

A spatially continuous max-flow and min-cut framework for binary labeling problems

Jing Yuan · Egil Bae · Xue-Cheng Tai ·
Yuri Boykov

Received: 10 October 2010 / Revised: 5 January 2013 / Published online: 16 June 2013
© Springer-Verlag Berlin Heidelberg 2013

Abstract We propose and investigate novel max-flow models in the spatially continuous setting, with or without *a priori* defined supervised constraints, under a comparative study of graph based max-flow/min-cut. We show that the continuous max-flow models correspond to their respective continuous min-cut models as primal and dual problems. In this respect, basic conceptions and terminologies from discrete max-flow/min-cut are revisited under a new variational perspective. We prove that the associated non-convex partitioning problems, unsupervised or supervised, can be solved globally and exactly via the proposed convex continuous max-flow and min-cut models. Moreover, we derive novel fast max-flow based algorithms whose convergence can be guaranteed by standard optimization theories. Experiments on image segmentation, both unsupervised and supervised, show that our continuous max-flow based algorithms outperform previous approaches in terms of efficiency and accuracy.

Mathematics Subject Classification 49M20 · 49M29 · 65K10 · 68U10

J. Yuan · Y. Boykov
Computer Science Department, Middlesex College, University of Western Ontario,
London, ONN6A 5B7, Canada
e-mail: cn.yuanjing@gmail.com

Y. Boykov
e-mail: yuri@csd.uwo.ca

E. Bae
Department of Mathematics, University of California, Los Angeles, CA, USA
e-mail: ebae@math.ucla.edu

X.-C. Tai (✉)
Department of Mathematics, University of Bergen, Bergen, Norway
e-mail: tai@math.uib.no

1 Introduction

Max-flow and min-cut are among the most fundamental pair of dual optimization problems defined over a graph. Such problems arise naturally in applications where one is interested in dividing a data set, represented by a set of nodes and edges, into two regions. Very efficient optimization algorithms are available for computing the maximal flow over a graph and consequently its minimal cut. In consequence, many optimization problems that can be viewed as a minimum cut problem can benefit from these algorithms. Perhaps the most prominent application area is image processing and computer vision, in which case the graph is typically a structured grid representing the pixels of the image. Many image processing and computer vision problems can be modeled in the form of energy minimization through Markov Random Fields (MRF) and solved numerically via min-cut and max-flow, see [38, 43] for good references. A long list of successful examples include image segmentation [2, 5, 10], stereo [31, 32], 3D reconstruction and shape-fitting [36, 37, 48], image synthesis and photomontage [1, 34], etc. There has been a vast amount of research on this topic during the last years, initiated partly by [8, 10]. Other discrete optimization methods include message passing [30, 49] and linear programming [33] etc. One main drawback of such graph-based approaches is the grid bias. The interaction potential penalizes some spatial directions more than other, which leads to visible artifacts in the computational results. Reducing such metrication errors can be done by considering more neighboring nodes [9, 29] or high-order interaction potentials [27, 28]. However, this results in a heavier memory load and higher computational cost. Recent studies [13] showed that formulating min-cut in the spatially continuous setting properly avoids metrication bias and leads to fast and global numerical solvers through convex optimization [11]. G. Strang [45, 46] was the first to formulate max-flow and min-cut problems over a continuous domain, with sources either within the domain, or on the boundary of the domain. Other related studies include Nozawa [41], which showed that for some special capacity functions there is a duality gap between the continuous max-flow and min-cut problems. In [2, 3], Appleton et al. proposed an edge-based continuous minimal surface approach to segmentation of 2D and 3D objects. A simultaneous work with ours [15], proposed a combinatorial optimization algorithm for solving a discretized version of the continuous max-flow problem.

In the variational community, there has been much interest in the problem of partitioning the domain Ω into two regions, a foreground region S and background region $\Omega \setminus S$, by minimizing

$$\min_S \int_{\Omega \setminus S} C_s(x) dx + \int_S C_t(x) dx + \alpha |\partial S|. \quad (1.1)$$

Here $C_t(x) \in \mathbb{R}$ is the cost of assigning x to foreground region and $C_s(x)$ is the cost of assigning x to the background region. The level set method [42] and piecewise constant variants [40, 39] are popular tools for solving the minimization problem numerically, but are not guaranteed to converge to a global minimizer. Chan et al. [13] showed that by relaxing the binary constraint set of the characteristic function $\lambda(x) \in \{0, 1\}$ of S

to the convex set $\lambda(x) \in [0, 1]$, the binary-constrained nonconvex problem (1.1) can be globally solved via the convex minimization problem

$$\min_{\lambda(x) \in [0, 1]} \int_{\Omega} (1 - \lambda(x)) C_s(x) dx + \int_{\Omega} \lambda(x) C_t(x) dx + \alpha \int_{\Omega} |\nabla \lambda| dx. \quad (1.2)$$

More specifically, solving (1.2) leads to a sequence of global binary optimizers by thresholding the solution $\lambda^*(x) \in [0, 1]$ at any value $t \in (0, 1]$. In case of non-uniqueness, a set of global binary solutions to the original nonconvex partition problem (1.1) can therefore be obtained. Traditional combinatorial max-flow algorithms do not offer such a flexibility and usually converge to one of the solutions. In this regard, (1.2) is also known as a *continuous min-cut model*. Fast numerical solvers for (1.2) were later developed through convex optimization in [11]. Recently, approach of Chan et al. was extended to more than two regions in [6, 35, 44], i.e. the continuous Potts model, although no simple thresholding scheme as above has been discovered for these relaxed models.

In the discrete setting, the duality between max-flow and min-cut have been exploited to design algorithms based on the max-flow [18, 14] formulations. Several very efficient algorithms, such as the Ford-Fulkerson algorithm [14], push-relabel algorithm [20], Dinitz blocking flow algorithm [16] are available to computing max-flow. In contrast, algorithms based on max-flow models over a continuous domain, as the dual formulation of (1.2), are still missing. For minimization problems involving total variation, like the ROF model [12], where the primal variable is unconstrained, dual formulations are also known and has been used to design fast algorithms [12]. However, if constraints like $\lambda \in [0, 1]$ are introduced, the dual formulation changes completely, as we will see. To tackle such constraints in research so far, algorithms which are designed for unconstrained total variation have been applied. They are modified such that the primal variable is forced to the feasible set every iteration, either by projections or by adding forcing terms [11, 13, 21]. Recently Bae et al. [6] studied the dual formulation of the continuous Potts problem with multiple labels, but not in the manner of maximizing flows. This motivates the contributions of this work. Moreover, we will also investigate the min-cut problem with a priori defined supervision constraints, by adapting the corresponding max-flow structures.

1.1 Contributions

In this paper we propose and study new max-flow models over a continuous domain, where a source and sink term are connected to every spatial point of the domain. Our work is a direct generalization of discrete max-flow models to the spatially continuous setting and consequently differs from Strang [45] in several respects: we introduce source and sink flows which are upper bounded by predefined capacities. This leads to new max-flow models, including the flow conservation constraints. We give a different proof of strong duality between the max-flow and min-cut models, where the indicator function of the cut acts as a Lagrange multiplier for the flow conservation constraint; We also consider max-flow and min-cut models with a priori defined supervision

constraints; Last, but not least, very efficient max-flow algorithms are constructed directly from the max-flow formulations based on the Augmented Lagrangian method.

We summarize our main contributions as follows:

First, we propose novel continuous max-flow models, which provide new equivalent representations of their respective continuous min-cut problems, unsupervised (1.2) or supervised (4.12), in terms of primal and dual.

Second, we revisit and give explanations of fundamental conceptions used in max-flow/min-cut, e.g. 'saturated'/'unsaturated' and 'cuts', through a new variational perspective. Via the equivalent max-flow formulation, we prove that the nonconvex partitioning problems, unsupervised (1.1) and supervised (to be introduced in Sec. (4.1)), can be solved exactly and globally.

Third, for the continuous min-cut model with supervised constraints, the proposed continuous max-flow formulation encodes the user-input constraints implicitly, and does not require to change flow capacities artificially as has been done previously.

Finally, new and fast max-flow based algorithms are proposed, which splits the optimization problem into simple subproblems over independent flow variables, where the labeling function λ works as a lagrange multiplier and can be updated at each iteration. Their convergence can be easily validated by classical optimization theories. The complexities of the supervised max-flow and min-cut algorithms are no worse than the unsupervised algorithms. Experiments show that our continuous max-flow algorithms significantly outperform previous continuous min-cut methods in terms of efficiency, e.g. [11]. They also outperform highly optimized graph cut implementations in terms of efficiency and accuracy (no metrication artifacts).

This work extends our preliminary conference paper [50]. More specifically, each section has been significantly elaborated, proofs of Proposition 3.2 and 4.1 have been included and significantly more experiments are performed to demonstrate the high efficiency of the new max-flow algorithms.

2 Revisit of discrete max-flow and min-cut

Many optimization problems can be formulated as max-flow/min-cut problems on appropriate graphs. This is particularly the case for many optimization problems arising in image processing and computer vision as first observed by Greig et. al. [23]. A graph \mathcal{G} is a pair $(\mathcal{V}, \mathcal{E})$ consisting of a vertex set \mathcal{V} and a directed edge set $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. The vertex set \mathcal{V} contains two distinguished vertices, the source $\{s\}$ and the sink $\{t\}$. A cost $C(e)$ is assigned to each edge $e \in \mathcal{E}$, which is assumed to be nonnegative i.e. $C(e) \geq 0$.

In this work, we focus especially on one type of graphs where the vertex set \mathcal{V} contains the nodes of a 2-D or N-D nested grid. The edge set is comprised of two types of edges: the spatial edges $e_n = (r, q)$, between pairwise neighboring nodes $r, q \in \mathcal{V} \setminus \{s, t\}$ on the grid; and the terminal edges $e_s = (s, r)$ and $e_t = (r, t)$, where $r \in \mathcal{V} \setminus \{s, t\}$, which link the source s and sink t to the grid node r . In case \mathcal{V} is a 2-D grid, such a graph is depicted Fig. 1 left. Such graphs have for instance become very popular for solving a wide range of problems in image processing and computer vision.

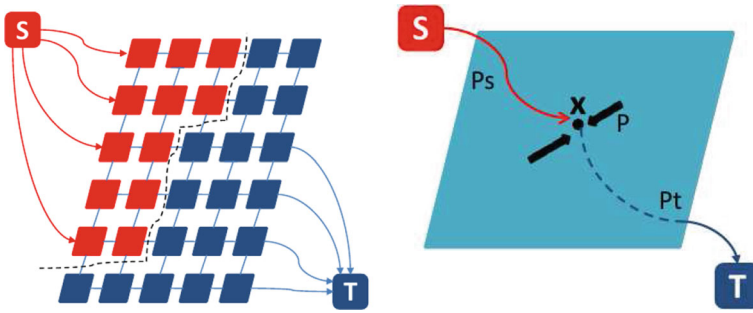


Fig. 1 Max-flow and min-cut in discrete setting (*left*) and continuous setting (*right*)

2.1 Min-cut

A cut on the graph \mathcal{G} is a division of the vertices into two disjoint groups V_s and V_t , such that: V_s contains the source $\{s\}$, V_t contains the sink $\{t\}$; for every vertex $v \in V_s$, there exists a path between s and v ; for every vertex $v \in V_t$, there exists a path between v and t ; i.e.

$$\mathcal{V} = \mathcal{V}_s \cup \mathcal{V}_t, \quad \mathcal{V}_s \cap \mathcal{V}_t = \emptyset.$$

The cost of the cut is defined as the sum of weights of edges with tail in V_s and head in V_t

$$C(V_s, V_t) = \sum_{(r,q) \in \mathcal{E}: r \in V_s, q \in V_t} C(e). \tag{2.1}$$

The min-cut problem is to find the cut (V_s, V_t) of minimum cost $C(V_s, V_t)$.

A cut on the type of graphs illustrated in Fig. 1 divides the spatial grid nodes $\mathcal{V} \setminus \{s\} \cup \{t\}$ into two disjoint groups: one is connected to the source s and the other is connected to the sink t (see Fig. 1 left). Therefore, the cut naturally forms an image segmentation into two regions, in case \mathcal{V} consists of the grid nodes of the image domain.

2.2 Max-flow

On the other hand, each edge $e \in \mathcal{E}$ can be viewed as a pipe and the edge cost $C(e)$ can be regarded as the capacity which bounds the maximal amount of flow $p(e)$ allowed on the pipe e , i.e. we get the constraint

$$0 \leq p(e) \leq C(e).$$

In addition, flow conservation is required at each vertex

$$\sum_{r \in \mathcal{V}: (r,q) \in \mathcal{E}} c(r, q) - \sum_{r \in \mathcal{V}: (q,r) \in \mathcal{E}} c(q, r) = 0 \quad \forall q \in \mathcal{V} \setminus \{s\} \cup \{t\}$$

The maximum flow problem is to find the largest amount of flow allowed to pass from the source s to the sink t . The total amount of flow from $\{s\}$ to $\{t\}$ can be expressed as the total amount of flow on the source edges, i.e. the problem can be expressed as

$$\max_p \sum_{v \in \mathcal{V}: (s,v) \in \mathcal{E}} p(s, v), \tag{2.2}$$

subject to the above flow constraints. It is well known that the max-flow problem (2.2) is equivalent to the min-cut problem (2.1). Edges with tail in V_s and head in V_t in the min-cut problem are saturated in the max-flow problem, i.e. the total amount of flow is bottlenecked by the 'saturated pipes'. With the graph-cut terminology, a flow $p(e)$ on the edge $e \in \mathcal{E}$ is said to be 'saturated' when it reaches the corresponding capacity $C(e)$; otherwise, it is called 'unsaturated'. We will revisit these conceptions under a variational perspective in the following sections.

In case of the graph structure we focus on in this work, and as depicted in Fig. 1 in 2-D, the constraints in the max-flow problem become

- Capacity of spatial flows p : for each $r, q \in \mathcal{V} \setminus \{s, t\}$, the directed spatial edges $(r, q) \in \mathcal{E}$ and $(q, r) \in \mathcal{E}$ have spatial flows $p(r, q)$ and $p(q, r)$ which are constrained by:

$$0 \leq p(r, q) \leq C(r, q), \quad 0 \leq p(q, r) \leq C(q, r). \tag{2.3}$$

If $C(q, r) = C(r, q) = \alpha$ for all $r, q \in \mathcal{V} \setminus \{s, t\}$, this corresponds to an anisotropic total-variation term. To simplify notation, one can define a single flow $\tilde{p}(r, q)$ for each edge pair $(r, q), (q, r)$ which is allowed to be negative, such that

$$\tilde{p}(r, q) = p(r, q) - p(q, r).$$

The flow capacity constraints (2.3) can then be merged in

$$-C(q, r) \leq \tilde{p}(r, q) \leq C(r, q).$$

The continuous generalization of spatial flow will be based on this simpler notation.

- Capacity of source flows p_s : for the edge $e_s(v) : s \rightarrow v$ linking the terminal s to a node $v \in \mathcal{V} \setminus \{s, t\}$, the source flow $p_s(v)$ is directed from s to v . Its capacity $C_s(v)$ indicates that

$$0 \leq p_s(v) \leq C_s(v); \tag{2.4}$$

- Capacity of sink flows p_t : for the edge $e_t(v) : v \rightarrow t$ linking a node $v \in \mathcal{V} \setminus \{s, t\}$ to the terminal t , $p_t(v)$ is directed from v to t . Its capacity $C_t(v)$ indicates that

$$0 \leq p_t(v) \leq C_t(v); \tag{2.5}$$

- Conservation of flows: at each node $v \in \mathcal{V} \setminus \{s, t\}$, incoming flows should be balanced by outgoing flows. In other words, all the flows passing through v , including spatial flows $p(e_n := (v, q))$ where $q \in N(v)$ is in the set of neighboring nodes of v , the source flow $p_s(v)$ and the sink flow $p_t(v)$, should be constrained by

$$\left(\sum_{q \in N(v)} \tilde{p}(q, v) \right) - p_s(v) + p_t(v) = 0. \tag{2.6}$$

3 Continuous max-flow and min-cut

In this section, we develop a direct continuous generalization of the max-flow model in Sect. 2.2 and show that it is dual to the min-cut problem (1.1).

3.1 Primal model: continuous max-flow

In the spatially continuous setting, let Ω be a closed spatial 2-D or N-D domain and s, t be the source and sink terminals, see Fig. 1 right. At each point $x \in \Omega$, we denote the spatial flow at x by $p(x)$; the directed source flow from s to x by $p_s(x)$; and the directed sink flow from x to t by $p_t(x)$. Now we consider the counterpart of the discrete max-flow problem (2.2) with constraints (2.3) (2.6) in the continuous limit as the number of grid nodes in \mathcal{V} goes to infinity. The max-flow model can be directly formulated in the same manner as in Sect. 2.

For each $x \in \Omega$ let $p_s(x) \in \mathbb{R}$ denote the flow from the source s to x and $p_t(x) \in \mathbb{R}$ denote the flow from x to the sink t . Define further the flow field $p \in C^\infty(\Omega)^N$ as the spatial flow within Ω , where N is the dimension of the domain Ω . In view of the flow constraints (2.3), (2.4), (2.5) and (2.6) in the discrete setting, the flows $p(x), p_s(x), p_t(x)$ are constrained by the capacities $C(x), C_s(x)$ and $C_t(x)$ as follows:

$$|p(x)| \leq C(x), \quad \forall x \in \Omega; \tag{3.1}$$

$$p_s(x) \leq C_s(x), \quad \forall x \in \Omega; \tag{3.2}$$

$$p_t(x) \leq C_t(x), \quad \forall x \in \Omega; \tag{3.3}$$

$$\operatorname{div} p(x) - p_s(x) + p_t(x) = 0, \quad \text{a.e. } x \in \Omega. \tag{3.4}$$

$$p \cdot n = 0, \quad \text{on } \partial\Omega \tag{3.5}$$

Constraint (3.4) is the continuous version of the flow capacity constraint. Here $\operatorname{div} p$ evaluates the total amount of incoming spatial flow locally at x , which is in analogy with the sum operator of (2.6) in the discrete setting. The notation a.e. stands for ‘‘for almost every’’. It means the constraint (3.4) should hold in the integrable, weak sense for every $x \in \Omega$, except possibly a subset of zero measure.

Here, the constraints on the source flow $p_s(x)$ (3.2) and the sink flow $p_t(x)$ (3.3) are changed a little in comparison to (2.4) and (2.5). This is because the flows $p_s(x)$ and $p_t(x)$ are not required to be positive as they are directed flows and their values

indicate how the flow is distributed from s to the point x or from x to t . Likewise, the capacities $C_s(x)$ and $C_t(x)$ are also not necessarily required to be positive.

In analogy with the discrete max-flow problem (2.2), the *continuous max-flow* model can be formulated as

$$\sup_{p_s, p_t, p} \left\{ P(p_s, p_t, p) = \int_{\Omega} p_s(x) dx \right\} \tag{3.6}$$

subject to the constraints (3.1), (3.2), (3.3) and (3.4). In this paper, we also call (3.6) the *primal model* and all flow variables p_s , p_t and p the *primal variables*.

3.2 Primal-dual model

By introducing the unconstrained lagrange multiplier function $\lambda : \Omega \mapsto \mathbb{R}$, also called the *dual variable*, to the linear equality of flow conservation (3.4), the continuous maximal flow model (3.6) can be formulated as its equivalent *primal-dual model*:

$$\begin{aligned} \inf_{\lambda} \sup_{p_s, p_t, p} & \left\{ E(p_s, p_t, p; \lambda) = \int_{\Omega} p_s(x) dx + \int_{\Omega} \lambda(x) (\operatorname{div} p - p_s + p_t) dx \right\} \\ \text{s.t.} & \quad p_s(x) \leq C_s(x), \quad p_t(x) \leq C_t(x), \quad |p(x)| \leq C(x) \quad \forall x \in \Omega. \end{aligned} \tag{3.7}$$

Rearranging the primal-dual formulation (3.7), we then get

$$\begin{aligned} \inf_{\lambda} \sup_{p_s, p_t, p} & \left\{ E(p_s, p_t, p; \lambda) = \int_{\Omega} \{(1 - \lambda)p_s + \lambda p_t + \lambda \operatorname{div} p\} dx \right\} \\ \text{s.t.} & \quad p_s(x) \leq C_s(x), \quad p_t(x) \leq C_t(x), \quad |p(x)| \leq C(x) \quad \forall x \in \Omega. \end{aligned} \tag{3.8}$$

Note that for the primal-dual model (3.8), the conditions of the minimax theorem (see e.g., [17] Chapter 6, Proposition 2.4) are all satisfied. That is, the constraints of flows are convex, and the energy function is linear in both the primal and dual functions $p_s(x)$, $p_t(x)$, $p(x)$ and $\lambda(x)$, hence convex l.s.c. for fixed λ and concave u.s.c. for fixed p_s , p_t and p . This also implies the existence of at least one saddle point, see [17]. Clearly, optimizing the primal-dual problem over the dual variable λ leads back to the primal max-flow model (3.6), i.e.

$$P(p_s, p_t, p) = \inf_{\lambda} E(p_s, p_t, p; \lambda).$$

3.3 Dual model: continuous min-cut

We show in this section that optimizing the primal-dual model (3.7) or (3.8) over the flow variables p_s, p_t and p leads to its equivalent dual model:

$$\min_{\lambda(x) \in [0,1]} \left\{ D(\lambda) = \int_{\Omega} \{ (1 - \lambda(x))C_s(x) + \lambda(x)C_t(x)dx + C(x) |\nabla\lambda| \} dx \right\}. \tag{3.9}$$

3.3.1 Optimization of flow variables

In order to optimize the flow variables of (3.8), let us first consider the following maximization problem

$$f(q) = \sup_{p \leq C} p \cdot q, \tag{3.10}$$

where p, q, C are scalars. When $q < 0$, p can be chosen to be negative infinity in order to maximize the value $p \cdot q$, which results in $f(q) = +\infty$. We further observe that

$$\begin{cases} \text{if } q = 0, \text{ then } p \leq C \text{ and } f(q) \text{ reaches maximum } 0 \\ \text{if } q > 0, \text{ then } p = C \text{ and } f(q) \text{ reaches maximum } q \cdot C \end{cases}. \tag{3.11}$$

Therefore, we can equivalently express $f(q)$ as

$$f(q) = \begin{cases} q \cdot C & \text{if } q \geq 0 \\ \infty & \text{if } q < 0. \end{cases} \tag{3.12}$$

Obviously, the function $f(q)$ given by (3.10) provides a prototype to maximize the primal-dual model (3.8) over the source flow $p_s(x)$ and sink flow $p_t(x)$. Define

$$f_s(x) = \sup_{p_s(x) \leq C_s(x)} (1 - \lambda(x)) \cdot p_s(x), \tag{3.13}$$

and

$$f_t(x) = \sup_{p_t(x) \leq C_t(x)} \lambda(x) \cdot p_t(x). \tag{3.14}$$

Then, by the discussion above, for each position $x \in \Omega$:

$$f_s(x) = \begin{cases} (1 - \lambda(x)) \cdot C_s(x) & \text{if } (1 - \lambda(x)) \geq 0 \\ \infty & \text{if } (1 - \lambda(x)) < 0 \end{cases} \tag{3.15}$$

and

$$f_t(x) = \begin{cases} \lambda(x) \cdot C_t(x) & \text{if } \lambda(x) \geq 0 \\ \infty & \text{if } \lambda(x) < 0. \end{cases} \quad (3.16)$$

For the maximization of (3.8) over the spatial flow $p(x)$, it is well known [19] that

$$\sup_{|p(x)| \leq C(x)} \int_{\Omega} \lambda \operatorname{div} p \, dx = \int_{\Omega} C |\nabla \lambda| \, dx. \quad (3.17)$$

By (3.15), (3.16) and (3.17), maximization of the primal-dual model (3.8) over flows p_s , p_t and p leads to its equivalent dual model (3.9). Observe that optimal λ must be contained in $[0, 1]$, otherwise the primal-dual energy would be infinite, contradicting the existence of at least one saddle point.

We summarize the above discussions by the following proposition:

Proposition 3.1 *The continuous max-flow model (3.6), the primal-dual model (3.7) or (3.8) and the dual model (3.9) are equivalent to each other.*

3.4 Global binary optimizers of the continuous min-cut

When $C(x)$ is constant over the whole domain Ω , e.g. $C(x) = \alpha$, the dual model (3.9) is reduced to

$$\min_{\lambda(x) \in [0,1]} \left\{ D(\lambda) = \int_{\Omega} \{ (1 - \lambda(x))C_s(x) + \lambda(x)C_t(x) + \alpha |\nabla \lambda| \} dx \right\} \quad (3.18)$$

which just coincides with the continuous min-cut model investigated by Chan et al. [13]. The function C satisfies $C(x) \geq 0$ for all $x \in \Omega$ and is assumed to be bounded and Borel measurable. For such spatial capacity functions there are no duality gap between the max-flow and min-cut models studied by Nozawa [41]. In image processing and computer vision C can for instance be an image edge detector, which takes small values at edges in the image and large values elsewhere. Such edge detector functions satisfy the above properties. In this case the model (3.9) becomes the geodesic segmentation model studied by Bresson et al. [11].

In this paper, we focus on the case that $C(x) = \alpha$ is constant for simplicity, and prove that there exists a series of binary optimums of (3.18) which are also globally optimal to the nonconvex min-cut problem (1.1) and can be obtained by thresholding. This is the same result that was shown by Chan et al. [13]. We demonstrate it in another way by duality through the continuous max-flow model (3.6). We show that every such minimal cut of (1.1) has the same energy as the maximum flow energy of (3.6). The results can easily be extended to the more general version of (3.9).

Proposition 3.2 *Let p_s^* , p_t^* , p^* and $\lambda^*(x)$ be a global optimizer of the primal-dual model (3.7) when $C(x) = \alpha$. Then each ℓ -upper level set $S^\ell := \{x \mid \lambda^*(x) \geq \ell, \ell \in$*

$(0, 1]$, $\ell \in (0, 1]$, of $\lambda^*(x)$ is a global minimizer of the nonconvex min-cut problem (1.1). The indicator function u^ℓ

$$u^\ell(x) := \begin{cases} 1, & \lambda^*(x) \geq \ell \\ 0, & \lambda^*(x) < \ell \end{cases},$$

is a global binary minimizer of (1.2).

Moreover, each cut energy given by S^ℓ has the same energy as its optimal max-flow energy, i.e.

$$P(p_s^*, p_t^*, p^*) = \int_{\Omega} p_s^*(x) dx = D(u^\ell).$$

Proof Let p_s^*, p_t^*, p^* and $\lambda^*(x)$ be an optimal primal-dual pair of (3.7), then p_s^*, p_t^*, p^* optimize the max-flow problem (3.6) and $\lambda^*(x)$ optimizes the dual problem (3.18). Clearly, the maximal flow energy of (3.6) is

$$P(p_s^*, p_t^*, p^*) = \int_{\Omega} p_s^*(x) dx \tag{3.19}$$

and satisfies

$$P(p_s^*, p_t^*, p^*) = E(p_s^*, p_t^*, p^*; \lambda^*) = D(\lambda^*).$$

For the max-flow problem (3.6), the flow conservation condition (3.4) is satisfied, i.e.

$$\operatorname{div} p^*(x) - p_s^*(x) + p_t^*(x) = 0, \quad \text{a.e. } x \in \Omega \tag{3.20}$$

Let S^ℓ be any level set of λ^* and $\ell \in (0, 1]$ and u^ℓ be its indicator function. In view of (3.11), for any point $x \in \Omega \setminus S^\ell$, i.e. where $\lambda(x) < \ell \leq 1$, it is easy to see that

$$p_s^*(x) = C_s(x), \quad \forall x \in \Omega. \tag{3.21}$$

Likewise, for any point $x \in S^\ell$, i.e. $\lambda(x) \geq \ell > 0$, we have

$$p_t^*(x) = C_t(x), \quad \forall x \in \Omega.$$

Then by (3.20), we have

$$p_s^*(x) = C_t(x) + \operatorname{div} p^*(x), \quad x \in S^\ell, \quad \text{a.e. } x \in \Omega \tag{3.22}$$

Therefore, by (3.21) and (3.22), the total energy defined in (3.19), for each level set S^ℓ , is

$$\begin{aligned}
 P(p_s^*, p_t^*, p^*) &= \int_{\Omega \setminus S^\ell} C_s(x) dx + \int_{S^\ell} (C_t(x) + \operatorname{div} p^*(x)) dx \\
 &= \int_{\Omega \setminus S^\ell} C_s(x) dx + \int_{S^\ell} C_t(x) dx + \int_{S^\ell} \operatorname{div} p^*(x) dx \\
 &= \int_{\Omega \setminus S^\ell} C_s(x) dx + \int_{S^\ell} C_t(x) dx + \alpha \left| \partial S^\ell \setminus (\partial S^\ell \cap \partial \Omega) \right|.
 \end{aligned}$$

The last term follows from the fact that $p_n^*(x) = \alpha \forall x \in \partial S^\ell \setminus (\partial S^\ell \cap \partial \Omega)$ and the Gaussian theorem

$$\int_{S^\ell} \operatorname{div} p^*(x) dx = \int_{\partial S^\ell} p_n^*(x) dl = \alpha \left| \partial S^\ell \setminus (\partial S^\ell \cap \partial \Omega) \right|. \tag{3.23}$$

It can also be verified by Prop. 4 of [6].

Therefore, the binary function u^ℓ , which is the indicator function of S^ℓ , solves the nonconvex min-cut problem (1.1) globally. This can be seen by the following facts: u^ℓ is obviously contained in the relaxed convex set $[0, 1]$ and its energy $P(p_s^*, p_t^*, p^*)$ is globally optimal to both the convex relaxed models (3.6) and (3.18). \square

In other words, the continuous max-flow formulation (3.6) implicitly leads to a segmentation of Ω with minimal length, i.e. the continuous min-cut given by the optimal multiplier function $\lambda^*(x)$. By solving the continuous max-flow model (3.6) one obtain a globally optimal solution to the partition problem (1.1). In Sec. 5.1, an algorithm is proposed for solving the continuous max-flow problem.

3.4.1 'Saturated'/'unsaturated' flows and cuts

By analyzing the new max-flow model, we can give a variational perspective to the connections between flows and cuts and also recover concepts such as 'saturated' and 'unsaturated' edges, which are used in the discrete max-flow/min-cut setting.

Let p^* be an optimizer of (3.10). By means of variations, if $p^* < C$ strictly, its variation δp can be both positive and negative. Observe that if $p^* + \delta p$ doesn't increase the value $f(q)$ for any δp , it directly follows that $q = 0$. On the other hand, for $p^* = C$, variations δp under the constraint must satisfy $\delta p < 0$. Again, any $p^* + \delta p$ doesn't increase the value $f(q)$, hence it follows that $q \geq 0$. In other words, if the flow $p^* < C$ does not reach its maximum capacity, then $q = 0$ and $f(q) = 0$ and hence there is no contribution to the total energy. We say the corresponding edge is 'unsaturated' and is therefore not part of the 'minimal cut'. These observations allow us to explain the relationships between flows and cuts in the spatially continuous setting. In the

following it is assumed that p_s^*, p_t^*, p^* and $\lambda^*(x)$ be an optimal primal-dual pair of (3.7).

Source flows, sink flows and cuts: observe from (3.2) that if the source flow $p_s^*(x) < C_s(x)$ at $x \in \Omega$ is 'unsaturated', we must have $1 - \lambda^*(x) = 0$, i.e.

$$p_s^*(x) < C_s(x) \implies \lambda^*(x) = 1.$$

At the position x , it is definitely labeled as 1. In addition, $f_s(x) = (1 - \lambda^*(x))p_s^*(x) = 0$, which means that at the position x , the source flow $p_s^*(x)$ has no contribution to the cut energy. It follows that $p_t^*(x) = C_t(x)$ is saturated and the minimal cut passes through the edge from x to the sink t .

Likewise, if the sink flow $p_t^*(x) < C_t(x)$ is 'unsaturated', we must have $\lambda^*(x) = 0$, i.e.

$$p_t^*(x) < C_t(x) \implies \lambda^*(x) = 0.$$

At the position x , it is labeled as 0. In addition, $f_t(x) = \lambda^*(x)p_t^*(x) = 0$, which means that at the position x , the sink flow $p_t^*(x)$ has no contribution to the cut energy. Hence, $p_s^*(x) = C_s(x)$ is saturated and the minimal cut passes through the edge from the source s to x .

As we see, only 'saturated' source and sink flows have contributions to the total energy.

Spatial flows and cuts: for the spatial flows $p^*(x)$, let

$$C_{TV}^\alpha := \{p \in C^\infty(\Omega)^N \mid \|p\|_\infty \leq \alpha, p_n|_{\partial\Omega} = 0\}.$$

Observe that

$$\sup_{p \in C_{TV}^\alpha} \langle \operatorname{div} p, \lambda \rangle = \sup_{p \in C_{TV}^\alpha} \langle p, \nabla \lambda \rangle, \tag{3.24}$$

where the inner product $\langle a, b \rangle$ is $\int_\Omega a(x)b(x)dx$. The extremum of the inner product $\langle p^*, \nabla \lambda^* \rangle$ in (3.24) is just the normal cone-based condition [24] of $\nabla \lambda^*$, i.e.

$$\nabla \lambda^* \in N_{C_{TV}^\alpha}(p^*). \tag{3.25}$$

Then we simply have:

$$\text{if } \nabla \lambda^*(x) \neq 0, \quad \text{then } |p^*(x)| = \alpha, \tag{3.26a}$$

$$\text{if } |p^*(x)| < \alpha, \quad \text{then } \nabla \lambda^*(x) = 0. \tag{3.26b}$$

In other words, at potential cut locations $x \in \Omega$ where $\nabla \lambda^*(x) \neq 0$ the spatial flow $p^*(x)$ is 'saturated'. At locations $x \in \Omega$ where $|p(x)| < \alpha$ is not saturated, we must have $\nabla \lambda^*(x) = 0$ and therefore the cut does not sever the spatial domain at x .

4 Supervised continuous max-flow and min-cut

In this section, we study continuous max-flow and min-cut models with priori given supervision constraints.

In contrast to the continuous max-flow and min-cut introduced above, the supervised max-flow/min-cut computes the optimal partition subject to given constraints on the region configurations, e.g. some grid nodes are labeled in advance as belonging to either the foreground or background. This gives a supervised partition problem which can be modeled as the following *supervised continuous min-cut problem*

$$\begin{aligned} \min_S \int_{S \setminus \Omega_f} C_s(x) dx + \int_{(\Omega \setminus \Omega_b) \setminus S} C_t(x) dx + \alpha |\partial S| \\ \text{s.t. } \Omega_f \subset S \subset \Omega \setminus \Omega_b, \end{aligned} \tag{4.1}$$

where $\Omega_f, \Omega_b \subset \Omega$ are the two disjoint areas marked a priori by the user: Ω_f belongs to the foreground and Ω_b belongs to the background.

The supervised continuous min-cut formulation can equivalently be written in terms of the binary characteristic function $\lambda(x) \in \{0, 1\}$ as:

$$\min_{\lambda(x) \in \{0,1\}} \int_{\Omega} (1 - \lambda(x)) C_s(x) dx + \int_{\Omega} \lambda(x) C_t(x) dx + \alpha \int_{\Omega} |\nabla \lambda(x)| dx. \tag{4.2}$$

subject to the labeling constraints

$$\lambda(\Omega_f) = 1, \quad \lambda(\Omega_b) = 0. \tag{4.3}$$

Considering the derivations in Sect. 3, we could simply set

$$C_s(\Omega_f) = +\infty, \quad C_t(\Omega_b) = +\infty. \tag{4.4}$$

These constraints say that the source flow $p_s(x)$ is unconstrained at $x \in \Omega_f$ and the sink flow $p_t(x)$ is unconstrained at $x \in \Omega_b$. In view of discussions of Sect. 3.3.1, the labeling constraints (4.3) would then follow. As in [8], this provides a direct way to couple the max-flow approach to the min-cut problem with supervised constraints (4.3).

In this work, we also propose new supervised max-flow and min-cut models without the artificial flow constraints (4.4), which implicitly encode the supervised information (4.3) and have the same complexity as the unsupervised formulations: (3.6) and (3.9). It is also flexible in case the supervised information is not given in an absolute sense as in (4.3): for example the marked areas Ω_f and Ω_b may be provided in a 'soft' manner by probabilities:

$$\lambda(\Omega_f) = t_f \in (0, 1), \quad \lambda(\Omega_b) = t_b \in (0, 1) \tag{4.5}$$

where t_f and t_b are positive constants but less than 1. It is easy to see that modification of the flows manually by (4.4) does not work in this case.

To motivate the following approach, we first define two characteristic functions corresponding to the label constraints (4.3):

$$u_f(x) = \begin{cases} 1, & x \in \Omega_f \\ 0, & x \notin \Omega_f \end{cases}, \quad u_b(x) = \begin{cases} 0, & x \in \Omega_b \\ 1, & x \notin \Omega_b \end{cases}. \tag{4.6}$$

Observe that Ω_f and Ω_b are disjoint. It follows that that

$$u_f(\Omega_b) = 0, \quad u_b(\Omega_f) = 1. \tag{4.7}$$

For the 'soft' version of the constraints (4.5), we define

$$u_f(x) = \begin{cases} t_f, & x \in \Omega_f \\ 0, & x \notin \Omega_f \end{cases}, \quad u_b(x) = \begin{cases} 1 - t_b, & x \in \Omega_b \\ 1, & x \notin \Omega_b \end{cases}. \tag{4.8}$$

It can be seen that the functions $u_f(x)$ and $u_b(x)$ describe lower and upper bounds on the probability of labeling the location $x \in \Omega$ as foreground and background respectively. This is shown in more detail in Sect. 4.3.

In the following discussion, we focus on (4.3) to simplify the derivations. The results are easily extended to the case of (4.5).

4.1 Primal model: supervised max-flow

Consider the flow $p_s(x)$ from the source s to each spatial location $x \in \Omega$: when $x \in \Omega_b$, the flow should have no contribution to the energy as it passes through the known background location; otherwise, it should be valued in the usual way. Therefore, in view of (4.6), which implies that $u_b(\Omega_b) = 0$ and $u_b(\Omega \setminus \Omega_b) = 1$, the total amount of source flow p_s in Ω is given by $\int_{\Omega} u_b(x) p_s(x) dx$. Similarly, concerning the total cost of the flow $p_t(x)$ from each spatial location x to the sink t : When $x \in \Omega_f$, the sink flow contributes to the cost by $-p_t(x)$, where the negative sign means it reduces the cost; otherwise, the sink flow contributes nothing. In view of (4.6), which implies $u_f(\Omega_f) = 1$ and $u_f(\Omega \setminus \Omega_f) = 0$, we therefore set the total cost of p_t in Ω as $-\int_{\Omega} u_f(x) p_t(x) dx$.

In contrast to the *continuous max-flow* problem (3.6), we formulate the related *supervised max-flow model* as

$$\sup_{p_s, p_t, p} P_S(p_s, p_t, p) = \int_{\Omega} u_b(x) p_s(x) dx - \int_{\Omega} u_f(x) p_t(x) dx \tag{4.9}$$

subject to the same flow constraints (3.1), (3.2), (3.3) and (3.4) on p_s, p_t and p . Likewise, (4.9) is also called the primal model of the supervised max-flow/min-cut problem.

In the special case that no priori information about foreground and background is given, the two characteristic functions satisfies $u_f(x) = 0$ and $u_b(x) = 1$ for $\forall x \in \Omega$. Therefore, the supervised max-flow problem (4.9) coincides with the max-flow problem (3.6).

4.2 Supervised primal-dual model

In analogy with (3.7), we can construct the equivalent primal-dual formulation of (4.9) by introducing the multiplier function λ

$$\begin{aligned} \sup_{p_s, p_t, p} \inf_{\lambda} E_S(p_s, p_t, p; \lambda) &= \int_{\Omega} u_b(x)p_s(x)dx - \int_{\Omega} u_f(x)p_t(x)dx \\ &\quad + \int_{\Omega} \lambda(x)(\operatorname{div} p(x) - p_s(x) + p_t(x))dx \\ \text{s.t. } p_s(x) \leq C_s(x), p_t(x) \leq C_t(x), |p(x)| \leq C(x), \end{aligned} \tag{4.10}$$

which can equivalently be formulated as

$$\begin{aligned} \sup_{p_s, p_t, p} \inf_{\lambda} E_S(p_s, p_t, p; \lambda) &= \int_{\Omega} (u_b - \lambda)p_s dx + \int_{\Omega} (\lambda - u_f)p_t dx \\ &\quad + \int_{\Omega} \lambda(x) \operatorname{div} p(x) dx \\ \text{s.t. } p_s(x) \leq C_s(x), p_t(x) \leq C_t(x), |p(x)| \leq C(x). \end{aligned} \tag{4.11}$$

As in Sect. 3.2, there exists at least one optimal primal-dual saddle point since all properties of the mini-max theorem are satisfied.

4.3 Dual model: supervised min-cut

Maximizing all the flow functions p_s, p_t and p in $E_S(p_s, p_t, p; \lambda)$ of (4.11), in the same manner as (3.15), (3.16) and (3.17), leads to the equivalent dual model corresponding to (4.9), also called the *supervised min-cut model* in this paper:

$$\begin{aligned} \min_{\lambda} D_S(\lambda) &= \int_{\Omega} (u_b - \lambda)C_s dx + \int_{\Omega} (\lambda - u_f)C_t dx + \int_{\Omega} C(x) |\nabla \lambda(x)| dx \\ \text{s.t. } u_f(x) \leq \lambda(x) \leq u_b(x). \end{aligned} \tag{4.12}$$

In this paper, we focus on the case that $C(x) = \alpha, \forall x \in \Omega$, in which case (4.12) can be written as

$$\begin{aligned} \min_{\lambda} D_S(\lambda) &= \int_{\Omega} (u_b - \lambda)C_s dx + \int_{\Omega} (\lambda - u_f)C_t dx + \alpha \int_{\Omega} |\nabla\lambda(x)| dx \\ \text{s.t.} & u_f(x) \leq \lambda(x) \leq u_b(x); \end{aligned} \tag{4.13}$$

or, observing that u_b and u_f are given in advance, be shortened as

$$\begin{aligned} \min_{\lambda} D_S(\lambda) &= \int_{\Omega} \lambda(C_t - C_s) dx + \alpha \int_{\Omega} |\nabla\lambda(x)| dx \\ \text{s.t.} & u_f(x) \leq \lambda(x) \leq u_b(x). \end{aligned} \tag{4.14}$$

We see that (4.14) is just the convex relaxed model of the nonconvex supervised min-cut problem (4.2), where the subset ordering

$$\Omega_f \subset S \subset \Omega \setminus \Omega_b$$

in (4.1) is expressed by the inequality ordering

$$u_f(x) \leq \lambda(x) \leq u_b(x), \quad x \in \Omega$$

in (4.14).

Moreover, the inequality constraint of λ in (4.14), in view of (4.6) and (4.7) implies that

$$\lambda(\Omega_f) = 1, \quad \lambda(\Omega_b) = 0. \tag{4.15}$$

This coincides with the priori information that Ω_f is already labeled as foreground objects and Ω_b is labeled as the background. It follows that the inequality constraint of $\lambda(x)$ implicitly encodes the priori supervision information.

In the special case when no priori information about foreground and background is given, i.e. $u_f(x) = 0$ and $u_b(x) = 1 \forall x \in \Omega$, the supervised min-cut problem (4.13) is equivalent to the continuous min-cut problem (1.2).

4.4 Global binary supervised min-cuts

Now we prove that global optimums of the nonconvex supervised min-cut model (4.1) can also be obtained by each upper level set of the global optimum λ^* to its convex relaxed version (4.13) or (4.14), in a similar manner as Proposition 3.2.

Proposition 4.1 *Let p_s^*, p_t^*, p^* and $\lambda^*(x)$ be a global optimum of the primal-dual problem (4.10) with $C(x) = \alpha$. Then each ℓ - upper level set $S^\ell := \{x \mid \lambda(x) \geq \ell\}$ of*

$\lambda^*(x)$ where $\ell \in (0, 1]$ is a global minimizer of (4.1). Its indicator function u^ℓ :

$$u^\ell(x) := \begin{cases} 1, & \lambda^*(x) \geq \ell \\ 0, & \lambda^*(x) < \ell \end{cases},$$

is a global solution of the nonconvex supervised min-cut problem (4.2).

Moreover, each supervised cut given by S^ℓ has the same energy as the optimal supervised max-flow energy, i.e.

$$P_S(p_s^*, p_t^*, p^*) = \int_{\Omega} u_b(x)p_s^*(x)dx - \int_{\Omega} u_f(x)p_t^*(x)dx = D_S(u^\ell).$$

Proof Let p_s^*, p_t^*, p^* and $\lambda^*(x)$ be a global optimum of (4.10). Then p_s^*, p_t^*, p^* optimize the primal problem (4.9) and $\lambda^*(x)$ optimizes (4.13) or (4.14) at the same time. Meanwhile, the two energies are equal, i.e.

$$P_S(p_s^*, p_t^*, p^*) = E_S(p_s^*, p_t^*, p^*, \lambda^*) = D_S(\lambda^*).$$

By the definition of u_b and u_f in (4.6), the optimal energy of (4.9) is

$$\begin{aligned} P_S(p_s^*, p_t^*, p^*) &= \int_{\Omega} u_b(x)p_s^*(x)dx - \int_{\Omega} u_f(x)p_t^*(x)dx \\ &= \int_{\Omega \setminus \Omega_b} p_s^*(x)dx - \int_{\Omega_f} p_t^*(x).dx \end{aligned} \tag{4.16}$$

Concerning the supervised min-cut problem, (4.15) indicates that

$$\lambda^*(\Omega_f) = 1, \quad \lambda^*(\Omega_b) = 0. \tag{4.17}$$

Then each level set S^ℓ of λ^*

$$S^\ell := \{x \mid \lambda^*(x) \geq \ell\},$$

with $\ell \in (0, 1]$, contains Ω_f and excludes Ω_b , i.e. we have

$$\Omega_f \subset S^\ell \subset \Omega \setminus \Omega_b. \tag{4.18}$$

As $\lambda^*(x)$ is the optimal multiplier, we must have the flow conservation condition (3.4), i.e.

$$\operatorname{div} p^*(x) - p_s^*(x) + p_t^*(x) = 0, \quad \text{a.e. } x \in \Omega. \tag{4.19}$$

For any point $x \in S^\ell$, where $\lambda^*(x) \geq \ell$, we have by (4.17) that $\lambda^*(x) \geq u_f(x)$, and therefore

$$p_t^*(x) = C_t(x).$$

Then by (4.19), we have

$$p_s^*(x) = C_t(x) + \operatorname{div} p^*(x), \quad \text{a.e. } x \in S^\ell \setminus \Omega_f. \tag{4.20}$$

For any point $x \in (\Omega \setminus \Omega_b) \setminus S^\ell$ we have that $\lambda^*(x) < \ell$ and hence $\lambda^*(x) < u_b(x)$. Hence

$$p_s^*(x) = C_s(x). \tag{4.21}$$

Therefore, in view of (4.21) and (4.20), the total optimal energy (4.16) is

$$\begin{aligned} P_S(p_s^*, p_t^*, p^*) &= \int_{(\Omega \setminus \Omega_b) \setminus S^\ell} C_s(x) dx + \int_{S^\ell} (C_t(x) + \operatorname{div} p^*(x)) dx - \int_{\Omega_f} p^*(x) dx \\ &= \int_{(\Omega \setminus \Omega_b) \setminus S^\ell} C_s(x) dx + \int_{S^\ell \setminus \Omega_f} C_t(x) dx + \int_{S^\ell} \operatorname{div} p^*(x) dx \\ &= \int_{(\Omega \setminus \Omega_b) \setminus S^\ell} C_s(x) dx + \int_{S^\ell \setminus \Omega_f} C_t(x) dx + \alpha \left| \partial S^\ell \setminus (\partial S^\ell \cap \partial \Omega) \right|, \end{aligned}$$

which obviously gives a solution u^ℓ of the nonconvex supervised min-cut problem (4.1). The last term follows from the observation of (3.23).

The above binary solution u^ℓ is contained in the relaxed convex set $\lambda(x) \in [0, 1]$ and reaches the globally optimal energy E^* . It follows that such binary function is globally optimal. □

5 Algorithms

In this section, we propose new algorithms for the continuous min-cut models (1.2) and (4.14) based on their respective max-flow formulations (3.6) and (4.9). In this section it is assumed the variables p , p_s , p_t and operators div , ∇ , \int are discretized. In experiments we apply a mimetic spatial discretization [25, 26], but any other discretization may be applied in the algorithms. Note that after discretization, the max-flow min-cut duality established in Propositions 3.2 and 4.1 does not necessarily hold exactly. This is because the coarea formula

$$\int_{\Omega} |\nabla u|_2 = \int_{-\infty}^{\infty} \int_{\Omega} |\nabla u^\gamma|_2 dx d\gamma$$

does not hold exactly after discretization in case of isotropic total variation above (but does so for the anisotropic variant of total variation). Importantly, as the grid size goes to zero, the discretized total variation γ converges to the continuous total variation. Consequently the duality gap goes to zero as the mesh size becomes smaller.

5.1 Continuous max-flow based algorithm

The energy function of the equivalent primal-dual model (3.7) is just the Lagrangian function of (3.6). For such a linear equality constrained optimization problem, we derive our fast max-flow based algorithm by means of the augmented Lagrangian method [7], which introduces an approach to compute both the flows and labeling function simultaneously. To this end, in view of the Lagrangian function (3.7), we define the respective augmented Lagrangian function as

$$L_c(p_s, p_t, p, \lambda) := \int_{\Omega} p_s dx + \int_{\Omega} \lambda (\operatorname{div} p - p_s + p_t) dx - \frac{c}{2} \|\operatorname{div} p - p_s + p_t\|^2, \tag{5.1}$$

where $c > 0$. Algorithm 1 shows the details of the proposed continuous max-flow based algorithm, where λ is updated as a multiplier at each iteration. Algorithm 1 is an example of the alternating direction method of multipliers. Convergence can be validated by standard convex optimization theories.

The sub-minimization problem (5.2) can also be solved by one step of the following iterative procedure:

$$p^{k+1} = \Pi_{\alpha} \left(p^k + c \nabla (\operatorname{div} p^k - F^k) \right) \tag{5.3}$$

where Π_{α} is the euclidian projection onto the convex set $C_{\alpha} = \{q \mid \|q\|_{\infty} \leq \alpha\}$. This requires much less computational efforts.

5.2 Supervised continuous max-flow based algorithm

Now we propose the algorithm for the supervision-constrained min-cut problem (4.14) based on its equivalent continuous max-flow formulation (4.9). The equivalent primal-dual formulation of (4.10) is the Lagrangian function of (4.9). We define its respective augmented lagrangian function as

$$L_c(p_s, p_t, p, \lambda) = \int_{\Omega} u_b p_s dx - \int_{\Omega} u_f p_t dx + \int_{\Omega} \lambda (\operatorname{div} p - p_s + p_t) dx - \frac{c}{2} \|\operatorname{div} p - p_s + p_t\|^2.$$

where $c > 0$. The supervised continuous max-flow based algorithm is provided in Algorithm 2.

Algorithm 1 Multiplier-based maximal-flow algorithm

Set the starting values p_s^1, p_t^1, p^1 and λ^1 , let $k = 1$ and start k -th iteration, which includes the following steps, until convergence:

- Optimizing p by fixing other variables

$$\begin{aligned}
 p^{k+1} &:= \arg \max_{\|p\|_\infty \leq \alpha} L_c(p_s^k, p_t^k, p, \lambda^k). \\
 &= \arg \max_{\|p\|_\infty \leq \alpha} -\frac{c}{2} \left\| \operatorname{div} p(x) - F^k \right\|^2,
 \end{aligned}
 \tag{5.2}$$

where F^k is a fixed variable. This problem can either be solved iteratively by Chambolle’s projection algorithm [12], or approximately by one step of (5.3).

- Optimizing p_s by fixing other variables

$$\begin{aligned}
 p_s^{k+1} &:= \arg \max_{p_s(x) \leq C_s(x)} L_c(p_s, p_t^k, p^{k+1}, \lambda^k) \\
 &:= \arg \max_{p_s(x) \leq C_s(x)} \int_{\Omega} p_s dx - \frac{c}{2} \left\| p_s - G^k \right\|^2
 \end{aligned}$$

where G^k is a fixed variable and optimizing p_s can be easily computed at each $x \in \Omega$ pointwise;

- Optimizing p_t by fixing other variables

$$\begin{aligned}
 p_t^{k+1} &:= \arg \max_{p_t(x) \leq C_t(x)} L_c(p_s^{k+1}, p_t, p^{k+1}, \lambda^k) \\
 &:= \arg \max_{p_t(x) \leq C_t(x)} -\frac{c}{2} \left\| p_t - H^k \right\|^2,
 \end{aligned}$$

where H^k is a fixed variable and optimizing p_t can be simply solved by

$$p_t(x) = \min(H^k(x), C_t(x));$$

- Update λ by

$$\lambda^{k+1} = \lambda^k - c(\operatorname{div} p^{k+1} - p_s^{k+1} + p_t^{k+1});$$

- Set $k = k + 1$ and repeat.

6 Experiments

We present two types of experiments for the proposed continuous max-flow/min-cut models: unsupervised image segmentation and supervised image segmentation. We start by introducing some related algorithms which will be used for comparison purposes.

6.1 Related algorithms

Bresson et al. [11] extended Chan et al.’s convex relaxation for two region partition problems by applying a weighted total-variation term. They also proposed a fast algorithm for (1.2) based on an approximation of (1.2):

Algorithm 2 Multiplier-based supervised max-flow

Set some starting values p_s^1, p_t^1, p^1 and λ^1 , let $k = 1$ and start k -th iteration, which includes the following steps, until convergence:

- Optimize p by fixing other variables

$$p^{k+1} := \arg \max_{\|p\|_\infty \leq \alpha} L_c(p_s^k, p_t^k, p, \lambda^k)$$

$$:= \arg \max_{\|p\|_\infty \leq \alpha} -\frac{c}{2} \left\| \operatorname{div} p - F^k \right\|^2;$$

where F^k is some fixed variable and results in a projection algorithm [12] or the gradient decent project (5.3);

- Optimizing p_s by fixing other variables

$$p_s^{k+1} := \arg \max_{p_s(x) \leq C_s(x)} L_c(p_s, p_t^k, p^{k+1}, \lambda^k)$$

$$:= \arg \max_{p_s(x) \leq C_s(x)} \int_{\Omega} u_b p_s dx - \frac{c}{2} \left\| p_s - G^k \right\|^2,$$

where G^k is a fixed variable and the optimal p_s can easily be computed at each $x \in \Omega$ pointwise;

- Optimize p_t by fixing other variables

$$p_t^{k+1} := \arg \max_{p_t(x) \leq C_t(x)} L_c(p_s^{k+1}, p_t, p^{k+1}, \lambda^k)$$

$$:= \arg \max_{p_t(x) \leq C_t(x)} - \int_{\Omega} u_f p_t dx - \frac{c}{2} \left\| p_t - H^k \right\|^2,$$

where H^k is a fixed variable and the optimal p_t can also be simply solved pointwise;

- Update λ by

$$\lambda^{k+1} = \lambda^k - c(\operatorname{div} p^{k+1} - p_s^{k+1} + p_t^{k+1});$$

- Let $k = k + 1$ and repeat.

$$\min_{\lambda, \mu} \left\{ \alpha \int_{\Omega} |\nabla \lambda(x)| dx + \frac{1}{2\theta} \|\lambda - \mu\|^2 + \int_{\Omega} \mu(x)(C_t(x) - C_s(x)) dx + \beta P(\mu) \right\} \tag{6.1}$$

where $P(\mu) := \int_{\Omega} \max\{0, 2|\mu - 0.5| - 1\} dx$ is an exact penalty function which forces $\mu(x)$ to the interval $[0, 1]$ pointwise. Clearly, when $\theta > 0$ is chosen small enough, it is expected that $\lambda \simeq \mu$, hence (6.1) solves (1.2) given $\mu(x) \in [0, 1]$. To this end, the convex constrained optimization problem (1.2) is approximated by a relatively simple unconstrained optimization formulation (6.1).

In view of (6.1), the authors introduced a fast alternating-descent scheme which includes two inner steps concerning the two variables λ and μ within each outer iteration, i.e. at the k -th iteration,

- fix μ^k and solve

$$\lambda^{k+1} := \arg \min_{\lambda} \left\{ \alpha \int_{\Omega} |\nabla \lambda(x)| dx + \frac{1}{2\theta} \|\lambda(x) - \mu^k(x)\|^2 \right\}$$

which can be computed by the standard Chambolle’s projection algorithm [12];

- fix λ^{k+1} and solve

$$\mu^{k+1} := \arg \min_{\mu} \left\{ \frac{1}{2\theta} \|\mu(x) - \lambda^{k+1}\|^2 + \int_{\Omega} \mu(x)(C_t(x) - C_s(x)) dx + \beta P(\mu) \right\}$$

which can be simply solved in closed form by shrinkage (see Proposition 4 of [11]).

Concerning the discrete min-cut problem, a very efficient implementation of the augmented paths method was proposed in [8], which was specialized for the type of graphs encountered in image processing and computer vision. It has now become a standard implementation that has been used in a large number of research papers during the last decade.

6.2 Experiments on unsupervised image segmentation

For image segmentation without user inputs, we adopt piecewise constant functions as the image model: i.e. two grayvalues f_1 and f_2 are chosen a priori for clues to build the data terms:

$$C_s(x) = D(f(x) - f_1(x)), \quad C_t(x) = D(f(x) - f_2(x)),$$

where $D(\cdot)$ is some penalty function.

Figures 2 and 3 depict two experiments, where the result of the proposed continuous max-flow based method Algorithm 1 is shown together with Bresson et al. [11] for comparisons. For the experiment shown in Fig. 2, we have chosen $\alpha = 0.4$ and threshold value $\ell = 0.5$. We have used the data term $D(s) = |s|$ in Fig. 2 and $D(s) = |s|^2$ in Fig. 3. Our method converges to a result (see graphs at the second row of Fig. 2), which takes the value 0 or 1 nearly everywhere. This is in contrast to the result of the method by Bresson et al. (see graphs at the first row of Fig. 2). For the experiment shown in Fig. 3, we chose $\alpha = 0.4$ and threshold value $\ell = 0.02$. Both results look almost the same, but our method converges significantly faster than the algorithm of Bresson et al. [11].

At each iteration we evaluate the following convergence criterion:

$$\text{err}^k = \left\| \lambda^{k+1} - \lambda^k \right\| / \left\| \lambda^{k+1} \right\|.$$

In contrast to Bresson et al. [11], the proposed algorithm converges within 100 iterations (with accuracy below 1×10^{-4}). It greatly outperforms [11] in terms of

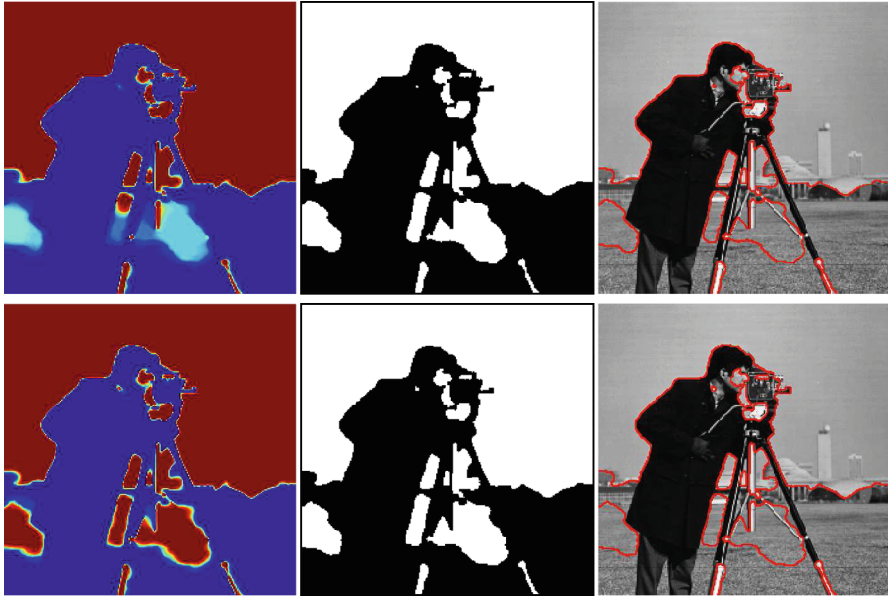


Fig. 2 In this experiment, we chose $\alpha = 0.4$ and $\ell = 0.5$. Graphs of the *first row* show the results by Bresson et al.: (*left*) computed $\lambda^*(x)$, (*middle*) thresholded $u^\ell(x)$, (*right*) segmented image. Graphs of the *second row* show the results by our method: (*left*) computed $\lambda^*(x)$, (*middle*) thresholded $u^\ell(x)$, (*right*) segmented image

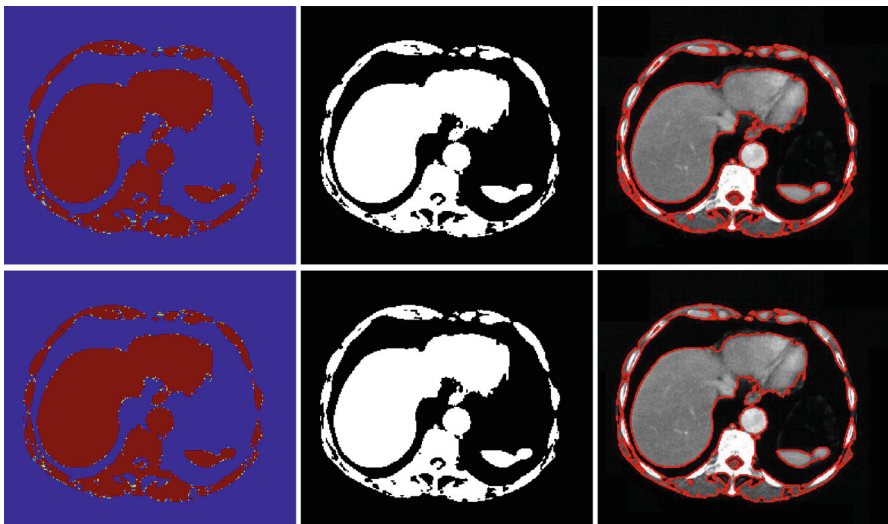


Fig. 3 In this experiment, we chose $\alpha = 0.02$ and $\ell = 0.5$. Graphs of the *first row* show the results by Bresson et al.: (*left*) computed $\lambda^*(x)$, (*middle*) thresholded $u^\ell(x)$, (*right*) segmented image. Graphs of the *second row* show the results by our method: (*left*) computed $\lambda^*(x)$, (*middle*) thresholded $u^\ell(x)$, (*right*) segmented image

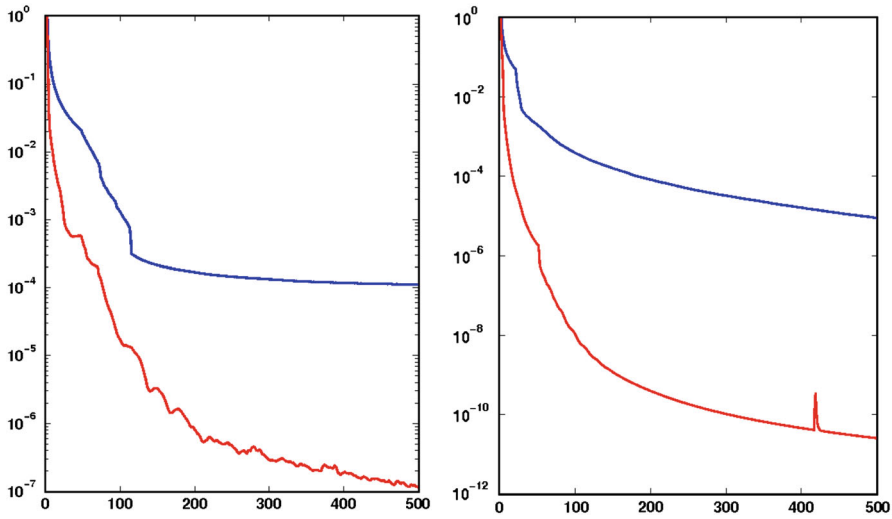


Fig. 4 Comparisons of convergence: for the experiments shown in Fig. 2 (left) and Fig. 3 (right), the method of Bresson et al. (blue line) converges much slower than the proposed continuous max-flow method (3.6) (red line) (color figure online)

Table 1 Time in seconds for proposed continuous max-flow algorithm vs discrete max-flow algorithm [8]

	Discrete max flow [8]	Continuous max-flow CPU	Continuous max-flow GPU
Figure 2	0.09	0.14	0.02
Figure 3	0.67	0.84	0.15

convergence rate, see Fig. 4: Bresson et al. (blue line) and ours (red line). In addition, our algorithm is reliable for a wide range of c .

Experimental comparison with discrete graph cut is provided in Table 1. We have used the highly optimized c++ implementation of the augmented paths method [8] specialized for imaging and vision applications. Our proposed continuous max-flow algorithm is implemented in C on both CPU and GPU. The computer used for experiments has an Intel Core i5-240M, 2.3 GHz CPU and NVIDIA GeForce GT 540M GPU. As stopping criteria, we have estimated the final energy E^* using a huge amount of iterations and terminated the algorithm when the relative energy precision at iteration k , $\frac{E^k - E^*}{E^*}$, falls below 10^{-3} . Note that, although the continuous algorithm will never terminate exactly, very little changes happen after a certain number of iterations. We are eventually interested in the binary thresholded solutions, and subsequent changes to the solution after $\frac{E^k - E^*}{E^*} < 10^{-3}$ have no impact on the final binarized result in our experience.

6.3 Supervised image segmentation

For supervised image segmentation, we use the Middlebury data set [47] for the experiments which are shown in Fig. 5. The corresponding data term, i.e. $C_s(x)$ and $C_t(x)$,



Fig. 5 *First row* The three given images, from the Middlebury data set, with pixels marked as foreground (*white*) and background (*black*). *Second row* computation result of λ^* to each image shown by color images, 0: blue and 1: red. *Third row* the black-white segmentation result by a threshold of λ^* . *Fourth and fifth rows* respective results computed from tree-reweighted message passing method [30,49] and α expansion algorithm [8,10]

is based on Gaussian mixture color models of foreground and background and are provided in advance. It is not required for us to put very large flow capacities artificially at the marked areas Ω_f and Ω_b as proposed in the supervised continuous max-flow method (4.9). This is in contrast to graph-based supervised image segmentation [10,30,49].

Here the tree-reweighted message passing method [30,49] and the α expansion method [8,10] are applied for comparisons. As we see, there are no visual artifacts, like metrication errors in our results (see details of the results, e.g. the left-bottom pedal of the flower (middle column)).

7 Conclusions and future topics

We have studied continuous max-flow and min-cut models, with or without supervised constraints. Dualities between the max-flow and min-cut models in the spatially continuous setting have been established and by variational techniques. Terminologies used by graph-cut based techniques have been revisited and explained under a new variational perspective. New optimization results on the exactness of the proposed convex models have been derived and discussed with aid of the continuous max-flow formulations. New continuous max-flow based algorithms have been proposed based on classical convex optimization theories, which provided fast and reliable numerical schemes. In contrast to discrete graph-based methods, the algorithms could be easily speeded up by highly parallel implementation on GPU.

The max-flow models can also be extended to other min-cut problems with multiple phases (see [4] and [51]). The continuous max-flow and min-cut models could also be defined over more general manifolds, which would correspond to more complex discrete graph structures. For instance, generalizations of the problem (1.2) with non-local operators could be formulated by such max-flow and min-cut models.

Recently, the Split-Bregman method [22] and some other efficient algorithm have been proposed for solving unconstrained total variation problems. The Split-Bregman method was also recently applied for solving the convex partition problem (1.2) by constraining the variable to the set $[0, 1]$ every iteration [21]. In a future work we will present a detailed experimental comparison with the Split-Bregman method and other recent algorithms for total variation minimization applied to the convex partition problem.

Acknowledgments This research has been supported by Natural Sciences and Engineering Research Council of Canada (NSERC) Accelerator Grant R3584A04, the Norwegian Research Council eVita project 166075 and eVita project 214889.

References

1. Agarwala, A., Dontcheva, M., Agrawala, M., Drucker, S., Colburn, A., Curless, B., Salesin, D., Cohen, M.: Interactive digital photomontage. *ACM Trans. Graph* **23**, 294–302 (2004)
2. Appleton, B., Talbot, H.: Globally optimal surfaces by continuous maximal flows. In: *DICTA*, pp 987–996 (2003)
3. Appleton, B., Talbot, H.: Globally minimal surfaces by continuous maximal flows. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(1), 106–118 (2006)
4. Bae, E., Yuan, J., Tai, X.-C., Boycov, Y.: A fast continuous max-flow approach to non-convex multi-labeling problems. Technical report CAM-10-62, UCLA (2010)
5. Bae, E., Tai, X.-C.: Efficient global minimization for the multiphase Chan-Vese model of image segmentation. In: *Energy Minimization Methods in Computer Vision and Pattern Recognition*, vol. 5681, pp. 28–41. *LNCS* (2009)

6. Bae, E., Yuan, J., Tai, X.-C.: Global minimization for continuous multiphase partitioning problems using a dual approach. *Int. J. Comput. Vision* **92**(1), 112–129 (2011)
7. Bertsekas, D.P.: *Nonlinear Programming*. Athena Scientific, Belmont (1999)
8. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**, 359–374 (2001)
9. Boykov, Y., Kolmogorov, V.: Computing geodesics and minimal surfaces via graph cuts. In: *ICCV*, pp 26–33 (2003)
10. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**, 1222–1239 (2001)
11. Bresson, X., Esedoglu, S., Vanderghenst, P., Thiran, J.-P., Osher, S.: Fast global minimization of the active contour/snake model. *J. Math. Imaging Vision* **28**(2), 151–167 (2007)
12. Chambolle, A.: An algorithm for total variation minimization and applications. *J. Math. Imaging Vision* **20**(1), 89–97 (2004)
13. Chan, T.F., Esedoglu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM J. Appl. Math.* 66(5):1632–1648 (electronic), (2006)
14. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: *Introduction to Algorithms*, 2nd edn. MIT Press, Cambridge (2001)
15. Couprie, C., Grady, L., Talbot, H., Najman, L.: Combinatorial continuous maximum flow. *SIAM J. Img. Sci* **4**(3), 905–930 (2011)
16. Dinitz, Y.: Dinitz’ algorithm: the original version and even’s version. *Theor. Comput. Sci.* **3859**, 218–240 (2006)
17. Ekeland, I., Těman, R.: *Convex analysis and variational problems*. Society for Industrial and Applied Mathematics, Philadelphia (1999)
18. Ford, L.R., Fulkerson, D.R.: *Flows in Networks*. Princeton University Press, Princeton (1962)
19. Giusti, E.: *Minimal surfaces and functions of bounded variation*. Australian National University, Canberra (1977)
20. Goldberg, A., Tarjan, R.E.: A new approach to the maximum-flow problem. *J. ACM* **35**(4), 921–940 (1988)
21. Goldstein, T., Bresson, X., Osher, S.: Geometric applications of the split bregman method: Segmentation and surface reconstruction. Technical report CAM09-06, UCLA, CAM (2009)
22. Goldstein, T., Osher, S.: The split bregman method for l1-regularized problems. *SIAM J. Imaging Sci* **2**(2), 323343 (2009)
23. Greig, D.M., Porteous, B.T., Seheult, A.H.: Exact maximum a posteriori estimation for binary images. *J. Royal Stat. Soc. Series B* **51**, 271–279 (1989)
24. Hiriart-Urruty, J.-B., Lemaréchal, C.: *Convex analysis and minimization algorithms*. I, volume 305 of *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Springer, Berlin (1993)
25. Hyman, J.M., Shashkov, M.J.: Adjoint operators for the natural discretizations of the divergence, gradient and curl on logically rectangular grids. *Appl. Numer. Math.* **25**(4), 413–442 (1997)
26. Hyman, J.M., Shashkov, M.J.: Natural discretizations for the divergence, gradient, and curl on logically rectangular grids. *Comput. Math. Appl.* **33**(4), 81–104 (1997)
27. Ishikawa, H.: Higher-order clique reduction in binary graph cut. In: *CVPR*, pp. 2993–3000, (2009)
28. Kohli, P., Pawan Kumar, M., Torr, P.H.S.: p^3 and beyond: Move making algorithms for solving higher order functions. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(9), 1645–1656 (2009)
29. Kolmogorov, V.: What metrics can be approximated by geo-cuts, or global optimization of length/area and flux. In: *ICCV*, pp. 564–571 (2005)
30. Kolmogorov, V.: Convergent tree-reweighted message passing for energy minimization. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(10), 1568–1583 (2006)
31. Kolmogorov, V., Zabih, R.: Multi-camera scene reconstruction via graph cuts. In: *European Conference on Computer Vision*, pp. 82–96 (2002)
32. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**, 65–81 (2004)
33. Komodakis, N., Tziritis, G.: Approximate labeling via graph-cuts based on linear programming. *Pattern Anal. Mach. Intell.* **29**, 1436–1453 (2007)
34. Kwatra, Vivek, Schoedl, Arno, Essa, Irfan, Turk, Greg, Bobick, Aaron: Graphcut textures: Image and video synthesis using graph cuts. *ACM Trans. Graphics SIGGRAPH* **22**(3), 277–286 (2003)

35. Lellmann, J., Kappes, J., Yuan, J., Becker, F., Schnörr, C.: Convex multi-class image labeling by simplex-constrained total variation. Technical report, HCI, IWR, Uni. Heidelberg, IWR, Uni. Heidelberg (2008)
36. Lempitsky, V., Boykov, Y.: Global optimization for shape fitting. In: CVPR (2007)
37. Lempitsky, Victor. S., Boykov, Y., Ivanov, D.V.: Oriented visibility for multiview reconstruction. In: ECCV'06, pp. 226–238 (2006)
38. Li, S.Z.: Markov random field modeling in image analysis. Springer, Secaucus (2001)
39. Lie, J., Lysaker, M., Tai, X.C.: A binary level set model and some applications to mumford-shah image segmentation. *IEEE Trans. Image Process.* **15**(5), 1171–1181 (2006)
40. Lie, J., Lysaker, M., Tai, X.C.: A variant of the level set method and applications to image segmentation. *Math. Comp.*, 75(255):1155–1174 (electronic), (2006)
41. Nozawa, R.: Examples of max-flow and min-cut problems with duality gaps in continuous networks. *Math. Program* **63**(2), 213–234 (1994)
42. Osher, S., Sethian, J.A.: Fronts propagating with curvature dependent speed: algorithms based on hamilton-jacobi formulations. *J. Comput. Phys.* **79**(1), 12–49 (1988)
43. Paragios, N., Chen, Y., Faugeras, O.: *Handbook Math Models Comput Vision*. Springer-Verlag New York, Inc., Secaucus (2005)
44. Pock, T., Chambolle, A., Bischof, H., Cremers, D.: A convex relaxation approach for computing minimal partitions. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 810–817, Miami (2009)
45. Strang, G.: Maximal flow through a domain. *Math. Programm.* **26**, 123–143 (1983)
46. Strang, G.: Maximum flows and minimum cuts in the plane. *Adv. Mech. Math.* **III**, 1–11 (2008)
47. Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Agarwala, A., Rother, C.: A comparative study of energy minimization methods for markov random fields. In *ECCV*, pp. 16–29 (2006)
48. Vogiatzis, G., Esteban, C.H., Torr, P.H., Cipolla, R.: Multi-view stereo via volumetric graph-cuts and occlusion robust photo-consistency. *PAMI* **29**(12), 2241–2246 (2007)
49. Wainwright, M., Jaakkola, T., Willsky, A.: Map estimation via agreement on (hyper)trees: Message-passing and linear programming approaches. *IEEE Trans. Inform. Theory* **51**, 3697–3717 (2002)
50. Yuan, J., Bae, E., Tai, X.C.: A study on continuous max-flow and min-cut approaches. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2217–2224. San Francisco (2010)
51. Yuan, Jing., Bae, Egil., Tai, Xue-Cheng., Boykov, Yuri.: A continuous max-flow approach to potts model. In: *European Conference on Computer Vision*, vol. 6316, pp. 379–392. LNCS (2010)