Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

On the application of the spectral projected gradient method in image segmentation

L. Antonelli, V. De Simone, D. di Serafino

RT-ICAR-NA-2015-01 **Febbraio 2015**

Consiglio Nazionale delle Ricerche, Istituto di Calcolo e Reti ad Alte Prestazioni (ICAR) – Sede di Napoli, Via P. Castellino 111, I-80131 Napoli, Tel: +39-0816139508, Fax: +39-0816139531, e-mail: napoli@icar.cnr.it, URL: *www.na.icar.cnr.it*

Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

On the application of the spectral projected gradient method in image segmentation*

L. Antonelli [1], *V. De Simone* [2], *D. di Serafino* [2,1]

Rapporto Tecnico N.:
RT-ICAR-NA-2015-01

Data:
Febbraio 2015

---

* Rapporto Tecnico sottomesso per la pubblicazione

[1] Istituto di Calcolo e Reti ad Alte Prestazioni, ICAR-CNR, Sede di Napoli, via P. Castellino n. 111, 80131 Napoli, Italy
[2] Dipartimento di Matematica e Fisica, Seconda Università degli Studi di Napoli, viale A. Lincoln n. 5, 81100 Caserta, Italy

On the application of the spectral projected gradient method in image segmentation

Laura Antonelli · Valentina De Simone · Daniela di Serafino

Abstract We investigate the application of the non-monotone spectral projected gradient (SPG) method to a region-based variational model for image segmentation. We consider a "discretize-then-optimize" approach and solve the resulting nonlinear optimization problem by an alternating minimization procedure that exploits an SPG algorithm by Birgin, Martínez and Raydan (SIAM J. Optim., 10(4), 2000). We provide a convergence analysis and perform numerical experiments on several images to evaluate the effectiveness of this procedure. The computational results show that our approach is competitive with a very efficient solver based on the Split Bregman method.

Keywords Image segmentation · Region-based variational model · Spectral Projected Gradient

Mathematics Subject Classification (2000) 68U10 · 65K05 · 90C30

Work partially supported by INdAM-GNCS (2014 Project *First-order optimization methods for image restoration and analysis*) and by MIUR (FIRB 2010 Project n. RBFR106S1Z002).

L. Antonelli
Institute for High-Performance Computing and Networking (ICAR), CNR, via P. Castellino 111, I-80131 Naples, Italy
E-mail: laura.antonelli@na.icar.cnr.it

D. di Serafino
Department of Mathematics and Physics, Second University of Naples, viale A. Lincoln 5, 81100 Caserta, Italy, and ICAR-CNR, via P. Castellino 111, I-80131 Naples, Italy
E-mail: daniela.diserafino@unina2.it

V. De Simone
Department of Mathematics and Physics, Second University of Naples, viale A. Lincoln 5, 81100 Caserta, Italy
E-mail: valentina.desimone@unina2.it

1 Introduction: background and motivations

Image segmentation is the process of partitioning an image into regions that are "homogeneous" according to some feature, such as intensity, texture or colour. This process is generally aimed at identifying objects in the image, thus it plays a fundamental role in computer vision, e.g., for object detection, recognition, measurement and tracking. Many successful methods for image segmentations are based on variational models where the regions of the desired partition, or their edges, are obtained by minimizing suitable cost functionals (see, e.g., [13,9,23,26] and the references therein).

Here we consider a region-based variational model by Chan, Esedoḡlu and Nikolova [12], henceforth referred to as CEN model, which was proposed to overcome difficulties associated with the minimization of the non-convex Chan-Vese (CV) model [14]. The latter is actually the Mumford-Shah model [27] restricted to the case of piecewise constant two-phase segmentation; its level-set formulation reads as follows:

$$\min_{c_1,c_2,\phi} E_{CV}(c_1,c_2,\phi), \tag{1}$$

where

$$\begin{aligned} E_{CV}(c_1,c_2,\phi) = &\int_\Omega |\nabla H(\phi(x))|\,dx \\ &+\lambda\left(\int_\Omega H(\phi(x))\,(c_1-\bar{u}(x))^2\,dx\right. \\ &\left.+\int_\Omega (1-H(\phi(x)))\,(c_2-\bar{u}(x))^2\,dx\right). \end{aligned} \tag{2}$$

Here $\Omega \subset \mathbb{R}^2$ is an open bounded set with Lipschitz boundary (generally a rectangle), $\bar{u}(x) : \Omega \to \mathbb{R}$ represents the image to be segmented, H is the Heaviside function, $\phi : \Omega \to \mathbb{R}$ is a Lipschitz-continuous function whose zero level set represents the boundary $\partial\Sigma$ of a set $\Sigma \subset \Omega$, $c_1, c_2 \in \mathbb{R}$, and $\lambda > 0$ is a suitable regularization

parameter. Solving (1) means finding the best approximation to $\bar{u}(x)$ among all the functions that take only two values; c_1 and c_2 represent these values, while Σ and $\Omega\backslash\Sigma$ are the sets where they are taken, which provide a two-phase partition of Ω. The first term in the right-hand side of (2) is a regularization term, which penalizes the size of $\partial\Sigma$. For any fixed ϕ, the values of c_1 and c_2 that minimize E_{CV} are given by

$$\begin{aligned} c_1 &= \frac{\int_\Omega \bar{u}(x)H(\phi(x))dx}{\int_\Omega H(\phi(x))dx}, \\ c_2 &= \frac{\int_\Omega \bar{u}(x)(1-H(\phi(x)))dx}{\int_\Omega (1-H(\phi(x)))dx}, \end{aligned} \tag{3}$$

i.e., by the mean values of $\bar{u}(x)$ in the regions Σ and $\Omega\backslash\Sigma$. Therefore, a natural approach to solve problem (1) is to alternate between the computation of c_1 and c_2 through (3) and the minimization of $E_{CV}(c_1, c_2, \phi)$ with respect to ϕ. This minimization is usually performed by using a gradient-descent scheme, where the first variation of $E_{CV}(c_1, c_2, \phi)$ is computed by using a smooth regularization of the Heaviside function [14]. However, because of the non-convexity of the functional E_{CV} with respect to ϕ, this method may get stuck into local minima, thus providing poor segmentations.

In order to overcome the previous problem, a convex relaxation approach is used in the CEN model, which is based on the idea of removing the constraint that $\bar{u}(x)$ be approximated by functions taking only two values. Specifically, in [12, Theorem 2] it is proved that, for any given (c_1, c_2), a global minimizer for the piecewise constant two-phase Mumford-Shah model can be obtained by solving the following convex problem:

$$\begin{aligned} &\min_u E_{CEN}(c_1, c_2, u), \\ &\text{s.t. } 0 \le u \le 1, \end{aligned} \tag{4}$$

where

$$\begin{aligned} E_{CEN}(c_1, c_2, u) = &\int_\Omega |\nabla u|\, dx \\ &+ \lambda \left(\int_\Omega u(x) \left(c_1 - \bar{u}(x)\right)^2 dx \right. \\ &\left. + \int_\Omega (1-u(x)) \left(c_2 - \bar{u}(x)\right)^2 dx \right), \end{aligned} \tag{5}$$

and by taking

$$\Sigma = \{x \in \Omega : u(x) \ge \mu\}, \tag{6}$$

for almost any $\mu \in (0, 1)$.

Different methods have been proposed to solve problem (4). In [12] the constraint $0 \le u \le 1$ is enforced by using a penalty term and then applying the gradient-descent technique to the resulting unconstrained problem. Regularized versions of $|\nabla u|$ and of the penalty term are used to deal with their non-differentiability. A penalty approach, combined with a suitable splitting of the resulting energy functional, is also used in [9]. In this case, minimization is performed by exploiting the dual formulation of the TV norm and applying a regularization that allows the application of an alternating solution procedure, based on a semi-implicit gradient-descent algorithm and an explicit minimization formula [1]. However, as observed in [23], the previous methods may introduce too much regularization, possibly yielding the elimination of fine segmentation details. Furthermore, when explicit discretizations of the gradient flow equation are used, the resulting numerical methods are generally slow, because of the time step limitations imposed by stability conditions.

The application of the Split Bregman (SB) method to problem (4) has been proposed in [23] to deal with the previous difficulties. This approach avoids regularization and splits the problem into subproblems that can be efficiently solved within an alternating minimization scheme. Although the convergence of the SB method has been proved when the subproblems are solved exactly [24,28], it has been observed that approximate solutions can be used in practice without deteriorating convergence speed [23]. It has been also shown in [23] that this approach can be faster and more accurate than the well-known algorithm for motion by mean curvature graph-cut technique [11,17].

We note that alternating minimization methods for problem (4), or for variants of it, can be also derived in the general context of Lagrangian methods (see, e.g., [29,26]). There are indeed strong connections between the alternating direction method of multipliers and the SB method, as shown in [20,28].

In this article we investigate the application of nonmonotone spectral projected gradient (SPG) methods, within an alternating minimization procedure, to the CEN model. The interest for SPG methods is due to their faster convergence with respect to classical gradient projection methods, which results from the combination of the spectral properties of the Barzilai and Borwein (BB) steplength [2] with the nonmonotone line-search technique by Grippo, Lampariello and Lucidi (GLL) [25]. We note that the idea of using steplengths related somehow to the spectrum of the Hessian of the objective function, in order to speedup the convergence of gradient methods, has gained widespread acceptance in the last years; therefore, an increasing amount of work has been devoted to designing and applying new gradient methods that employ such steplengths (see, e.g., [6,8,16,18,19,21,22,31]). In the context of image processing, the BB steplength has been used, e.g., in [30] to accelerate Chambolle's gradient projection method for total variation image restoration [10]. In this case, the nonmonotone line-search technique by Dai and Flet-

cher [15] has been applied to ensure global convergence. Finally, it has been observed in [9] that the regularization of $|\nabla u|$ may impose strong limitations on the steplength in classical explicit gradient-descent algorithms. The use of the BB steplength, with a nonmonotone line-search technique, offers a possibility to overcome this problem.

We consider a "discretize-then-optimize" approach, i.e., we first consider a discretization of the CEN model and then solve the resulting optimization problem. We focus of the SPG2 algorithm proposed in [4]. The k-th iteration of our minimization procedure consists of two steps:

- computation of an approximation u^k to u by applying an SPG2 step, with a suitable GLL-type line search, to the discretized CEN functional where c_1 and c_2 are equal to the previously computed values c_1^{k-1} and c_2^{k-1};
- computation of c_1^k and c_2^k by exact minimization, by using a discrete version of (3) where u is set equal to u^k.

This method is proved to be globally convergent and its effectiveness is illustrated through numerical experiments on a large set of test problems.

This paper is organized as follows. In Section 2 we provide a discrete formulation of the CEN model. In Section 3 we describe our SPG-based minimization algorithm and prove its convergence. In Section 4 we discuss the results of numerical experiments performed by applying our algorithm to several images. We also make a comparison with the SB technique presented in [23]. Finally, in Section 5, we draw some conclusions.

2 Discrete formulation

The image domain Ω is discretized by using a $m \times n$ grid of pixels

$$\Gamma = \{(i,j) : 0 \le i \le m-1, 0 \le j \le n-1\}.$$

We identify each pixel with its center and denote by $v_{i,j}$ the value in (i,j) of any function v defined in Ω. Now we introduce some finite-difference operators that are useful to present the discretization of the functional (5) over the grid Γ. For any $v_{i,j}$ with $(i,j) \in \Gamma$, we set

$$\begin{aligned}
&\delta_+^x v_{i,j} = v_{i+1,j} - v_{i,j}, \quad \delta_+^y v_{i,j} = v_{i,j+1} - v_{i,j},\\
&\delta_-^x v_{i,j} = v_{i,j} - v_{i-1,j}, \quad \delta_-^y v_{i,j} = v_{i,j} - v_{i,j-1},\\
&\delta_0^x v_{i,j} = \frac{v_{i+1,j} - v_{i-1,j}}{2}, \; \delta_0^y v_{i,j} = \frac{v_{i,j+1} - v_{i,j-1}}{2},
\end{aligned}$$

where we assume

$$v_{i-1,j} = v_{i,j} \;\text{ for } i = 0, \;\; v_{i,j-1} = v_{i,j} \;\text{ for } j = 0,$$
$$v_{i+1,j} = v_{i,j} \;\text{ for } i = m-1, \;\; v_{i,j+1} = v_{i,j} \;\text{ for } j = n-1,$$

i.e., we define by replication the values of v with indices outside Γ. To discretize the total variation term in (5) we proceed as in [10]; however, since $|\nabla u|$ is nondifferentiable in 0, we use a classical regularization of it, i.e., we set

$$|\nabla u|_{i,j} = \sqrt{\left(\delta_+^x u_{i,j}\right)^2 + \left(\delta_+^y u_{i,j}\right)^2 + \varepsilon},$$

where ε is a small positive parameter. Then we consider the following discretization of the functional E_{CEN}:

$$\begin{aligned}
E(c_1, c_2, u) = &\sum_{i,j} |\nabla u|_{i,j}\\
&+ \lambda \sum_{i,j} \left((c_1 - \bar{u}_{i,j})^2 u_{i,j} + (c_2 - \bar{u}_{i,j})^2 (1 - u_{i,j}) \right),
\end{aligned}$$

where the indices i and j in the sum run from 0 to $m-1$ and from 0 to $n-1$, respectively.

Obviously, the function $E(c_1, c_2, u)$ is continuously differentiable. Its derivatives take the following form:

$$\begin{aligned}
&\frac{\partial E}{\partial c_1} = 2\lambda \sum_{i,j} (c_1 - \bar{u}_{i,j})\, u_{i,j},\\
&\frac{\partial E}{\partial c_2} = 2\lambda \sum_{i,j} (c_2 - \bar{u}_{i,j})\, (1 - u_{i,j}),\\
&\frac{\partial E}{\partial u_{i,j}} = a_0 u_{i,j}\\
&\qquad - a_1 u_{i+1,j} - a_2 u_{i-1,j} - a_3 u_{i,j+1} - a_4 u_{i,j-1}\\
&\qquad + \lambda \left((c_1 - \bar{u}_{i,j})^2 - (c_2 - \bar{u}_{i,j})^2 \right),
\end{aligned} \tag{7}$$

where

$$\begin{aligned}
&a_1 = \left(\sqrt{\left(\delta_+^x u_{i,j}\right)^2 + \left(\delta_+^y u_{i,j}\right)^2 + \varepsilon} \right)^{-1},\\
&a_2 = \left(\sqrt{\left(\delta_-^x u_{i,j}\right)^2 + \left(\delta_+^y u_{i-1,j}\right)^2 + \varepsilon} \right)^{-1},\\
&a_4 = \left(\sqrt{\left(\delta_+^x u_{i,j-1}\right)^2 + \left(\delta_-^y u_{i,j}\right)^2 + \varepsilon} \right)^{-1},\\
&a_3 = a_1, \qquad a_0 = a_1 + a_2 + a_3 + a_4.
\end{aligned} \tag{8}$$

By imposing $\partial E/\partial c_1 = \partial E/\partial c_2 = 0$ we trivially obtain the values of c_1 and c_2 that minimize E for any fixed u:

$$\begin{aligned}
&c_1 = \frac{\sum_{i,j} \bar{u}_{i,j} u_{i,j}}{\sum_{i,j} u_{i,j}},\\
&c_2 = \frac{\sum_{i,j} \bar{u}_{i,j} (1 - u_{i,j})}{\sum_{i,j} (1 - u_{i,j})}.
\end{aligned} \tag{9}$$

From (9) and $0 \le u_{i,j} \le 1$ it follows that

$$u_{min} \le c_1, c_2 \le u_{max},$$

where $u_{min} = \min_{i,j} \bar{u}_{i,j}$ and $u_{max} = \max_{i,j} \bar{u}_{i,j}$. Note also that (9) provides a discretization of (3) where $H(\phi(x))$ has been replaced by $u(x)$.

In the following we focus on the solution of the problem

$$\begin{aligned} &\min\ E(c_1, c_2, u), \\ &\text{s.t.}\ \ 0 \le u \le 1, \\ &\qquad u_{min} \le c_1, c_2 \le u_{max}, \end{aligned} \tag{10}$$

where $0 \le u \le 1$ is intended componentwise.

Henceforth, to simplify the notation, we denote by G the vector ∇E, by G_c the vector with components $\partial E/\partial c_1$ and $\partial E/\partial c_2$, and by G_u the vector with components $\partial E/\partial u_{i,j}$ (we implicitly assume some ordering in the grid of pixels). For any vectors v and w, we use the notation (v, w) to denote $(v^T, w^T)^T$; furthermore, we denote by $\langle \cdot, \cdot \rangle$ the Euclidean inner product and by $\| \cdot \|$ the induced vector norm.

3 An SPG-based alternating method for the discretized CEN model

We compute a solution to problem (10) by exploiting the SPG2 spectral projected gradient method proposed in [4] within an alternating minimization framework. Therefore, we first provide a short description of SPG2. Given a closed convex set $X \subset \mathbb{R}$ and a function $f : X \to \mathbb{R}$ that is continuously differentiable in an open set containing X, the SPG2 method attempts to compute a minimizer of f in X by building a sequence of approximate solutions as follows:

$$x^{k+1} = x^k + \theta^k d^k.$$

Here the search direction d^k is defined as

$$d^k = P_X\left(x^k - \alpha^k \nabla f(x^k)\right) - x^k,$$

where P_X denotes the orthogonal projection on X, α^k is obtained through the BB formula

$$\alpha^k_{BB} = \frac{\langle s^{k-1}, s^{k-1} \rangle}{\langle s^{k-1}, y^{k-1} \rangle}, \tag{11}$$

with $s^{k-1} = x^k - x^{k-1}$ and $y^{k-1} = \nabla f(x^k) - \nabla f(x^{k-1})$, and the steplength θ^k is such that the Grippo-Lampariello-Lucidi (GLL) condition [25] holds, i.e.,

$$f(x^{k+1}) \le \max_{0 \le j \le \min\{k, \nu-1\}} f(x^{k-j}) + \gamma\, \theta_k \langle \nabla f(x^k), d^k \rangle \tag{12}$$

with $\nu \in \mathbb{N}$ and $\gamma \in (0, 1)$. More precisely, given a "small" parameter $\alpha_{min} > 0$ and a "large" parameter $\alpha_{max} > 0$, the value of α^k is set as

$$\alpha^k = \begin{cases} \min\left\{\alpha_{max}, \max\left\{\alpha_{min}, \alpha^k_{BB}\right\}\right\}, & \text{if } \alpha^k_{BB} > 0, \\ \alpha_{max}, & \text{otherwise.} \end{cases} \tag{13}$$

A steplength θ^k satisfying (12) is computed by using an iterative procedure, starting from the trial value $\theta^k_0 = 1$. That is, if θ^k_{j-1} is such that (12) does not hold, then a new candidate steplength θ^k_j is computed by using a classical one-dimensional quadratic interpolation technique; furthermore, given the safeguard parameters σ_1 and σ_2, with $0 < \sigma_1 < \sigma_2 < 1$, if θ^k_j does not belong to $[\sigma_1,\ \sigma_2 \theta^k_{j-1}]$, then $\theta^k_j = \theta^k_{j-1}/2$.

It can be proved that the SPG2 algorithm is convergent in the sense that any limit point $\bar{x}$ of the sequence $\{x^k\}$ generated by it is a stationary point of f in X, or equivalently

$$\|P_X(\bar{x} - \nabla f(\bar{x})) - \bar{x}\| = 0.$$

Our alternating minimization method based on SPG2 has the following structure: given (u^k, c^k_1, c^k_2) such that (9) holds, we first compute u^{k+1} by applying an SPG step to $E(u, c^k_1, c^k_2)$ with (c^k_1, c^k_2) fixed, and then compute c^{k+1}_1 and c^{k+1}_2 by (9), using u^{k+1}. However, in the computation of u^{k+1} we require that

$$\begin{aligned} &E\left(c^k_1, c^k_2, u^{k+1}\right) \\ &\qquad \le \max_{0 \le j \le \min\{k, \nu-1\}} E\left(c^{k-j}_1, c^{k-j}_2, u^{k-j}\right) \\ &\qquad + \gamma\, \theta_k \left\langle G_u\left(c^k_1, c^k_2, u^k\right), d^k_u \right\rangle, \end{aligned} \tag{14}$$

where d^k_u is the SPG search direction, i.e., the GLL condition involves values of c_1 and c_2 different from the fixed ones c^k_1 and c^k_2 (unless $\nu = 1$). On the other hand, since $G_c\left(c^k_1, c^k_2, u^k\right) = 0$, each iteration of the alternating minimization procedure can be regarded as the application of a projected gradient step to the function E with respect to all variables, followed by an "improvement" of the approximate solution resulting from that step. Furthermore, letting $d^k = \left(d^k_u, c^{k+1}_1 - c^k_1, c^{k+1}_2 - c^k_2\right)$, the GLL condition holds with respect to all variables, i.e.,

$$\begin{aligned} &E\left(c^{k+1}_1, c^{k+1}_2, u^{k+1}\right) \\ &\qquad \le \max_{0 \le j \le \min\{k, \nu-1\}} E\left(c^{k-j}_1, c^{k-j}_2, u^{k-j}\right) \\ &\qquad + \gamma\, \theta_k \left\langle G\left(c^k_1, c^k_2, u^k\right), d^k \right\rangle, \end{aligned} \tag{15}$$

although $\left(c^{k+1}_1, c^{k+1}_2, u^{k+1}\right)$ is not obtained through a line search along d^k.

The SPG-based Alternating procedure described so far, henceforth referred to as SPG-A, is summarized in Algorithm 3.1, where $V = [u_{min}, u_{max}]^2 \times [0, 1]^{mn}$ and $W = [0, 1]^{mn}$. We note that, since $G_c\left(c^k_1, c^k_2, u^k\right) = 0$ for all k, the stopping condition reduces to

$$\left\|P_W\left(u^k - G_u\left(c^k_1, c^k_2, u^k\right)\right) - u^k\right\| < tol.$$

We also observe that Algorithm SPG-A differs from the inexact block coordinate descent methods presented

Algorithm 3.1 (SPG-A)

given $tol > 0,\ \alpha_{max} > \alpha_{min} > 0,\ \alpha^0 \in [\alpha_{min},\ \alpha_{max}],\ 0 < \sigma_1 < \sigma_2 < 1,\ \nu \in \mathbb{N},\ \gamma \in (0,1)$

choose u^0 and compute c_1^0 and c_2^0 through (9), using u^0

set $k = 0$

while $\left\| P_V\left(\left(c_1^k, c_2^k, u^k\right) - G\left(c_1^k, c_2^k, u^k\right)\right) - \left(c_1^k, c_2^k, u^k\right)\right\| \leq tol$ **do**

compute $d_u^k = P_W\left(u^k - \alpha^k G_u\left(c_1^k, c_2^k, u^k\right)\right) - u^k$

compute $u^{k+1} = u^k + \theta^k d_u^k$, where θ^k satisfying (14) is obtained by safeguarded quadratic interpolation

compute c_1^{k+1} and c_2^{k+1} through (9), using u^{k+1}

compute α^{k+1} according to (13) and (11), where $s^k = u^{k+1} - u^k$ and $y^k = G_u(c_1^{k+1}, c_2^{k+1}, u^{k+1}) - G_u(c_1^k, c_2^k, u^k)$

set $k = k + 1$

end while

in [7], because it does not satisfy conditions (3.5) in [7] and uses a nonmonotone line-search technique.

Now we prove that Algorithm SPG-A preserves the convergence properties of SPG2. To this end, we first observe that for all $(c_1, c_2, u) \in V$ and $t \in (0, \alpha_{max}]$,

$$\begin{aligned}\langle G(c_1,c_2,u), G^t(c_1,c_2,u)\rangle &\leq -\frac{1}{t}\|G^t(c_1,c_2,u)\|^2 \\ &\leq -\frac{1}{\alpha_{max}}\|G^t(c_1,c_2,u)\|^2,\end{aligned} \tag{16}$$

where the notation

$$G^t(c_1,c_2,u) = P_V\left((c_1,c_2,u) - t\,G(c_1,c_2,u)\right) - (c_1,c_2,u)$$

has been used for simplicity. This is a known result [4, Lemma 2.1]), which follows straightforwardly from the properties of the projection on a convex set (see, e.g., [3, Proposition 2.1.3, (b)]). In order to shorten the notation, we also define

$$G_u^t(c_1,c_2,u) = P_W\left(u - t\,G_u(c_1,c_2,u)\right) - u.$$

The convergence of Algorithm SPG-A is stated in the following theorem.

Theorem 1 *Algorithm SPG-A is well defined and any limit point of the sequence $\left\{\left(c_1^k, c_2^k, u^k\right)\right\}$ generated by it is a stationary point of E in V.*

Proof This proof closely follows the proof of Theorem 2.4 in [4], however we outline it for the sake of completeness.

If $\left(c_1^k, c_2^k, u^k\right)$ is not a stationary point for E in V, then, by (16), for all $t \in (0, \alpha_{max}]$ we have

$$\begin{aligned}\left\langle G\left(c_1^k, c_2^k, u^k\right),\ G^t\left(c_1^k, c_2^k, u^k\right)\right\rangle &\leq -\frac{1}{t}\left\|G^t\left(c_1^k, c_2^k, u^k\right)\right\|^2 \\ &\leq -\frac{1}{\alpha_{max}}\left\|G^t\left(c_1^k, c_2^k, u^k\right)\right\|^2 < 0.\end{aligned}$$

It follows that

$$\begin{aligned}&\left\langle G_u\left(c_1^k, c_2^k, u^k\right),\ d_u^k\right\rangle \\ &\qquad = \left\langle G_u\left(c_1^k, c_2^k, u^k\right),\ G_u^t\left(c_1^k, c_2^k, u^k\right)\right\rangle < 0,\end{aligned}$$

because $G_c\left(c_1^k, c_2^k, u^k\right) = 0$ for all k. Therefore, d_u^k is a descent direction for $E\left(c_1^k, c_2^k, u\right)$ in u^k, and a step-size θ^k such that (14) holds can be computed with a finite number of trials. Thus, Algorithm SPG-A is well defined.

Now we suppose that $(\bar{c}_1, \bar{c}_2, \bar{u}) \in V$ is a limit point of the sequence generated by Algorithm SPG-A. Without loss of generality, we can assume that $\left\{\left(c_1^k, c_2^k, u^k\right)\right\}$ converges to $(\bar{c}_1, \bar{c}_2, \bar{u})$. We distinguish two cases.

Case 1: $\inf \theta^k = 0$.

By contradiction we suppose that $(\bar{c}_1, \bar{c}_2, \bar{u})$ is not a stationary point, i.e., $G^\alpha(\bar{c}_1, \bar{c}_2, \bar{u}) \neq 0$ for all $\alpha > 0$. From $G_c\left(c_1^k, c_2^k, u^k\right) = 0$ it follows that $G_c(\bar{c}_1, \bar{c}_2, \bar{u}) = 0$ and hence $G_u^\alpha(\bar{c}_1, \bar{c}_2, \bar{u}) \neq 0$. Then, by reasoning as in the proof of Theorem 2.4 in [4], we find that there exists $\delta > 0$ such that for all sufficiently large k,

$$\left\langle G_u\left(c_1^k, c_2^k, u^k\right),\ \frac{G_u^\alpha\left(c_1^k, c_2^k, u^k\right)}{\left\|G_u^\alpha\left(c_1^k, c_2^k, u^k\right)\right\|}\right\rangle < -\frac{\delta}{2}, \quad \text{for all } \alpha \in [\alpha_{min}, \alpha_{max}]. \tag{17}$$

Since $\inf \theta^k = 0$, there exists $K \subseteq \mathbb{N}$ such that

$$\lim_{k \in K} \theta^k = 0.$$

Then, moving from the way θ^k is chosen to satisfy the GLL condition (14) and proceeding again as in [4], we find that for all sufficiently large $k \in K$, there exists ρ^k, with $0 < \sigma_1 \leq \rho^k \leq \sigma_2$, and $t^k \in [0, \theta^k/\rho^k]$ such that

$$\left\langle G_u\left(c_1^k, c_2^k, u^k + t^k d_u^k\right),\ d_u^k\right\rangle > \gamma\left\langle G_u\left(c_1^k, c_2^k, u^k\right),\ d_u^k\right\rangle.$$

Hence, by taking a subsequence $\left\{d_u^k\right\}_{k\in K_1\subseteq K}$ such that $d_u^k/\|d_u^k\|$ is convergent with limit $\bar{d}_u$, we get

$$\left\langle G_u(\bar{c}_1,\bar{c}_2,\bar{u}),\, \bar{d}_u\right\rangle = 0.$$

From $d_u^k = G_u^{\alpha^k}\left(c_1^k, c_2^k, u^k\right)$ it follows, by continuity, that for all sufficiently large $k \in K_1$,

$$\left\langle G_u\left(c_1^k, c_2^k, u^k\right),\, \frac{G_u^{\alpha}\left(c_1^k, c_2^k, u^k\right)}{\left\|G_u^{\alpha}\left(c_1^k, c_2^k, u^k\right)\right\|}\right\rangle > -\frac{\delta}{2},$$

which contradicts (17).

Case 2: $\inf\theta^k \geq \rho > 0$.

By contradiction, we suppose that $(\bar{c}_1,\bar{c}_2,\bar{u})$ is not a constrained stationary point. Then $\|G^{\alpha}(\bar{c}_1,\bar{c}_2,\bar{u})\| > 0$ for all $\alpha \in (0,\alpha_{max}]$ and hence there exists $\delta > 0$ such that $\|G^{\alpha}(\bar{c}_1,\bar{c}_2,\bar{u})\| \geq \delta$ for all $\alpha \in [\rho,\alpha_{max}]$.

Let $l(k)$ be an integer such that $k-\min\{k,\nu-1\} \leq l(k) \leq k$ and

$$E\left(c_1^{l(k)}, c_2^{l(k)}, u^{l(k)}\right) = \max_{0\leq j\leq \min\{k,\nu-1\}} E\left(c_1^{k-j}, c_2^{k-j}, u^{k-j}\right).$$

By reasoning as in the proof of Theorem 2.4 in [4], we find that $\left\{E\left(c_1^{l(k)}, c_2^{l(k)}, u^{l(k)}\right)\right\}$ is a nonincreasing sequence; furthermore, for k sufficiently large, we have

$$\left\|G^{\alpha}\left(c_1^{l(k)}, c_2^{l(k)}, u^{l(k)}\right)\right\| \geq \delta/2.$$

Then, by (15) and (16), we get

$$\begin{aligned} E\left(c_1^{l(k)}, c_2^{l(k)}, u^{l(k)}\right) &\leq E\left(c_1^{l(k)-1}, c_2^{l(k)-1}, u^{l(k)-1}\right) \\ &\quad -\frac{\gamma\rho}{\alpha_{max}}\left\|G^{\alpha^{l(k)-1}}\left(c_1^{l(k)-1}, c_2^{l(k)-1}, u^{l(k)-1}\right)\right\|^2 \\ &\leq E\left(c_1^{l(k)-1}, c_2^{l(k)-1}, u^{l(k)-1}\right) - \frac{\gamma\delta^2\rho}{4\alpha_{max}}. \end{aligned}$$

It follows that

$$\lim_{k\to\infty} E\left(c_1^{l(k)}, c_2^{l(k)}, u^{l(k)}\right) = -\infty,$$

which is a contradiction because, by continuity,

$$\lim_{k\to\infty} E\left(c_1^{l(k)}, c_2^{l(k)}, u^{l(k)}\right) = E\left(\bar{c}_1,\bar{c}_2,\bar{u}\right).$$

This completes the proof. □

It is worth noting that Algorithm SPG-A can be applied in a more general context than the one considered in this work. More precisely, let $X \times Y \subset \mathbb{R}^p \times \mathbb{R}^q$ be a closed convex set and $F(x,y)$ a real-valued function that is continuously differentiable in an open set containing $X \times Y$. If we assume that for all $y \in Y$ $\operatorname{argmin}_{x\in X} F(x,y)$ exists and can be exactly computed, then we can apply Algorithm SPG-A to $F(x,y)$, with x playing the role of (c_1,c_2) and y the role of u. Of course, the convergence of the algorithm still holds.

4 Computational experiments

We performed numerical experiments to evaluate the effectiveness of our approach. To this aim, we applied the Algorithm SPG-A to several gray-scale images with different sizes and features, which are widely used in the literature of image segmentation (see Figure 1). We considered both real scenes (a–f) and synthetic pictures (g–l), including noisy (b, k, l) or blurred (i) ones. The images have either uniform or nonuniform background; furthermore, they show a single object or multiple ones.

Algorithm SPG-A was implemented in C, exploiting the implementation of Algorithm SPG2 [5] available as a part of the TANGO project (`http://www.ime.usp.br/~egbirgin/tango/`). It was run under Matlab (R2011b, v. 7.13), through a MEX-interface, using the Image Processing Toolbox to read and display images. The C code was compiled by using gcc (v. 4.7.2) and the experiments were performed on an Intel Core i5 processor with clock frequency of 2.7 GHz and 8 GB of RAM.

The parameters in the objective function $E(c_1,c_2,u)$ were set as $\lambda = 1$ and $\epsilon = 10^{-6}$, and the threshold parameter in (6) as $\mu = 0.5$. Following the implementation of SPG2, the ℓ_∞ norm was used instead of the ℓ_2 norm in the SPG-A stopping criterion, with $tol = 10^{-6}$. A maximum number of iterations, $maxit = 1000$, and of function evaluations, $fmax = 10000$, was also considered. The values $\nu = 10$ and $\gamma = 10^{-4}$ were used in the GLL condition (14). The parameters α_{min}, α_{max}, σ_1 and $\sigma_2$were left equal to their default values in SPG2. The initial value of α was also chosen as in SPG2, i.e., $\alpha^0 = \min\{\alpha_{max}, \max\{\alpha_{min}, 1/\|G_u^1(c_1^0,c_2^0,u^0\|_\infty\}\}$.

For each test problem we used three choices of the initial function u_0:

- $\bar{u}_{[0,1]}$, the image to be segmented, with intensity values scaled between 0 and 1;
- u_{ws}, a small white square ($p \times q$ pixels, with $p \ll m$ and $q \ll n$) on black background;
- u_{bs}, a small black square($p \times q$ pixels, with $p \ll m$ and $q \ll n$) on white background.

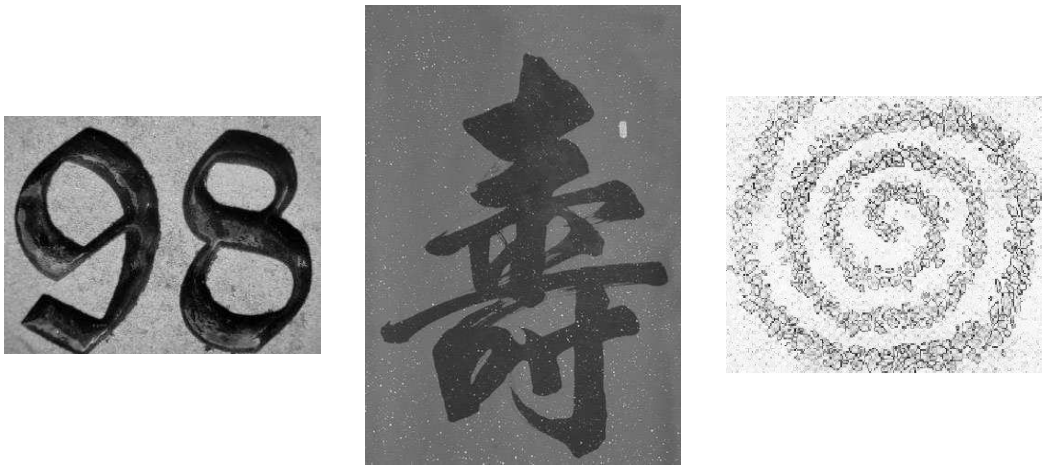
Since the three functions practically led to the same segmentations, we report here the results obtained with $u_0 = \bar{u}_{[0,1]}$.

For comparison purpose, we also applied to the CEN model the SPG2 algorithm and the SB-based algorithm described in [23], for all the test problems. The former was chosen to evaluate the behaviour of our approach with respect to a straightforward application of the spectral projected gradient algorithm; the latter is known to be very efficient on the segmentation problem considered in this work and therefore it was taken as a reference algorithm for our analysis. The SB approach

(a) cameraman – 204 × 204 (b) carplate – 285 × 224 (c) granite – 225 × 225

(d) squirrel – 230 × 167 (e) watercoins – 252 × 312 (f) zebras – 640 × 512

(g) blava – 228 × 266 (h) hoffman – 200 × 200 (i) circles – 320 × 240



(j) ninetyeight – 750 × 562 (k) shou – 344 × 500 (l) spiral – 329 × 289

Fig. 1 Test images: real scenes (a-f) and synthetic pictures (g-l). The number of pixels ($m \times n$) of each image is specified after the name. The images have been resized using different scalings to fit the figure layout.

computes an approximation u^{k+1} to u by performing the following steps:

$$(u^{k+1}, d^{k+1}) = \underset{0\le u\le 1;\, d}{\operatorname{argmin}} \int_\Omega \left(|d| + \lambda u r + \frac{\eta}{2} |d - \nabla u - b^k|^2 \right) dx, \quad (18)$$

$$b^{k+1} = b^k + \nabla u^k - d^k, \quad (19)$$

where $r = (c_1 - \bar{u})^2 - (c_2 - \bar{u})^2$. Problem (18) is approximately solved by an alternating minimization procedure, where u^{k+1} is obtained by applying the Gauss-Seidel method to the Euler-Lagrange equations (with $d = d^k$) and projecting the computed solution in $[0, 1]$, while d^{k+1} is obtained by applying the so-called shrink operator (see [23] for more details).

The above-mentioned implementation of SPG2 was used in the experiments, with the same values of the parameters as in SPG-A. Concerning the SB approach, the code available from `http://www.xavier-bresson.tk` was applied. This code was modified to employ double precision as in SPG-A and SPG2; furthermore, its native stopping criterion for the outer iterations was substituted by the stopping criterion used in SPG-A and SPG2, in order to make a fair comparison. The parameter η in (18) was set equal to 1; the stopping criterion for the Gauss-Seidel iterations was unchanged.

Details about the behaviour of the algorithms SPG-A, SPG2 and SB are given in Table 1. For all the algorithms we show the number of iterations, *nit*, performed to satisfy the stopping criterion. For SPG-A and SPG2 we also show the number of objective function evaluations, *nf*, each requiring $O(mn)$ floating-point operations, where $m \times n$ is the size of the image. The number of gradient evaluation is equal to the number of iterations, therefore, we do not report it (each gradient evaluation costs $O(mn)$ floting-point operations plus the computation of the square roots in (8)). For the SB algorithm we also show the number of Gauss-Seidel iterations, n_{GS}; note that each of these iterations has a computational cost comparable with the function evaluation performed in SPG-A and SPG2, i.e., $O(mn)$. As for the gradient evaluations in SPG-A and SPG2, we do not report the number of updates of d^{k+1} and b^{k+1}, since it is equal to n_{GS} (each update is $O(mn)$). We also do not report the objective function values at the computed solution, since they generally agree up to 7 significant digits (there are some exceptions for SPG-2). As a consequence, the resulting segmentations are practically the same. Therefore, we show only the segmentations obtained with SPG-A (see Figure 2).

We see that SPG-A is much faster than SPG2. SPG-A generally achieves the required accuracy in at most 7 iterations, with at most 15 objective function evaluations, while SPG2 requires a much larger number of iterations and objective function evaluations. This behaviour is more evident for images with particular texture, such as "granite", "zebras" or "ninetyeight", or corrupted by non-negligible noise, such as "carplate", "shou" and "spiral". The large increase in the number of SPG2 iterations because of noise is confirmed by further numerical experiments, performed adding noise to some images. For example, in the last two rows of Table 1 we report the results obtained after corrupting the real image "cameraman" and the synthetic one "ninethyeight" with Gaussian noise having zero mean and standard deviation values of 15 and 25, respectively (the corrupted images are shown in the left column of Figure 3). We see that the noise strongly deteriorates the performance of SPG-2, while it slightly affects the behaviour of SPG-A.

IMAGE	SPG-A		SPG2		SB	
	nit	*nf*	*nit*	*nf*	*nit*	n_{GS}
(a) cameraman	5	11	72	104	5	10
(b) carplate	3	7	118	209	4	8
(c) granite	7	15	242	381	7	14
(d) squirrel	4	9	60	76	4	8
(e) watercoin	3	7	46	57	3	6
(f) zebras	5	11	336	571	5	10
(g) blava	3	7	25	29	3	6
(h) hoffman	2	5	42	55	2	4
(i) circles	3	7	44	53	3	6
(j) ninetyeight	3	7	168	309	4	8
(k) shou	3	7	142	212	3	6
(l) spiral	6	13	454	788	7	15
noisy cameraman	7	15	283	527	8	16
noisy ninetyeight	5	11	—	—	5	10

Table 1 Numerical comparison of SPG-A, SPG2 and SB ("—" indicates that the required accuracy has not been achieved within the maximum number of iterations).

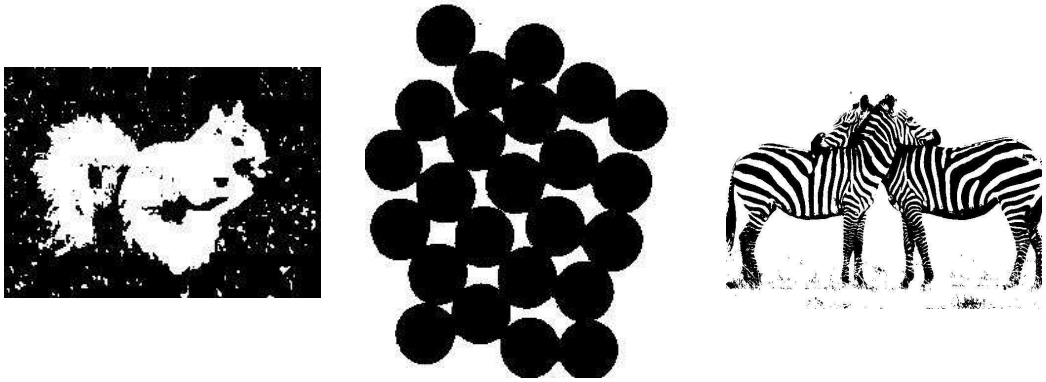
The SB-based algorithm shows about the same behaviour as SPG-A in terms of iterations and hence of computational cost. As previously observed, both methods produce the same segmentations of the images of the original test set. This is also true for the images obtained by adding Gaussian noise, whose segmentations are shown in the right column of Figure 3.

We conclude this section by observing that the quality of the segmentations produced by SPG-A is generally good, since the "meaningful" regions of each image are identified.

5 Conclusions

We presented an alternating minimization procedure called SPG-A, which exploits the nonmonotone SPG method for solving the formulation of segmentation problems provided by the CEN model. The choice of the SPG method was motivated by its steplength selection

(a) cameraman (b) carplate (c) granite

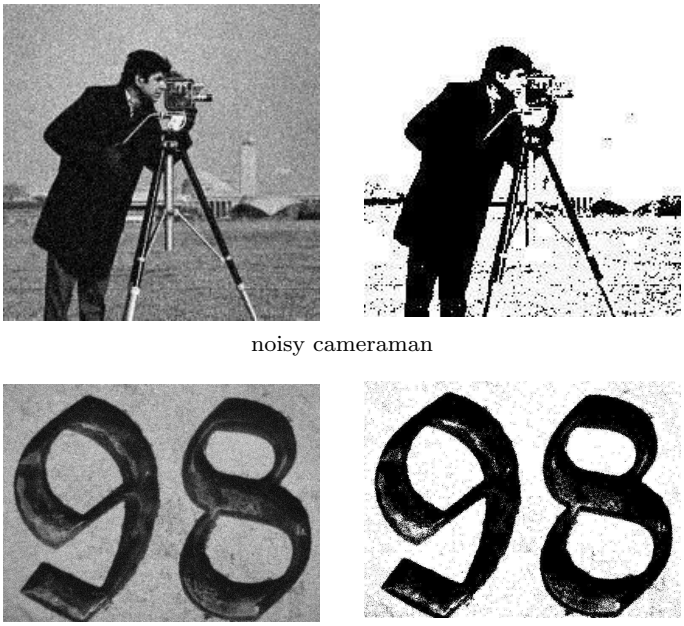


(d) squirrel (e) watercoins (f) zebras

(g) blava (h) hoffman (i) circles

(j) ninetyeight (k) shou (l) spiral

Fig. 2 Segmentations of the test images using SPG-A.



noisy cameraman

noisy ninetyeight

Fig. 3 Images corrupted by Gaussian noise and corresponding segmentations by SPG-A.

strategy, allowing faster convergence than classical gradient methods. We proved that the SPG-A algorithm is globally convergent to a constrained stationary point of the discretized CEN model. Numerical experiments on several images widely used in the literature of image segmentation show that SPG-A is competitive with the SB algorithm presented in [23], which is very effective on the segmentation model considered in this work.

Acknowledgements We wish to thank Giovanni Pisante for useful discussions concerning variational models for image segmentation.

References

1. Aujol, J.F., Chambolle, A.: Dual norms and image decomposition models. International Journal of Computer Vision **63**(1), 85–104 (2005)
2. Barzilai, J., Borwein, J.: Two-point step size gradient methods. IMA Journal of Numerical Analysis **8**, 141–148 (1988)
3. Bertsekas, D.: Nonlinear Programming, 2nd edn. Athena Scientific, Belmont, MA, USA (1999)
4. Birgin, E., Martínez, J., Raydan, M.: Nonmonotone spectral projected gradient methods on convex sets. SIAM Journal on Optimization **10**(4), 1196–1211 (2000)
5. Birgin, E., Martínez, J., Raydan, M.: Algorithm 813: SPG – software for convex-constrained optimization. ACM Transactions on Mathematical Software **27**(3), 340–349 (2001)
6. Birgin, E., Martínez, J., Raydan, M.: Spectral projected gradient methods: review and perspectives. Journal of Statistical Software **60**(3) (2014)
7. Bonettini, S.: Inexact block coordinate descent methods with application to non-negative matrix factorization. IMA Journal of Numerical Analysis **31**(4), 1431–1452 (2011)
8. Bonettini, S., Zanella, R., Zanni, L.: A scaled gradient projection method for constrained image deblurring. Inverse Problems **25**(1), 015,002 (25 pp.) (2009)
9. Bresson, X., Esedoḡlu, S., Vandergheynst, P., Thiran, J.P., Osher, S.: Fast global minimization of the active contour/snake model. Journal of Mathematical Imaging and Vision **28**(2), 151–167 (2007)
10. Chambolle, A.: An algorithm for total variation minimization and applications. Journal of Mathematical Imaging and Vision **20**(1–2), 89–97 (2004)
11. Chambolle, A., Darbon, J.: On total variation minimization and surface evolution using parametric maximum flows. International Journal of Computer Vision **84**(3), 288–307 (2009)
12. Chan, T., Esedoḡlu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM Journal on Applied Mathematics **66**(5), 1632—1648 (2006)
13. Chan, T., Sandberg, B., Moelich, M.: Some recent developments in variational image segmentation. In: X.C. Tai, K.A.. Lie, T. Chan, F. Osher (eds.) Image Processing based on Partial Differential Equations, Mathematics and Visualization Series, pp. 175–201. Springer, Heidelberg (2005)
14. Chan, T., Vese, L.: Active contours without edges. IEEE Transactions on Image Processing **10**(2), 266–277 (2001)
15. Dai, Y.H., Fletcher, R.: Projected Barzilai–Borwein methods for large–scale box–constrained quadratic programming. Numerische Mathematik **100**(1), 21–47 (2005)
16. Dai, Y.H., Yuan, Y.: Analysis of monotone gradient methods. Journal of Industrial and Management Optimization **1**(2), 181–192 (2005)

17. Darbon, J., Sigelle, M.: A fast and exact algorithm for total variation minimization. In: J. Marques, N. Pérez de la Blanca, P. Pina (eds.) Pattern Recognition and Image Analysis, *Lecture Notes in Computer Science*, vol. 3522, pp. 351–359 (2005)
18. De Asmundis, R., di Serafino, D., Hager, W., Toraldo, G., Zhang, H.: An efficient gradient method using the Yuan steplength. Computational Optimization and Applications **59**(3), 541—563 (2014)
19. De Asmundis, R., di Serafino, D., Riccio, F., Toraldo, G.: On spectral properties of steepest descent methods. IMA Journal of Numerical Analysis **33**(4), 1416–1435 (2013)
20. Esser, E.: Applications of Lagrangian–based alternating direction methods and connections to split Bregman. CAM Technical Report 09-31, UCLA, Los Angeles, CA, USA, available from ftp://ftp.math.ucla.edu/pub/camreport/cam09-31.pdf (2009)
21. Figueiredo, M., Nowak, R., Wright, S.: Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems. IEEE Journal of Selected Topics in Signal Processing **1**(4), 586–598 (2007)
22. Fletcher, R.: A limited memory steepest descent method. Mathematical Programming, Series A **135**(1–2), 413–436 (2012)
23. Goldstein, T., Bresson, X., Osher, S.: Geometric applications of the split Bregman method: segmentation and surface reconstruction. Journal of Scientific Computing **45**(1–3), 272–293 (2010)
24. Goldstein, T., Osher, S.: The split Bregman method for l1-regularized problems. SIAM Journal on Imaging Sciences **2**(2), 323–343 (2009)
25. Grippo, L., Lampariello, F., Lucidi, S.: A nonmonotone line search technique for Newton's method. SIAM Journal on Numerical Analysis **23**(4), 707–716 (1986)
26. Jung, M., Kang, M., Kang, M.: Variational image segmentation models involving non-smooth data-fidelity terms. Journal of Scientific Computing **59**(2), 277–308 (2014)
27. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. Communications on Pure and Applied Mathematics **45**(2), 577–685 (1989)
28. Setzer, S.: Operator splittings, Bregman methods and frame shrinkage in image processing. International Journal of Computer Vision **92**(3), 265–280 (2011)
29. Wang, Y., Yang, J., Yin, W., Zhang, Y.: A new alternating minimization algorithm for total variation image reconstruction. SIAM Journal on Imaging Sciences **1**(3), 248–272 (2008)
30. Yu, G., Qi, L., Dai, Y.H.: On nonmonotone Chambolle gradient projection algorithms for total variation image restoration. Journal of Mathematical Imaging and Vision **35**(2), 143–154 (2005)
31. Zhu, M., Wright, S., Chan, T.: Duality–based algorithms for total–variation–regularized image restoration. Computational Optimization and Applications **47**(3), 377–400 (2010)