Available online at www.sciencedirect.com
ScienceDirect
Computer Vision and Image Understanding 107 (2007) 38–55
Computer Vision and Image Understanding
www.elsevier.com/locate/cviu

Graph regularization for color image processing

Olivier Lezoray [a,*], Abderrahim Elmoataz [a], Sébastien Bougleux [b]

[a] LUSAC EA 2607, Vision and Image Analysis, IUT SRC, 120 Rue de l'exode, F-50000 Saint-Lô, France
[b] GREYC CNRS UMR 6072, Image Group, ENSICAEN, 6 Bd. Maréchal Juin, F-14050 Caen, France

Received 12 April 2006; accepted 16 November 2006
Available online 26 January 2007
Communicated by Rastislav Lukac

Abstract

Nowadays color image processing is an essential issue in computer vision. Variational formulations provide a framework for color image restoration, smoothing and segmentation problems. The solutions of variational models can be obtained by minimizing appropriate energy functions and this minimization is usually performed by continuous partial differential equations (PDEs). The problem is usually considered as a regularization matter which minimizes a smoothness plus a regularization term. In this paper, we propose a general discrete regularization framework defined on weighted graphs of arbitrary topologies which can be seen as a discrete analogue of classical regularization theory. The smoothness term of the regularization uses a discrete definition of the p-Laplace operator. With this formulation, we propose a family of fast and simple anisotropic linear and nonlinear filters which do not involve PDEs. The proposed approach can be useful to process color images for restoration, denoising and segmentation purposes.

Keywords: Color image processing; Regularization; Weighted graphs; Denoising; Smoothing; Segmentation

1. Introduction

Today color images are everywhere; indeed it is very easy for people to produce a huge amount of color images thanks to the advances in digital technologies. Color images allow us to perform a lot of complex tasks on a daily basis but to communicate and transmit information too. Just try to think about the amount of multimedia documents in which color plays an extremely important role; for instance in printing, photographs, digital television, computer displays, cinema movies. In addition, as with color multimedia documents are extremely easy to create, share, and reproduce, the processing of color images is becoming a crucial problem in the field of image processing [54]. Numerous approaches can be found to process color images and among those, variational models have been extremely successful in a wide variety of computer vision problems such as image restoration and image segmentation.

Variational formulations provide a framework that can handle such problems and provide algorithms for their solutions. Solutions of variational models can be obtained by minimizing appropriate energy functions, and this minimization is usually performed by designing continuous partial differential equations (PDEs). PDEs [4] are written in a continuous setting referring to images, and once the existence and the uniqueness of the solution have been proved, they are discretized in order to have a numerical solution. One typical use of PDEs is image restoration and a lot of authors have proposed color image enhancement or restoration with PDEs [17,59,64,67].

Let $f : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ be an initial image (for the sake of clarity we assume that Ω is a rectangle of $\mathbb{R}^2$) and let f^0 be an observed degraded (noisy) image of f so that:

$$f^0 = f + \eta \tag{1}$$

* Corresponding author. Fax: +33 0 233771166.
E-mail addresses: olivier.lezoray@unicaen.fr (O. Lezoray), elmoataz@chbg.unicaen.fr (A. Elmoataz), sebastien.bougleux@greyc.ensicaen.fr (S. Bougleux).
URL: http://www.info.unicaen.fr/~lezoray (O. Lezoray).

1077-3142/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2006.11.015

where η is additive noise of variance σ^2 induced by the recording system. To restore the degraded image f^0 is equivalent to reconstruct f from f^0. Therefore, we have to solve (1) for f which is an ill-posed problem [5]. This can be solved by variational approaches and finding f can be formulated as minimizing an energy functional of the following form:

$$E(f, f^0, \lambda) = E_{\text{smooth}}(f) + \lambda E_{\text{fidelity}}(f, f^0) \tag{2}$$

This idea was initially introduced by Tikhonov [60]. The term E_{smooth} measures the smoothness of the regularized image f (regularization term), E_{fidelity} measures the closeness of the regularized image f to the input image f^0 and $\lambda \geqslant 0$ is a constant scalar weight (the so-called Lagrange multiplier [4], which can be adaptive [30]) which controls the ratio between fidelity and regularization terms. Without loss of generality, we restrict ourselves to the following family of variational problems:

$$\min_{f \in \mathscr{H}} \left\{ E_p(f, f^0, \lambda) = \int_\Omega (\|\nabla f\|^p + \lambda \|f - f^0\|^2_{\mathscr{H}}) \right\}, p = 1 \text{ or } 2. \tag{3}$$

The smoothness term in E_p measures the p-smoothness of f on the image domain Ω. The fidelity term is the square norm of a given Hilbert space $\mathscr{H}$. The regularized image f which minimizes (3) can be solved using gradient-descent method and the Euler-Lagrange equation of E_p. The resulting PDEs are discretized and solved by numerical calculus algorithms. Over the last decade, PDE-based methods have been widely used for image regularization (see in [17] for a complete review) and can be classified into major categories according to whether they use or not the fidelity term. The most famous work using only the regularization term was introduced by Perona and Malik to perform nonlinear diffusion [48]. The most famous work using both regularization and fidelity terms is the total variation (TV) model proposed by Rudin, Osher and Fatemi [50] to perform edge preserving image denoising. In (3), when $p = 1$ and $\mathscr{H} = L^2$, E_{smooth} is the total variation energy (TV), and E_1 corresponds to the Rudin, Osher and Fatemi (ROF) model. It is a nonlinear method which reduces the image blurring in the denoising process. When $p = 2$, E_2 corresponds to the Tikhonov regularization.

Several approaches have extended variational models to vector valued images $f : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^m$ and more particularly to color images ($m = 3$) [53,8,32]. Sapiro and Ringach [53] regard an image as a parametrized two dimensional surface in $\mathbb{R}^m$ and use the difference between eigenvalues of the first fundamental form as a measure of edge strength. Blomgren and Chan [8] generalized TV regularization to vector data as the Euclidean norm of the vector of total variations. Keren and Gotlib [31] propose to augment standard regularization by adding a correlation term which measures color smoothness. This correlation term relies on geometric measures on the angle between adjacent pixels. Kimmel, Malladi and Sochen [32] regard an image as a surface in $\mathbb{R}^{m+2}$ and propose an area minimizing flow [32] (called the Beltrami flow) to perform edge preserving denoising [11]. Smolka and Lukac [57,56] propose to combine forward and backward anisotropic diffusion with incorporated cooling process, which makes it possible for them to remove image noise while enhancing image edges. Levine, Stanich and Chen [35] propose to combine isotropic and anisotropic TV-based diffusion depending on the local image information. This approach is effective to remove noise and retains sharp edges while avoiding the staircasing effect. Finally, Tschumperlé [65] has recently proposed a generic constrained regularization formalism capable of anisotropically smoothing color images with PDEs while preserving natural curvature constraints.

Such variational models have been extensively studied in theory and in practice (see in [17] for a recent review). However, they have several drawbacks. First, they are defined on a continuous setting and require a discretization. It might be more suitable to have a direct discrete formulation for images since an image is a discrete grid. Moreover, a discrete formulation of regularization is quite interesting since a lot of data can be represented by a graph of arbitrary topology and those data often need to be regularized too (point clouds for instance). Second, PDEs have a very local behavior and even with efficient numerical schemes, they can be very time consuming algorithms. Thirdly, the numerical schemes needed for the discretization of PDEs require that the data to regularize have a grid topology. Therefore, PDEs are not well suited to perform the regularization of data organized in a totally arbitrary topology: with data represented by graphs of arbitrary topology, the continuous regularization is not directly applicable.

Discrete methods might therefore be more suitable than PDEs in some cases. For instance, vector median filters (VMF), introduced by Astola [3], are a widely used discrete approach to vector-valued signal and color image denoising [39,36,44,49,61]. The definition of the multichannel median is a direct extension of the ordinary single-channel median definition [3]. The output of these filters is defined as the lowest ranked vector according to a specific ordering technique, typically the reduced ordering based on aggregated distances [55,38]. VMF are robust estimators due to the employed minimization concept [38]: the VMF output is the sample that minimizes the aggregated distance to the other samples. For further details, one can refer to the most recent overviews of vector processing for color image filtering and enhancement in [37,38]. Caselles et al have shown [12] that for the scalar case, there is a close relationship between median filtering, inf-sup morphological operations, and partial differential equations. For the color case, Caselles considers a lexicographic ordering of color vectors and obtains the same connections.

In this paper, we transcribe the continuous regularization framework of PDEs presented above into a discrete domain defined on weighted graphs of arbitrary topologies. The discrete minimization problem analogue to problem (3) is formalized by:

$$\min_{f\in\mathscr{H}(V)}\left\{E_p(f,f^0,\lambda)=\sum_{v\in V}\|\nabla f(v)\|^p+\lambda\|f-f^0\|^2_{\mathscr{H}(V)}\right\},\quad p=1 \text{ or } 2, \tag{4}$$

where V is a set of vertices, $\mathscr{H}(V)$ is a given Hilbert space defined on V and $\|\nabla f(v)\|$ is the norm of a given discrete gradient.

The main advantage is that we do not have to solve a PDE and we can apply this method on any data assuming they can be represented by a graph of the arbitrary topology. A formalism is therefore proposed to perform regularization on graphs structures. This regularization is expressed directly in a discrete setting. Therefore, unlike the PDE case, no discretization problems have to be overcomed. Discrete regularization on unweighted graphs has already been proposed for images represented as grid graphs such as the TV digital filter [16]. In [16] Chan, Osher and Shen show that iterating such an unweighted filter for $p=1$ leads to the minimization of the total variation. One can find several papers treating of the exact minimization of (4) in the case of $p=1$ [14,19,20,24,25]. For instance, the methods introduced by Chambolle [14] and Darbon [20] can be considered as discrete regularization on unweighted graphs which yield to Markov random fields where only binary variables are involved in the minimization. Discrete regularization on graphs has also been used for semi-supervised data classification [6,69,70]. Let's use those works as our bases: we propose a regularization framework on weighted graphs of arbitrary topologies which defines a family of simple and fast anisotropic linear and nonlinear filters. The proposed framework can be applied to a wide range of color image processing applications.

The paper is organized as follows. In Section 2, after having recalled basic definitions on weighted graphs and quoted some graph structures for the case of color image processing, we present differential geometry on graphs which is similar to the one proposed by Benssoussan [7] and Zhou [69,70]. In Section 3, we present a general framework for discrete regularization on graphs [9]. In Section 4, we show how regularization on graphs can be applied to color image restoration, denoising, and segmentation. Finally, main ideas, results, and future objectives are summarized in Section 5.

2. Differential geometry on graphs

Differential geometry on graphs is a discretization of classical differential geometry in the continuous case.

2.1. Preliminaries on graphs

A graph is a structure used to describe a set of objects and the pairwise relations between those objects (links between objects). The objects are called vertices (or nodes) and a link between two objects is called an edge. We provide some basic definitions on graph theory (further details can be found in [21]). A graph $\mathscr{G}$ is a couple $\mathscr{G}=(V,E)$ where V is a finite set of vertices and E is a subset $E\subseteq V\times V$. The elements of V are the vertices of the graph and the elements of E are the edges of the graph. Two vertices u and v are adjacent vertices if the edge $(u,v)\in E$. In the rest of this paper, we only consider simple graphs which are always assumed to be connected, undirected with no self loops (see in [9,21,70] for details on these notions).

A graph as defined above is said to be weighted if it is associated with a weight function $w:E\rightarrow\mathbb{R}^+$ satisfying $w(u,v)>0$ for $(u,v)\in E$, $w(u,v)=0$ for $(u,v)\notin E$ and $w(u,v)=w(v,u)$ for all edges in E since we consider undirected graphs. The degree function of a vertex $\delta:V\rightarrow\mathbb{R}^+$ is defined to be $\delta(v)=\sum_{u\sim v}w(u,v)$ where $u\sim v$ denotes all vertices u connected to v by the edges $(u,v)\in E$.

Now we can define the space of functions on graphs. Let $\mathscr{H}(V)$ denote the Hilbert space of real-valued functions on vertices, in which each $f:V\rightarrow\mathbb{R}^+$ assigns a real value $f(v)$ to each vertex v. The function space $\mathscr{H}(V)$ can be endowed with the usual inner product $\langle f,g\rangle_{\mathscr{H}(V)}=\sum_{v\in V}f(v)g(v)$ where f and g denote two functions in $\mathscr{H}(V)$. A function f in $\mathscr{H}(V)$ can be thought as a column vector in $\mathbb{R}^{|V|}$. The norm of a function f induced from the inner product is $\|f\|=\sqrt{\langle f,f\rangle}$. Similarly, one can define $\mathscr{H}(E)$ as the space of real-valued functions on edges, in which each $h:E\rightarrow\mathbb{R}^+$ assigns a real value to each edge e. This function space can be endowed with the usual inner product $\langle h,l\rangle_{\mathscr{H}(E)}=\sum_{(u,v)\in E}h(u,v)l(u,v)$ where h and l denote two functions in $\mathscr{H}(E)$ defined on $E\rightarrow\mathbb{R}^+$.

Any general discrete domain can be modeled by a graph. Therefore, graphs are very well adapted to computer vision problems. We can quote many examples where graphs can be used to represent image data:

- Grid graphs [16] which are natural structures corresponding to the definition of digital images: vertices represent pixels and edges represent pixel adjacency relationship.
- Region Adjacency Graphs (RAG) [62] which provide very useful and common ways of describing the structure of a picture: vertices represent regions and edges represent region adjacency relationship.
- Color adjacency graphs [43] which represent color clusters in the chromatic histogram: vertices represent a set of colors belonging to a same cluster and edges represent the adjacency of color clusters.
- Nearest neighbor (k-NN) graphs [27] are proximity graphs which (for a set of points in a metric space) associate to every vertex a single vector. Edges correspond to links to neighbor vectors. Every vertex has exactly k edges to the k nearest vectors according to a given distance function.

Therefore, graph structures are extremely useful and occur naturally while processing digital images. A graph can be associated to any color image representation according to the definition of a distance or a similarity. Processing color images comes to processing graphs, the

vertices models and edges weights of which depend on colorimetric properties of the image. The topology of the graph depends on the problem under consideration: grid graphs for image restoration, region adjacency graphs for image segmentation, color adjacency graphs for image clustering, and nearest neighbor graphs for image categorization. In this paper, discrete regularization for graphs of the arbitrary topology is proposed. Since the topology can be arbitrary, the regularization can be adapted to each computer vision problem by defining specific functions $f \in \mathscr{H}(V)$ on vertices and functions $h \in \mathscr{H}(E)$ on edges.

2.2. Gradient and divergence operators

The difference operator $d : \mathscr{H}(V) \to \mathscr{H}(E)$ on $\mathscr{G} = (V, E)$ of a function $f \in \mathscr{H}(V)$ on an edge (u,v) linking two vertices u and v is defined for all $(u,v) \in E$ as

$$(\mathrm{d}f)(u,v) = (\mathrm{d}f)_{uv} = \sqrt{w(u,v)}(f(v) - f(u)) \tag{5}$$

In the lattice case, the difference degrades to $(f(v) - f(u))$ which is the standard difference definition in numerical analysis.

The edge derivative $\frac{\partial f}{\partial e}|_v : \mathscr{H}(V) \to \mathbb{R}^+$ of a function f at vertex v along the edge $e = (u,v)$ is defined as

$$\left.\frac{\partial f}{\partial e}\right|_v = (\mathrm{d}f)(u,v)$$

It is also denoted by $\partial_v f(u)$. This definition is consistent with the continuous definition of the derivative of a function, e.g., if $f(v) = f(u)$ then $\frac{\partial f}{\partial e}|_v = 0$. Clearly one has $\frac{\partial f}{\partial e}|_v = -\frac{\partial f}{\partial e}|_u$ $(\partial_v f(u) = \partial_u f(v))$ and $\partial_u f(u) = 0$.

Given a function $f \in \mathscr{H}(V)$ and a vertex v, the gradient of f at vertex v is the vector operator defined as $\nabla : V \to \mathbb{R}^N$ by

$$\nabla f(v) = \nabla_v f = (\partial_v f(u) : (u,v) \in E, u \sim v)^T \tag{6}$$

where $u \sim v$ means that u and v are connected by an edge. Then, the norm of the graph gradient ∇f at vertex v or the local variation of f at vertex v is defined as $\|\nabla\| : \mathbb{R}^N \to \mathbb{R}^+$

$$\begin{aligned}\|\nabla_v f\| &= \sqrt{\sum_{u\sim v}(\partial_v f(u))^2} \\ &= \sqrt{\sum_{u\sim v} w(u,v)(f(v) - f(u))^2}\end{aligned} \tag{7}$$

The norm of the gradient measures the roughness of a function around a vertex. Let $\mathscr{R}$ denote a functional on $\mathscr{H}(V)$, for any $p \in [1,+\infty)$ which is defined by $\mathscr{R}_p(f) = \sum_{v\in V}\|\nabla_v f\|^p$. This functional $\mathscr{R}_p$ can be seen as the measure of the smoothness of f since it is the sum of the local variations at each vertex.

The graph divergence operator is an operator $div : \mathscr{H}(E) \to \mathscr{H}(V)$ which satisfies

$$\langle \mathrm{d}f, h\rangle_{\mathscr{H}(E)} = \langle f, -\mathrm{div}(h)\rangle_{\mathscr{H}(V)}$$

with $f \in \mathscr{H}(V)$ and $h \in \mathscr{H}(E)$. This operator $-\mathrm{div}$ is therefore the adjoint operator d^* of the difference operator d. From the definition of the inner products in $\mathscr{H}(V)$ and $\mathscr{H}(E)$ and Eq. (5), one can prove that the graph divergence of a function $h \in \mathscr{H}(E)$ at a vertex v can be expressed as

$$\begin{aligned}(d^*h)(v) &= (-\mathrm{div}(h))(v) \\ &= \sum_{u\sim v}\sqrt{w(v,u)}(h(u,v) - h(v,u))\end{aligned} \tag{8}$$

The divergence operator measures the net outflow of function $h \in \mathscr{H}(E)$ at each vertex v of $\mathscr{G}$.

2.3. p-Laplace operator

The classical graph Laplacian can be considered as a discrete analogue of the Laplace–Beltrami operator on Riemannian manifolds. The graph Laplacian is an operator $\Delta : \mathscr{H}(V) \to \mathscr{H}(V)$ defined as $\Delta f = -\mathrm{div}(\mathrm{d}f) = d^*(\mathrm{d}f)$. The graph Laplacian is a linear operator because both the gradient and divergence operator are linear. Furthermore, it is self-adjoint:

$$\langle \Delta f, g\rangle_{\mathscr{H}(V)} = \langle \mathrm{d}f, \mathrm{d}g\rangle_{\mathscr{H}(E)} = \langle f, \Delta g\rangle_{\mathscr{H}(V)}$$

and positive semi-definite:

$$\langle \Delta f, f\rangle_{\mathscr{H}(V)} = \langle \mathrm{d}f, \mathrm{d}f\rangle_{\mathscr{H}(E)} = \mathscr{R}_p(f) \geqslant 0$$

which implies that

$$\Delta f = \frac{\partial \mathscr{R}_p(f)}{\partial f} \tag{9}$$

The classical graph curvature is a nonlinear operator $\kappa : \mathscr{H}(V) \to \mathscr{H}(V)$ defined as $\kappa f = -div(\frac{df}{\|\nabla f\|}) = d^*(\frac{df}{\|\nabla f\|})$.

We can generalize the graph Laplacian and curvature to an operator which can be considered as a discrete analogue of the p-Laplacian in the continuous case [63]. The graph p-Laplacian is an operator $\Delta_p : \mathscr{H}(V) \to \mathscr{H}(V)$ with $p \in [1,+\infty)$ defined as

$$\Delta_p f = -\mathrm{div}\left(\|\nabla f\|^{p-2}\mathrm{d}f\right) = d^*\left(\|\nabla f\|^{p-2}\mathrm{d}f\right) \tag{10}$$

Clearly, one has $\Delta_1 = \kappa$ and $\Delta_2 = \Delta$. Substituting (5) and (8) into the definition (10) of $\Delta_p f$, we obtain:

$$(\Delta_p f)(v) = \sum_{u\sim v}\gamma(u,v)(f(v) - f(u)) \tag{11}$$

where $\gamma(u,v)$ is the function defined by

$$\gamma(u,v) = w(u,v)\left(\|\nabla f(v)\|^{p-2} + \|\nabla f(u)\|^{p-2}\right) \tag{12}$$

which generalizes the classical graph Laplacian and curvature. In general Δ_p is nonlinear (except in the case of $p = 2$) and it is positive semi-definite.

Practically, to avoid having a zero denominator to compute the curvature, the graph gradient $\|\nabla_v f\|^{p-2}$ has to be replaced by its regularized version $\|\nabla_v f\|_\varepsilon^{p-2} = (\varepsilon + \|\nabla_v f\|^{p-2})$ where $\varepsilon > 0$ is a small positive parameter called the regularization parameter [16], fixed to 10^{-5} in our experimentation. As stated in [16], the performance of the regularization is insensitive to ε as long as it is kept

small. The use of ε is needed to reduce degeneracies in flat regions where $\|\nabla_v f\| \approx 0$. This is a commonly used technique [17] and this leads to the minimization of the following energy (note the presence of ε):

$$\min_{f\in\mathscr{H}(V)}\left\{E_p^{\varepsilon}(f,f^0,\lambda)=\sum_{v\in V}\|\nabla f(v)\|_{\varepsilon}^{p}+\lambda\|f-f^0\|^2_{\mathscr{H}(V)}\right\},\quad p=1 \text{ or } 2, \tag{13}$$

The obtained solution is therefore an approximation of the TV solution. For the sake of clarity, we keep the notation $\|\nabla_v f\|^{p-2}$ instead of $\|\nabla_v f\|_{\varepsilon}^{p-2}$ in the rest of the paper. As pointed in [18], when ε is close to 0, the domain of convergence is small. To overcome this, some works consider an evolving value for ε (e.g. [18,23]). In this study neither do we consider this issue, nor the kind of approximation that is introduced by ε. One way to avoid the use of ε is to consider the dual formulation (see in [15] and the references therein).

3. Regularization on graphs

In this section, we propose a general framework to regularization on graphs. We recall that a graph is considered as a discrete representation of data. Therefore, the proposed framework is devoted to the regularization of data living on graphs. We do not make any assumptions on the topology of the graph and on the dimensionality of the data: they can be arbitrary. For the sake of clarity we present this framework for scalar images which are, in this case, represented as a grid graph, but the principle is the same for color images. In this paper, color images are considered as red-green-blue images. A color image f is therefore composed of three channels f_1,f_2,f_3 where the term f_i denotes red (for $i=1$), green (for $i=2$) and blue (for $i=3$). The regularization framework is applied in a component-wise manner (each channel is processed independently). Section 3.4 will provide further details on how we manage the case of color image regularization to have a coupling between the independent channel regularizations.

3.1. General framework

Given a graph $\mathscr{G}=(V,E)$ associated with a weighting function $w: E\to\mathbb{R}^+$, we want to perform the p-Laplace regularization of a function $f^0\in\mathscr{H}(V)$ using the p-Laplace operator. It consists in seeking for a function f^* which is smooth and simultaneously close to the function f^0. This comes to consider general variational problems on graphs. The goal is, given a function $f^0\in\mathscr{H}(V)$, to find a function $f^*\in\mathscr{H}(V)$ which is not only smooth enough on $\mathscr{G}$ but also close enough to the given function f. This can be formalized by minimizing a weighted sum of two energy terms:

$$f^*=\min_{f\in\mathscr{H}(V)}\left\{E_p=\mathscr{R}_p(f)+\lambda\|f-f^0\|^2=\sum_{v\in V}\|\nabla_v f\|^p+\lambda\sum_{v\in V}\|f-f^0\|^2\right\} \tag{14}$$

The first term is the smoothness term or regularizer, which requires f not to change too much between closely related objects. The second term is the fitting term, which says that f should not be far away from f^0. The parameter $\lambda\geqslant 0$ is a fidelity parameter called the Lagrange multiplier which specifies the trade-off between the two competing terms. Both terms of the energy E_p are strictly convex functions of f [16,13], therefore, by standard arguments in convex analysis, this optimization problem has a unique solution for $p=1$ or $p=2$ which satisfies:

$$\left.\frac{\partial E_p}{\partial f}\right|_v=0,\quad \forall v\in V \tag{15}$$

Using the property (9) of the p-Laplace operator to compute the derivative of the first term in E_p, Eq. (15) can be rewritten as follows:

$$(\Delta_p f^*)(v)+2\lambda(f^*(v)-f^0(v))=0,\quad \forall v\in V \tag{16}$$

The solution f of (14) is also the solution of (16). Substituting the expression of the p-Laplace operator into (16), we obtain:

$$\left(2\lambda+\sum_{u\sim v}\gamma_{uv}\right)f^*(v)-\sum_{u\sim v}\gamma(u,v)f^*(u)=2\lambda f^0(v),\forall v\in V. \tag{17}$$

Among the existing methods [10,28] which can be used to solve (17), we use the Gauss–Jacobi iterative algorithm [26,33]. Let t be the iteration step, and let $f^{(t)}$ be the solution of (17) at the step t. The initial function $f^{(0)}$ can be initialized to f^0. The corresponding linearized Gauss–Jacobi iteration is given by:

$$f^{(t+1)}(v)=\frac{2\lambda}{2\lambda+\sum_{u\sim v}\gamma^{(t)}(u,v)}f^0(v)+\frac{\sum_{u\sim v}\gamma^{(t)}(u,v)f^{(t)}(u)}{2\lambda+\sum_{u\sim v}\gamma^{(t)}(u,v)},\forall v\in V, \tag{18}$$

where $\gamma^{(t)}$ is the function $\gamma(u,v)$ at the step t. One can note that the value of $f(v)$ for a given iteration $(t+1)$ depends on two quantities: the original value of f at v ($f^0(v)$) and the values for iteration t in the neighborhood of v. Coefficients are associated to those quantities which depend on the sum of weighted local variations.

The global algorithm to compute the solution of (14) has the following inputs: the graph $\mathscr{G}=(V,E)$ and its weight function w, the parameter λ, the smoothness degree p, the initial function $f^0=f^{(0)}$, and the number of iterations i. First, the weights are initialized for each edge of E. Then, for each step $t=0,\ldots,i$ and:

(a) for each $v \in V$, the term $f^{(t+1)}(v)$ is computed according to (18),
(b) for each edge $(u,v) \in E$, the function $\gamma^{(t+1)}(u,v)$ is updated according to (12).

The case of an arbitrary p is not considered in this paper and we reduce ourselves to $p = 1$ and $p = 2$. The proposed filters are nonlinear in the case of $p = 1$ and linear in the case of $p = 2$. The γ coefficients are updated for each iteration only in the case of $p = 1$. In the case of $p = 2$ and if $\lambda \neq 0$, the optimal solution can be obtained by iterating (a) until convergence, e.g. until $|f^{(t+1)} - f^{(t)}| < \varepsilon$, with $\varepsilon \to 0$.

3.2. Regularization Algorithm for $p = 2$

When $p = 2$, it follows from (16) that the solution of (14) is based on the Laplacian operator and satisfies

$$\Delta f^* + 2\lambda(f^* - f^0) = 0 \tag{19}$$

This can be considered as a discrete analogue of the Euler–Lagrange equation. In this case, the filter presented in Section 3.1 is linear and for $\lambda \neq 0$ converges to the solution of (14). The iterative scheme used to solve this is expressed by:

$$\begin{cases} f^0 = f \\ f^{(t+1)}(v) = \frac{2\lambda}{2\lambda + \sum_{u \sim v} w(u,v)} f^0(v) + \frac{\sum_{u \sim v} w(u,v) f^t(u)}{2\lambda + \sum_{u \sim v} w(u,v)}, \forall v \in V, \end{cases} \tag{20}$$

where t indicates the iteration step. We define the function $c : V \to \mathbb{R}^+$ by $c(v) = \frac{1}{2\lambda + \sum_{u \sim v} w(u,v)}$ and the iterative scheme is then expressed by:

$$f^{(t+1)}(v) = 2\lambda c(v) f^0(v) + c(v) \sum_{u \sim v} w(u,v) f^t(u), \forall v \in V, \tag{21}$$

For each iteration, the new value of f at v, only depends on a weighted average of the existing values in the direct neighborhood of v, and the original value at v. The obtained filtering is a low-pass filter, the behavior of which is adaptable to a given image according to the values of $w(u,v)$ computed from f^0. When $\lambda = 0$ and $w(u,v) = 1 \forall (u,v) \in E$, then it is the discrete analogue of the diffusion flow on Riemannian manifolds [53].

Fig. 1(a) presents an initial image from the Berkeley Segmentation Dataset and Benchmark (BSDB) [42]. The behavior of the filter on this image is illustrated by Fig. 2(a–d) for different numbers of iterations and values of λ. The image is represented by a grid graph. To demonstrate the differences between the considered solutions, an

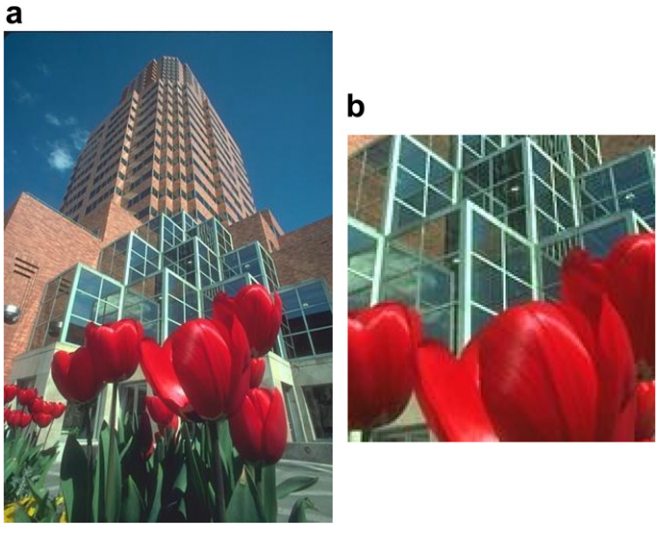

Fig. 1. Test image. (a) Initial test image. (b) Cropped and zoomed part of (a).

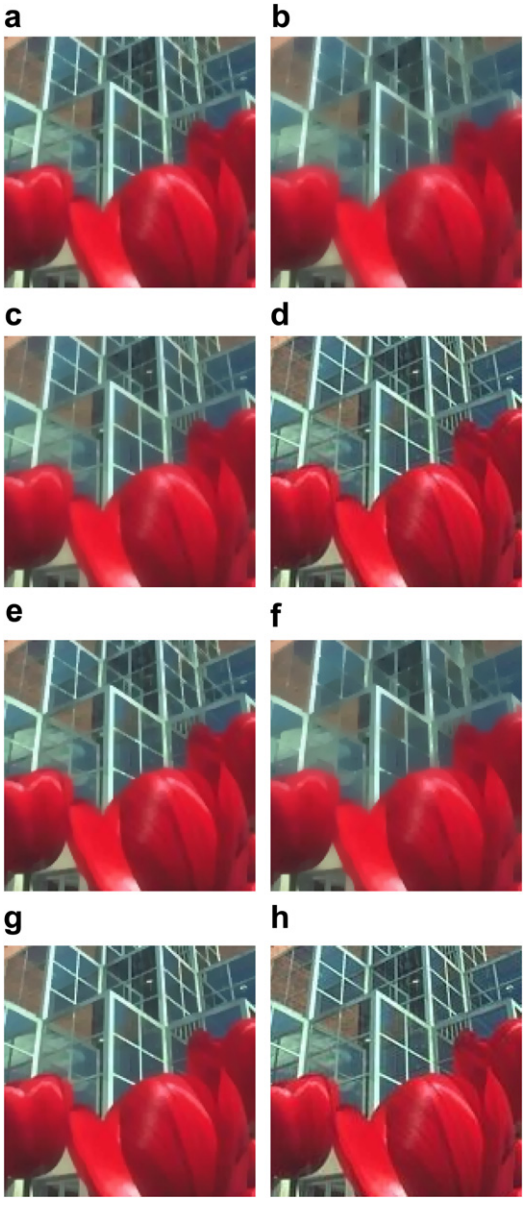

Fig. 2. Zoomed results of regularization related to Fig. 1 for $p = 2$ and $p = 1$. (a) $t = 10$, $\lambda = 0$, $p = 2$. (b) $t = 50$, $\lambda = 0$, $p = 2$. (c) $t = 50$, $\lambda = 0.01$, $p = 2$. (d) $t = 50$, $\lambda = 0.25$, $p = 2$. (e) $t = 10$, $\lambda = 0$, $p = 1$. (f) $t = 50$, $\lambda = 0$, $p = 1$. (g) $t = 50$, $\lambda = 0.01$, $p = 1$. (h) $t = 50$, $\lambda = 0.25$, $p = 1$.

area of Fig. 1(a) with significant structural content has been cropped and zoomed (Fig. 1(b)). Fig. 2 presents results for this cropped area. One can see the influence of the λ term which levels the influence of the data fidelity term of the energy in the regularization. For $\lambda = 0$ the image becomes smooth, and one recovers a discrete analogue of the diffusion flow (Fig. 2(a) and (b)). The data fidelity term enables to level the smoothness of the result because of the competition between the two terms of the energy (Fig. 2(c) and (d)). The computational complexity of the proposed filters in the case of $p = 2$ is $O(n)$. Indeed, for each iteration one has to compute the weights $w(u,v)$ whose computational complexity is $O(n)$ and one has to apply Eq. (20) whose computational complexity is $O(n)$.

3.3. Regularization algorithm for p = 1

When $p = 1$, it follows from (16) that the solution of (14) is based on the nonlinear curvature operator κ and satisfies

$$\kappa f^* + 2\lambda(f^* - f^0) = 0 \tag{22}$$

The iterative scheme used to solve this problem is expressed as follows:

$$\begin{cases} f^0 = f \\ \gamma^{(t+1)}(u,v) = w(u,v)\left(\frac{1}{\varepsilon+\|\nabla f^{(t+1)}(v)\|}+\frac{1}{\varepsilon+\|\nabla f^{(t+1)}(u)\|}\right), \forall (u,v)\in E, \\ f^{(t+1)}(v) = \frac{2\lambda}{2\lambda+\sum_{u\sim v}\gamma^t(u,v)} f^0(v) + \frac{\sum_{u\sim v}\gamma^t(u,v) f^t(u)}{2\lambda+\sum_{u\sim v}\gamma^t(u,v)}, \forall v \in V, \end{cases} \tag{23}$$

Similarly as for $p = 2$, we can define the function $c : V \to \mathbb{R}^+$ by $c^t(v) = \frac{1}{2\lambda+\sum_{u\sim v}\gamma^t(u,v)}$ and the iterative scheme used to compute new values of f^t is then expressed by:

$$f^{(t+1)}(v) = 2\lambda c^t(v) f^0(v) + c^t(v)\sum_{u\sim v}\gamma^t(u,v) f^t(u), \forall v \in V, \tag{24}$$

Compared with the iterative algorithm in the case of $p = 2$, it is also a low pass filter but the coefficients are adaptively updated for each iteration in addition of updating the function f. It is worth to note the connection between the proposed filter and the TV digital filter [16] (TV+L^2). Indeed,

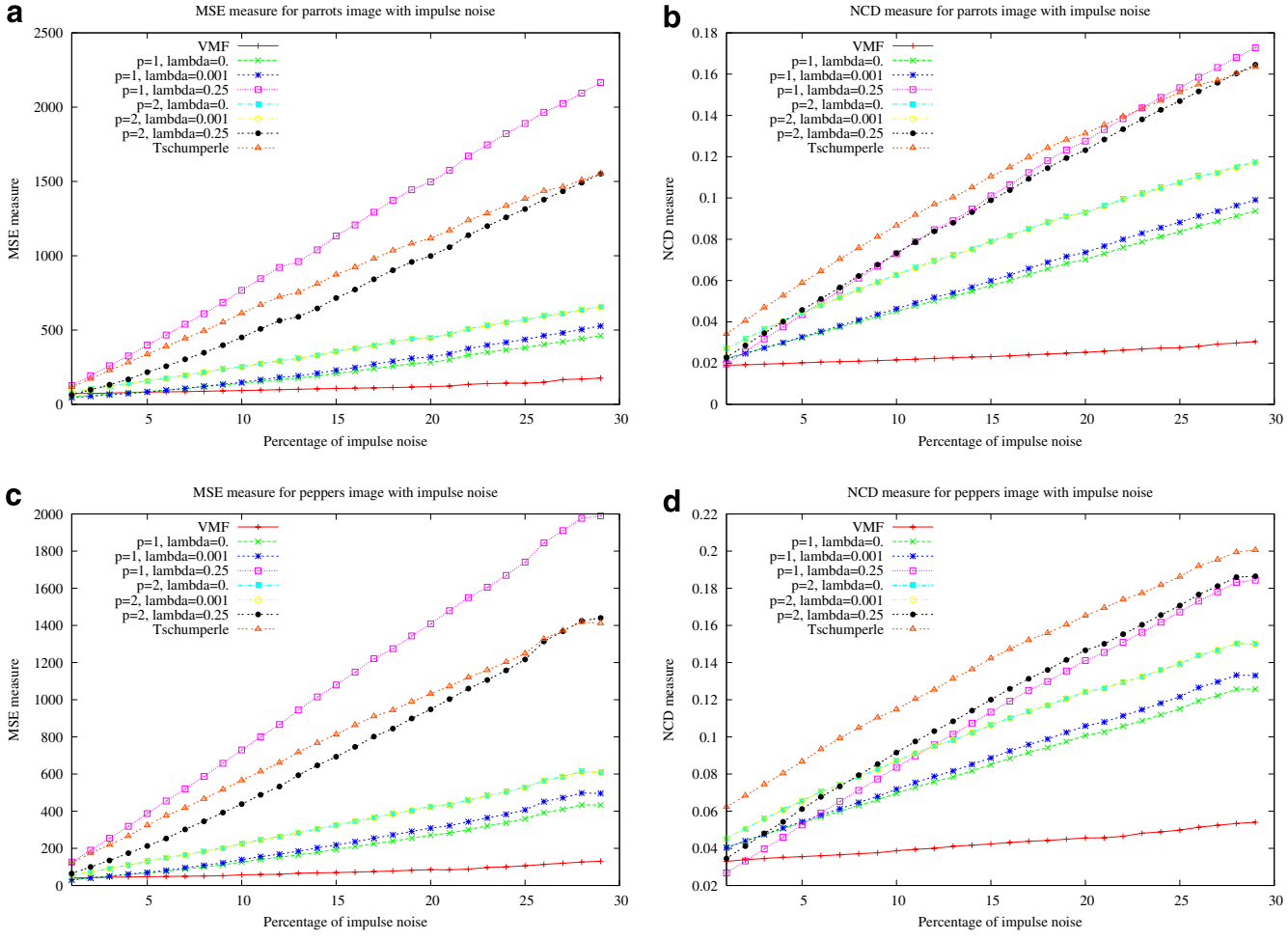

Fig. 3. Comparison of filters efficiency in terms of MSE (first column) and NCD (second column) on parrots (first row) and peppers (second row) images for different amounts of impulse noise.

with $p = 1$, if $\forall (u,v) \in E, w(u,v) = 1$ i.e. the edges have all the same weights, one recovers exactly the same iterative filtering performed on a regular grid represented by a graph [16].

The behavior of the filter is illustrated by Fig. 2(e)–(h) for different numbers of iterations and values of λ. In order to compare the results with $p = 2$, the same initial image (Fig. 1(a)) and parameters are considered. In Fig. 2(e)–(h), we observe that the smoothing process for the regularization with $p = 1$ is more appropriate to preserve sharp features during a smoothing process. Indeed, compared with the solutions provided by Fig. 2(a)–(d), less smoothed edges are obtained. The computational complexity of the proposed filters in the case of $p = 1$ is $O(n^2)$. As for the case of $p = 2$, one has to compute the weights $w(u,v)$ the computational complexity of which is $O(n)$. However, when we apply the iterative scheme in Eq. (23), that has a computational complexity of $O(n^2)$. On the one hand, the computation of new values for the vertices has a computational complexity of $O(n)$. On the other hand, the coefficients $\gamma(u,v)$ are updated for each iteration and the computational complexity of this update is $O(n^2)$.

3.4. The case of color images

Let $f : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^3$ be a color image of pixels. As previously mentioned, we consider red–green–blue images. To perform graph regularization on a color image, a graph structure has to be associated to the color image. As defined in Section 2.1, we can associate a vector to each vertex of a graph. For the case of color images, we define $f \in \mathscr{H}(V), f : V \to \mathbb{R}^3$ which associates a red-green-blue color to each vertex. Similarly, we can associate a weight to each edge of a graph, $w \in \mathscr{H}(E), w : E \to \mathbb{R}^3$. The choice of the weight function depends on the application of the regularization process, we will come back on its definition in Section 4.1. To perform the regularization on a grid representing a color image, the iteration of Eq. (18) is considered. Since a color image is composed of three channels, three independent regularization processes are considered (each channel is processed independently). This comes to have three independent iterative regularization schemes (one per channel f_i with $i \in [1,3]$):

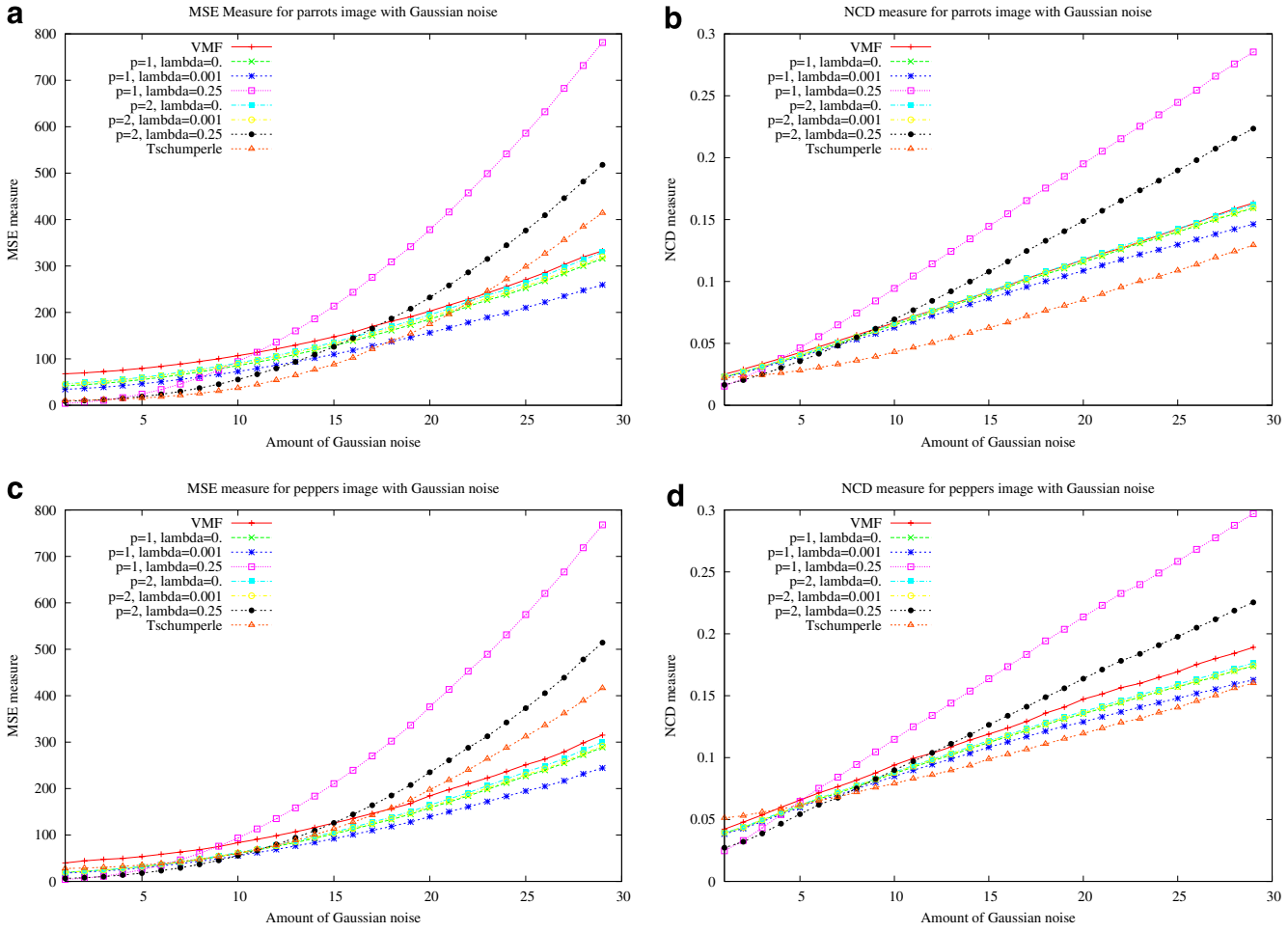

Fig. 4. Comparison of filters efficiency in terms of MSE (first column) and NCD (second column) on parrots (first row) and peppers (second row) images for different amounts of Gaussian noise.

$$f_i^{(t+1)}(v) = \frac{2\lambda}{2\lambda + \sum_{u\sim v}\gamma_i^{(t)}(u,v)} f_i^0(v) + \frac{\sum_{u\sim v}\gamma_i^{(t)}(u,v) f_i^{(t)}(u)}{2\lambda + \sum_{u\sim v}\gamma_i^{(t)}(u,v)} \tag{25}$$

The p-Laplace operator is different for each channel and one has

$$\gamma_i(u,v) = w(u,v)\left(\|\nabla f_i(v)\|^{p-2} + \|\nabla f_i(u)\|^{p-2}\right) \tag{26}$$

for the ith channel. Applying the regularization in a component-wise manner is interesting to develop a computational efficient solution. However, component-wise processing of color images can have serious drawbacks contrary to vector processing solutions. For instance, component-wise median filters can introduce false colors [38]. To overcome this limitation, a regularization process acting on vector-valued images needs to be driven by equivalent geometric attributes, taking the coupling between vector channels into account [64,67]. Therefore, component-wise regularization does not have to use different local geometries but a vector one. In the case of $p = 2$ the p-Laplace operator is the same for the three channels, but in the case of $p = 1$ it is different. As we have just mentioned, this is not interesting for color images. Indeed, in this case, the three regularization processes can be totally independent if $w(u,v)$ does not incorporate any inter-channel information resulting in no coupling between the three regularizations. To overcome this limitation and in order to take into account the inner correlation aspect of color vector data, the p-Laplace operator is considered as being the same for the three channel regularizations (channel coupling) and is defined by

$$\gamma(u,v) = w(u,v)\left(\|\nabla f(v)\|_{3D}^{p-2} + \|\nabla f(u)\|_{3D}^{p-2}\right) \tag{27}$$

The term $\|\nabla f(v)\|_{3D}^{p-2}$ is only used in the case of $p = 1$. The norm of a color vector is defined as its multi-dimensional Euclidean norm and remains the same whatever the color channel under consideration is. This is required to have a global vector geometry:

$$\|\nabla f(v)\|_{3D} = \sqrt{\|\nabla f_1(v)\|^2 + \|\nabla f_2(v)\|^2 + \|\nabla f_3(v)\|^2} \tag{28}$$

The proposed regularization applies therefore to each channel of a color image with a weighting of edges and a vector gradient norm acting both as a coupling between channels to avoid the drawbacks of applying the regularization in a component-wise manner. The above formulations have been applied in the processing of the images of Fig. 2.

4. Applications

In this Section, we show how the previously proposed regularization can be used to perform color image restoration, denoising, smoothing or segmentation. For $p = 1$ or $p = 2$, since we consider color images, three iterative schemes ((23) or (21)) are simultaneously considered (one for each color channel).

4.1. Color image restoration/denoising

In the context of image restoration and denoising, the similarity between two neighbor vertices u and v (two adjacent pixels) can be estimated by the following weight function (but any other formulation can be considered, depending on the intended application):

$$w(u,v) = \frac{1}{\varepsilon + \|f(u) - f(v)\|}, \text{ with } \varepsilon \to 0 \tag{29}$$

where the distance norm between two vertices (two colors) is defined as

$$\|f(u) - f(v)\| = \sqrt{\sum_{i=1}^{3} \|f_i(u) - f_i(v)\|^2} \tag{30}$$

with $i \in [1,3]$ denoting the number of the color channel. It is worth to note that any weighting w quantifying the similarity between adjacent vertices (i.e. pixels) can be considered. In this paper, Eq. (29) is considered for its simplicity but other similarity measures are also of interest [41]. To evaluate the abilities of the proposed family of filters, several experimentations have been carried out. We consider three different filters: the classical vector median filter [3], the PDE-based color regularization of Tschumperlé [65] (obtained from the author web page[1].) and our proposed family of filters. For the latter, we consider grid graphs (one vertex per pixel) with the city-block distance (4-neighborhood connectivity). The parameters p and λ are the only parameters of our filters and provide control over the performance of the family of filters we propose, therefore different values of these parameters are considered. For the PDE-based color regularization of Tschumperlé, default parameters are used (more than ten parameters are used in this method and we cannot test all of them). To have an equivalent comparison of the proposed filters, only one iteration of these ones is considered. To compare the filtering performances of the different methods mentioned above, we consider two types of noise: Gaussian and impulse noise. Corruption by impulse noise is expressed as [39]:

$$x_{i,j} = \begin{cases} v & \text{with probability } p_v \\ o_{i,j} & \text{with probability } 1 - p_v \end{cases} \tag{31}$$

where i,j characterize the sample position, $o_{i,j}$ is the original sample, $x_{i,j}$ represents the sample from the noisy image, p_v is a corruption probability and $v = (v_R, v_G, v_B)$ is a noise vector of intensity random values [37]. For the experiments, the considered degree of the impulse noise corruption p_v ranges from 1% to 30%. To evaluate the achieved results, objective criteria such as mean square error (MSE) and

[1] http://www.greyc.ensicaen.fr/~dtschump/.

normalized color difference (NCD) have been considered. These objective quality measures can be expressed by the following formulas:

$$MSE = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}(o_{i,j} - x_{i,j})^2 \tag{32}$$

$$NCD = \frac{\sum_{i=1}^{N}\sqrt{(L^o_{i,j} - L^x_{i,j})^2 + (a^o_{i,j} - a^x_{i,j})^2 + (b^o_{i,j} - b^x_{i,j})^2}}{\sum_{i=1}^{N}\sqrt{(L^o_{i,j})^2 + (a^o_{i,j})^2 + (b^o_{i,j})^2}} \tag{33}$$

where L represents lightness values and a,b chrominance values corresponding to the original $o_{i,j}$ and noisy $x_{i,j}$ samples expressed in the CIE LAB color space. The noise attenuation properties of the different filters are examined by utilizing the standard images parrots and peppers [55].

Fig. 3 shows the performances of the examined filters expressed through MSE and NCD according to the degree of impulse noise corruption. PDE-based color regularization of Tschumperlé is not a good candidate for impulse noise elimination and its behavior quickly degrades even with small quantities of impulse noise. PDEs expressed when separating chromaticity from brightness [59] seem more appropriate for this kind of noise. This also is the case of our filter for high values of λ since it accentuates fidelity to the content of the (noisy) image. For small portions of impulse noise (less than 3%) our filters outperforms the VMF when very small values of λ are used. However, as soon as noise corruption increases, the performance of our proposed filters is not as good as the VMF. For our

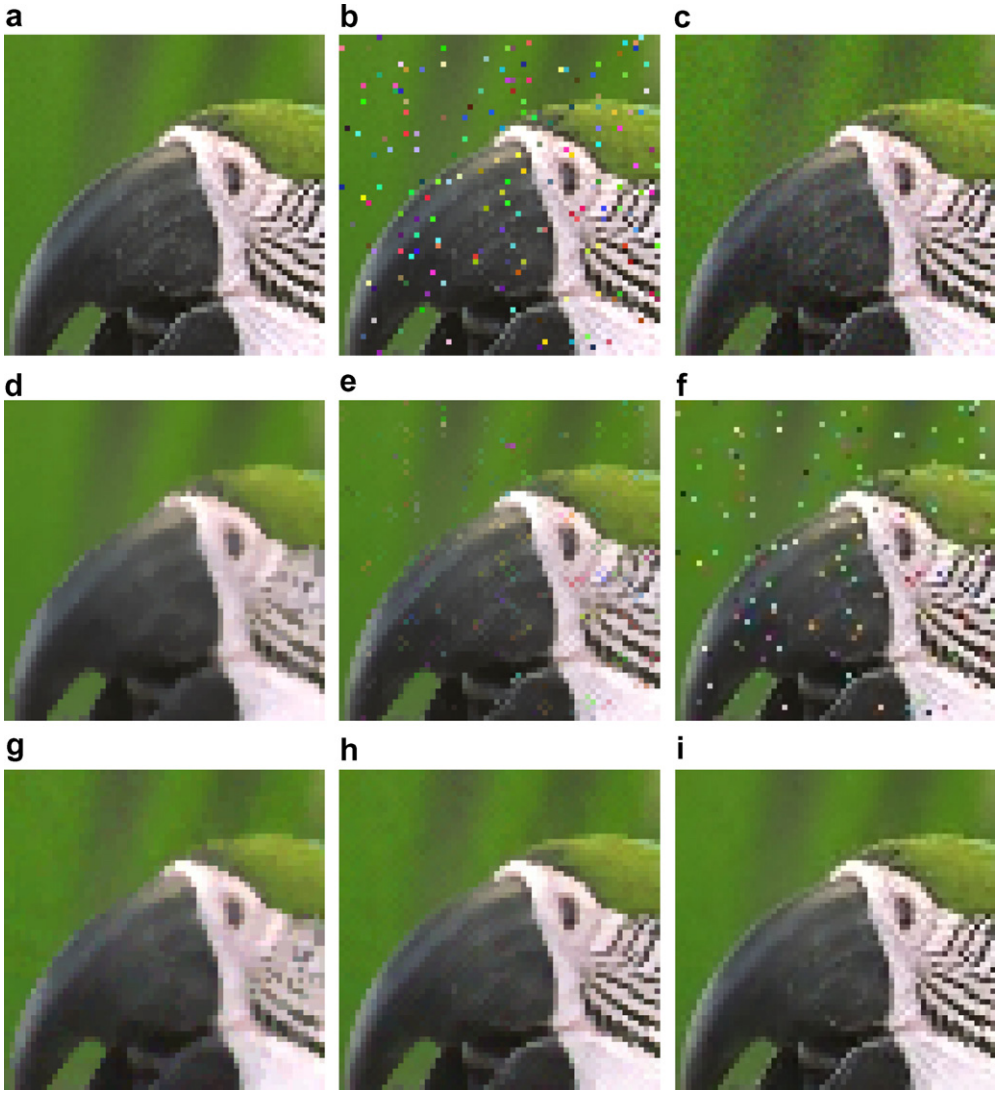

Fig. 5. Illustrative examples of the filtering efficiency. First row: (a) zoomed part of the parrots test image, (b) image distorted by 5% impulse noise, (c) image distorted by Gaussian noise ($\sigma = 5$). Second row concerns the restoration of impulse noise: (d) restoration achieved with the VMF, (e) restoration achieved using the proposed filtering ($p = 1$, $\lambda = 0$). (f) Restoration achieved with the PDE-based color regularization of Tschumperlé. Third row concerns the restoration of Gaussian noise: (g) restoration achieved with the VMF, (h) restoration achieved using the proposed filtering ($p = 1$, $\lambda = 0.001$), (i) restoration achieved with the PDE-based color regularization of Tschumperlé.

family of filters, good parameters seem to be $p = 1$ and $\lambda = 0$. Figs. 5 and 6 present restored zoomed parts of the parrots and peppers images with the VMF (Fig. 5(d) and 6(d)), our new filter with $p = 1, \lambda = 0$ (Fig. 5(e) and 6(e)) and the one of Tschumperlé (Fig. 5(f) and 6(f)). Fig. 5(a) and 6(a) present the zoomed part of the original image and Fig. 5(b) and 6(b) their 5% impulse noise corrupted versions. One can see that our filter better preserves edges while suppressing noise.

Fig. 4 shows the performances of the examined filters expressed through MSE and NCD according to the degree of Gaussian noise corruption. The PDE-based color regularization of Tschumperlé now is much more competitive, but this also is the case of our proposed filters when small values of λ are used. The VMF results always are worse than the results produced using the proposed filters and the method by Tschumperlé. For our family of filters, good parameters seem to be $p = 1$ and $\lambda = 0.001$. Figs. 5 and 6 present restored zoomed parts with the VMF (Fig. 5(g) and 6(g)), our new filter with $p = 1, \lambda = 0.001$ (Fig. 5(h) and 6(h)) and the one of Tschumperlé (Fig. 5(i) and 6(i)). Fig. 5(c) and 6(c) present the corrupted versions with Gaussian noise ($\sigma = 5$) of the original zoomed parts. Results for our filter and Tschumperlé's one are very similar, the latter providing slightly more contrasted results.

We have seen that the family of filters proposed in this paper is a good candidate both for Gaussian and impulse noise filtering. However, we have considered, in the above experiments, only one iteration of the filters. They can achieve better performance when iterated since their minimization process is based on such an iterated one. Fig. 7 presents convergence depending on the iteration of one filter with given parameters ($p = 1$, $\lambda = 0.01$) for both different amounts of Gaussian and impulse noise. The

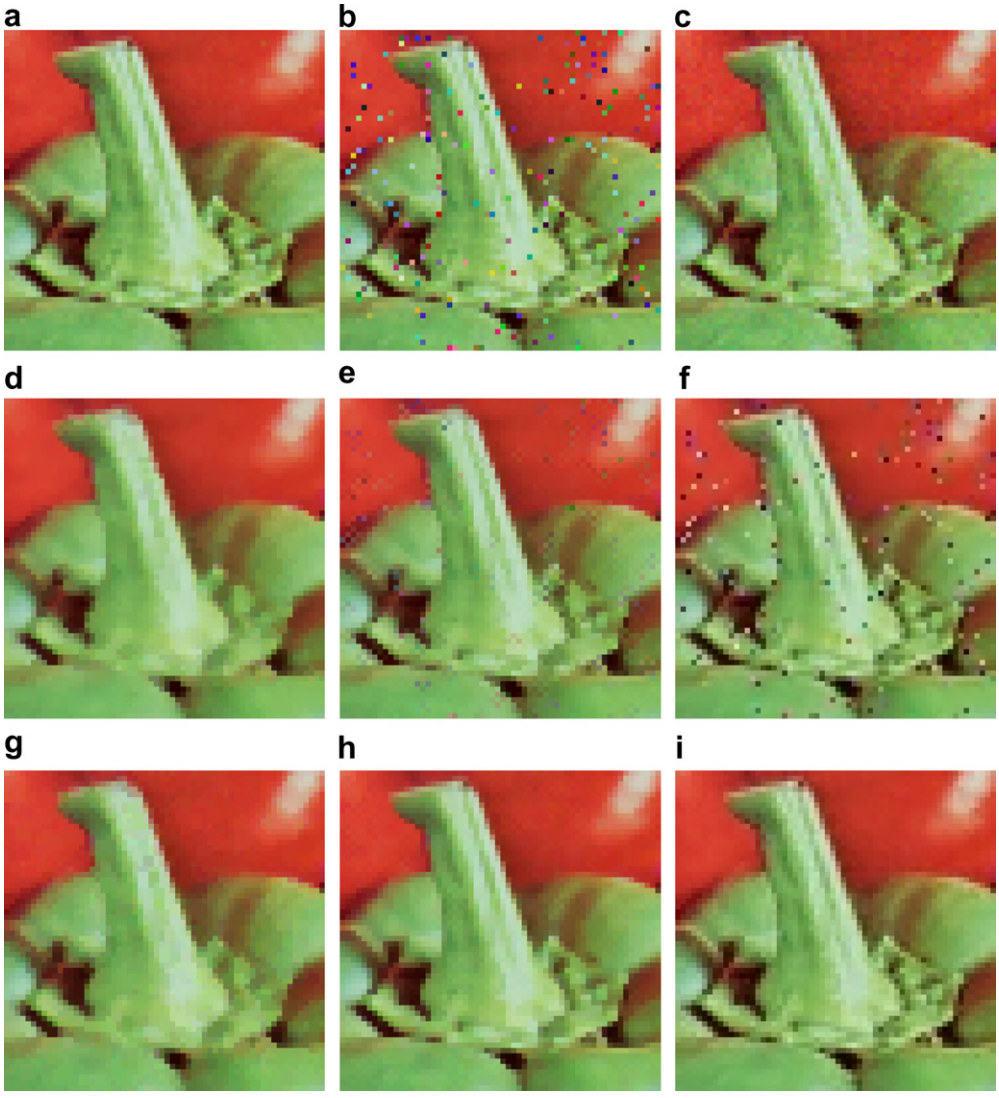

Fig. 6. Illustrative examples of the filtering efficiency. First row: (a) zoomed part of the peppers test image, (b) image distorted by 5% impulse noise, (c) image distorted by Gaussian noise ($\sigma = 5$). Second row concerns the restoration of impulse noise: (d) restoration achieved with the VMF, (e) restoration achieved using the proposed filtering ($p = 1$, $\lambda = 0$). (f) Restoration achieved with the PDE-based color regularization of Tschumperlé. Third row concerns the restoration of Gaussian noise: (g) restoration achieved with the VMF, (h) restoration achieved using the proposed filtering ($p = 1$, $\lambda = 0.001$), (i) restoration achieved with the PDE-based color regularization of Tschumperlé.

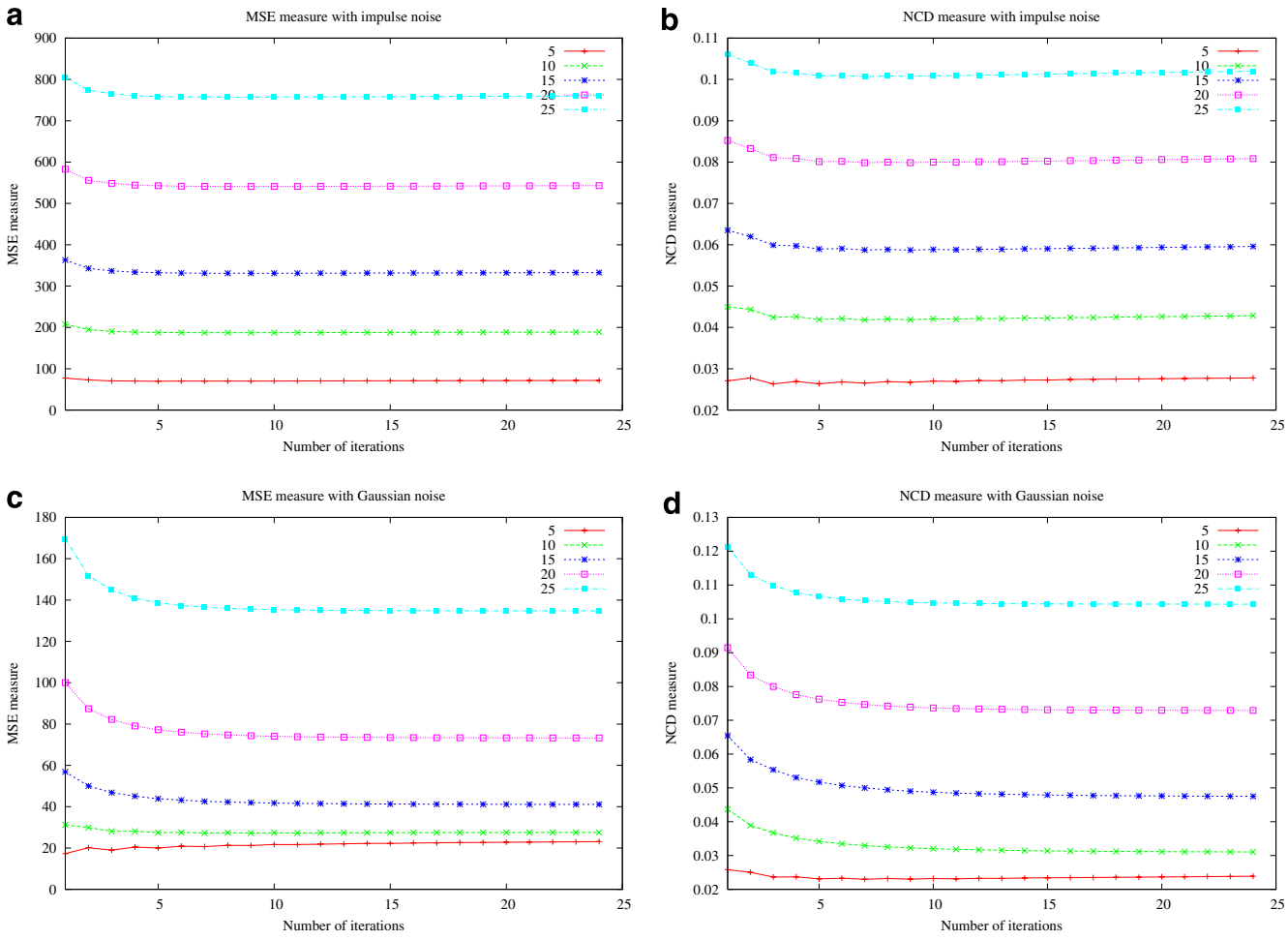

Fig. 7. Efficiency of the proposed filters in successive iterations using the parrots image corrupted with different amounts of Gaussian (second row) and impulse (first row) noise in terms of MSE (first column) and NCD (second column). Used parameters are ($p = 1$, $\lambda = 0.01$) for both Gaussian and impulse noise. Each curve corresponds to a given amount of introduced noise.

convergence is evaluated in terms of MSE and NCD changes in subsequent iterations of the filter. These objective quality criteria for both Gaussian and impulse noise are shown for different amount of noise introduced in the parrots test image. We only consider the case of $p = 1$, since for $p = 2$, convergence is very quickly obtained (less than 5 iterations). For $p = 1$, one can see that optimal results can also be obtained in a low number of iterations (from 5 to 10 iterations). As expected, the iterative application of the proposed filters enables to accentuate their filtering abilities. The number of required iterations increases with the level of noise while remaining quite small; and for small noise portions, very few iterations are needed. Finally, the use of 8-neighborhood connectivity with the proposed filters will surely provide better results but this has to be assessed.

4.2. Color image simplification

Image simplification can be performed by graph regularization on grid graphs but that requires a higher number of iterations. Another graph structure can be considered since the pixel grid is not a natural representation of visual scenes. Indeed, it is much more natural, and presumably more efficient, to work with perceptually meaningful entities obtained from a low-level grouping process: one can consider what is called fine partitions. A fine partition is an over-segmented partition of an image [1,68]. Since it is over-segmented, the reconstructed color image associated to the fine partition is very close to the initial image. Constructing this so-called mosaic image [2] (each pixel has the average color of its surrounding region) is similar to achieving an image simplification the result of which is piecewise constant. The graph structure associated to an image partition is the region adjacency graph. Therefore, image simplification can be performed on that graph and not on the pixel grid. That's quite interesting since the RAG has less vertices than the grid graph, and so the graph regularization can be computed much faster.

Fig. 8 presents results of image simplification. The initial image (Fig. 8(a)) has been segmented by two different segmentation methods (quasi flat zones [52] and homogeneous zones [46]) which can produce fine partitions (Fig. 8(b) and 8(c)). Those partitions have a number of regions which are

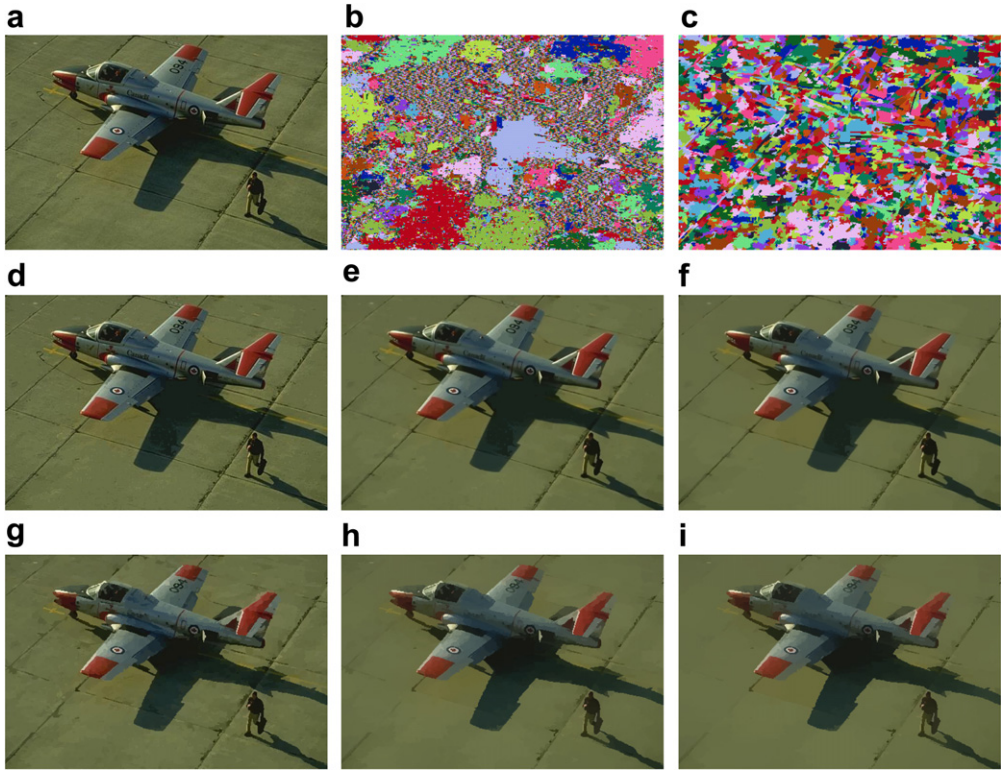

Fig. 8. Smoothing of a color image depending on the initial image fine partition. (a) Initial image (150072 pixels). (b) Quast flat zones (51882 regions). (c) Homogeneous zones (3011 regions). (d) Colorized image of (b) (PSNR = 31.75 dB. (e) $t = 5$, $p = 1$, $\lambda = 0.01$. (f) $t = 100$, $p = 1$, $\lambda = 0.01$. (g) Colorized image of (c) (PSNR = 31.80 dB. (h) $t = 5$, $p = 1$, 0.01. (i) $t = 100$, $p = 1$, $\lambda = 0.01$.

significantly less important than the pixels of the initial image (around 3 times less for the quasi flat zones and 50 times for the homogeneous zones), but their visual content still is close to the image initial content (see the colorized version of each fine partition in Fig. 8(d) and 8(g)). Since the simplification is performed on the Region Adjacency Graph of a fine partition, each region is described by its mean color; and the graph regularization operates on these models. The used weight function of edges $w(u,v)$ is the same as for image restoration. The simplification effect primarily depends on the fineness of the fine partition. For very fine partitions, the simplification is small. For coarser fine partitions, the simplification is higher. This is illustrated in Fig. 8(e)–(f) and (h)–(i) for two different fine partitions with the same simplification parameters. Therefore, this image simplification method is an interesting alternative to classical image simplification: it is much faster, the acceleration factor depends on the fineness of the partition used to construct the region adjacency graph and the simplification depends on the number of iterations and on λ.

4.3. Color image segmentation

We have seen that fine partitions (over-segmented segmentations) are of interest for image smoothing within the proposed graph regularization framework. First, this method is a faster alternative to classical image simplification, since it operates on a set which is much less important than the whole pixel grid. Second, classical scale-space generation by image simplification implies a loss of resolution and a displacement of edges across the scales which has to be solved by specific linking schemes [66], this is not the case with our approach to color image smoothing. Third, the simplified RAG can be used to obtain easily an improved segmentation as compared to the original fine one. Therefore, we propose to couple graph regularization with graph decimation by merging vertices after each RAG regularization step. Since the model attached to each RAG vertex is simplified by regularization for each iteration, similar regions tend to similar models and can be merged. This can be observed on Fig. 8(i) where a lot of regions have come to those having same color, although it was not the case on the original reconstructed mosaic image (Fig. 8(g)). A graph decimation will further decrease the computation time since the simplification will operate on a restricted RAG after the merging of similar vertices. This is the core of our proposal for color image segmentation which produces an irregular pyramid since it is based on graph decimation (i.e. region merging).

Our graph regularization method is a good way to obtain simplified versions of the models associated to vertices of the RAG of a fine partition. Indeed, it produces a hierarchy of images simplified at different scale levels. As stated above, we can make the most of the regularization to simplify the structure of the graph too. In fact, graph regularization tends to bring similar models closer

and similar regions can merge then. The idea of merging regions in a partition is quite old [29] and is the basis of a lot of hierarchical methods such as irregular pyramids [45]. For a complete merging strategy based on a RAG, several notions have to be defined [51]:

- The region model M_R: a model defines how to represent a region and also the union of two of them.
- The merging order $O(R_i,R_j)$: it associates to each edge of the RAG a similarity measure (a weight) between adjacent vertices. This order is a function defined for each couple of neighbor regions and its values belong to a totally ordered set $\mathscr{A}$ which provides the set of scales.
- The merging predicate $C(R_i,R_j)$: this criterion defines if two regions have to merge or not.

Creating a hierarchy of partitions by a region merging algorithm simply consists in pairwise merging of regions and in updating the RAG structure [40]. For each threshold $\alpha \in \mathscr{A}$, one can define a contraction kernel [34] on the graph which merges regions, the edge weight of which is lower than a given α threshold. This provides a partition P_α for each scale α. The construction of P_α is equivalent to finding the maximal connected components on the graph, the similarity of which is under the scale level α and $H = \{P_\alpha\}_{\alpha\in\mathscr{A}}$ is a hierarchy of nested partitions (since each region of $P_{\alpha+1}$ is a disjoint union of regions of P_α). We propose to combine this type of hierarchical segmentation which proceeds to a graph decimation with the previously proposed graph regularization. The principle is iterative and consists in alternating regularization and decimation of the RAG. For each iteration, models attached to each vertex are simplified and similar regions are merged according to a given merging criterion [22]. Each region is modeled by its average color: $M_{R_i} = f(v_i)$. M_{R_i} is therefore a vector of 3 components. The union of regions having to be computed fast, it is defined directly from the two models merging: $M_{R_i} \cup M_{R_j} = f(v_i) \cup f(v_j) = f(v_i) + f(v_j)$. The merging order is directly based upon the similarity between regions: $O(R_i,R_j) = O(v_i,v_j) = \|f(v_i) - f(v_j)\|$. To perform the merging of regions that fulfills the merging criterion, edges are ranked into a hierarchical priority queue according to the edge weights. For each merging, the edge of minimum cost is removed from the hierarchical queue. The next step

Fig. 9. Initial images (first column), fine partitions of the original images (second column), the reconstructed color images from the fine partitions (third column).

consists in merging the information of both regions associated to the edge. As a result, region models and edge weights are updated. Algorithm 1 lists the pseudo-code where α is the level of the hierarchy, one goes from one scale level α to the next one $(\alpha+1)$ only if regions have merged. The symbol $|.|$ denotes the cardinality when applied to a set.

Algorithm 1 (Algorithm for color image segmentation.).

$\alpha \leftarrow 0$; Define α_{end}
$\mathcal{G}^{\alpha} = (V^{\alpha}, E^{\alpha})$ for an initial partition P_{α}.
while $\alpha \leqslant \alpha_{end}$ **do**
  Regularize the graph $\mathcal{G}_{\alpha}$
  for all the edges $E_l = (v_i, v_j) \in V^{\alpha} \times V^{\alpha}$ **do**
    if $C(v_i, v_j)$ **then**
      Add E_l to the contraction kernel $\mathcal{K}^{\alpha}$
    end if
  end for
  if $|\mathcal{K}^{\alpha}| > 0$ **then**
    Contract the graph $\mathcal{G}^{\alpha}$ with the contraction kernel $\mathcal{K}^{\alpha} : \mathcal{G}^{\alpha+1} = \mathcal{K}^{\alpha}(\mathcal{G}^{\alpha})$
    $\alpha \leftarrow \alpha + 1$
  end if
end while

However, one thing remains to be defined: the merging predicate $C(v_i, v_j)$. Several possible choices include: a fixed threshold, an evolving threshold and an adaptive threshold. For fixed threshold the merging predicate is $O(v_i, v_j) < \alpha$. For an evolving threshold, the merging predicate is the same but after each iteration the threshold is increased via $\alpha = \alpha + \delta$ (δ is fixed to 0.25 in this paper). For an adaptive threshold, the threshold is dynamically

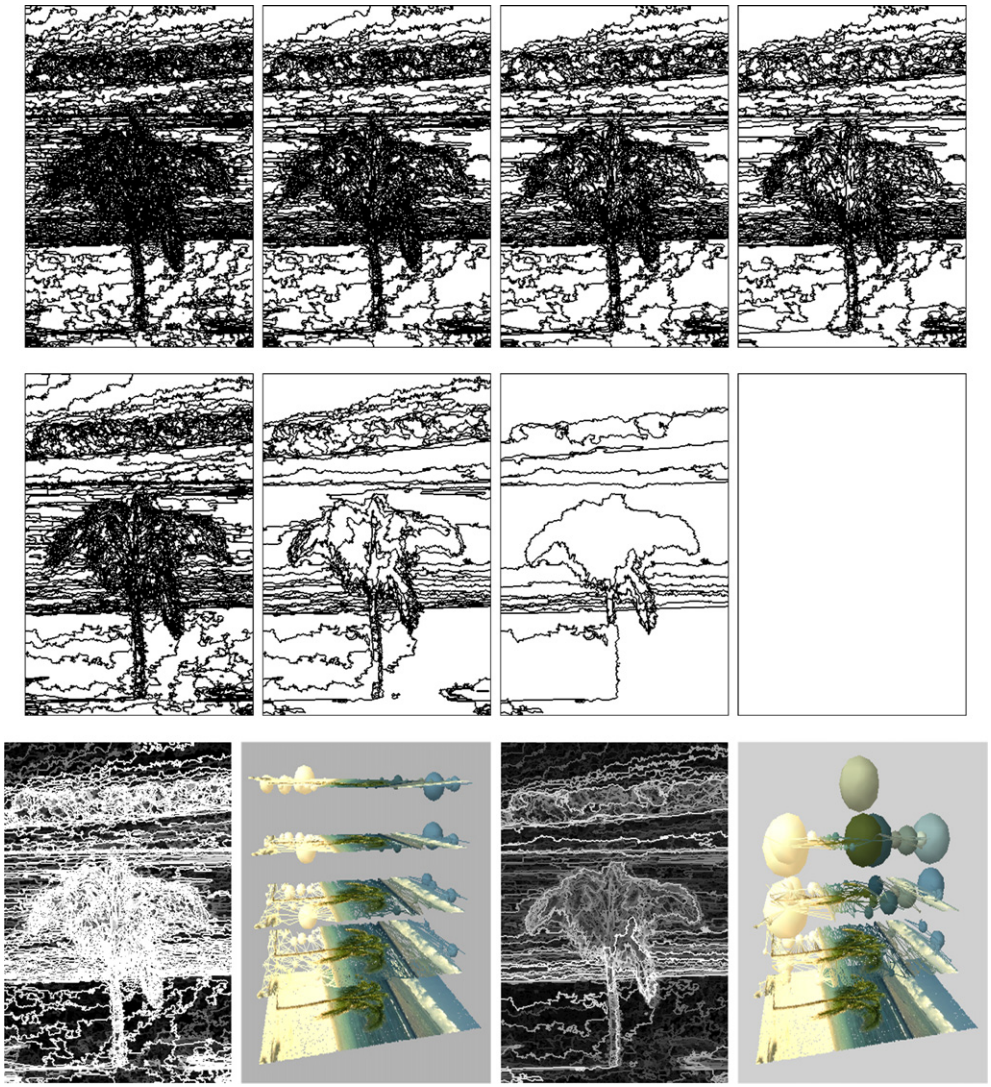

Fig. 10. Multi-scale segmentations in a hierarchy of partitions for scales 5, 10, 15 and 20 with two different merging criteria: fixed threshold (first row) and adaptive threshold (second row). Third row, columns 1 and 3 show the saliency map of the produced hierarchy. Third row, columns 2 and 4 present a hierarchical view of the region adjacency graphs. Hierarchies were obtained on Fig. 9(a) with the fine partition of Fig. 9(b).

determined from the image distribution (such as in [47] for instance) but for the sake of brevity, we do not consider that issue in this paper.

Figs. 10 and 11 present, for two color images from the BSDB [42] (shown in Fig. 9), multi-scale segmentations obtained by the proposed approach with two different merging predicates for the levels 5, 10, 15 and 20. We consider a regularization with $p = 1$ and $\lambda = 0$. The initial fine partition was obtained by extracting homogeneous zones using a connective criterion defined in [46]. Fig. 9 presents the initial images together with the corresponding fine partitions and reconstructed colorized images. As seen in Fig. 9, fine partitions are very close to the initial images contents (the corresponding PSNR is of 117.33 and 112.42) but their number of regions is much lower (resp. 41212 and 24725) than the whole number of pixels on the image grid (154401). For each image, two segmentation hierarchies are provided (illustrated by the produced partitions in Figs. 10 and 11), one for a fixed threshold equal to 1 (first rows of Fig. 10 and 11) and one for an evolving threshold (second rows of Fig. 10 and 11). To assess the influence of the merging predicate on the hierarchical segmentations, saliency maps are provided for the different predicates (third row, columns 1 and 3 of Fig. 10 and 11). A saliency map illustrates the importance of each pixel among the levels, the saliency of a pixel being defined as the highest level for which it occurs at the boundary between two regions (in the saliency map images, the gray level corresponds to the hierarchy level, therefore the highest the brightest). To illustrate the decimation processes, hierarchical views of the produced irregular pyramids are provided (third row, columns 2 and 4 of Figs. 10 and 11). A sphere is associated to each vertex (i.e. region) of the pyramid. The radius of the sphere is proportional to the area of the region associated to the vertex. The color of the sphere depends on $f(v_i)$. With a fixed threshold,

Fig. 11. Multi-scale segmentations in a hierarchy of partitions for scales 5, 10, 15 and 20 (columns 1–5) with two different merging criteria: fixed threshold (first row) and adaptive threshold (second row). Third row, columns 1 and 3 show the saliency map of the produced hierarchy. Third row, columns 2 and 4 present a hierarchical view of the region adjacency graphs. Hierarchies were obtained on Fig. 9(d) with the fine partition of Fig. 9(e).

the number of scales is high and the produced image segmentation is very close to the initial image even if the number of regions is significantly less important than the initial number of pixels. For the other predicate, less levels and coarser segmentations are obtained faster. This is related to the combination of simplification and decimation when the decimation criterion changes across scales. The produced hierarchies are good and extract the primary visual components from the image. Contrary to several multiscale methods which first simplify the image and then segment it [58], our approach makes it possible to combine these two approaches in a single faster algorithm.

5. Conclusions

We have considered a discrete regularization framework based on graph differential geometry. A family of linear and nonlinear filters was derived from this general framework. The discrete framework we used to process color images relies on the use of weighted graphs of the arbitrary topology. The proposed framework has been successfully applied to color image processing problems such as restoration, denoising and segmentation. There are a lot of possible extensions to this work. For example, one can process color adjacency graphs representing color images or color histograms for color image clustering purposes. Moreover, it is also known that better processing results can be obtained on color images when one separates chromaticity from brightness; the proposed framework has therefore to integrate such a representation. Finally, the proposed discrete regularization framework can be applied and adapted to a large number of applications such as color active contours, image inpainting, structure-texture image decomposition, etc.

Acknowledgments

The authors thank the reviewers for their valuable comments and suggestions to improve the quality of this paper.

This work was supported by the University of Caen Low Normandy under grants of FEDER and FNADT funds.

References

[1] J. Angulo, J. Serra, Morphological coding of color images by vector connected filters, in: Proceedings of ISSPA. vol. I, 2003, pp. 69–72.
[2] J. Angulo, J. Serra, Color segmentation by ordered mergings, in: IEEE Internat. Conf. on Image Processing, vol. 2, 2004, pp. 125–128.
[3] J. Astola, P. Haavisto, Y. Neuvo, Vector median filters, Proc. IEEE 74 (4) (1990) 678–689.
[4] G. Aubert, P. Kornprobst, Mathematical Problems in Image Processing, Springer-Verlag, 2002.
[5] A. Bakushinsky, A. Goncharsky, Ill-Posed Problems: Theory and Applications, Kluwer Academic Publishers, 1994.
[6] M. Belkin, P. Niyogi, V. Sindhwani, On manifold regularization, in: Proc. Tenth Internat. Workshop on Artificial Intelligence and Statistics, 2005, pp. 17–24.
[7] A. Bensoussan, J.-L. Menaldi, Difference equations on weighted graphs, J. Convex. Anal. 12 (2005) 13–44.
[8] P. Blomgren, T. Chan, Color TV: total variation methods for restoration of vector-valued images, IEEE Trans. Image Process. 7 (3) (1998) 304–309.
[9] S. Bougleux, A. Elmoataz, Image smoothing and segmentation by graph regularization, in: Proc. Intern. Symp. on Visual Computing, 2005, pp. 745–752.
[10] I. Bronshtein, K. Semendyayev, Handbook of Mathematics, 3rd ed., Springer-Verlag, New York, 1997.
[11] A. Brook, R. Kimmel, N. Sochen, Variational restoration and edge detection for color images, J. Math Imaging and Vis. 18 (2003) 247–268.
[12] V. Caselles, G. Sapiro, D. Chung, Vector median filters, inf-sup operations, and coupled PDEs: theoretical connections, J. Math. Imaging Vis. 8 (2000) 109–119.
[13] A. Chambolle, P.-L. Lions, Image recovery via total variation minimization and related problems, Numer. Math. 76 (2) (1997) 167–188.
[14] A. Chambolle, An algorithm for total variation minimization and applications, J. Math. Imaging Vis. 20 (1) (2004) 89–97.
[15] T. Chan, S. Esedoglu, F. Park, M. Yip, Total variation image restoration: overview and recent developments, in: Y.N. Paragios, O.F. Chen (Eds.), Handbook of Mathematical Models in Computer Vision, Springer Verlag, 2005, pp. 17–32, Ch. 2.
[16] T. Chan, S. Osher, J. Shen, The digital TV filter and nonlinear denoising, IEEE Trans. Image Process. 10 (2001) 231–241.
[17] T. Chan, J. Shen, Image Processing and Analysis—Variational, PDE, Wavelet, and Stochastic Methods, SIAM, 2005.
[18] T. Chan, H. Zhou, R. Chan, Continuation method for total variation denoising problems, in: Proceedings of the SPIE Conference onAdvanced Signal Processing Algorithms, 1995, pp. 314–325.
[19] P. Combettes, J. Luo, An adaptive level set method for nondifferentiable constrained image recovery, IEEE Trans. Image Process. 11 (11) (2002) 1295–1304.
[20] J. Darbon, M. Sigelle, Image restoration with discrete constrained total variation part i: Fast and exact optimization, J. Math. Imaging Vis. 26 (3) (2006) 261–276.
[21] R. DiestelGraph Theory, vol. 173, Springer-Verlag, 2005.
[22] P. Felzenswalb, D. Huttenlocher, Efficient graph-based image segmentation, Int. J. Comput. Vis. 59 (2) (2004) 167–181.
[23] C. Frohn-Schauf, S. Henn, K. Witsch, Nonlinear multigrid methods for total variation image denoising, Comput. Visual. Sci. 7 (3-4) (2004) 199–206.
[24] H. Fu, M. Ng, M. Nikolova, J. Barlow, Efficient minimization methods of mixed Ll–Ll and L2–L1 norms for image restoration, SIAM J. Scientific Comput. 27 (6) (2006) 1881–1902.
[25] D. Goldfarb, W. Yin, Second-order cone programming based methods for total variation image restoration, SIAM J. Sci. Comput. 27 (2005) 622–645.
[26] A. Greenbaum, Iterative Methods for Solving Linear Systems, vol. 17 of Frontiers in Applied Mathematics. SIAM, Philadelphia, 1997.
[27] H. Hacid, A. Zighed, Neighborhood graphs for image databases indexing and content-based retrieval, in: First IEEE International Workshop on mining Complex Data (IEEE MCD05), 2005, pp. 45–52.
[28] L. Hageman, D. Young, Applied Iterative Methods, Academic Press, New York, 1981.
[29] S. Horowitz, T. Pavlidis, Picture segmentation by a tree traversal algorithm, J. Assoc. Comput. Mach. 2 (23) (1976) 368–388.
[30] M.C. Kang, A.K. Katsaggelos, General choice of the regularization functional in regularized image restoration, IEEE Trans. Image Process. 4 (5) (1995) 594–602.
[31] D. Keren, A. Gotlib, Denoising color images using regularization and correlation terms, J. Vis. Commun. Image Represent. 9 (4) (1998) 352–365.
[32] R. Kimmel, R. Malladi, N. Sochen, Images as embedded maps and minimal surfaces: movies, color, texture, and volumetric medical images, Int. J. Comput. Vis. 39 (2) (2000) 111–129.

[33] D. Kincaid, W. Cheney, Numerical Analysis: Mathematics of Scientific Computing, 3rd ed., Brooks/Cole, Pacific Grove, 2002.
[34] Kropatsch, Building irregular pyramids by dual graph contraction, in: IEEE Proceedings of Vision, Image and Signal Processing, vol. 142, 1995, pp. 366–374.
[35] S. Levine, Y. Chan, J. Stanich, Image Restoration via Nonstandard Diffusion, Technical Report 04-01, Dept of Mathematics and Computer Science, Duquesne University, 2004.
[36] R. Lukac, K. Plataniotis, B. Smolka, A. Venetsanopoulos, Generalized selection weighted vector filters, EURASIP Journal on Applied Signal Processing: Special Issue on Nonlinear Signal and Image Processing 2004 (12) (2004) 1870–1885.
[37] R. Lukac, K. Plataniotis, A taxonomy of color image filtering and enhancement solutions, in: P. Hawkes (Ed.), Advances in Imaging and Electron Physics, vol. 140, Elsevier, 2006, pp. 187–264.
[38] R. Lukac, B. Smolka, K. Martin, K. Plataniotis, A. Venetsanopoulos, Vector filtering for color imaging, IEEE Signal Process. Mag. Special Issue Color Image Process. 22 (I) (2005) 74–86.
[39] R. Lukac, Adaptive vector median filtering, Pattern Recognit. Lett. 24 (12) (2003) 1889–1899.
[40] S. Makrogiannis, G. Economou, S. Fotopoulos, A graph theory approach for automatic segmentation of color images, in: International Workshop on Very Low Bit-rate Video, 2001, pp. 162–166.
[41] J. Malik, S. Belongie, T.K. Leung, J. Shi, Contour and texture analysis for image segmentation, Int. J. Comput. Vis. 43 (1) (2001) 7–27.
[42] D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: Proc. Eighth Internat. Conf. on Computer Vision, vol. 2, 200L, pp. 416–423.
[43] J. Matas, R. Marik, J. Kittler, The Color Adjacency Graph Representation of Multi-coloured Objects, Tech. Rep. VSSP-TR-1/95, University of Surrey, Surrey, Great Britain, 1995.
[44] Z. Ma, H. Wu, B. Qiu, A robust structure-adaptive hybrid vector filter for color image restoration, IEEE Trans. Image Process. 14 (12) (2005) 1990–2001.
[45] P. Meer, Stochastic image pyramids, Comput. Vis. Graph. Image Process. 45 (3) (1990) 269–294.
[46] C. Meurie, Color Image Segmentation by Pixel Classification and Partition Hierarchies. Ph.D. Thesis, Université de Caen Basse Normandie, 2005.
[47] R. Nock, F. Nielsen, Statistical region merging, IEEE Trans. Pattern Anal. Mach. Intell. 26 (11) (2004) 1452–1458.
[48] P. Perona, J. Malik, Scale-space and edge detection using anisotropic diffusion, IEEE Trans. Pattern Anal. Mach. Intell. 12 (1990) 629–639.
[49] K. Plataniotis, A. Venetsanopoulos, Color Image Processing and Applications, Springer Verlag, 2000.
[50] L. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Phys. D 60 (1992) 259–268.
[51] P. Salembier, L. Garrido, Binary partition tree as an efficient representation for image processing, segmentation and information retrieval, IEEE Trans. Image Process. 9 (4) (2000) 561–576.
[52] P. Salembier, J. Serra, Flat zone filtering, connected operators and filters by reconstruction, IEEE Trans. Image Process. 4 (1995) 1153–1160.
[53] G. Sapiro, D. Ringach, Anisotropic diffusion of multivalued images with applications to color filtering, IEEE Trans. Image Process. 10 (5) (1996) 1582–1586.
[54] G. Sharma, H. Trussell, Digital color imaging, IEEE Trans. Image Process. 6 (7) (1997) 901–932.
[55] B. Smolka, A. Chydzinski, Fast detection and impulsive noise removal in color images, Real-Time Imaging 11 (5-6) (2005) 389–402.
[56] B. Smolka, R. Lukac, K.N. Plataniotis, A.N. Venetsanopoulos, Modified anisotropic diffusion framework, in: Proceedings of SPIE, Visual Communications and Image Processing, vol. 5150, 2003, pp. 1657–1666.
[57] B. Smolka, R. Lukac, On the combined forward and backward anisotropic diffusion scheme for the multispectral image enhancement, in: Proceedings of Photogrammetric Computer Vision, 2002, pp. 249–254.
[58] B. Sumengen, B.S. Manjunath, Multi-scale edge detection and image segmentation, in: European Signal Processing Conference (EUSIPCO). 2005, pp. CD-proceedings.
[59] B. Tang, G. Sapiro, V. Caselles, Color image enhancement via chromaticity diffusion, IEEE Trans. Image Process. 10 (2001) 701–707.
[60] A. Tikhonov, V. Arsenin, Solution of Ill-posed Problems, Winston & Sons, 1977.
[61] P. Trahanias, D. Karakos, A. Venetsanopoulos, Directional processing of color images: theory and experimental results, IEEE Trans. Image Process. 5 (1996) 868–880.
[62] A. Tremeau, P. Colantoni, Regions adjacency graph applied to color image segmentation, IEEE Trans. Image Process. 9 (4) (2000) 735–744.
[63] M. Troyanov, Solving the p-Laplacian on manifolds, in: Proc. Am. Math. Soc., vol. 128, 2000, pp. 541–545.
[64] D. Tschumperlé, R. Deriche, Vector-valued image regularization with PDEs: a common framework for different applications, IEEE Trans. Pattern Anal. Mach. Intell. 17 (4) (2005) 506–517.
[65] D. Tschumperlé, Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE's, Int. J. Comput. Vis. 68 (1) (2006) 65–82.
[66] I. Vanhamel, I. Pratikakis, H. Sahli, Multiscale gradient watersheds of color images, IEEE Trans. Image Process. 12 (6) (2003) 617–626.
[67] J. Weickert, Coherence-enhancing diffusion of colour images, Image Vis. Comput. 17 (3-4) (1999) 201–212.
[68] F. Zanoguera, B. Marcotegui, F. Meyer, A tool-box for interactive image segmentation based on nested partions, in: IEEE Internat. Conf. on Image Processing, 1999, pp. 21–25.
[69] D. Zhou, B. Scholkopf, A regularization framework for learning from graph data, in: ICML Workshop on Statistical Relational Learning and Its Connections to Other Fields, 2004, pp. 132–137.
[70] D. Zhou, B. Scholkopf, Regularization on discrete spaces, in: Proceedings of the 27th DAGM Symposium, 2005, pp. 361–368.