Index Terms

Partial differential equations, graph-theoretic methods, edge and feature detection

Globally minimal surfaces by continuous maximal flows

Ben Appleton, Hugues Talbot

Globally minimal surfaces by continuous maximal flows

Abstract

In this paper we address the computation of globally minimal curves and surfaces for image segmentation and stereo reconstruction. We present a solution, simulating a continuous maximal flow by a novel system of partial differential equations. Existing methods are either grid-biased (graph-based methods) or sub-optimal (active contours and surfaces).

The solution simulates the flow of an ideal fluid with isotropic velocity constraints. Velocity constraints are defined by a metric derived from image data. An auxiliary potential function is introduced to create a system of partial differential equations. It is proven that the algorithm produces a globally maximal continuous flow at convergence, and that the globally minimal surface may be obtained trivially from the auxiliary potential. The bias of minimal surface methods toward small objects is also addressed. An efficient implementation is given for the flow simulation.

The globally minimal surface algorithm is applied to segmentation in 2D and 3D as well as to stereo matching. Results in 2D agree with an existing minimal contour algorithm for planar images. Results in 3D segmentation and stereo matching demonstrate that the new algorithm is robust and free from grid bias.

I. Introduction

Geometric optimisation methods provide an exciting approach to solving image analysis problems. They have been applied with great success to image segmentation and to stereo reconstruction. They explicitly acknowledge the uncertainty commonly present in the extraction of geometric structures from images due to noise, occlusions and background clutter, and can in some cases obtain provably best estimates according to a measure of quality appropriate to the application.

Broadly speaking there are two classes of geometric optimisation techniques. One class is the active contours methods, including snakes [1], level sets [2], [3] and geodesic active contours and surfaces [4], [5]. Another class of methods taking a very different approach is the graph-based methods including shortest paths [6] and graph cuts [7].

Active contour methods model the evolution of a curve or surface toward a structure of interest in an image. They are usually based on a variational approach, performing a gradient descent flow to locally minimise an energy function whose minima ideally correspond to the objects of interest in the image. Unfortunately there are usually an extremely large number of local minima due to noise and irrelevant objects, and as a result active contours are highly dependent upon their initialisation. A wide array of heuristics have been proposed to assist in avoiding or overcoming these irrelevant minima, including pressure forces designed to overcome shallow minima [8], multiresolution approaches designed to focus on objects which persist at high scales, and methods which modify the gradient descent to favor more significant contours [9]. Despite the advent of these heuristics active contours typically require manual intervention which limits their application.

Graph-based methods are well known in image analysis and in stereo matching. [10] and [11] were among the first to propose stereo matching by shortest paths. Shortest paths remain competitive in current stereo research as they form the core of a number of minimal surface methods [12], [13]. Graph cuts have also been applied to 3D reconstruction, sacrificing speed for improved accuracy [14]. These methods are also used in image segmentation. [15] segmented cell nucleii using a polar trellis centered on the nucleus. They computed shortest paths using a Viterbi or dynamic programming approach. Graph-based methods may obtain optimal solutions to the associated minimisation problem. However their use is restricted in practice because they suffer from discretisation artifacts. These typically result in a preference for contours and surfaces to travel along the grid directions. See [16] for a good introduction to these methods.

Ideally geometric optimisation methods used in image analysis should be free of these problems, being both isotropic and optimal. In recent years several advances have been made in the extension of optimal methods from discrete graphs to continuous spaces. Dijkstra's classic shortest path algorithm [6] was extended in [17] and [18] to compute minimal geodesics and continuous distance functions. These have found broad application to optimal control, wave propagation and computer vision. The problem of continuous graph cuts has also received some attention. [19] described a method for approximating continuous minimal surfaces by a cut in a vertex weighted graph. [20] recently proposed a method for computing edge weights which approximate continuous graph cuts, toward the goal of computing globally minimal surfaces for segmentation and stereo vision.

In this paper we present an algorithm to compute globally minimal curves, surfaces and partitionings in arbitrary Riemannian spaces. Section II introduces Geodesic Active Contours and Surfaces. Section III introduces discrete weighted graphs and continuous Riemannian spaces along with a number of relationships between geometric optimisation problems. Section IV then presents an algorithm for obtaining continuous maximal flows in arbitrary Riemannian spaces with scalar metric. Also presented is a proof of correctness and an efficient implementation. Section V presents a solution to the inherent bias of minimal surfaces toward small objects. Section VI presents the results of the application of this new algorithm and Section VII concludes.

II. Geodesic Active Contours and Surfaces

Caselles et. al. introduced Geodesic Active Contours [4] and Geodesic Active Surfaces [5] for segmentation in 2D and 3D images. They are closed curves or surfaces which evolve to minimise their weighted length or area:

$$E\left[S\right] = \oint_S g(S) \mathrm{d}S. \tag{1}$$

$E[S]$ is often termed the *energy* of the surface S. In segmentation $g \geq \varepsilon > 0$ is a soft edge indicator function, tending toward zero where local image features suggest the presence of an object boundary. [4] also proposed the following form for the metric

$$g = \frac{1}{1 + \left|\nabla G_\sigma \star I\right|^p} + \varepsilon. \tag{2}$$

$\left|\nabla G_\sigma \star I\right|$ is the magnitude of the gradient at scale σ. It is usually raised to a power $p = 1$ or 2. ε is an arc length or surface area penalty which effectively regularises the minimal surface. They also demonstrated that all local minima are smooth surfaces for $\varepsilon > 0$.

Geodesic Active Contours and Surfaces evolve an initial surface via a gradient descent flow toward a local minima of the energy functional. We may derive the gradient descent flow by variational calculus, giving the Euler-Lagrange equation:

$$\frac{\partial S}{\partial \tau} = -\left(g\kappa - \nabla g \cdot \vec{N}\right)\vec{N}. \tag{3}$$

Here τ is the evolution parameter or *time*, $\vec{N} = \frac{\nabla \phi}{|\nabla \phi|}$ is the surface normal and $\kappa = \nabla \cdot \vec{N}$ the mean curvature.

The evolution of this surface may be implemented using a level set embedding due to [2]. For a function $\phi : I\!R^N \rightarrow I\!R$ whose zero level set is $S = \{\mathbf{x} \,|\phi(\mathbf{x}) = 0\}$, we may evolve ϕ so as to implement the gradient descent flow for S given in (3):

$$\frac{\partial \phi}{\partial \tau} = \nabla \cdot \left(g \frac{\nabla \phi}{|\nabla \phi|} \right) |\nabla \phi| \, .$$

A more efficient, implicit update scheme has also been presented in [21]. Unfortunately as we pointed out earlier these gradient descent flows usually converge to local minima with no guarantee on the quality of the resulting segmentation.

III. Weighted graphs and Riemannian spaces

A number of optimal methods have been proposed for computer vision based on discrete graphs [14], [15] and later continuous Riemannian spaces [22], [23]. Here we review the basic theory and definitions of these closely related frameworks.

A. Minimal Paths and Geodesics

A graph G is a pair (V, E) consisting of a vertex set V and an edge set $E \subseteq V \times V$. Vertices may be interpreted as points while edges are lines connecting pairs of points. A weighted graph includes vertex costs $C_V : V \rightarrow I\!R$ and edge costs $C_E : E \rightarrow I\!R$. In this paper we consider only positive cost functions.

A simple path P is defined as a sequence of unique vertices, while a cycle has equal endpoints so as to form a loop. The length L of a path P is the sum of vertex and edge costs along the path,

$$L\,[P] = \sum_{v \in P} C_V(v) + \sum_{e \in P} C_E(e).$$

The length of a cycle is defined analogously.

A path between two points s and t is a minimal or shortest path if there exists no connected path of lower length. Such paths may be computed using Dijkstra's shortest path algorithm [6], which first computes the distance of each vertex from s before backtracking from t to s.

A Riemannian space R is the continuous equivalent of a weighted graph. It consists of an N-manifold Ω and an associated metric $g : \Omega \rightarrow I\!R$. Here we consider only positive scalar

metrics $g \in I\!R^+$. A simple curve in a Riemannian space is a 1-manifold embedded in Ω which does not pass through itself. A curve C with parameter ξ in the range $[a, b]$ has length

$$L[C] = \int_a^b g(C(\xi)) \left| \frac{\partial C}{\partial \xi} \right| \mathrm{d}\xi.$$

A simple curve between two points s and t is a minimal geodesic if there exists no such curve of lower length. Minimal geodesics may be computed using the Fast Marching Method [18], which first computes a distance function from s by wavefront propagation before backtracking by gradient descent from t to s.

B. Minimal Cuts and Minimal Surfaces

A partitioning of a graph G decomposes its vertex set into a collection $\Gamma_G = \{V_1, V_2, \ldots\}$ of disjoint subsets:

$$\bigcup_{V_i \in \Gamma_G} V_i = V, \quad V_i \cap V_j = \emptyset \quad \text{for} \quad i \neq j$$

To each partition Γ_G we associate a cost $C(\Gamma_G)$ which is the total cost of the edges whose endpoints lie in different partitions.

$$C(\Gamma_G) = \sum_{e \in E^*} C_E(e)$$

Here the *cut* $E^* \subseteq E$ denotes the set of edges crossing the partition. The s-t minimal cut problem seeks the partitioning of minimal cost such that the disjoint vertex sets $s, t \subseteq V$ lie in different partitions. A good introduction to algorithms solving this problem is [16].

A partitioning of a Riemannian space R decomposes the space into a collection $\Gamma_R = \{\Omega_1, \Omega_2, \ldots\}$ of compact subsets whose pairwise intersection has zero Lebesgue measure:

$$\bigcup_{\Omega_i \in \Gamma_R} \Omega_i = \Omega, \quad \mathcal{L}(\Omega_i \cap \Omega_j) = 0 \quad \text{for} \quad i \neq j.$$

Similarly to the discrete case, to each partition Γ_R we associate a cost $C(\Gamma_R)$ which is the integral of the metric g over the partition surfaces $\partial\Omega_i$,

$$C(\Gamma_R) = \frac{1}{2} \sum_{\Omega_i \in \Gamma_R} \oint_{\partial\Omega_i} g \mathrm{d}(\partial\Omega_i).$$

The potentially confusing term $d(\partial\Omega_i)$ denotes an infinitesimal component of the partition surface $\partial\Omega_i$. Fig. 1 depicts a binary partitioning of the plane Ω. In this paper we will only consider binary partitionings.

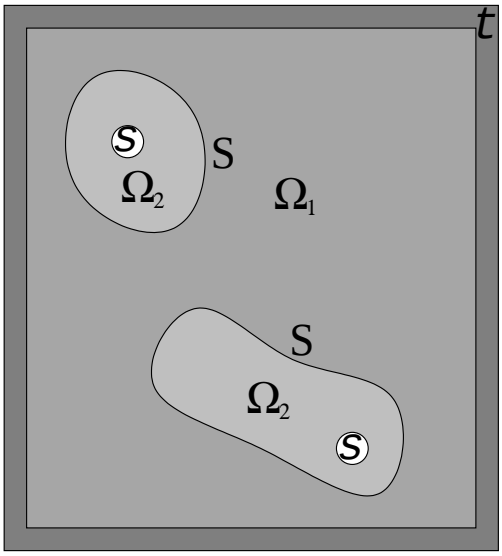


Fig. 1. A binary partitioning of the space Ω. Note that the seeds and the resulting partitions are not necessarily connected.

In this continuous case, the s-t minimal cut problem seeks the partition Γ_R of minimal total cost such that the point sets $s, t \subseteq \Omega$ fall in different partitions. To the authors' best knowledge this paper is the first to solve this problem in continuous spaces with more than 2 dimensions.

C. Maximal Flows

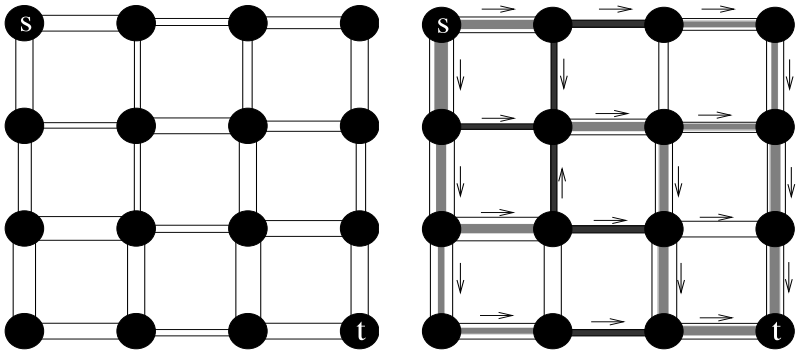
1) Discrete case: Let G be a graph with edge costs C_E now reinterpreted as *capacities*. A flow $F_G : E \to I\!R$ from a *source* set $s \subseteq V$ to a *sink* set $t \subseteq V$ has the following properties:

- Conservation of flow: The total (signed) flow in and out of any vertex is zero.
- Capacity constraint: The flow along any edge is less than or equal to its capacity:

$$\forall e \in E, \quad F(e) \leq C_E(e).$$

An edge along which the flow is equal to the capacity is decribed as *saturated*. [7] demonstrated that the maximal s-t flow equals the minimal s-t cut, with the flow saturated uniformly on the cut. Fig. 2 gives an example of a capacitated graph and an s-t maximal flow through this graph. In this example s and t are single vertices.

[16] describes how to convert the problem of computing a maximum flow between the sets s and t to an equivalent problem of computing a maximum flow between single vertices s' and t'. First we add to the graph G two new vertices s' and t', which become the new source and



(a) (b)

Fig. 2. An example of the minimal cut — maximal flow duality. (a) A capacitated graph. Edge thickness corresponds to capacity. (b) An s-t maximum flow. The set of saturated edges form a minimal cut.

sink respectively. Then from s' to each source vertex in s we add an edge of infinite capacity, and from each sink vertex in t to t' we do likewise. A maximum flow from s' to t' directly corresponds to a maximum flow from s to t in the original graph G. In this paper we will make implicit use of this direct correspondence between the two viewpoints.

A second convenience which we adopt in this paper is to add an implicit edge connecting $t \to s$ (equivalently $t' \to s'$) with infinite capacity to conserve flow uniformly throughout G. This ensures that the flow is conserved at every vertex in the graph, rather than treating the source and sink vertices as special cases. With this viewpoint, a maximal flow in a capacitated graph G then maximises the flow through the $t \to s$ edge. We denote this flow by F_{st}, and its maximisation is the objective of the maximal flow problem.

2) Continuous case: [24] and [25] explored the theoretical extension of maximal flows to continuous domains. A continuous flow $\vec{F}$ is a vector field over a continuous domain. It has the following properties:

- Conservation of flow: $\nabla \cdot \vec{F} = 0$.
- Capacity constraint: $\left|\vec{F}\right| \leq g$.

In the continous case the source s and the sink t become compact subsets of the continuous domain.

Let $\vec{F}$ be any flow and S be any simple, closed and smooth surface containing the source s. Let N_S denote the normal to the surface S. The net flow out of the source is denoted F_{st} as in

the discrete case. Then, combining the two properties stated above, we obtain:

$$F_{st} = \oint_S \vec{F} \cdot \vec{N}_S \mathrm{d}S \leq \oint_S g \mathrm{d}S. \tag{4}$$

Therefore, all flows are bounded from above by all smooth, simple and closed surfaces separating the source and sink, and all simple closed surfaces have weighted area bounded from below by all flows from source to sink. In fact, [25] showed that under very general continuity assumptions the maximal flow F_{max} is strictly equal to the minimal surface S_{min}. For such a flow and surface, the flow saturates the surface uniformly:

$$\forall \mathbf{x} \in S_{min}, \quad F_{max}(\mathbf{x}) = g(\mathbf{x}) \vec{N}(\mathbf{x}). \tag{5}$$

The minimal surface algorithm presented in this paper makes explicit use of this duality.

The duality between maximal flows and minimal cuts and surfaces has a simple interpretation. Any cut forms a bottleneck for a flow, limiting the flow to be less than the capacity of that cut. The maximal flow is limited by all possible cuts, and therefore must be less than or equal to the cost of the minimal cut. These dualities state that the maximal flow is indeed equal to the minimal cut, and therefore that a maximal flow saturates a minimal cut.

D. The Planar Case

For planar graphs and spaces some special equivalences exist between, on the one hand, minimal paths and geodesics, and on the other hand, minimal cuts and surfaces. In the discrete case of a graph embedded in the plane, a minimal cut in this *primal* graph is identical to a shortest path in the *dual* graph whose vertices correspond to the faces of the primal graph. Fig. 3 presents an example of this planar duality between paths and cuts. A similar duality has been noted in the continuous case between geodesics which are manifolds of dimension 1, and minimal surfaces which are manifolds of co-dimension 1 (and hence also dimension 1).

These dualities are important in the design of planar minimal cut algorithms because the computation of shortest paths is more efficient compared to general maximal flow methods. They are used in Weihe's discrete maximal flow algorithm [26] and in Mitchell's continuous maximal flow algorithm [27].

The authors' have previously presented an algorithm for 2D image segmentation, Globally Optimal Geodesic Active Contours (GOGAC), which can be interpreted under the planar duality

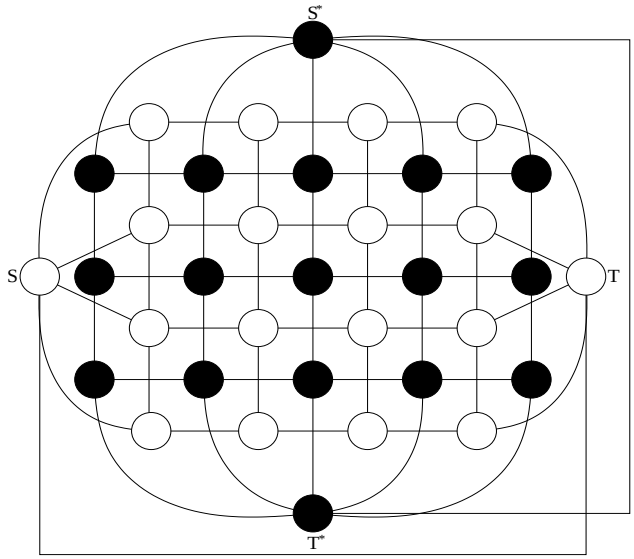


Fig. 3. A planar graph (white) and its dual (black). The vertices of the dual graph correspond to the faces of the primal graph. Edges connect adjacent faces in the dual graph and correspond uniquely to edges in the primal graph.

as a solution to the minimal surface problem in 2D spaces. However despite this connection the GOGAC algorithm is fundamentally based on the computation of geodesics and cannot be extended beyond 2 dimensions. The method presented in this paper is based on flows and may therefore be applied to higher dimensional spaces.

E. Approximating minimal surfaces by graph cuts

A number of approaches have been proposed to compute approximate minimal surfaces by transforming the problem to a graph cut. These approaches obtain a polyhedral surface of minimal weighted area, where the weighting is derived from the metric of the original Riemannian space.

[19] presented a formulation of the minimal cut problem in a graph with vertex capacities rather than edge capacities. Under this alternate formulation a cut becomes a set of vertices whose removal disconnects the source and sink. The cost of a cut is the sum of the capacities of these vertices. The continuous problem is modelled as a grid of square vertices of sidelength h. The vertex capacities are sampled directly from the metric of the continuous domain. All vertices are connected within a radius $r \gg h$. It was shown that, in the limit as $h \to 0$, $r \to 0$ and $\frac{h}{r} \to 0$ the minimal cut converges to a surface of minimal weighted area. As presented this method only approximates isotropic metrics.

[20] presents an approximation to the minimal surface problem using a graph with edge capacities derived from the metric of the continuous domain. Their approach is able to handle all convex metrics. Edge capacities are derived from the metric of the continuous domain using the Cauchy-Crofton formula from integral geometry.

In both of these approximations the theoretical convergence of a minimal cut to a minimal surface depends upon the degree of each vertex increasing toward infinity. In practice the number of directions that each segment of the polyhedral approximation can take on is proportional to the degree of each vertex. For an angular precision of $\Delta\theta$, the degree of each vertex is proportional to $(\frac{1}{\Delta\theta})^N$ in [19] and $(\frac{1}{\Delta\theta})^{N-1}$ in [20]. Consequently the time and memory required by these algorithms grows rapidly with the desired angular resolution, particularly in higher dimensions.

IV. Minimal surfaces in higher dimensions

In this section we present a non-linear system of partial differential equations (PDEs) to compute continuous maximal flows and hence obtain globally minimal surfaces. This extends the previous presentation by the same authors in [28], giving a detailed description of the implementation on regular grids and deriving the necessary and sufficient stability conditions.

The development of the following system of PDEs was motivated by considering existing discrete maximum flow algorithms. Two of the more popular maximum flow algorithms are the augmenting-path algorithm of Ford and Fulkerson and the pre-flow push algorithm of Goldberg and Tarjan [16]. The augmenting-path algorithm maintains a conservative flow at each step, repeatedly searching for paths along which the flow may be increased. However the direct extension of this algorithm to continuous spaces seems problematic. Not the least of these problems would be the requirement for a non-local system, corresponding to augmenting the flow along curves. This implies that it would not be possible to obtain a partial differential equation framework. Primarily this is due to the conservation constraint which imposes infinite 'stiffness' in the flow, making it difficult to modify the flow locally.

On the other hand, the pre-flow push algorithm relaxes the conservation constraint, allowing more flow into a vertex than out of it. This algorithm introduces an additional variable at each vertex which in some sense ensures that the system converges toward an incompressible flow. This is the approach we take in developing a solution to the continuous maximal flow problem. We allow the flow to have non-zero divergence during its evolution, but introduce a scalar

potential field which stores this excess flow. The potential field is then used to drive the flow to become incompressible at convergence.

A. A continuous maximal flow algorithm

The continuous maximal flow system developed in [28] is described by the following system:

$$\frac{\partial P}{\partial \tau} = -\nabla \cdot \vec{F} \tag{6}$$

$$\frac{\partial \vec{F}}{\partial \tau} = -\nabla P \tag{7}$$

subject to

$$\left|\vec{F}\right| \leq g \tag{8}$$

$P = P(\mathbf{x}, \tau) : (\Omega, I\!R^+) \rightarrow I\!R$ is a scalar potential field over the domain Ω evolving over time τ. $\vec{F} = \vec{F}(\mathbf{x}, \tau) : (\Omega, I\!R^N) \rightarrow I\!R^N$ is the vector flow field, also over the N-dimensional domain Ω and evolving over time τ. For boundary conditions we fix the scalar field P at the source s and sink t: $P(\mathbf{x}) = 1$ for $\mathbf{x} \in s$ and $P(\mathbf{x}) = 0$ for $\mathbf{x} \in t$. These values are chosen arbitrarily and without loss of generality. Initial conditions may be chosen as $P = 0$ except at the source and sink, and $\vec{F} = 0$ everywhere. However suitably selected initial conditions may lead to faster convergence, as we will discuss further in Section IV-D.

(6) relaxes the conservation constraint, instead storing excess flow in the potential field P. (7) couples the flow $\vec{F}$ to the potential P such that gradients in the potential drive the flow. (6) and (7) form a simple system of wave equations. They may be viewed as a linear model of the dynamics of an idealised fluid with pressure P and velocity $\vec{F}$, ignoring convection terms. (8) constitutes a hard constraint on the magnitude of the flow velocity $\vec{F}$.

B. Properties of the continuous maximal flow algorithm

1) Conservation of potential P: Let $P_A = \int_A P \mathrm{d}A$ denote the total integral of P in a given region A not including s, t. Then, for smooth P and $\vec{F}$,

$$\frac{\partial P_A}{\partial \tau} = -\oint_{\partial A} \vec{F} \cdot \vec{N}_{\partial A} \mathrm{d}\left(\partial A\right) \tag{9}$$

So P is conserved in the interior of any *sourceless* region A (any region not including the source s or sink t).

2) Monotonic reduction of energy $\frac{1}{2}(P^2 + \|\vec{F}\|^2)$*:* Consider the temporal rate of change of the total quantity of $\frac{1}{2}(P^2 + \|\vec{F}\|^2)$ in a given region A not including s, t. For smooth P and $\vec{F}$,

$$\frac{\partial}{\partial \tau} \int_A \frac{1}{2} \left(P^2 + \|\vec{F}\|^2 \right) \mathrm{d}A = - \oint_{\partial A} P \vec{F} \cdot \vec{N}_{\partial A} \mathrm{d}\left(\partial A\right) \tag{10}$$

Note that we have momentarily ignored the magnitude constraint (8). Consequently $\frac{1}{2}(P^2+\|\vec{F}\|^2)$ is conserved in the interior of any sourceless region A. Including the magnitude constraint may only decrease $\|\vec{F}\|^2$ and hence the energy $\frac{1}{2}(P^2 + \|\vec{F}\|^2)$ must monotonically decrease in the interior of a sourceless region. Since the energy is positive it must converge. To ensure smoothness and convergence of P and $\vec{F}$ independently, a dissipative term can be added to the equations. In practice this term is not necessary.

C. Correctness at convergence

At convergence any isosurface of P may be taken as the globally minimal surface S_{min} separating s and t.

Proof: Setting temporal derivatives to zero at convergence, we may restate the system (6), (7), (8):

$$\nabla \cdot \vec{F} = 0$$

$$\begin{aligned} \nabla P &= 0 && && \text{if } \left|\vec{F}\right| < g \\ \nabla P &= -\lambda \vec{F} \qquad \text{where} \qquad \lambda \geq 0 && && \text{if } \left|\vec{F}\right| = g \end{aligned}$$

The first equation simply restates the conservation of flow. The second equation is derived from (7), (8). It states that where $\vec{F}$ is not saturated P must be constant, and where $\vec{F}$ is saturated ∇P must be such that $\vec{F}$ cannot change direction or decrease in magnitude. Consequently $\nabla P \cdot \vec{F} \leq 0$, indicating that P is a (non-strictly) monotonic function along the flow lines of $\vec{F}$. As $\vec{F}$ is divergence-free, flow lines may only initiate at s and terminate at t. Therefore there are no local extrema in P.

Now consider the closed region A_p obtained from P by the application of a threshold $0 < p < 1$:

$$A_p = \{\mathbf{x} \,|\, P(\mathbf{x}) \geq p\}$$

Due to the monotonicity of P this is a connected region containing the source s. On the isosurface $S = \partial A_p$ we have $\nabla P \neq \vec{0}$ by construction. Therefore the flow is uniformly saturated outward

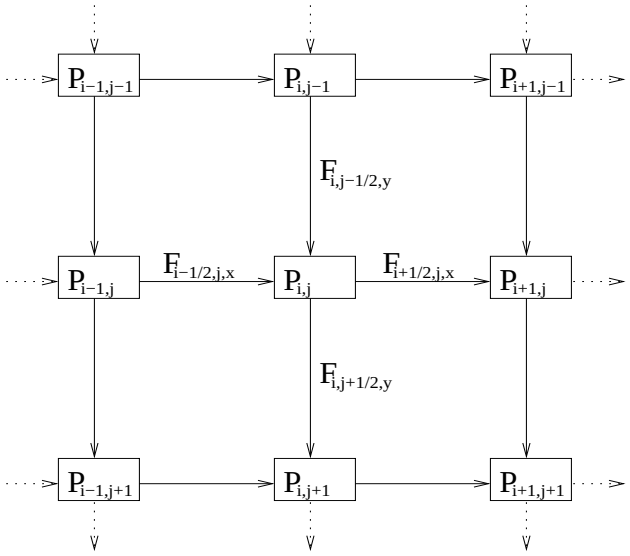


Fig. 4. The discrete representation of the numerical scheme presented here. P is stored at grid vertices while $\vec{F}$ is stored by component on grid edges.

on this surface and we obtain:

$$\nabla \cdot \vec{F}_s = \oint_S \vec{F} \cdot N_S \mathrm{d}S = \oint_S g \mathrm{d}S.$$

Hence $\vec{F}$ and S satisfy (5) for optimality. Therefore at convergence any isosurface of P is a globally minimal surface. In the usual case of a unique minimal surface, S_{min} will be the only isosurface at convergence and hence P will approach an indicator function for the interior of S_{min}. ■

D. Implementation

(6), (7) are discretised on a staggered grid using an explicit first-order scheme in time and space. The scalar field P is stored on grid points while the vector field $\vec{F}$ is stored by component on grid edges, depicted in Fig. 4. The system of equations is iterated sequentially with the flow magnitude constraint (8) enforced after each timestep.

Here for simplicity we describe the update scheme for a single iteration in 2 dimensions. The spatial grid step is set to $h = 1$. We first consider the linear portion of the update scheme, implementing (6), (7). We begin by defining the notation that we will use to describe this discrete system. Let n denote the iteration number and $\Delta\tau$ the timestep. Let $P^n_{i,j}$ denote the value of

the potential at time $n\Delta\tau$ and grid point (i,j) and let $g_{i,j}$ denote the value of the metric at the point (i,j). Let $F^n_{i\pm\frac{1}{2},j,x}$ and $F^n_{i,j\pm\frac{1}{2},y}$ represent the components of the flow along the four edges incident on the point (i,j) at time $n\Delta\tau$. Then we may give the explicit discretisation of the PDE system as:

$$P^{n+1}_{i,j} = P^n_{i,j} - \Delta\tau\left((F^n_{i+\frac{1}{2},j,x} - F^n_{i-\frac{1}{2},j,x}) + (F^n_{i,j+\frac{1}{2},y} - F^n_{i,j-\frac{1}{2},y})\right), \tag{11}$$

$$\begin{aligned} F'^{n+1}_{i+\frac{1}{2},j,x} &= F^n_{i+\frac{1}{2},j,x} - \Delta\tau(P^{n+1}_{i+1,j} - P^{n+1}_{i,j}) \\ F'^{n+1}_{i,j+\frac{1}{2},y} &= F^n_{i,j+\frac{1}{2},y} - \Delta\tau(P^{n+1}_{i,j+1} - P^{n+1}_{i,j}). \end{aligned} \tag{12}$$

The magnitude constraint is applied immediately following the update of the flow velocity field by (12). Here we describe the application of the magnitude constraint at point (i,j) for time $(n+1)\Delta\tau$, consisting of three stages:

1) Determine the maximal outward flow along each axis:

$$\begin{aligned} \left|F^{n+1}_{i,j,x}\right|' &= \max\left(-F'^{n+1}_{i-\frac{1}{2},j,x},\ 0,\ F'^{n+1}_{i+\frac{1}{2},j,x}\right) \\ \left|F^{n+1}_{i,j,y}\right|' &= \max\left(-F'^{n+1}_{i,j-\frac{1}{2},y},\ 0,\ F'^{n+1}_{i,j+\frac{1}{2},y}\right) \end{aligned}$$

2) Compare the absolute maximal outward velocity to the metric $g_{i,j}$:
If $v'^{n+1}_{i,j} \equiv \sqrt{\left(\left|F^{n+1}_{i,j,x}\right|'\right)^2 + \left(\left|F^{n+1}_{i,j,y}\right|'\right)^2} > g_{i,j}$ then

$$|F^{n+1}_{i,j,x}| = |F^{n+1}_{i,j,x}|'\frac{g_{i,j}}{v'^{n+1}_{i,j}} \quad \text{and} \quad |F^{n+1}_{i,j,y}| = |F^{n+1}_{i,j,y}|'\frac{g_{i,j}}{v'^{n+1}_{i,j}}.$$

3) Apply the magnitude constraint to each outward velocity component:

$$\begin{aligned} F^{n+1}_{i-\frac{1}{2},j,x} &= \max\left(F'^{n+1}_{i-\frac{1}{2},j,x},\ -|F^{n+1}_{i,j,x}|\right), \qquad F^{n+1}_{i+\frac{1}{2},j,x} = \min\left(F'^{n+1}_{i+\frac{1}{2},j,x},\ |F^{n+1}_{i,j,x}|\right), \\ F^{n+1}_{i,j-\frac{1}{2},y} &= \max\left(F'^{n+1}_{i,j-\frac{1}{2},y},\ -|F^{n+1}_{i,j,y}|\right), \qquad F^{n+1}_{i,j+\frac{1}{2},y} = \min\left(F'^{n+1}_{i,j+\frac{1}{2},y},\ |F^{n+1}_{i,j,y}|\right). \end{aligned}$$

Despite the complexity of its formal description, this update scheme is simple enough that a single implementation is used to handle input data of arbitrary dimension. This explicit scheme is also simple to parallelise by domain decomposition.

Several heuristics have been found to increase the speed of convergence. The fields P and $\vec{F}$ are rapidly initialised using the pre-flow push discrete maximal flow algorithm with both global and gap relabelling [16]. A multiscale approach is also applied recursively for rapid convergence at a fine grid resolution from a coarse grid initialisation. Computation may be avoided in the

interior of the source s and sink t, yielding great savings when they occupy a significant portion of the space.

At convergence in the continuous system, in the usual case of a single surface of globally minimal value, the potential field P is theoretically perfectly binary with value 1 within the volume bounded by the minimal surface, and 0 outside. However in the discrete implementation convergence is deemed to be attained if the sum of the relative areas of potential $|A_{P\geq 1-\gamma}|$ and $|A_{P\leq\gamma}|$ is greater than $\mu\%$. For example, $\gamma = 0.03$ and $\mu = 99$. Once convergence has been obtained, the minimal surface is extracted from P as the isosurface of value $\frac{1}{2}$ using a bilinear interpolation.

E. Stability

In this section we derive the maximum timestep for which the update scheme described by (11), (12) is stable. In this analysis we neglect the magnitude constraint as it may only reduce the magnitudes of the variables of interest and hence cause the system to tend toward stability. For simplicity we perform the derivation in the 2-dimensional case and then give the general solution.

By an appropriate combination of (11), (12) we may obtain the discrete update equation solely for P:

$$P_{i,j}^{n+2} - 2P_{i,j}^{n+1} + P_{i,j}^{n} = (\Delta\tau)^2 \left(-4P_{i,j}^{n+1} + P_{i+1,j}^{n+1} + P_{i-1,j}^{n+1} + P_{i,j+1}^{n+1} + P_{i,j-1}^{n+1}\right) \tag{13}$$

This is a discrete analogue to the wave equation $\frac{\partial^2 P}{\partial \tau^2} = \nabla^2 P$ which may be derived from (6), (7).

(13) describes a linear system and so is amenable to spectral analysis. Specifically, consider the Z-transform over $z_x,\ z_y,\ z_\tau \in \mathbb{C}$ with $|z_x| = |z_y| = 1$ for a bounded field P:

$$P(z_x, z_y, z_\tau) = \sum_{i,j,n} P_{i,j}^{n} z_x^i z_y^j z_\tau^n \tag{14}$$

For $P \neq 0$ substitution into (13) gives

$$\left(z_\tau^2 - 2z_\tau + 1\right) \;=\; (\Delta\tau)^2 \left(-4 + z_x + z_x^{-1} + z_y + z_y^{-1}\right) \tag{15}$$

For a stable and causal system we require $|z_\tau| < 1$. Now the right side of (15) takes values in the range $[-8(\Delta\tau)^2,\ 0]$ over the entire spatial spectrum. The left side has range $(-4,\ 0]$. Therefore

in order that this equation have a solution for all spatial frequency components, we require that it have a solution when the right side equals $-8(\Delta\tau)^2$, *i.e.*

$$\begin{aligned} -4 &< -8(\Delta\tau)^2 \\ \therefore \Delta\tau &< \frac{1}{\sqrt{2}} \end{aligned}$$

More generally when the update is performed in N dimensions it is simple to show that $\Delta\tau < \frac{1}{\sqrt{N}}$ as before. The same argument may be applied to the evolution of $\vec{F}$ but is not pursued here for reasons of space. This condition on the timestep is therefore necessary and sufficient to obtain a stable discrete implementation.

V. Metric weighting functions

Minimal surface methods have an inherent bias in favor of small surfaces. In many applications this is undesirable, resulting in incorrect or even trivial solutions. In this section we present a technique to automatically remove this bias.

A. Construction

Consider a metric that is uniformly constant throughout the domain, $g = 1$. This metric conveys no preference for any particular point through which the partition surface should pass. Intuitively then, every point in the domain should belong to some (globally) minimal surface. Unfortunately as the minimal surface problem is posed this is not the case. In order to improve the behavior of the solutions to this problem then, we replace the metric g by $g' = gw$, introducing an appropriate weighting function w. This weighting function will account for the geometry of the sources and sinks, so that the minimal surface depends only on the data as represented by g.

[23] considered the special case of a single point source $\mathbf{p}$ in a planar image. Here it was demonstrated that the introduction of the weighting function $w(\mathbf{x}) = \frac{1}{|\mathbf{x}-\mathbf{p}|}$ resulted in a continuum of minimal surfaces, the set of all circles centered on $\mathbf{p}$. In N dimensions it is simple to see that the modified weighting function $w(\mathbf{x}) = \frac{1}{|\mathbf{x}-\mathbf{p}|^{N-1}}$ will behave similarly, ensuring that each point in the domain belongs to a minimal surface (a hypersphere centered on $\mathbf{p}$). These weighting functions may be extended to other seed geometries. For a line source in 3 dimensions we obtain the weighting function $w(\mathbf{x}) = \frac{1}{|\mathbf{x}-\mathbf{p}|}$ where $\mathbf{p}$ is the nearest point to $\mathbf{x}$ on the line. More

generally for a set of seeds which form an M-dimensional manifold embedded in N dimensions we should expect a weighting function that decays as $\frac{1}{|\mathbf{x}-\mathbf{p}|^{N-M-1}}$ in the neighborhood of the manifold.

We wish to derive an *unbiased* flow $\vec{F}$ from which we may define the weighting function $w = \|\vec{F}\|$. This flow will be produced by the source set s and absorbed by the sink set t,

$$\nabla \cdot \vec{F} = \rho \tag{16}$$

where ρ is a distribution that is zero in the interior of the domain, positive on the source set s and negative on the sink set t, with total source weight $\int_s \rho \mathrm{d}V = 1$ and sink weight $\int_t \rho \mathrm{d}V = -1$. There will naturally be many such flows; here we select a flow to minimise a measure of the weighting function

$$E[w] = \int_V \frac{1}{2} w^2 \mathrm{d}V = \int_V \frac{1}{2} \|\vec{F}\|^2 \mathrm{d}V.$$

In this way we will ensure that the weighting function is not arbitrarily large at any particular point in space, as it could be for example for some flows with large rotational components.

We may minimise the measure $E[\vec{F}] \equiv E[w]$ by variational calculus: consider adding a minimisation parameter τ to obtain $w \equiv w(\mathbf{x}, \tau)$. Then we may compute the first variation with respect to τ to determine the local minima of $E[\vec{F}]$:

$$\frac{\delta E[\vec{F}]}{\delta \tau} = \int_V \vec{F}_\tau \cdot \vec{F} \mathrm{d}V = 0$$

Here we have set the first variation to 0 to obtain a local minimum condition on $E[w]$. This minimisation must be carried out subject to the incompressibility constraint expressed in (16). Taking the time derivative of the constraint, we obtain an equivalent constraint on $\vec{F}_\tau$:

$$\nabla \cdot \vec{F}_\tau = 0$$

Therefore $\vec{F}_\tau$ may be decomposed into cyclic components, and $\vec{F}$ is a local minimum of $E[\vec{F}]$ if it is locally minimal with respect to all cyclic flows. Consider then $\vec{F}_\tau = \vec{T}_C$ the unit tangent vector over the tube formed from the set of all points within a vanishing radius r of the smooth closed curve C, with $\vec{F}_\tau = 0$ elsewhere. For $\vec{F}$ a local minimum of $E[\vec{F}]$ we have

$$\begin{aligned} \frac{\delta E[\vec{F}]}{\delta \tau} &= \int_V \vec{F}_\tau \cdot \vec{F} \mathrm{d}V \\ &= A_{N-1}(r) \oint_C \vec{F}_\tau \cdot \vec{T}_C \mathrm{d}C \\ &= 0 \end{aligned}$$

where $A_{N-1}(r)$ is the volume of the $N-1$ dimensional sphere of radius r. So we find that the vector field is a potential flow. Set $\vec{F}_\tau = \nabla\phi_\tau$ then, and replace the divergence of the flow $\vec{F}$ in (16) by the Laplacian of ϕ to obtain

$$\nabla^2\phi = \rho$$

We choose boundary conditions $\lim_{|\mathbf{x}|\to\infty} \nabla\phi(\mathbf{x}) = 0$ so that the flow is zero at infinity. ϕ is then determined up to the addition of a constant which will not affect the weighting function $w = |\nabla\phi|$.

Observe now that all isosurfaces S of ϕ in the interior of the domain have constant net flux $\oint_S \nabla\phi \cdot \vec{N}_S \mathrm{d}S = 1$. As $w = |\nabla\phi|$ we then obtain $\oint_S w\mathrm{d}S = 1$ over all isosurfaces of ϕ, with $\oint_S w\mathrm{d}S \geq 1$ for all closed surfaces S containing the seeds. So the isosurfaces of ϕ form the set of minimal surfaces under the metric w. In general we have $\nabla\phi \neq 0$ almost everywhere, therefore almost every point in the domain belongs to some minimal weighted surface under the metric w as desired.

B. Implementation

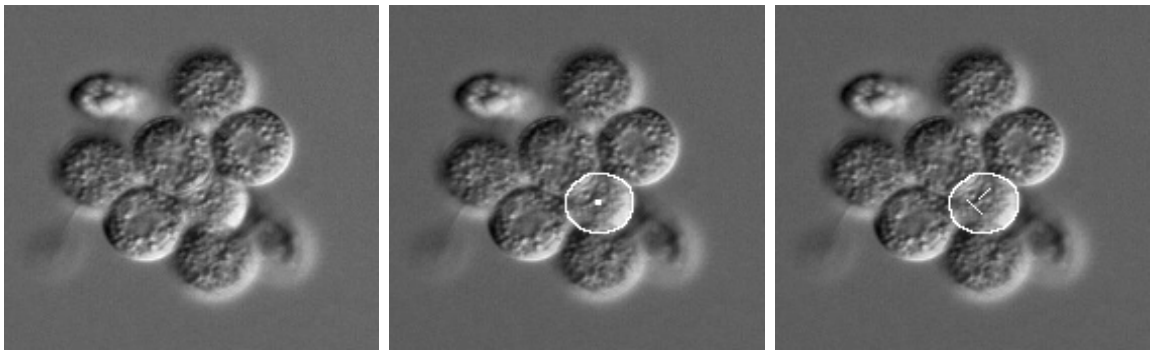
For the regular grids considered in this paper the weighting functions may be computed by convolving the distribution ρ with the Green's function $\phi_\odot$ with the property $\nabla^2\phi_\odot = \delta(\mathbf{x})$. In $I\!R^2$ this is $\phi_\odot = \frac{1}{2\pi}\ln(|\mathbf{x}|)$ while in $I\!R^3$ this is $\phi_\odot = -\frac{1}{4\pi|\mathbf{x}|}$ [29]. This convolution may be efficiently computed on discrete images using the Fast Fourier Transform. The gradient of ϕ may then be numerically estimated in the discrete grid to obtain the weighting function w.

Fig. 5 shows an example of a set of seed points and the process of computing an appropriate weighting function. The weighting function is highest in the neighborhood of point sources and at the endpoints of line sources.

Fig. 6 depicts the application of metric weighting in the segmentation of a microscope image of a protist, *Chilomonas Paramecium*. Presented are segmentations using a simple seed geometry and a complex seed geometry. The metric weighting scheme proposed in this section produces similar results on the two examples, demonstrating that it does not significantly bias the segmentation.

(a) (b) (c)

Fig. 5. Metric weighting example. (a) The seed geometry. Source points are depicted, while the sink points are the image boundary. (b) The function ϕ computed by convolution. (c) The metric weighting w, computed from the numerical derivative of ϕ.



(a) (b) (c)

Fig. 6. An example of the application of metric weighting. (a) A microscope image of a protist, *Chilomonas Paramecium*. (b) Segmentation using a single internal seed. (c) Segmentation with complex seed geometry.

VI. Results

In this section we demonstrate the results of using globally minimal surfaces for 2D and 3D medical image segmentation and for stereo matching. All applications were performed using the metric weighting scheme introduced in Section V.

All tests were performed on a quad 2.2GHz AMD Opteron Processor 848 under the Linux operating system. The algorithm presented here has been implemented in C with no assembly

optimisations. Timings for minimum cuts have been obtained using the Boost Graph Library implementation of the preflow-push algorithm [30]. The preflow-push algorithm is generally accepted as a fast general-purpose maximum flow algorithm, although a faster image-specific maximum flow algorithm presented in [31] has not been considered here.

A. 2D Image Segmentation

Object boundaries are often difficult to detect along transitions to adjacent objects with similar features. Segmentation via minimal contours uses the regularisation of the segmentation contour to avoid leaking across such gaps. The authors have previously developed an algorithm, Globally Optimal Geodesic Active Contours (GOGAC) [23], which efficiently computes globally minimal contours in planar Riemannian spaces using the planar duality in Section III-D. Here we apply discrete minimal cuts, GOGAC [23], and the algorithm presented in this paper to segment a microscope image of a cluster of cells (Fig. 7(a)) and compare the results. In spite of its apparent simplicity this problem demonstrates the challenge of delineating faint boundaries between cells without leaking.

We compute a metric (Fig. 7(b)) from the microscope image as described in (2), with default parameters $p = 1$ and $\varepsilon = 0$ and with blurring at scale $\sigma = 1$. Low metric regions are dark while high metric regions are bright. Observe that the regions of low metric correspond to the boundaries of the cells, except where the cells overlap. The metric has been weighted according to the method described in Section V. This is not displayed due to the extremely large range of values. All methods used the same (weighted) metric.

The segmentation of each cell is performed independently in sequence for each method. The source sets are depicted in (Fig. 7(b-d)) while the sink is the image boundary. The discrete minimal cut solves a discretised minimal surface problem, resulting in a clear grid bias and a poor segmentation. GOGAC and the continuous maximal flow algorithm solve the same continuous optimisation problem and are in close agreement. Note that the continuous segmentations follow the perceived cell contours despite the weakness of local cues.

The image depicted in Fig. 7(a) has dimensions 231×221. We reduce the amount of computation required by expanding the sink to include only the cells of interest, a region of size 150×100.

The discrete minimal cuts required 0.41 seconds to compute in total. GOGAC required 0.73

seconds to compute in total. The continuous minimal surface algorithm presented here required 0.68 seconds in total to converge.

B. 3D Image Segmentation

Here we demonstrate the application of globally minimal surfaces to a 3D segmentation problem. Fig. 8 depicts a Computed Tomography scan of a chest in which the two lungs are segmented. We compare the results from the application of globally minimal surfaces to those obtained using geodesic active surfaces and discrete minimal cuts. All segmentations use the same data, *i.e.* the weighted metric as well as the seeds. The sources are large spheres inside each lung while the sinks are the volume boundaries. Large spheres were used because the geodesic active surfaces are initialised using these same spheres, and such variational methods require good initialisation in order to obtain a reasonable result. An artificial inflation term was also used to drive the level sets to fill the lungs. The lungs are segmented independently in all three methods.

The top row of Figure 8 depicts corresponding 2D slices of the original CT data, the metric derived from this data, and the weighted metric. The weighted metric has been displayed on a log scale due to its large range of values. The middle row of Figure 8 shows corresponding 2D slices of segmentations by each of the methods considered here: geodesic active surfaces, minimal cuts, and globally minimal surfaces. The bottom row of Figure 8 provides identical 3D views of segmentations obtained by the three different methods.

The geodesic active surfaces give a poor segmentation. At the base of the lung the inflation term is too weak for the surface to completely fill the lung, becoming trapped on edges due to alveoli. At the top of the lung the inflation term is too strong, causing the surface to leak through the weak edges of the lung. This behavior is common in the application of active contour methods.

The discrete minimal cuts also produce inaccurate segmentations. Observe the bias toward the grid directions, which can be clearly seen as the flat boundaries in the interior surfaces at the top of the lungs. By contrast, the continuous minimal surface does not exhibit such directional bias, giving a faithful segmentation.

The CT data shown in Fig. 8 has dimensions $200 \times 160 \times 90$. The Geodesic Active Surfaces required 279 seconds to converge to the final result. The discrete minimal cuts required 44

seconds to compute using the pre-flow push algorithm. The continuous minimal surface algorithm required only 28.8 seconds. Note that the minimal surface algorithm uses a multiscale framework to obtain a good initialisation from the solution at a coarser scale. At the coarsest scale the fields P and $\vec{F}$ are initialised using a minimum cut. In this example the minimal surface segmentation is faster than the minimum cut segmentation due to the multiscale framework.

C. 3D Scene Reconstruction from Stereo Images

The reconstruction of a 3D scene from two or more images is often performed using an energy minimisation approach [13], [14]. Here we adapt the framework of [14], replacing their discrete graph cut by a globally minimal surface.

In general the application of globally minimal surfaces requires a suitable choice of metric appropriate to the application. In stereo matching a number of metrics have been proposed for real and synthetic images. Here we use the zero-mean normalised cross correlation (ZNCC) window based matching score, which performs well on natural scenes with lighting variation and specular reflections and may be computed very efficiently [12]. We set $g = 1 - ZNCC$ to convert high matching scores to low metrics suitable for a minimal surface approach. Matching scores are computed using a 5×5 window. The stereo pair being analysed has a disparity range of $[-15,\ 0]$. Following [14] the source and sink are connected to the first and last layers of the disparity volume. Both the discrete minimal cut and the globally minimal surface are computed from the same metric.

The results of the stereo matching are depicted in Fig. 9. These results are shown as disparity maps (depth maps) as well as surface meshes. We observe that the discrete minimal cut produces large flat regions due to the small number of disparities and hence poor depth resolution of discrete methods. Compared to the graph cut we can see a great deal more detail in the disparity map computed by the globally minimal surface. This includes the surface texture of the bushes as well as the third parking meter. In addition to the apparent improvement in depth resolution is the apparent spatial isotropy of the continuous method. This can be seen on the frame of the car, where the discrete method produces a ‘rectangular’ curve while the minimal surface produces a straight line.

The stereo image pair used here has dimensions 256×240. The discrete minimal cut required 11.3 seconds to compute. The continuous minimal surface algorithm required only 8.3 seconds.

D. Accuracy

In the continuous theory the system of PDEs presented in (6), (7), (8) was proven to obtain the globally minimal surface at convergence. However in order to develop a practical algorithm it was necessary to discretise these equations in Section IV-D. It is natural to question whether this discretisation introduces grid bias into the solution surfaces.

To address this question, we compare the surface obtained by the algorithm presented in this paper with the analytic solution on a simple problem in 3 dimensions. Consider two circles of equal radius R' whose centers lie along the z-axis. These circles lie parallel to each other in the planes $z = -H$ and $z = H$ respectively. Then the minimal surface which connects these two circles is a catenoid. Representing the surface points by the triplet (x, y, z) we may parameterise by $\theta \in [0, 2\pi)$ and $z \in [-H, H]$ as follows:

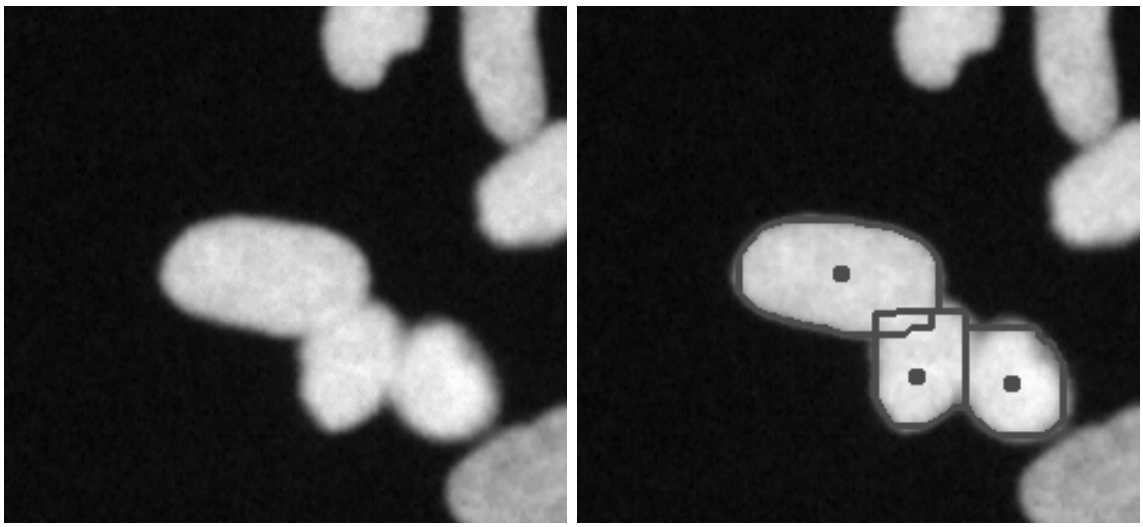
$$\begin{aligned} x &= R' \cosh\left(\frac{z}{R'}\right) \cos(\theta) \\ y &= R' \cosh\left(\frac{z}{R'}\right) \sin(\theta) \end{aligned}$$

Here the value R' is selected such that $R' \cosh\left(\frac{H}{R'}\right) = R$ in order to meet the boundary conditions.

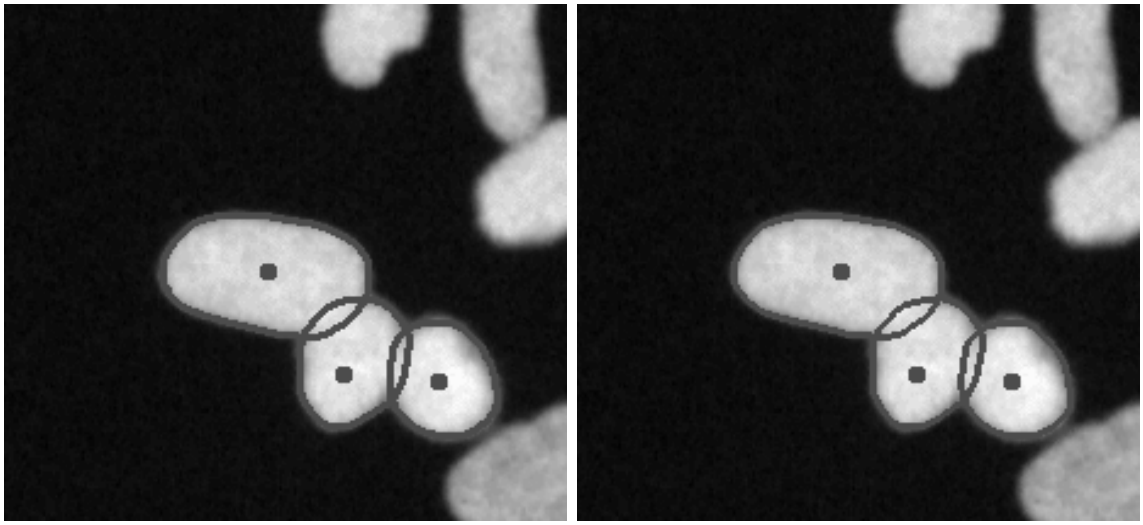
We select $R = 35$ and $H = 15$ and make a comparison between the analytic and numeric solutions. The algorithm proposed in this paper is discretised on a regular grid with grid step $h = 1$. For the Euclidean metric we set $g = 1$ throughout the grid, without metric weighting. Boundary conditions are enforced by placing disk-shaped sources of radius R on the vertical boundaries $z = H$ and $z = -H$, and sinks elsewhere on the volume boundary. The results for this comparison are presented in Fig. 10. Fig. 10(a) depicts the analytic solution while Fig. 10(b) depicts the solution obtained by the method proposed in this paper. The numeric and analytic solutions are in clear visual agreement. Fig. 10(c) depicts a horizontal slice $z = 0$ through the potential function P computed by the new algorithm, overlayed with the corresponding cross-section of the analytic solution (a circle of radius R'). Fig. 10(d) depicts a vertical slice $x = 0$ through P, overlayed with the corresponding cross-section of the analytic solution (two catenaries). In both Fig. 10(c) and (d) the analytic solution closely coincides with the isosurface $P = 0.5$.

In order to make a quantitative comparison we measured the average distance between the two surfaces for 6000 points, consisting of 30 values for z and 200 values for θ. The mean distance between the surfaces was 0.09 while the root-mean-square distance was 0.11. To summarise, in

this simple example the new algorithm obtains a result which is accurate to approximately 10% of the grid step.



(a) (b)



(c) (d)

Fig. 7. Segmentation of a cluster of cells in a microscope image. (a) The original image. (b) Discrete minimal cuts. (c) Globally Optimal Geodesic Active Contours. (d) Globally minimal surfaces.

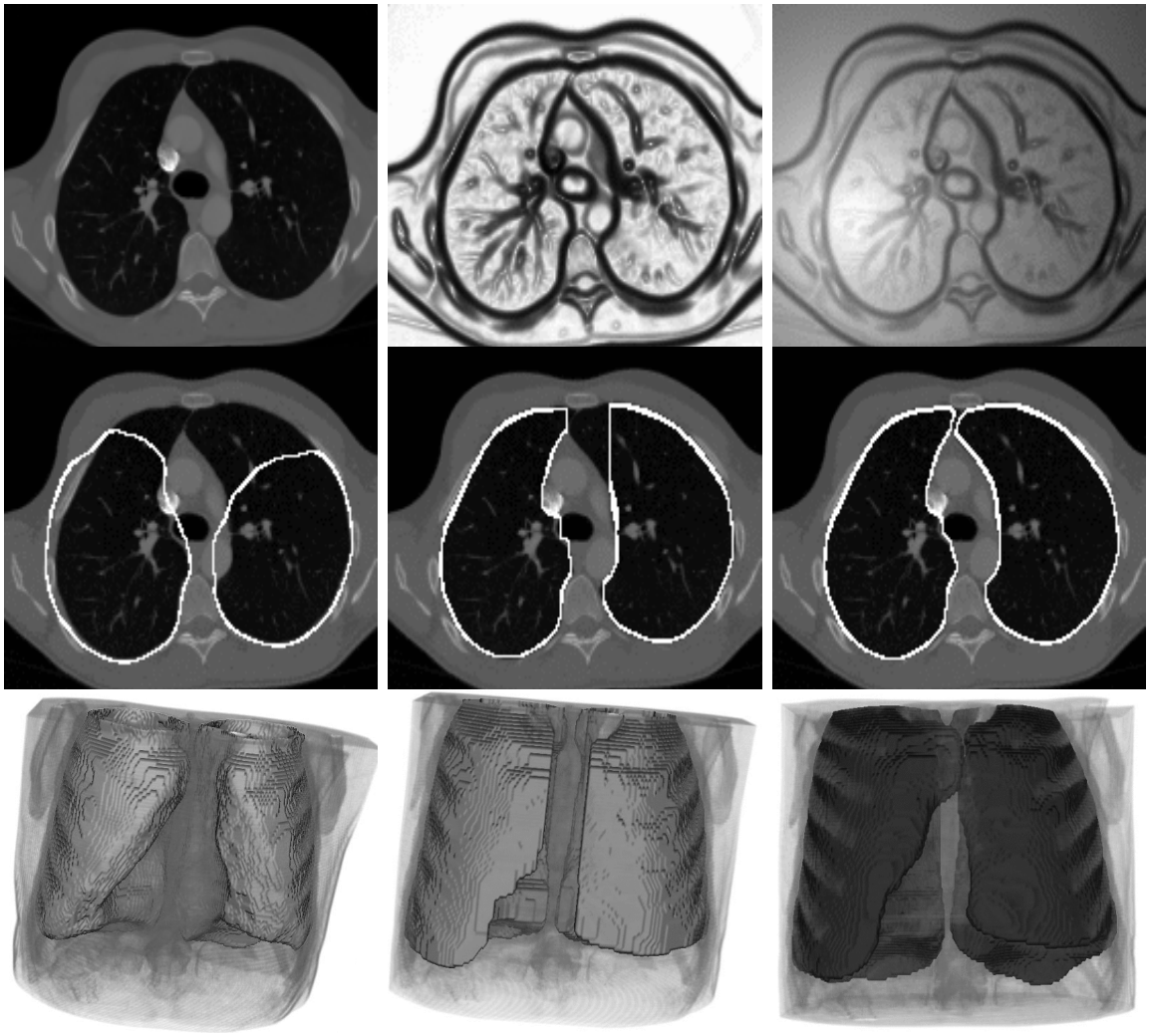
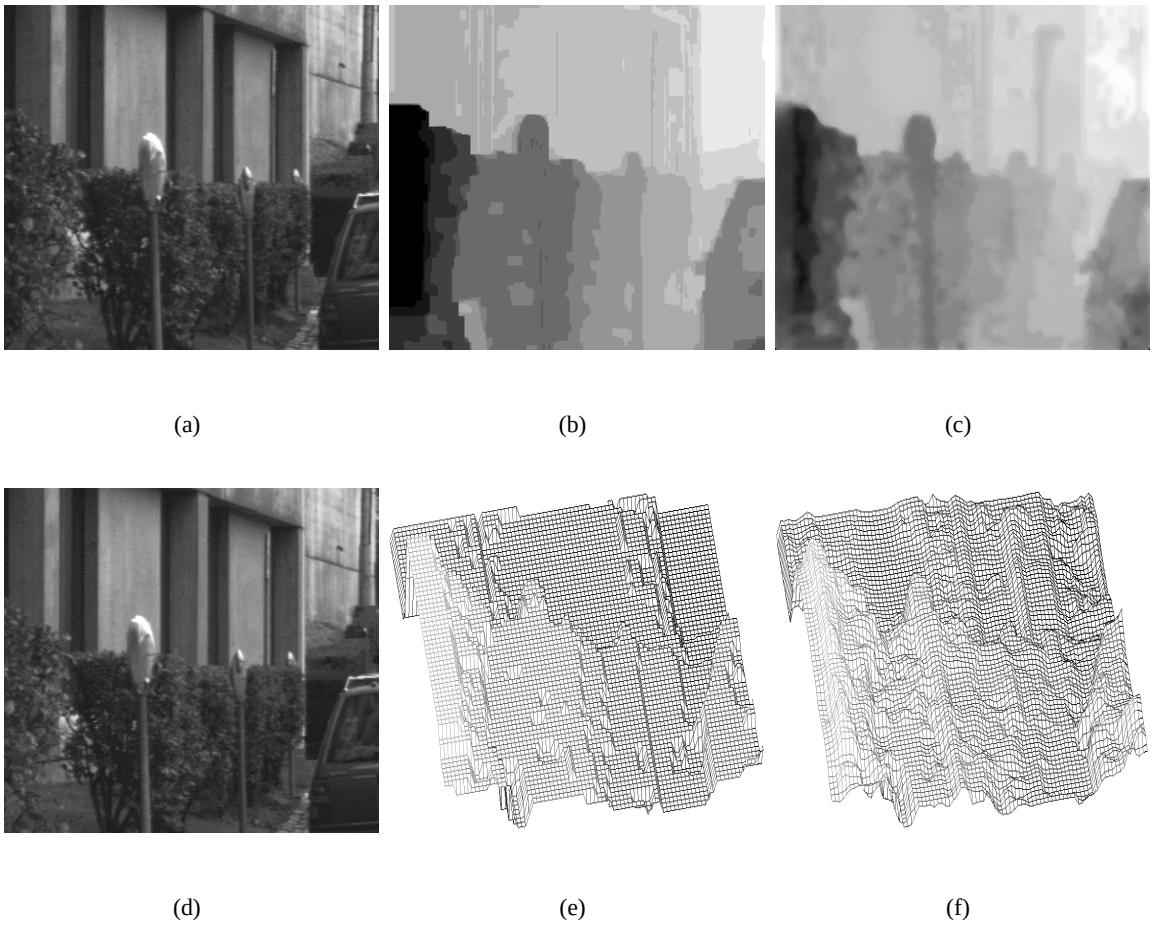


Fig. 8. Segmentation of the lungs in a CT image of a chest. Top row: 2D slices of original data, metric, and weighted metric (log scale). Middle row: 2D slices of segmentations by geodesic active surfaces, minimal cuts, and globally minimal surfaces. Bottom row: 3D views of segmentations by geodesic active surfaces, minimal cuts, and globally minimal surfaces.



(a) (b) (c)

(d) (e) (f)

Fig. 9. Stereo matching from two views. (a, d) The original images. (b, e) Disparity map and corresponding mesh obtained by a discrete maximal flow. (c, f) Disparity map and corresponding mesh obtained by a globally minimal surface. Note the improved detail.

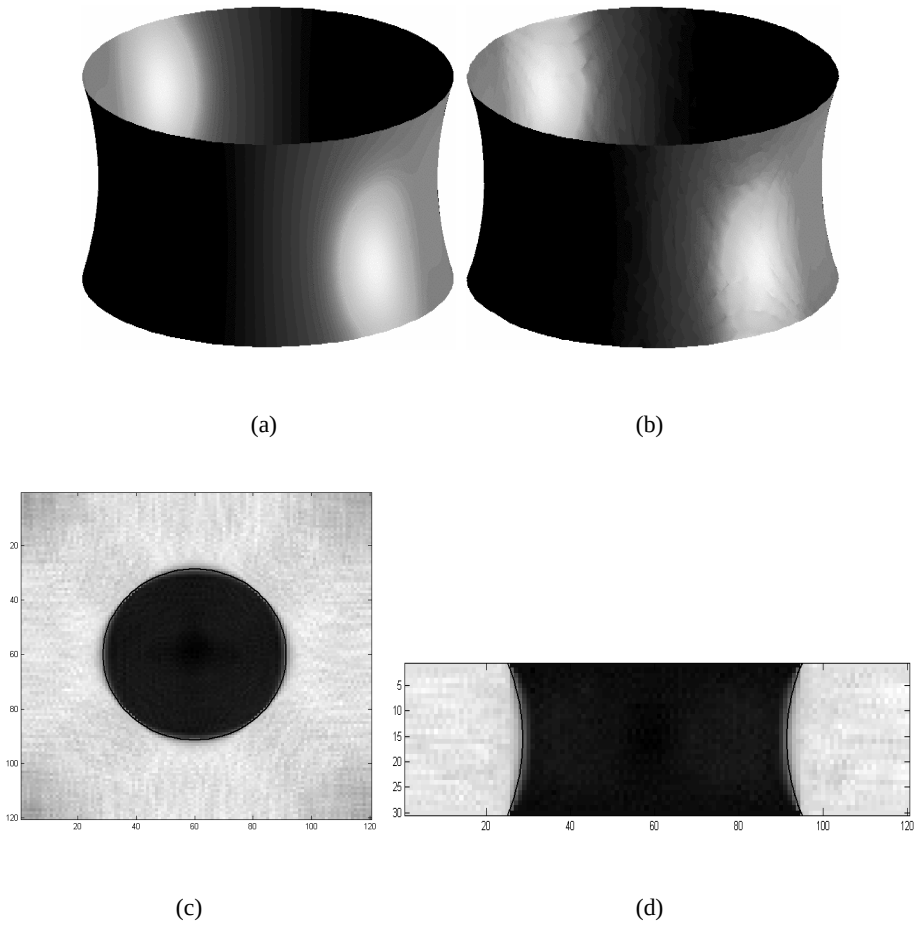


Fig. 10. The catenoid test problem. (a) The correct minimal surface, constructed analytically. (b) The minimal surface computed using the method described in this paper. (c) A horizontal slice through P. The correct cross-section is overlayed in black. (d) A vertical slice through P. The correct cross-section is overlayed in black.

VII. Conclusions

In this paper we have developed a new algorithm to compute globally minimal weighted surfaces for image segmentation and stereo matching.

We obtain these surfaces using a non-linear system of partial differential equations which simulate an ideal fluid flow. The velocity of the flow is constrained in magnitude by a spatially varying metric, itself derived from the image or images being analysed. This simulation is performed using a simple finite difference scheme with explicit update step. To improve efficiency a multiresolution scheme is used to reduce computational costs and the solution is approximated at the coarsest scale by a discrete maximal flow.

A proof is given that at convergence the algorithm produces a globally maximal flow. A simple threshold of the auxiliary potential function produces the corresponding minimal surfaces. A proof that the system converges is left to future work.

Results are given demonstrating the application of globally minimal surfaces to 2D and 3D segmentation and to stereo matching. Comparison to an existing optimal geodesic active contour for 2D images demonstrates close similarity. Comparisons for 3D segmentation and stereo matching demonstrate that globally minimal surfaces overcome the existing problems with graph-based approaches and with active contours. On a simple test problem with known analytic solution, the discrete implementation of this algorithm was shown to be accurate to 10% of the grid step size. The algorithm is also efficient when compared to previous methods. These results suggest that many existing applications using geodesic active surfaces or graph cuts will benefit from the improved accuracy and robustness of globally minimal surfaces.

References

[1] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1998.

[2] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations," *Journal of Computational Physics*, vol. 79, pp. 12–49, 1988. [Online]. Available: citeseer.nj.nec.com/osher88fronts.html

[3] J. A. Sethian, *Level Set Methods and Fast Marching Methods - Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science.* Cambridge University Press, 1999.

[4] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *IJCV*, vol. 22, no. 1, pp. 61–79, 1997.

[5] V. Caselles, R. Kimmel, G. Sapiro, and C. Sbert, "Minimal surfaces based object segmentation," *IEEE Trans. on PAMI*, vol. 19, pp. 394–398, 1997.

[6] E. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.

[7] J. L. R. Ford and D. R. Fulkerson, *Flows in Networks*. Princeton University Press, Princeton, NJ, 1962.

[8] L. D. Cohen, "On active contour models and balloons," *Computer Vision, Graphics, and Image Processing. Image Understanding*, vol. 53, no. 2, pp. 211–218, 1991. [Online]. Available: citeseer.nj.nec.com/cohen91active.html

[9] C. Xu and J. L. Prince, "Snakes, shapes and gradient vector flow," *IEEE Transaction on Image Processing*, vol. 7, no. 3, pp. 359–369, March 1998.

[10] S. Lloyd, "Stereo matching using intra- and inter-row dynamic programming," *Pattern Recognition Letters*, vol. 4, pp. 273–277, September 1986.

[11] Y. Ohta and T. Kanade, "Stereo by intra- and inter-scanline search using dynamic programming," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 7(2), pp. 139–154, March 1985.

[12] C. Sun, "Fast stereo matching using rectangular subregioning and 3D maximum-surface techniques," *International Journal of Computer Vision*, vol. 47, no. 1/2/3, pp. 99–117, May 2002.

[13] C. Leung, B. Appleton, and C. Sun, "Fast stereo matching by iterated dynamic programming and quadtree subregioning," in *British Machine Vision Conference*, S. B. A. Hoppe and T. Ellis, Eds., vol. 1, Kingston, United Kingdom, September 2004, pp. 97–106.

[14] S. Roy and I. J. Cox, "A maximum-flow formulation of the n-camera stereo correspondence problem," in *Int. Conf. on Computer Vision (ICCV'98)*, Bombay, India, January 1998, pp. 492–499.

[15] P. Bamford and B. Lovell, "Unsupervised cell nucleus segmentation with active contours," *Signal Processing (Special Issue: Deformable models and techniques for image and signal processing)*, vol. 71, no. 2, pp. 203–213, 1998.

[16] R. Sedgewick, *Algorithms in C*, 3rd ed. Addison-Wesley, 2002, no. 5.

[17] J. N. Tsitsiklis, "Efficient algorithms for globally optimal trajectories," *IEEE Transactions on Automatic Control*, vol. 40, no. 9, pp. 1528–1538, September 1995.

[18] J. Sethian, "A fast marching level set method for monotonically advancing fronts," in *Proceedings of the National Academy of Sciences*, vol. 93(4), 1996, pp. 1591–1595. [Online]. Available: citeseer.nj.nec.com/sethian95fast.html

[19] T. C. Hu, *Integer Programming and Network Flows*. Addison-Wesley, Reading, MA, 1969.

[20] Y. Boykov and V. Kolmogorov, "Computing geodesics and minimal surfaces via graph cuts," in *International Conference on Computer Vision*, Nice, France, October 2003, pp. 26–33.

[21] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, "Fast geodesic active contours," *IEEE Trans. On Image Processing*, vol. 10, no. 10, pp. 1467–1475, 2001.

[22] L. D. Cohen and R. Kimmel, "Global minimum for active contour models: A minimal path approach," *International Journal of Computer Vision*, vol. 24, no. 1, pp. 57–78, August 1997. [Online]. Available: citeseer.nj.nec.com/cohen97global.html

[23] B. Appleton and H. Talbot, "Globally optimal geodesic active contours," *Journal of Mathematical Imaging and Vision*, July 2005, in press.

[24] G. Strang, "Maximal flow through a domain," *Mathematical Programming*, vol. 26, pp. 123–143, 1983.

[25] M. Iri, *Survey of Mathematical Programming*. North-Holland, Amsterdam, 1979.

[26] K. Weihe, "Maximum (s,t)-flows in planar networks in $O(|V|\log|V|)$ time," *Journal of Computer and System Sciences*, vol. 55, no. 3, pp. 454–475, December 1997.

[27] J. S. Mitchell, "On maximum flows in polyhedral domains," in *Proc. 4th annual symposium on computational geometry*, Urbana-Champange, Illinois, United States, 1988, pp. 341–351.

[28] B. Appleton and H. Talbot, "Globally optimal surfaces by continuous maximal flows," in *Digital Image Computing:*

Techniques and Applications, Proc. VIIth APRS conference, C. Sun, H. Talbot, S. Ourselin, and T. Adriaansen, Eds., vol. 2. Sydney: CSIRO publishing, December 2003, pp. 987–996.

[29] G. Strang, *Introduction to Applied Mathematics*. Wellesley-Cambridge Press, 1986.

[30] J. Siek, L.-Q. Lee, and A. Lumsdaine, *The Boost Graph Library: User Guide and Reference Manual*. Addison-Wesley, 2002.

[31] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. PAMI*, vol. 26, no. 9, pp. 1124–1137, Sep 2004.