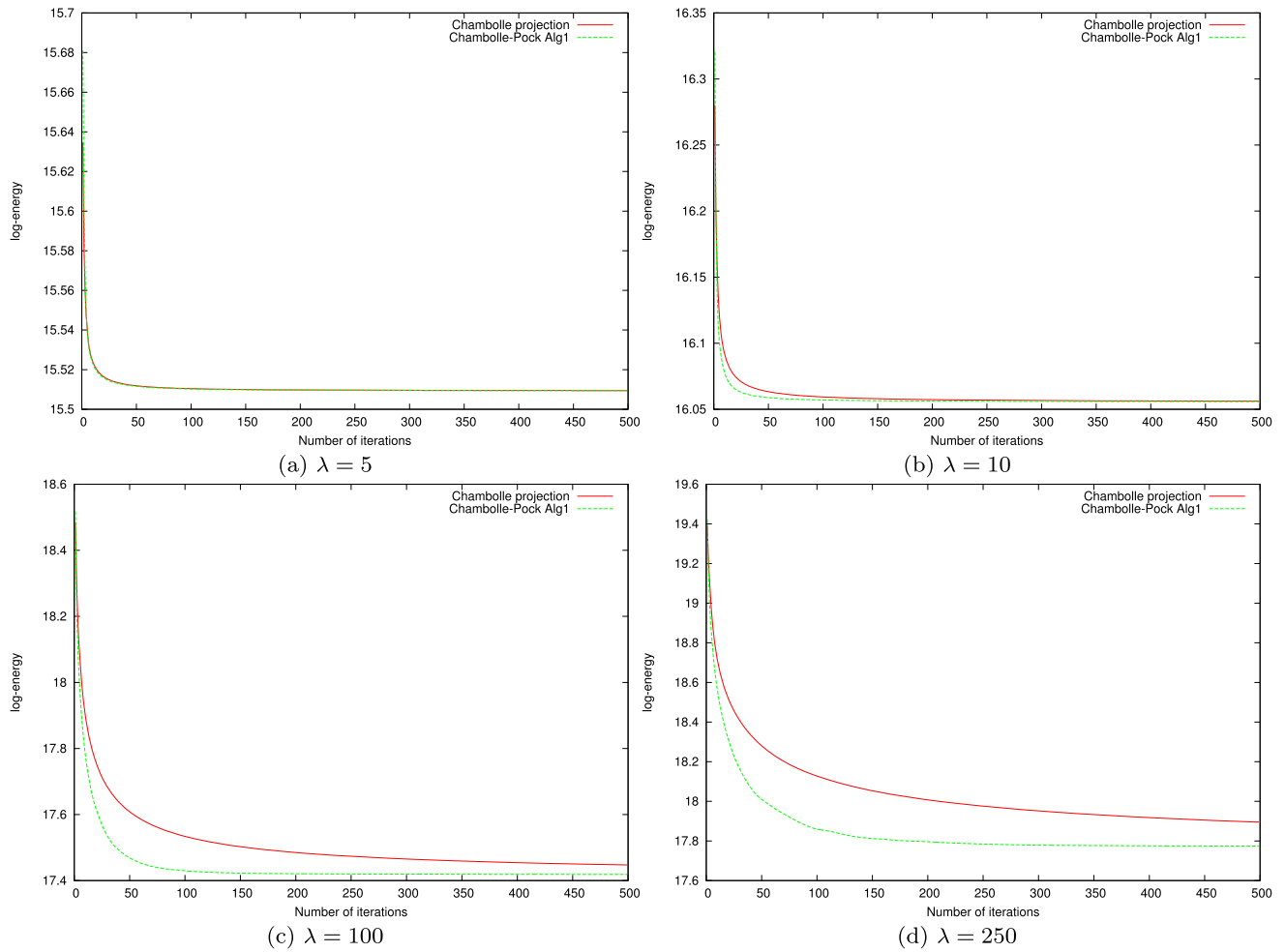
and hence, since we are in a finite-dimensional setting, we readily get the convergence for any other norm. In particular, we have

$$\lim_{n \rightarrow \infty} \left\| f - \sum_{i=0}^n u_i \right\|_X = 0. \tag{51}$$

□

*Remark 1* Equation (48) relates the rate of convergence of the decomposition (47) to that of the sequence of scales  $(\lambda_n)_{n \geq 0}$ .

We also have the following energy decomposition statement.



**Fig. 5** Energy profiles in log-scale obtained when applying algorithms (37) and (42) with different values of  $\lambda$  to the standard Jetplane image

**Proposition 4** Consider a positive and decreasing sequence  $(\lambda_n)_{n \geq 0}$ . Let  $f \in X$  and  $(u_i)_{i \geq 0}$  obtained by applying algorithm (46). Then the series whose general term is  $2\lambda_i J_w(u_i) + \|u_i\|_X^2$  converges to the energy of  $f$ :

$$\sum_{i=0}^{\infty} (2\lambda_i J_w(u_i) + \|u_i\|_X^2) = \|f\|_X^2. \tag{52}$$

In order to prove Proposition 2 we need the following lemma.

**Lemma 1** Let  $f \in X, \lambda > 0$ . Let  $u$  denote the minimizer of problem (28)–(29) and  $v = f - u$  the residual. We have

$$\langle u, v \rangle_X = \lambda J_w(u). \tag{53}$$

*Proof* Equation (35) shows that  $v = \text{proj}_{G_w^\lambda}(f)$ . Applying the characterization of a projection onto a closed convex set, we see that  $v$  is the unique vector in  $G_w^\lambda$  satisfying

$$\forall w \in G_w^\lambda, \quad \langle w - v, f - v \rangle_X \leq 0. \tag{54}$$

Hence, we have for all  $w \in G_w^\lambda$

$$\langle w - v, u \rangle_X \leq 0, \tag{55}$$

so that

$$\begin{aligned} \langle u, v \rangle_X &= \sup_{w \in G_w^\lambda} \langle u, w \rangle_X = \lambda \sup_{w \in G_w^1} \langle u, w \rangle_X, \\ &= \lambda \sigma_{G_w^1}(u), \\ &\stackrel{(27)}{=} \lambda J_w(u). \end{aligned} \tag{56}$$

□

We now give the proof of Proposition 2.

*Proof* For all  $i \geq -1$ , the decomposition  $v_i = u_{i+1} + v_{i+1}$  leads the identity

$$\|v_i\|_X^2 = \|v_{i+1}\|_X^2 + \|u_{i+1}\|_X^2 + 2\langle u_{i+1}, v_{i+1} \rangle_X. \tag{57}$$

Using Eq. (53), we have

$$\langle u_{i+1}, v_{i+1} \rangle_X = \lambda_{i+1} J_w(u_{i+1}),$$

so that

$$\|v_i\|_X^2 - \|v_{i+1}\|_X^2 = \|u_{i+1}\|_X^2 + 2\lambda_{i+1}J_w(u_{i+1}). \tag{58}$$

Summing for  $i$  from  $-1$  (with  $v_{-1} = f$ ) to  $n - 1$  leads to

$$\|f\|_X^2 - \|v_n\|_X^2 = \sum_{i=0}^n (2\lambda_i J_w(u_i) + \|u_i\|_X^2). \tag{59}$$

We achieve the result by taking the limit when  $n \rightarrow \infty$  and using Proposition 3.  $\square$

### 4.3 Choosing the Sequence of Scales

In the rest of the paper, we consider a dyadic progression of scales:  $\lambda_{i+1} = \frac{\lambda_i}{2}$ , with  $\lambda_0$  given. We are now reduced to the choice of the first scale  $\lambda_0$  which is an important issue since it determines the performance of the decomposition (47). In particular, if  $\lambda_0$  is chosen too small, then many levels of the hierarchy will be needed in order to recover the original data. For large graphs whose number of nodes and/or edges is important, the cost associated with such a decomposition can be prohibitive. In contrast, if  $\lambda_0$  is too important, the final decomposition might miss some of the scales present in the graph-signal  $f$ . The parameter  $\lambda_0$  is thus related to the greatest value  $\lambda$  that yields a nontrivial decomposition.<sup>3</sup> This choice can be made clear after the following proposition.

**Proposition 5** *Let  $f \in X$  and  $\lambda > 0$ . Let  $\bar{f}$  denote the vector whose components are all equal to the mean of  $f$ :  $\bar{f}_i = \frac{1}{N} \sum_{j=1}^N f_j$ , and  $u_\lambda$  the solution of (28)–(29). Then*

$$\begin{aligned} \|f - \bar{f}\|_{G_w} &\geq \lambda \|f - u_\lambda\|_{G_w} = \lambda, \\ \|f - \bar{f}\|_{G_w} &\leq \lambda u_\lambda = \bar{f}. \end{aligned} \tag{60}$$

Proposition 5 is the analogue of Theorem 3 in [28]. In our setting, this result is an immediate consequence of (35). It provides an interpretation of the  $G_w$  norm, namely that the open interval  $]0, \|f - \bar{f}\|_{G_w}[$  is the scale range which yields a nontrivial decomposition for the ROF model. Adopting a dyadic progression of scales  $\lambda_{i+1} = \lambda_i/2$ , and considering the result in Proposition 5 we can deduce the following bounds for  $\lambda_0$

$$\frac{1}{2} \|f - \bar{f}\|_{G_w} \leq \lambda_0 < \|f - \bar{f}\|_{G_w}. \tag{61}$$

We decide to chose the middle point estimate:  $\lambda_0 = \frac{\|f - \bar{f}\|_{G_w}}{4}$ . We are now led to the computation of  $\|f - \bar{f}\|_{G_w}$ . The details are given in the next sections.

<sup>3</sup>By a trivial decomposition we mean the decomposition  $f = u + v$  where  $u = \bar{f}$  is the vector whose all components are equal to the mean of  $f$ .

### 4.4 Multilayered Representation with the Presence of Noise

We consider here the case where the original data  $f$  is corrupted by additive noise. Let us fix the initial scale parameter  $\lambda_0$  according to (61), and denote  $(u_i)_{i \geq 0}$  the successive layers extracted by dyadic progression as explained above. According to Proposition 3, the series  $\sum_{i=0}^\infty u_i$  converges to the noisy data  $f$ . Due to the fact that the representation evolves in inverse scale order (coarse scales are recovered first), we see that in order to avoid integrating noise to the reconstruction, the expansion  $\sum_{i=0}^\infty u_i$  should be truncated at an adequate stopping time  $\tau$ . If the corrupting noise is additive white Gaussian with known variance  $\sigma^2$ , then one can use the discrepancy principle proposed in [32], which amounts to stopping at the first time  $\tau$  satisfying  $\|f - \sum_{i=0}^\tau u_i\|_X^2 \leq N\sigma^2$ .

## 5 Computing the $G_w$ Norm

In [4], the authors proposed to compute the  $G$  norm of a digital image by dichotomy, exploiting the result of Proposition 5. However, they report that it does not yield a precise approximation. A very close approach has also been taken in [38].

We propose in this section to use the recent primal-dual algorithm of [13] in order to compute the  $G_w$  norm. In the sequel, the sets  $X$  and  $Y$  are the ones defined in Sect. 3.1. Let  $u \in G_w$ . Computing the  $G_w$  norm of  $u$  amounts to solving the following convex problem (23):

$$\begin{aligned} &\underset{p \in Y}{\text{minimize}} \quad \|p\|_{\infty,2}, \\ &\text{subject to} \quad \text{div}_w(p) = u. \end{aligned} \tag{62}$$

Problem (62) can be rewritten in the form

$$\underset{p \in Y}{\text{minimize}} \quad \|p\|_{\infty,2} + \iota_{\{u\}} \circ \text{div}_w(p). \tag{63}$$

Applying algorithm (43) with  $F \equiv \iota_{\{u\}}$ ,  $K \equiv \text{div}_w$ ,  $G = \|\cdot\|_{\infty,2}$ , and  $\theta = 1$  yields the following iteration:

$$\begin{cases} p^{n+1} = \text{prox}_{\tau\|\cdot\|_{\infty,2}}(p^n + \tau \nabla_w \bar{v}^n), \\ v^{n+1} = \text{prox}_{\sigma\langle \cdot, u \rangle_X}(v^n + \sigma \text{div}_w p^{n+1}), \\ \bar{v}^{n+1} = 2v^{n+1} - v^n. \end{cases} \tag{64}$$

Note that the dual problem associated to (62) is the following

$$\underset{v \in X}{\text{minimize}} \quad \langle u, v \rangle_X, \tag{65}$$

subject to  $J_w(v) \leq 1$ ,

which involves a constraint on the total variation of the optimization variable  $v$ .

### 5.1 Computing Proximity Operators

The computation of the second proximity operator in (64) is straightforward because of the linearity of the involved function. The expression is given by

$$\forall x \in X, \quad \text{prox}_{\sigma(\cdot, u)_X}(x) = x - \sigma u. \tag{66}$$

In order to compute the first proximity operator in (64), we apply Moreau’s identity (14), use the fact that the conjugate of a norm is the support function of its dual norm unit ball (8), and that  $\|\cdot\|_{\infty, 2}$  is the dual norm of  $\|\cdot\|_{1, 2}$ . Hence, we have

$$\begin{aligned} \text{prox}_{\tau \|\cdot\|_{\infty, 2}}(y) &\stackrel{(14)}{=} y - \tau \text{prox}_{\|\cdot\|_{\infty, 2}^* / \tau}(y/\tau) \\ &\stackrel{(9)}{=} y - \tau \text{prox}_{B_{1, 2}(\tau)}(y/\tau) \\ &\stackrel{(6)}{=} y - \underset{B_{1, 2}(\tau)}{\text{proj}}(y), \end{aligned} \tag{67}$$

where  $B_{1, 2}(\tau)$  denotes the  $\|\cdot\|_{1, 2}$  closed ball of radius  $\tau$ .

### 5.2 Computing the Projection onto an $\|\cdot\|_{1, 2}$ ball

We have just seen that computing the first proximity operator in (64) is related to the computation of the projection of a matrix  $y \in Y$  onto a  $\|\cdot\|_{1, 2}$  ball. We propose in this section to extend the projection algorithm proposed in [8, 19], which computes the  $\ell_2$  projection of a vector in the Euclidean space  $\mathbb{R}^n$  onto an  $\ell_1$  ball. In order to do so, we need to introduce the soft-thresholding operator acting on matrices  $y \in Y$ . A similar extension to vector fields has recently been proposed in [21] and is used as an intermediate step in order to compute the projection of a vector onto a TV ball of pre-specified radius.

The soft-thresholding operator of level  $\mu \geq 0$  on  $Y$ ,  $\mathcal{S}_\mu : Y \rightarrow Y$ , is defined by

$$(\mathcal{S}_\mu y)_{i, j} = \begin{cases} 0 & \text{if } \|y_{i, \cdot}^\top\|_X \leq \mu, \\ \left(1 - \frac{\mu}{\|y_{i, \cdot}^\top\|_X}\right) y_{i, j} & \text{if } \|y_{i, \cdot}^\top\|_X > \mu. \end{cases} \tag{68}$$

Thus we have

$$\begin{aligned} \|\mathcal{S}_\mu y\|_{1, 2} &= \sum_{i=1}^N \|(\mathcal{S}_\mu y)_{i, \cdot}^\top\|_X = \sum_{i=1}^N \left( \sum_{j=1}^N (\mathcal{S}_\mu y)_{i, j}^2 \right)^{1/2} \\ &= \sum_{\substack{i=1 \\ \|y_{i, \cdot}^\top\|_X \geq \mu}}^N \left(1 - \frac{\mu}{\|y_{i, \cdot}^\top\|_X}\right) \|y_{i, \cdot}^\top\|_X \\ &= \sum_{\substack{i=1 \\ \|y_{i, \cdot}^\top\|_X \geq \mu}}^N (\|y_{i, \cdot}^\top\|_X - \mu). \end{aligned} \tag{69}$$

The following proposition relates the soft-thresholding operator defined in (68) to the computation of the projection onto an  $\|\cdot\|_{1, 2}$  ball.

**Proposition 6** *Let  $y \in Y$ ,  $R > 0$ . If  $\|y\|_{1, 2} \leq R$ , then  $\text{proj}_{B_{1, 2}(R)}(y) = y$ . Otherwise, there exists a unique  $\mu > 0$  such that  $\text{proj}_{B_{1, 2}(R)}(y) = \mathcal{S}_\mu y$ .*

*Proof* The first assumption follows from the definition of the projection operator. Suppose that  $\|y\|_{1, 2} > R$ . First observe that in view of Eq. (69), the mapping  $\mu \mapsto \|\mathcal{S}_\mu y\|_{1, 2}$  from  $[0, \infty[$  to  $\mathbb{R}$  is piecewise linear, continuous and non-increasing with  $\|\mathcal{S}_0 y\|_{1, 2} = \|y\|_{1, 2}$  and  $\|\mathcal{S}_\mu y\|_{1, 2} = 0$  for all  $\mu \geq \max_{i=1, \dots, N} \|y_{i, \cdot}^\top\|_X$ . Thus there exists a unique  $\mu > 0$  such that  $\|\mathcal{S}_\mu y\|_{1, 2} = R$ . To show that  $\mathcal{S}_\mu y = \text{proj}_{B_{1, 2}(R)}(y)$  it suffices to observe that  $\mathcal{S}_\mu y$  is the unique solution (see [11] for the vectorial case) of the problem

$$\underset{z \in Y}{\text{minimize}} \|z - y\|_Y^2 + \mu \|z\|_{1, 2}. \tag{70}$$

Hence

$$\forall z \in B_{1, 2}(R), \|\mathcal{S}_\mu y - y\|_Y^2 \leq \|z - y\|_Y^2 + \mu (\|z\|_{1, 2} - R), \tag{71}$$

so that

$$\forall z \in B_{1, 2}(R), \|\mathcal{S}_\mu y - y\|_Y^2 \leq \|z - y\|_Y^2, \tag{72}$$

which concludes the proof.  $\square$

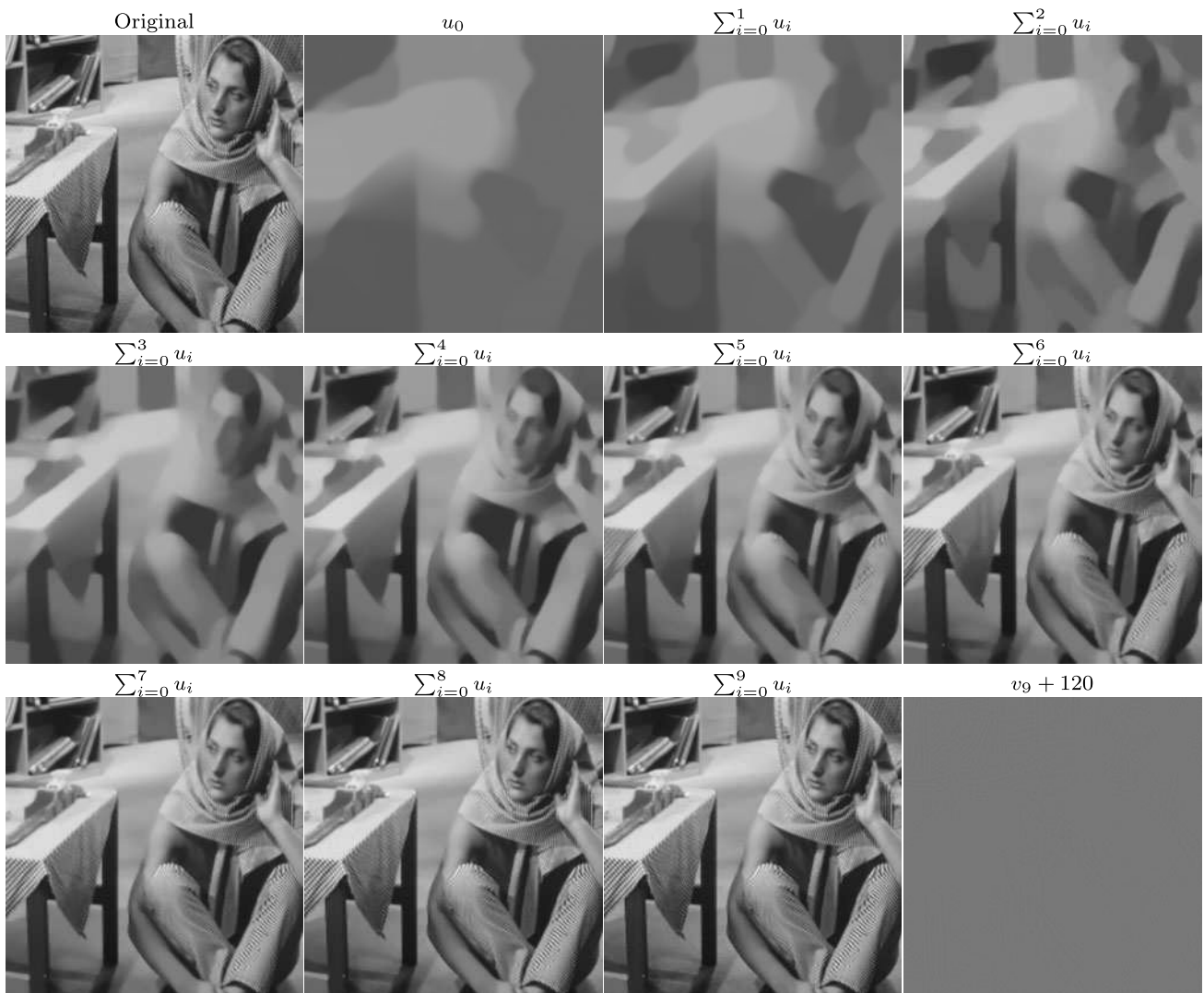
Coming back to Eq. (69), we see that the slope of the piecewise linear mapping  $\mu \mapsto \|\mathcal{S}_\mu y\|_{1, 2}$  changes at the points  $\|y_{i, \cdot}^\top\|_X$ ,  $i = 1, \dots, N$ . This leads to the following steps to compute the projection summarized in Algorithm 1. Algorithm 2 summarizes the computation of the  $G_w$  norm. Finally, Algorithm 3 summarizes the proposed multilayered representation we propose.

---

#### Algorithm 1 Projection onto $\|\cdot\|_{1, 2}$ ball of radius $R$

---

- 1: INPUT:  $y \in Y$ ,  $0 < R \leq \|y\|_{1, 2}$ .
  - 2: OUTPUT:  $\text{proj}_{B_{1, 2}(R)}(y)$ .
  - 3: Compute:  $\|y_{i, \cdot}^\top\|_X$  for  $i = 1 \dots n$ .
  - 4: Sort  $\|y_{i, \cdot}^\top\|_X$  in descending order to obtain  $a_n \leq a_{n-1} \leq \dots \leq a_1$ .
  - 5: Compute  $b_i := \|\mathcal{S}_{a_i} y\| = \sum_{j=1}^i (a_j - a_i)$  for  $i = 1 \dots n$ .
  - 6: Find  $k$  such that  $b_k \leq R < b_{k+1}$ .
  - 7:  $\mu \leftarrow a_k + \frac{b_k - R}{k}$ .
  - 8:  $\text{proj}_{B_{1, 2}(R)}(y) \leftarrow \mathcal{S}_\mu y$ .
-



**Fig. 6** Sum of the successive layers extracted from the standard image Barbara. Results are obtained by considering a 4-connectivity unweighted graph. The *bottom right image* represents the last residual

---

**Algorithm 2** Computing the  $G_w$  norm of  $y \in G_w$

---

- 1: INPUT:  $y \in G_w$ .
- 2: OUTPUT:  $\|y\|_{G_w}$ .
- 3: set  $p^0 = 0, v^0 = \bar{v}^0 = 0$
- 4: **while** (not converged) **do**
- 5:    $p^{n+1} = p^n + \tau \nabla_w \bar{v}^n - \text{proj}_{B_{1,2}(\tau)}(p^n + \tau \nabla_w \bar{v}^n)$ ,
- 6:    $v^{n+1} = v^n + \sigma(\text{div}_w p^{n+1} - u)$ ,
- 7:    $\bar{v}^{n+1} = 2v^{n+1} - v^n$ .
- 8: **end while**
- 9:  $\|y\|_{G_w} = \text{div}_w p^\infty$ .

---

## 6 Experiments

We present in this section results obtained by applying the proposed multilayered representation to three types of digi-

---

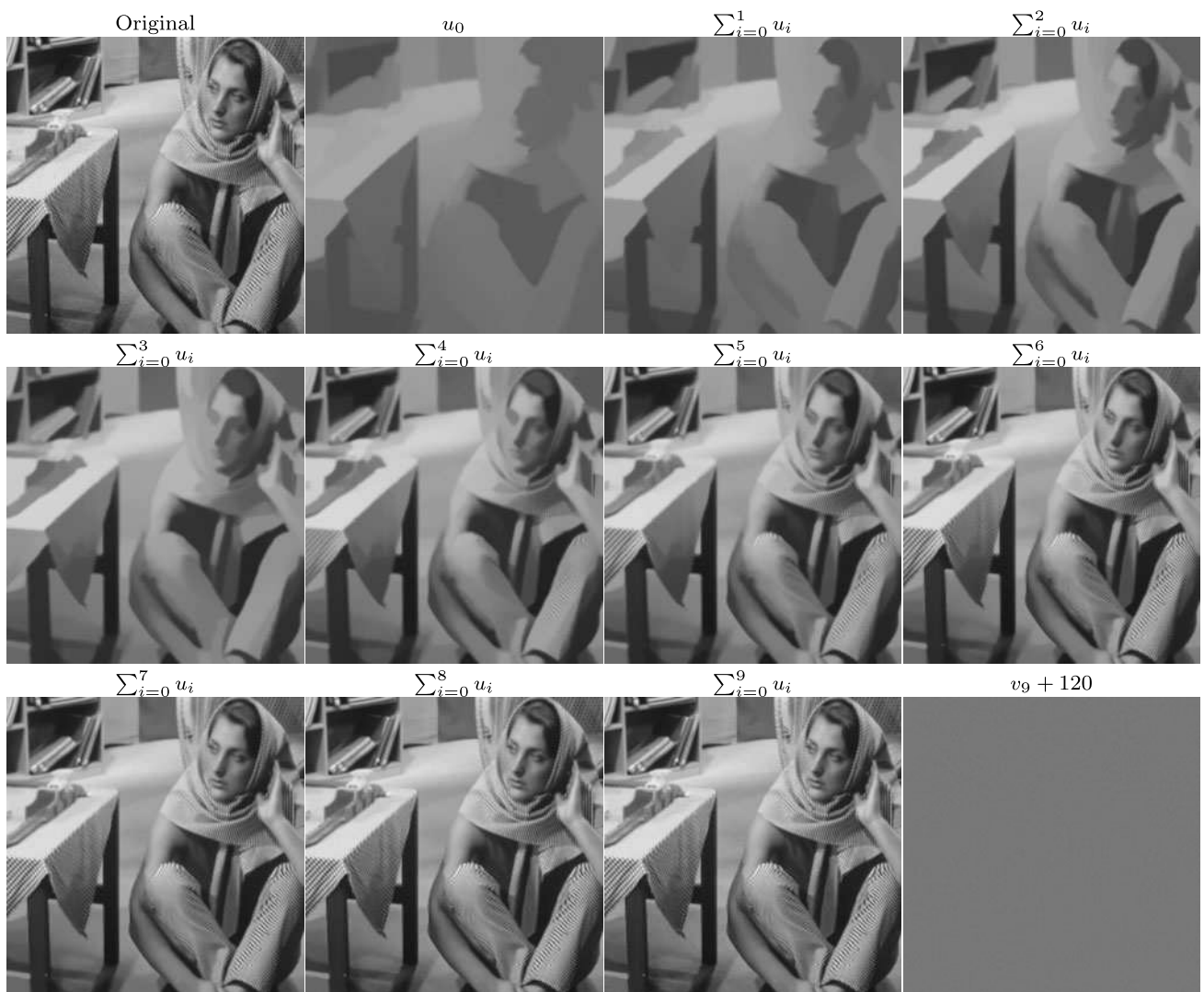
**Algorithm 3** Multilayered representation of graph-signals

---

- 1: INPUT:  $f \in X, N \geq 0$ .
- 2: OUTPUT:  $(u_i)_{0 \leq i \leq N}$ .
- 3:  $v_{-1} \leftarrow f$
- 4: Compute  $\lambda_0 = \|f - \bar{f}\|_{G_w}$  using Algorithm 2.
- 5: **for**  $i = -1 \rightarrow N - 1$  **do**
- 6:   Decompose  $v_i$  into  $u_{i+1}$  and  $v_{i+1}$  using model (28)–(29) using  $\lambda_{i+1}$ .
- 7:    $\lambda_{i+2} \leftarrow \frac{1}{2} \lambda_{i+1}$ .
- 8: **end for**

---

tal data, namely digital images, 3-D triangular meshes, and point clouds. In order to obtain the representations, appropriate similarity graphs are associated with each data type. While the construction of similarity graphs is a task by it-



**Fig. 7** Sum of the successive layers extracted from the standard image Barbara. Results are obtained by considering a 10-nearest neighbors unweighted graph. The *bottom right image* represents the last residual

self, we try to give specific advices concerning each particular data type. The next subsection explains the general setting.

### 6.1 General Setting

Let  $F$  denote the input data we want to decompose. We consider  $F$  to be in the space  $\mathbb{R}^{n \times m}$ , where  $n$  is the number of instances, and  $m$  the number of components. When  $m > 1$ , the data is multivalued: to each vertex  $i$  is associated a vector of observations  $F_{i,\cdot}^T \in \mathbb{R}^m$ . In the latter case, the decomposition is performed component-wise, the scale parameters  $(\lambda_i)_{i \geq 0}$  being common to all the components.

For all the experiments, the first scale parameter  $\lambda_0$  in (46) is computed automatically, except for meshes with color attributes of Sect. 6.3.1 where it has been set manually. When the input data  $F$  is in  $\mathbb{R}^{n \times 1}$ , the value of  $\lambda_0$  is

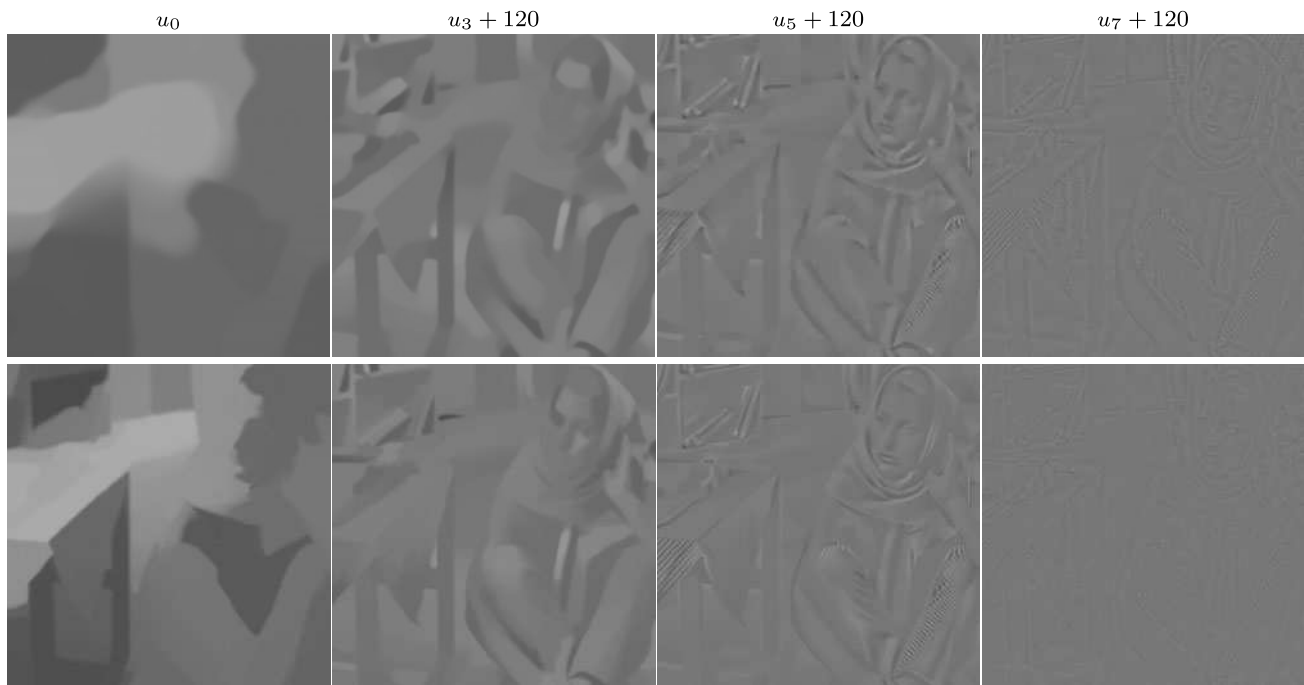
set to  $\|F - \bar{F}\|_{G_w}/4$  as explained in Sect. 4.3. This is for example the case for grayscale images. When  $F$  is multivalued,  $\lambda_0$  is set to the maximum of the  $G_w$  norms of the components:  $\lambda_0 = \max_{1 \leq i \leq m} \|F_{i,\cdot}^T - \bar{F}_{i,\cdot}^T\|_{G_w}/4$ . In all cases,  $G_w$  norms are computed using Algorithm 2.

Finally, each decomposition is performed using Algorithm 1 of Chambolle and Pock [13] for which the actual derivation in our case was given in Sect. 3.5.2.

### 6.2 Images

With each digital image we associate a weighted graph whose set of vertices is the set of pixels of the given image. Different strategies can be adopted in order to construct the weighted adjacency matrix. The first natural choice consists in adopting a 4-connectivity lattice graph, where each vertex





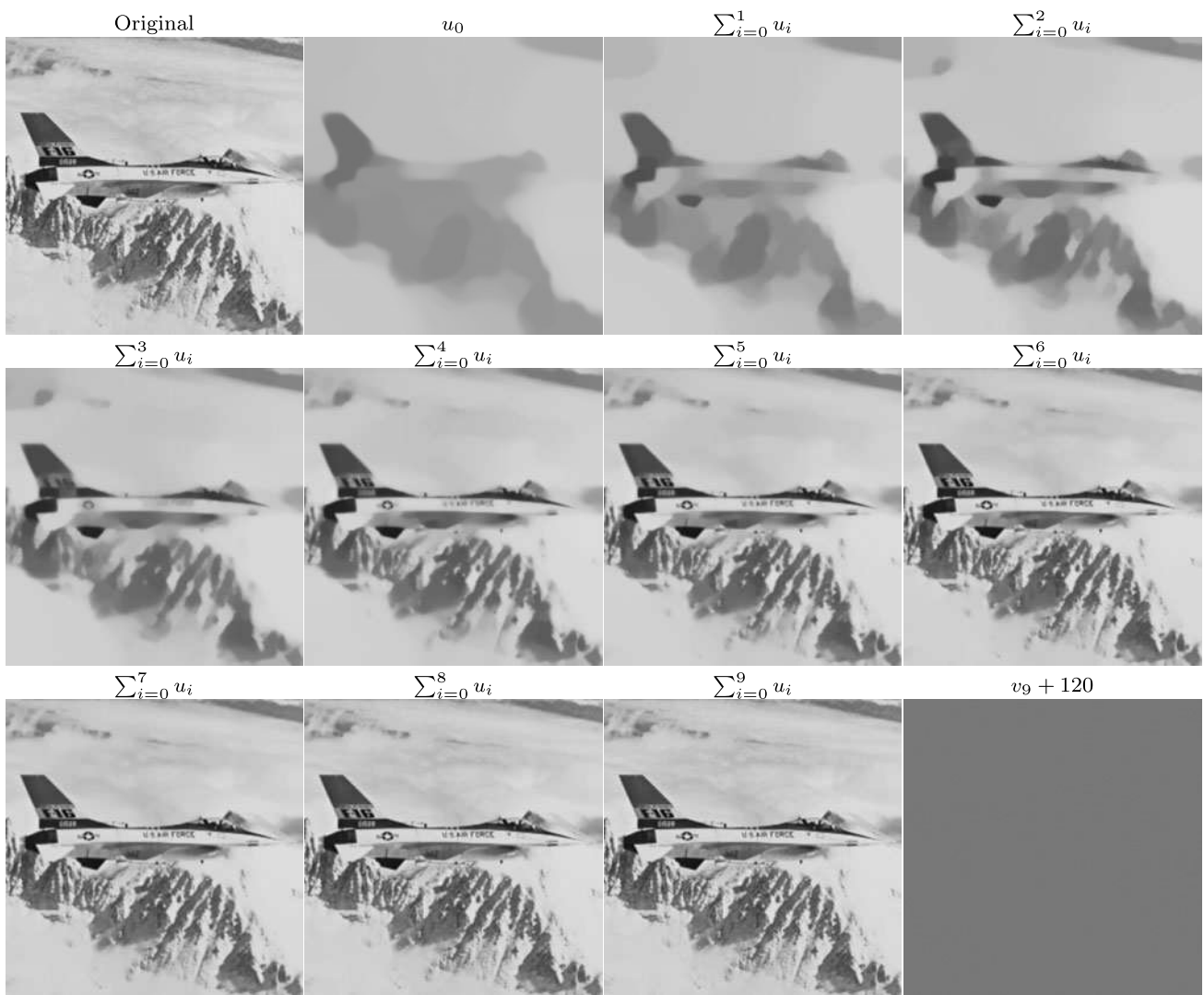
**Fig. 8** Layers extracted from Barbara image. *Top row* is obtained using a 4-connectivity unweighted graph. *Second row* is obtained using a 10-nearest neighbors unweighted graph

is connected to its 4 neighbors and weights are set to 1. Figures 6 and 9 show the sums of the successive layers obtained when using the latter graph construction on the two standard images Barbara and Jetplane of Fig. 1. One can see clearly the reconstruction being performed as the successive layers are extracted and summed up, in accordance with Proposition 1. First levels exhibit a very rough description of the original images. A first geometrical description is obtained at the fifth level ( $\sum_{i=0}^4 u_i$ ) for the image Barbara, and at the sixth level ( $\sum_{i=0}^5 u_i$ ) for the Jetplane image. First rows of Figs. 8 and 11 show the successive layers extracted. Therein, we can see how texture is iteratively resolved as the process evolves. The second to last layers are enhanced by adding a factor of 120 for better visualization.

Though the lattice graph construction is natural for digital images, other graph constructions can be considered. The general class of similarity graphs is often used to model digital images. In this setting, a given distance measure is first computed, then transformed into a similarity measure upon which the weighted graph adjacency matrix is computed [20, 27]. Since the introduction of the NL-Means algorithm [7], patch-based metrics have been popular in image processing. In this setting, the distance between two pixels  $u$  and  $v$  does not depend solely on the values at  $u$  and  $v$  but rather on the values taken inside square windows centered at each pixel, called patches. The distance between  $u$  and  $v$ ,  $d(u, v)$ , is then computed as a weighted Euclidean distance between the vectors collecting the values of each

patch. In this work, we consider  $5 \times 5$  square patches, and the distance between two vertices is set to the Euclidean distance between the vectors collecting the values inside each patch. Based on the latter distance, the most used graph construction consists in fixing a search window  $S_u$  around each pixel  $u$  (typically a  $21 \times 21$  square window), connecting  $u$  to all  $v \in S_u - \{u\}$ , and weighting the edges  $(u, v)$  using a Gaussian kernel:  $w_{u,v} = e^{-\frac{d(u,v)^2}{2h^2}}$ . While the latter approach proved its efficiency for the denoising task [7], where the parameter  $h$  is related to the noise level, its adaption to the decomposition task is not straightforward. We instead propose to form a nearest-neighbors graph. In this setting we fix a search window (a  $21 \times 21$  square window in the experiments) and connect its center to its  $N$  nearest-neighbors inside the search window. The value of  $N$  is set to 10 in the experiments. The latter construction leads to a directed graph (the matrix  $W$  is not symmetric). We cast it into an undirected graph simply by ignoring the orientation of the edges, so that we are led to a graph where each vertex is connected to *at least*  $N$  neighbors. Once the neighbors are found, the associated edges are given a unit weight. Furthermore, the nearest-neighbors graph is coupled with the standard 4-connectivity lattice graph to ensure that the resulting graph is connected. Notice that the initial scale  $\lambda_0$  changes with respect to the graph topology.

Figures 7 and 10 show the sums of the successive layers obtained when using the latter graph construction on the two standard images Barbara and Jetplane. Again one can



**Fig. 9** Sum of the successive layers extracted from the standard image Jetplane. The results are obtained by considering a 4-connectivity un-weighted graph. The *bottom right image* represents the last residual

see the reconstruction being performed as the successive layers are extracted and summed up. However, in contrast with the lattice graph results, we see that the first levels exhibit a more geometric description of the input images so that the geometric part of both images are reconstructed earlier. To quantify this observation, we show in Fig. 12 the mean square errors (MSE) between the original images and the sums of the successive layers. For the partly textured image Barbara, the nonlocal graph construction clearly outperforms the lattice graph construction. Finally, second rows of Figs. 8 and 11 show the successive layers extracted when using a nearest-neighbors graph.

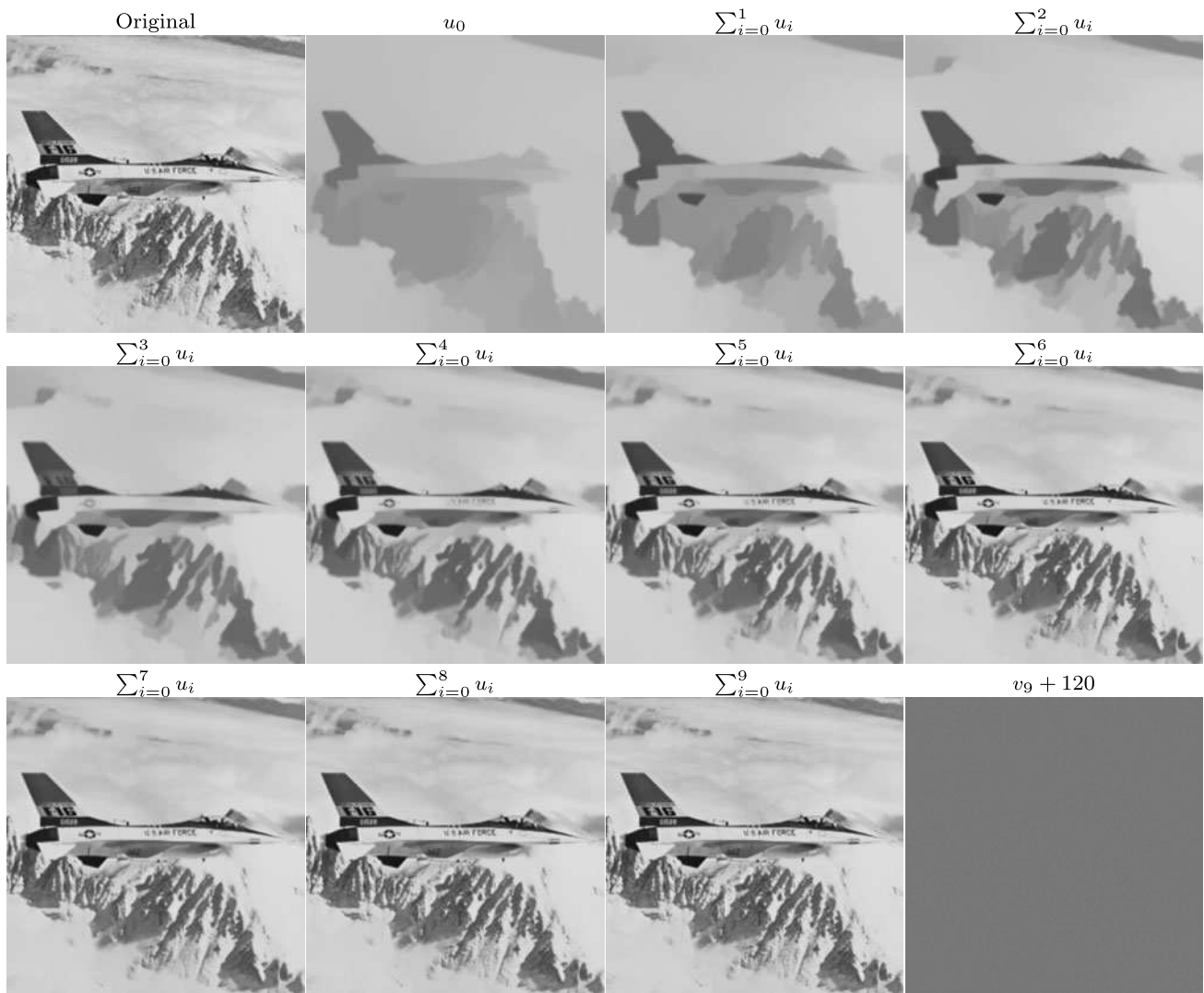
### 6.3 Nonuniformly Sampled Data

We show in this section that the proposed approach can be used to generate multiscale descriptions of nonuniformly

sampled data. We illustrate this capability on two types of data, namely, 3-D triangular meshes and point clouds.

#### 6.3.1 3-D Triangular Meshes

A triangular mesh consists of a set of vertices along with a triangle-vertex incidence graph. Each vertex is described by a set of geometrical attributes and optionally by a set of photometric attributes. The geometrical attributes are the coordinates of each vertex and optionally the normals, while the photometric attributes can be colors or texture coordinates per vertex. For a given mesh, each of the aforementioned attributes leads to a specific signal on the triangle-vertex incidence graph of the mesh, thus leading to a specific graph-signal. We concentrate in this subsection on two specific instances of such graph-signals, namely meshes with coordinates attributes, and meshes with vertex colors attributes.



**Fig. 10** Sum of the successive layers extracted from the standard image Jetplane. The results are obtained by considering a 10-nearest neighbors unweighted graph. The *bottom right image* represents the last residual

Let us begin with multiscale descriptions of meshes with coordinates attributes. Here, the aim is to enhance the description of a geometric model by describing its surface at different smoothness levels [34]. This is different from multiresolution representations where the goal is to describe the original model with varying sampling resolution.

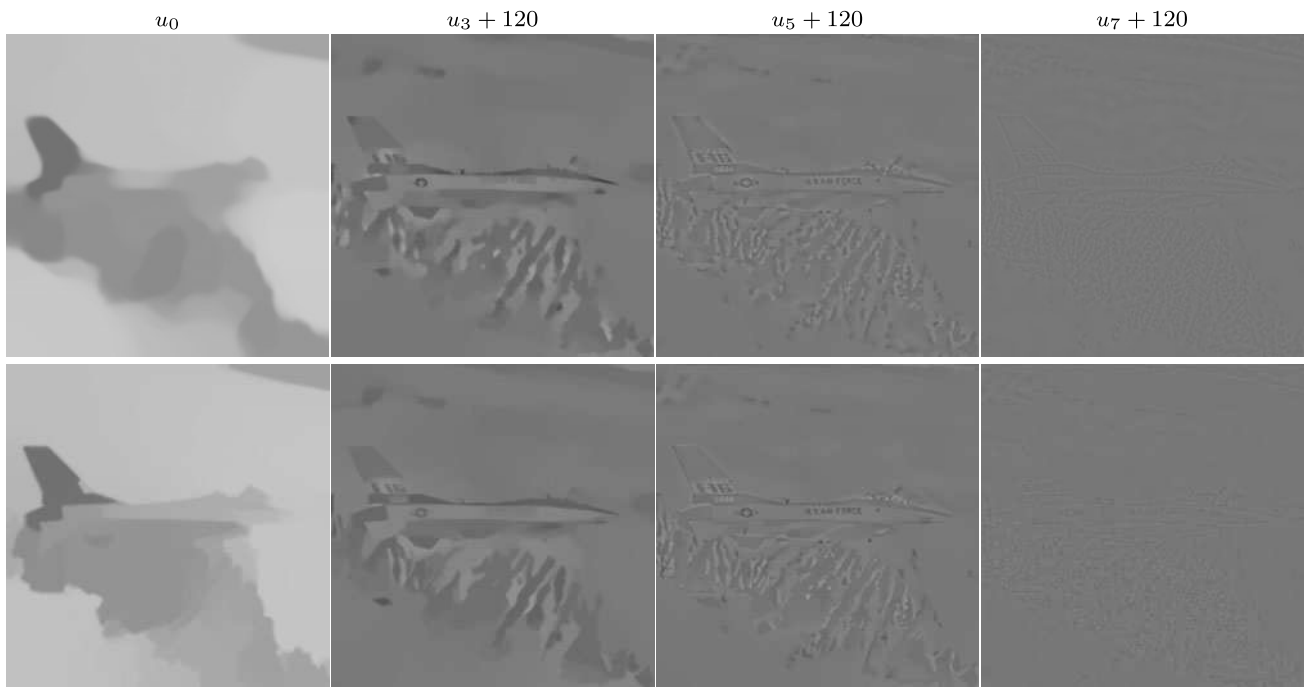
Figure 13 shows the sum of the successive layers extracted from the top left original head 3-D model. The result is obtained by applying our multilayered representation to the coordinates signal. Since the latter signal is multivalued, the successive regularizations are performed component-wise. Graph edges have been assigned weights using the inverse distance between their endpoints:

$$w_{u,v} = 1/(d(u, v) + \epsilon),$$

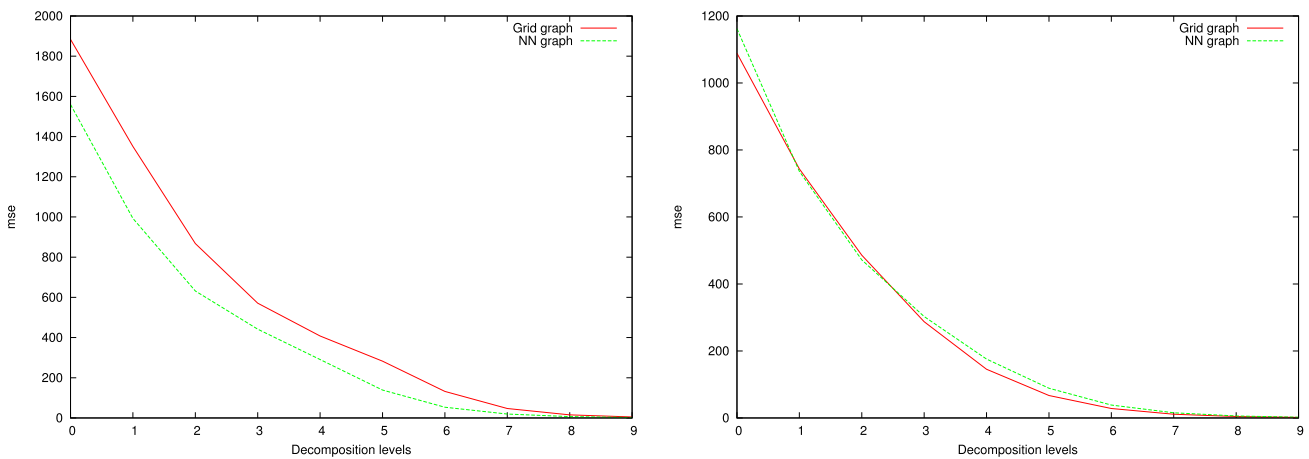
where  $d(u, v)$  is the Euclidean distance between  $u$  and  $v$ , and  $\epsilon = 10^{-4}$ .

Once again, the representation starts with a very rough description of the input data, and the geometrical details are recovered as the successive layers are extracted. At the twelfth level ( $\sum_{i=0}^{11}$ ), the original model is completely recovered. Notice however how the volume of the model corresponding to  $u_0$  has been shrunk in comparison to that of the original model. This suggests that the volume preservation property, if desired, should be built into the regularizing functional by means of a constraint.

We now move on to multiscale descriptions of meshes with vertex colors attributes. In this setting, each vertex has, in addition to its three spatial coordinates, three color attributes corresponding to RGB values. The top model of



**Fig. 11** Layers extracted from Jetplane image. *Top row* is obtained using a 4-connectivity unweighted graph. *Second row* is obtained using a 10-nearest neighbors unweighted graph

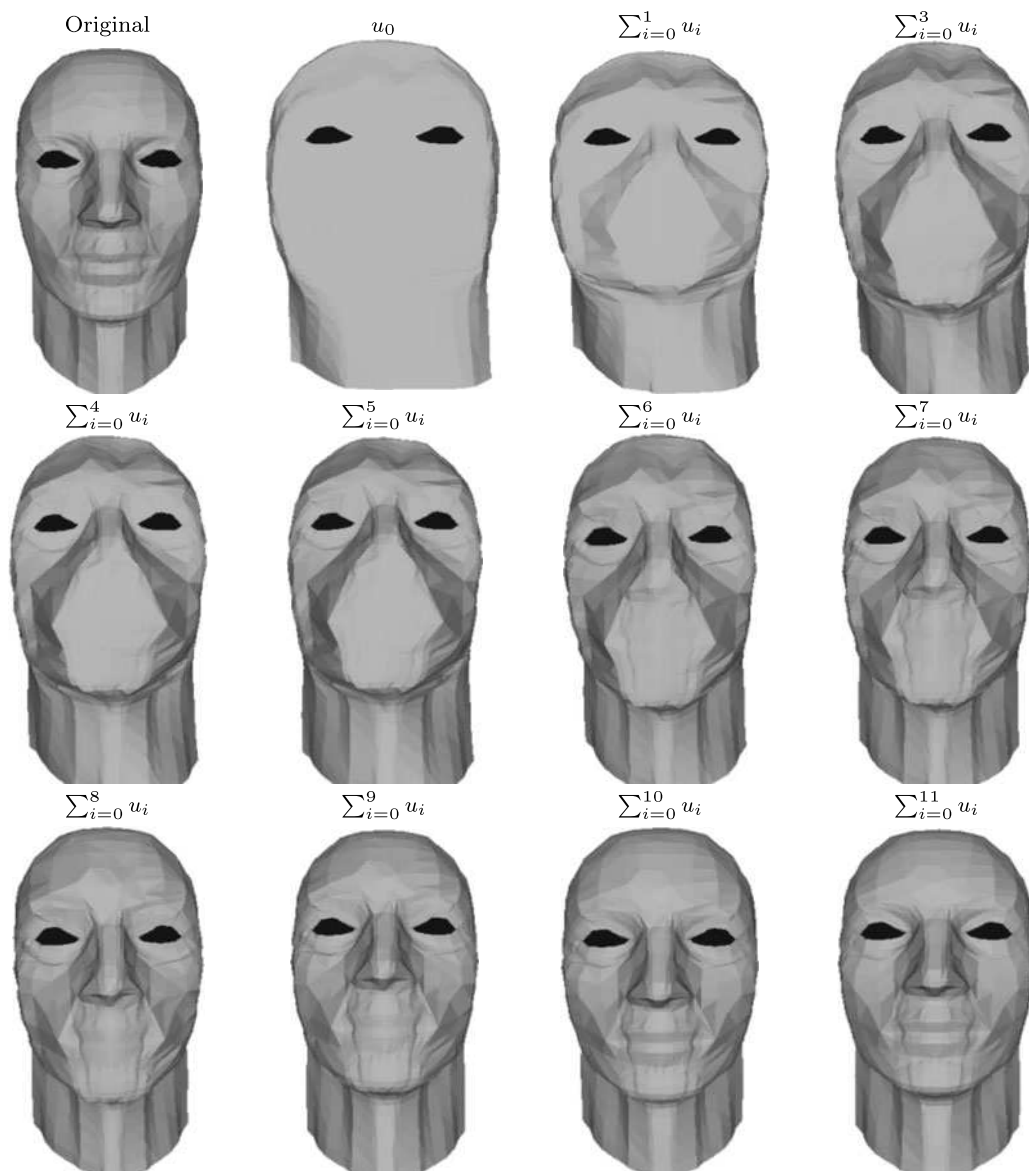


**Fig. 12** Mean square error (MSE) between the sum of the layers and the original image for two graph constructions. *Left figure*: Barbara image, 4-connectivity unweighted graph in *solid line*, 10-nearest neighbors unweighted graph in *dashed line*. *Right figure*: Jetplane image, 4-connectivity unweighted graph in *solid line*, 10-nearest neighbors unweighted graph in *dashed line*

Fig. 14 shows an example of such a mesh.<sup>4</sup> The graph we consider is the triangle-vertex incidence graph given by the mesh, and the signal is composed of the color attributes of each vertex. Our multilayered representation manipulates the color attributes without changing the coordinates of the

vertices. It leads to a multilayered representation of the colors of the model, similar to the ones presented above for digital images, the difference being that the support of the signal is now nonuniform. In addition, the processing is done component-wise. Note that the last residual  $v_9$  still contains some texture and that the sum of the six first layers,  $\sum_{i=0}^5 u_i$ , can be interpreted as the structure part of a structure-texture decomposition of the original model. In particular,  $\sum_{i=0}^5 u_i$  can be used to segment the 3-D model into different mean-

<sup>4</sup>The model is taken from the sample dataset of the Cyberware Head & Face Color 3D Scanner available at: <http://www.cyberware.com/>.



**Fig. 13** Sum of the successive layers extracted from the top left 3-D triangular mesh. The graph structure is given by the input mesh. The edges are weighted as the inverse of the Euclidean distance. The signal is the coordinates of each vertex

ingful components, based on the homogeneity of the color signal. This segmentation can be transferred back to the original model since there is a one to one correspondence between the models.

We now give an example of decomposition in the presence of noise. The top model of Fig. 15 is obtained by adding Gaussian white noise of standard deviation  $\sigma = 20$  to the color attributes of the model of Fig. 14. The noise is added to each of the three RGB components. This example is given to support the remark made in Sect. 4.4. We see that as the multilayered algorithm evolves, it extracts more details from the successive residuals. The key observation here is that, moving from level 8 ( $\sum_{i=0}^7$ ) to level 9 ( $\sum_{i=0}^8$ ), we see that

part of the noise has been added back to the reconstruction. Thus, the stopping time in this example should be  $\tau = 7$ . The bottom right image shows indeed that at level 8, the residual is composed mainly of noise, and no further decomposition should be performed.

### 6.3.2 Point Clouds

Finally, we consider the decomposition of a noisy toroidal helix in Fig. 16. Each point in 3-D space is mapped to a vertex. A nearest-neighbors graph is constructed based on the Euclidean distance between two points. For  $N = 10$ , the resulting graph is connected. Finally, the edges are

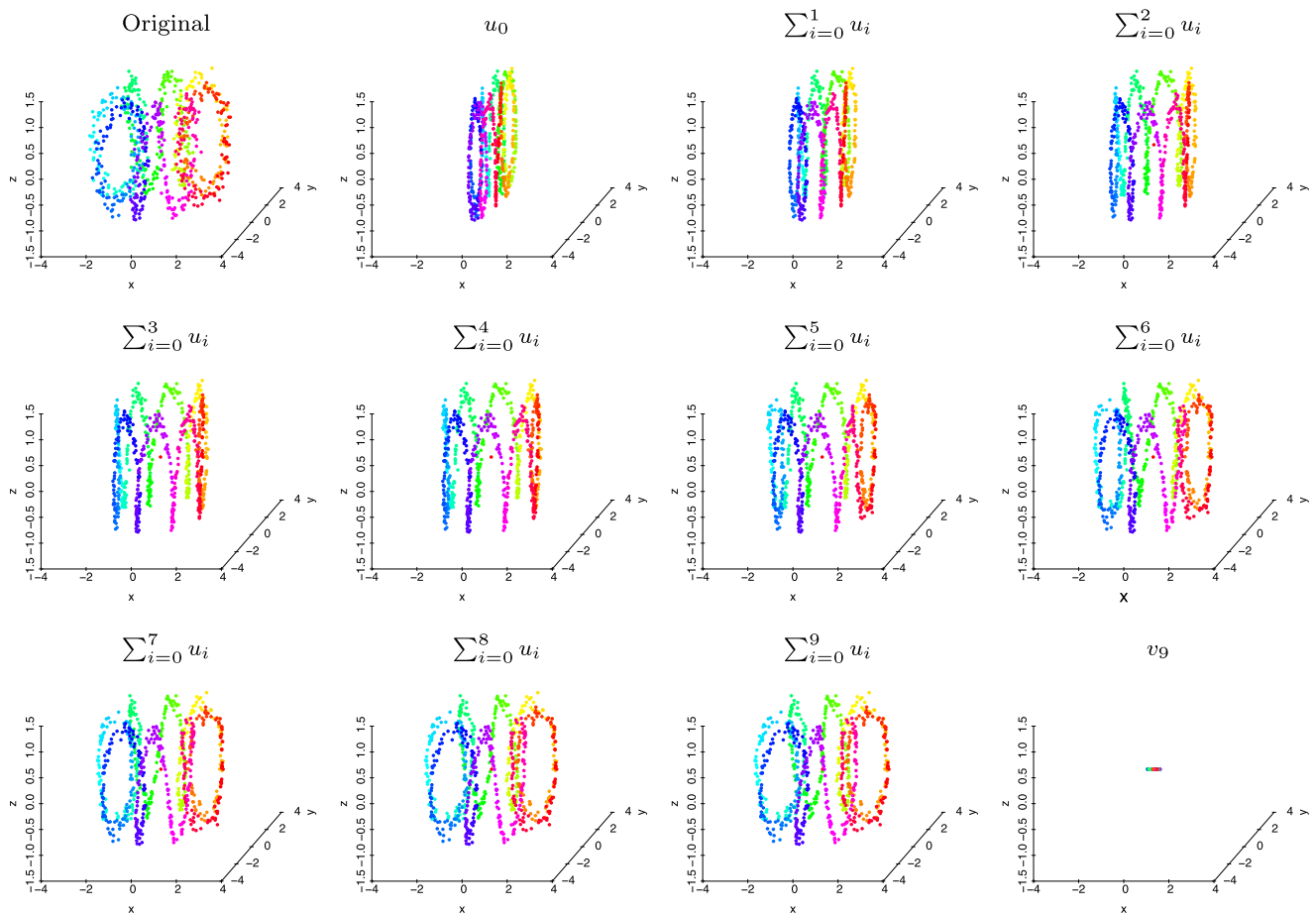


**Fig. 14** Sum of the successive layers extracted from the top 3-D triangular mesh. The graph structure is given by the input mesh. The edges are unweighted. The signal is the RGB color values associated with each vertex. The *bottom right model* represents the tenth residual



**Fig. 15** Sum of the successive layers extracted from the top noisy 3-D triangular mesh. The graph structure is given by the input mesh. The edges are unweighted. The signal is the RGB color values associated

with each vertex. The *bottom right model* represents the eighth residual and is composed mainly of noise



**Fig. 16** Sum of the successive layers extracted from the top left noisy toroidal helix. The results are obtained by considering a 10-nearest neighbors graph. The edges are weighted as the inverse of the Euclidean distance. The *bottom right plot* represents the last residual

weighted as for meshes, taking the inverse of the Euclidean distance. The sums of the successive layers are displayed in Fig. 16 where a unique color has been associated with each vertex of the graph. The bottom right plot shows the tenth residual ( $v_9$ ) which is almost reduced to a single point.

### 7 Conclusion

We have presented a method that allows to generate adaptive multiscale descriptions of general digital data sets that can be supported by a weighted graph structure. The proposed method works by extracting successive layers from the input data in a nonlinear way. Those layers are obtained by successive minimizations of a graph variant of the classical TV regularization approach. The behavior of the method has been studied and quantified. Issues regarding the choice of the scale parameters have been formulated as a convex optimization problem and solved using the recent primal-dual algorithm of [13]. The latter formulation draws connections

with previous works concerning the  $G$  norm introduced in [28]. We have shown how the proposed method can be applied to generate multiscale representations of digital images. In this setting, we have shown how the incorporation of nonlocal interactions can improve the multiscale description. We have also shown how the proposed method can be applied to nonuniformly sampled data and gave examples for 3-D triangular meshes and point clouds multiscale representation.

**Acknowledgements** The authors would like to thank Jalal Fadili for advices and fruitful discussions.

### References

1. Alvarez, L., Guichard, F., Lions, P.L., Morel, J.M.: Axioms and fundamental equations of image processing. *Arch. Ration. Mech. Anal.* **123**, 199–257 (1993)
2. Arias, P., Caselles, V., Sapiro, G.: A variational framework for non-local image inpainting. In: *Energy Minimization Methods in Computer Vision and Pattern Recognition*. LNCS, vol. 5681, pp. 345–358 (2009)



3. Aujol, J.F., Aubert, G., Blanc-Féraud, L., Chambolle, A.: Image decomposition into a bounded variation component and an oscillating component. *J. Math. Imaging Vis.* **22**, 71–88 (2005)
4. Aujol, J.F., Chambolle, A.: Dual norms and image decomposition models. *Int. J. Comput. Vis.* **63**(1), 85–104 (2005)
5. Borwein, J.M., Lewis, A.S.: *Convex Analysis and Nonlinear Optimization: Theory and Examples*. Springer, Berlin (2006)
6. Bougleux, S., Elmoataz, A., Melkemi, M.: Discrete regularization on weighted graphs for image and mesh filtering. In: *Scale Space and Variational Methods in Computer Vision*. LNCS, vol. 4485, pp. 128–139 (2007)
7. Buades, A., Coll, B., Morel, J.M.: Image denoising methods. A new non-local principle. *SIAM Rev.* **52**(1), 113–147 (2010)
8. Candes, E.J., Romberg, J.K.: Signal Recovery from Random Projections, pp. 76–86. SPIE, Bellingham (2005)
9. Chambolle, A.: An algorithm for total variation minimization and applications. *J. Math. Imaging Vis.* **20**, 89–97 (2004)
10. Chambolle, A.: Total variation minimization and a class of binary mrf models. In: *Energy Minimization Methods in Computer Vision and Pattern Recognition*. LNCS, vol. 3757, pp. 136–152 (2005)
11. Chambolle, A., De Vore, R., Lee, N.Y., Lucier, B.: Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage. *IEEE Trans. Image Process.* **7**(3), 319–335 (1998)
12. Chambolle, A., Lions, P.L.: Image recovery via total variation minimization and related problems. *Numer. Math.* **76**(2), 167–188 (1997)
13. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vis.* **40**, 120–145 (2011)
14. Chan, R., Chan, T., Yip, A.: Numerical methods and applications in total variation image restoration. In: *Handbook of Mathematical Methods in Imaging*, pp. 1059–1094. Springer, Berlin (2011)
15. Chan, T., Osher, S., Shen, J.: The digital TV filter and nonlinear denoising. *IEEE Trans. Image Process.* **10**(2), 231–241 (2001)
16. Chan, T.F., Golub, G.H., Mulet, P.: A nonlinear primal-dual method for total variation-based image restoration. *SIAM J. Sci. Comput.* **20**(6), 1964–1977 (1999)
17. Combettes, P.L., Pesquet, J.C.: Proximal splitting methods in signal processing. In: *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, vol. 49, pp. 185–212. Springer, Berlin (2011)
18. Combettes, P.L., Wajs, V.R.: Signal recovery by proximal forward-backward splitting. *Multiscale Model. Simul.* **4**(4), 1168–1200 (2005)
19. Daubechies, I., Fornasier, M., Loris, I.: Accelerated projected gradient method for linear inverse problems with sparsity constraints. *J. Fourier Anal. Appl.* **14**, 764–792 (2008)
20. Elmoataz, A., Lézoray, O., Bougleux, S.: Nonlocal discrete regularization on weighted graphs: a framework for image and manifold processing. *IEEE Trans. Image Process.* **17**(7), 1047–1060 (2008)
21. Fadili, J., Peyré, G.: Total variation projection with first order schemes. *IEEE Trans. Image Process.* **20**(3), 657–669 (2011)
22. Getreuer, P.: Image zooming with contour stencils. In: *Proceedings of SPIE*, vol. 7246 (2009)
23. Gilboa, G., Osher, S.: Nonlocal operators with applications to image processing. *Multiscale Model. Simul.* **7**(3), 1005–1028 (2008)
24. Hidane, M., Lézoray, O., Ta, V.T., Elmoataz, A.: Nonlocal multiscale hierarchical decomposition on graphs. In: *Computer Vision ECCV 2010*. LNCS, vol. 6314, pp. 638–650 (2010)
25. Hiriart-Urruty, J.B., Lemaréchal, C.: *Fundamentals of Convex Analysis*. Springer, Berlin (2001)
26. Lézoray, O., Ta, V.T., Elmoataz, A.: Partial differences as tools for filtering data on graphs. *Pattern Recognit. Lett.* **31**(14), 2201–2213 (2010)
27. von Luxburg, U.: A tutorial on spectral clustering. *Stat. Comput.* **17**, 395–416 (2007)
28. Meyer, Y.: *Oscillating Patterns in Image Processing and Nonlinear Evolution Equations*. University Lecture Series. American Mathematical Society, Providence (2001)
29. Mohar, B.: The Laplacian spectrum of graphs. *Graph Theory Comb. Appl.* **2**(6), 871–898 (1991)
30. Moreau, J.J.: Fonctions convexes duales et points proximaux dans un espace hilbertien. *C. R. Acad. Sci. Paris Sér. A Math.* **255**, 2897–2899 (1962)
31. Moreau, J.J.: Proximité et dualité dans un espace hilbertien. *Bull. Soc. Math. Fr.* **93**, 273–299 (1965)
32. Osher, S., Burger, M., Goldfarb, D., Xu, J., Yin, W.: An iterative regularization method for total variation-based image restoration. *Multiscale Model. Simul.* **4**(2), 460–489 (2005)
33. Osher, S., Sole, A., Vese, L.: Image decomposition and restoration using total variation minimization and the  $H^{-1}$  norm. *Multiscale Model. Simul.* **1**(3), 349–370 (2003)
34. Pauly, M., Kobbelt, L.P., Gross, M.: Point-based multiscale surface representation. *ACM Trans. Graph.* **25**(2), 177–193 (2006)
35. Peyré, G., Bougleux, S., Cohen, L.: Non-local regularization of inverse problems. In: *Computer Vision ECCV 2008*. LNCS, vol. 5304, pp. 57–68 (2008)
36. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Physica D* **60**(1–4), 259–268 (1992)
37. Scherzer, O., Groetsch, C.: Inverse scale space theory for inverse problems. In: *Scale-Space and Morphology in Computer Vision*. LNCS, vol. 2106, pp. 317–325 (2006)
38. Strong, D.M., Aujol, J.F., Chan, T.F.: Scale recognition, regularization parameter selection, and Meyer’s  $g$  norm in total variation regularization. *Multiscale Model. Simul.* **5**(1), 273–303 (2006)
39. Tadmor, E., Nezzar, S., Vese, L.: A multiscale image representation using hierarchical  $(BV, L^2)$  decompositions. *Multiscale Model. Simul.* **2**(4), 554–579 (2004)
40. Tadmor, E., Nezzar, S., Vese, L.: Multiscale hierarchical decomposition of images with applications to deblurring, denoising and segmentation. *Commun. Math. Sci.* **6**(2), 281–307 (2008)
41. Vese, L.A., Osher, S.J.: Modeling textures with total variation minimization and oscillating patterns in image processing. *J. Sci. Comput.* **19**, 553–572 (2003)
42. Weickert, J.: *Anisotropic Diffusion in Image Processing*. ECI Series. Teubner, Leipzig (1998)
43. Yun, S., Woo, H.: Linearized proximal alternating minimization algorithm for motion deblurring by nonlocal regularization. *Pattern Recognit.* **44**(6), 1312–1326 (2011)
44. Zhou, D., Schölkopf, B.: Regularization on discrete spaces. In: *Pattern Recognition*. LNCS, vol. 3663, pp. 361–368 (2005)