

A Hybrid Eulerian–Lagrangian Approach for Thickness, Correspondence, and Gridding of Annular Tissues

Kelvin R. Rocha, *Student Member, IEEE*, Anthony J. Yezzi, Jr., *Member, IEEE*, and Jerry L. Prince

Abstract—We present a novel approach to efficiently compute thickness, correspondence, and gridding of tissues between two simply connected boundaries. The solution of Laplace’s equation within the tissue region provides a harmonic function whose gradient flow determines the *correspondence trajectories* going from one boundary to the other. The proposed method uses and expands upon two recently introduced techniques in order to compute thickness and correspondences based on these trajectories. Pairs of partial differential equations are efficiently computed within an Eulerian framework and combined with a Lagrangian approach so that correspondences trajectories are partially constructed when necessary. Examples are presented in order to compare the performance of this method with those of the pure Lagrangian and pure Eulerian approaches. Results show that the proposed technique takes advantage of both the speed of the Eulerian approach and the accuracy of the Lagrangian approach.

Index Terms—Correspondence, correspondence trajectory, partial differential equations (PDEs), thickness.

I. INTRODUCTION

MANY parts of the human body have an annular tissue structure comprising two or more quasi-homogeneous tissues nested within one another. For example, the cerebral cortex is comprised of gray matter “sandwiched” between white matter on the inside and cerebrospinal fluid on the outside [1]–[4]. Another example is the left ventricular myocardium, which is intrinsically annular when viewed from cross section images such as those obtained using magnetic resonance imaging (MRI) or computed tomography (CT) [5], [6].

The thickness of such annular structures is often associated with functional performance or disease. For example, an increased thickness of the cerebral cortex may be associated with cortical dysplasias and lissencephaly [7], and decreased thickness may be related to Alzheimer’s disease [8]–[11] and anorexia nervosa [12]. In the heart, adequate thickening of the myocardium during systole is associated with a healthy heart, whereas overall thickening of the myocardium over time is associated with many cardiac diseases including hypertrophic cardiomyopathy, hypertension, and Marfan’s syndrome [13].

Manuscript received June 19, 2005; revised September 8, 2006. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Attila Kuba.

K. R. Rocha and A. J. Yezzi, Jr., are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332-1100 USA (e-mail: krocha@ece.gatech.edu; ayezzi@ece.gatech.edu).

J. L. Prince is with the Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218 USA (e-mail: prince@jhu.edu).

Digital Object Identifier 10.1109/TIP.2007.891072

Sometimes, it is also useful to subdivide (or grid) annular regions for regional characterization, labeling, or for finite element analysis [14]. The cortex, for example, is divided into lobes and gyres characterizing regions that are often functionally different. As well, the left ventricular myocardium is frequently divided into basal, midventricle, and apical regions along the long axis; septal, anterolateral, posteroseptal, and free regions around the circumference; and subendocardial, midwall, and subepicardial regions transmurally [15]. Finite-element analysis can be used to understand cortical deformation during surgery or myocardial strain during normal heartbeats or those occurring under stress. A principled method for the gridding of annular tissues would be useful for these applications.

A variety of methods have been described and used to measure thickness within annular regions. Most methods are *ad hoc*, often manually-assisted, and have accuracies that are highly dependent on the boundary shapes and on the person analyzing the images. Jones *et al.* [16] proposed an approach based on solving Laplace’s equation on the annular region and defining thickness as the length of unique correspondence trajectories running between the boundaries on lines orthogonal to equipotential contours. This approach yields unique measures of thickness without paradoxes (see [17] and [18]), and provides a strong initial basis for making thickness, corresponding, and gridding of annular regions unique, accurate, and repeatable. The Jones’ approach is Lagrangian, meaning that paths from one boundary to the other are explicitly traced (using numerical integration of the gradient field of Laplace’s solution). An Eulerian approach in which partial differential equations (PDEs) are solved for the desired lengths, thereby avoiding the explicit construction or tracing of any correspondence trajectory was presented in [17] and [18].

In this paper, we first extend the Eulerian thickness approach both to find boundary correspondences and to gridding. These two new methods are based on the same general framework of [17] and [18] and are, therefore, very computationally fast, but there are inherent limitations in their accuracy. We next introduce a hybrid approach, in which the Eulerian approaches are carefully modified so that they can use the Lagrangian approach where more precision is needed. The new methods are significantly faster than the pure Lagrangian approaches and yet approach their accuracy.

In Section II, we briefly review how thickness is defined and briefly summarize both the Lagrangian and Eulerian PDE approach with their respective numerical implementations. In Section III, we extend the Eulerian approach in [17] to compute

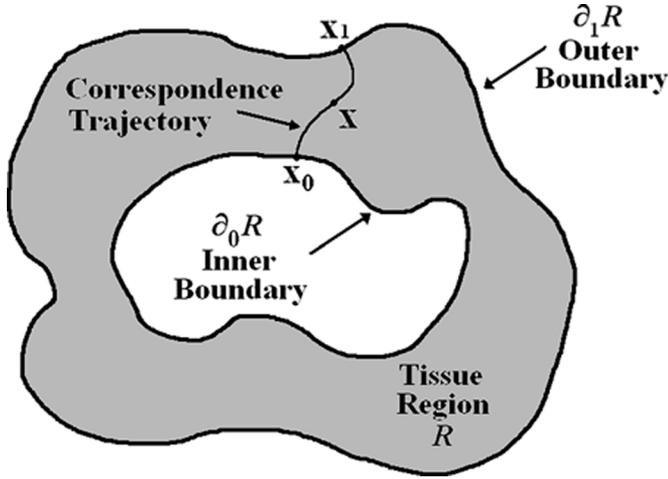


Fig. 1. Inner and outer boundaries of the tissue region R and a correspondence trajectory.

correspondences and gridding. Next, in Section IV, we describe in detail the proposed hybrid Eulerian–Lagrangian approach together with its numerical implementation. Finally, in Section V, we compare the three approaches in terms of the precision of their results and their computational times using two synthetic examples and a cardiac MR image.

II. PREVIOUS WORK

Although there have been many attempts to properly define thickness within an annular region [19]–[21], most of them present certain problems [16]–[18]. For instance, the simple definition of thickness as the smallest Euclidean distance from the point in one surface to any point in the opposite surface has the problem that it lacks reciprocity, that is, the thickness may be different in the case where the surfaces are interchanged. Jones *et al.* [16] defined thickness as the length of the flow lines of a harmonic function that is equal to 0 in one of the surfaces and equal to 1 in the other. The advantage of Jones’ method is that these flow lines, which we call *correspondence trajectories*, have the highly desirable properties that they are orthogonal to each one of the surfaces, they do not intersect each other, and they are nominally parallel.

Let $R \subset \mathbb{R}^n$ for $n = 2, 3$, be a spatial region with a simply connected inner boundary $\partial_0 R$ and outer boundary $\partial_1 R$ (see Fig. 1). These boundaries have a sub-voxel resolution and are usually given as level set representations of some given functions. Now let u be a harmonic function in R such that $u(\partial_0 R) = 0$ and $u(\partial_1 R) = 1$. The normalized gradient vector field of u coincides with the tangent vector field of the correspondence trajectories and is given by

$$\vec{T} = \frac{\nabla u}{\|\nabla u\|}. \quad (1)$$

For each point $\mathbf{x} \in R$, let the functions $L_0(\mathbf{x})$ and $L_1(\mathbf{x})$ be defined as the lengths of the correspondence trajectories that go from \mathbf{x} to $\partial_0 R$ and from \mathbf{x} to $\partial_1 R$, respectively. Accordingly, the thickness $W(\mathbf{x})$ of R at \mathbf{x} is just

$$W(\mathbf{x}) = L_0(\mathbf{x}) + L_1(\mathbf{x}). \quad (2)$$

There exist several numerical methods that can be used to compute the harmonic interpolant u given by

$$\Delta u = 0 \text{ subject to } u(\partial_0 R) = 0 \text{ and } u(\partial_1 R) = 1 \quad (3)$$

where Δ is the Laplace operator [22]–[24]. The amount of computation needed to solve this partial differential equation depends, of course, on the nature of method used. For instance, if the preconditioned conjugated gradient (PCG) or the multigrid technique is used to solve (3), it would take less than 1 s (in 2-D) or under 10 s (in 3-D). If, instead, a relaxation technique such as Laplacian over Gaussian (LOG) or Gauss–Seidel is used, it would require a couple of seconds (in 2-D) or in the order of 10–20 s in 3-D (depending upon the desired accuracy). Once we have u , the tangent vector field \vec{T} is computed using (1). Fig. 2 shows the harmonic interpolant u and the tangent vector field \vec{T} corresponding to the annulus between a circle and an ellipse. Now we will briefly review the two previous methods, the Lagrangian and the Eulerian PDE approaches, that have been used to compute thickness according to the model in [16].

A. Lagrangian Approach

In Jones’ approach, $L_1(\mathbf{x})$ is computed by integrating \vec{T} from \mathbf{x} to $\partial_1 R$, whereas $L_0(\mathbf{x})$ is computed by integrating $-\vec{T}$ from \mathbf{x} to $\partial_0 R$. Then the thickness of R at \mathbf{x} is obtained using (2). Integration can be carried out using a variety of methods such as Euler and Runge–Kutta integration. For instance, if the Euler integration method is used, then given an initial point \mathbf{x}_0 in R and a chosen step size Δt , a point \mathbf{x} that is closer to the outer boundary $\partial_1 R$ and that is located on the correspondence trajectory passing through \mathbf{x}_0 can be approximated by

$$\mathbf{x} = \mathbf{x}_0 + \Delta \mathbf{x}, \quad \text{for } \Delta \mathbf{x} = \Delta t \vec{T}(\mathbf{x}_0)$$

where $\vec{T}(\mathbf{x}_0)$ represents the value of \vec{T} at \mathbf{x}_0 . This new point can then be used to estimate another one which is even closer to $\partial_1 R$, and so forth. If we continue this way we will obtain a point which is sufficiently close to $\partial_1 R$. Then, the product of the step size Δt and the number of times we had to repeat this procedure will give us an estimate for $L_1(\mathbf{x}_0)$. Similarly, if instead of \vec{T} we use $-\vec{T}$, we will get an estimate for $L_0(\mathbf{x}_0)$. As it can be seen, the overall approach is simple, but computationally expensive since \vec{T} (and $-\vec{T}$) must be repeatedly interpolated in order to extend the current trajectory.

B. Eulerian PDE Approach

In the Eulerian approach, thickness is computed by solving a pair of PDEs that are constructed from the geometry of the problem. From the differential structure of L_0 and L_1 , the following set of PDEs must be satisfied for each point $\mathbf{x} \in R$:

$$\langle \nabla L_0, \vec{T} \rangle = 1, \quad \text{with } L_0(\partial_0 R) = 0 \quad (4)$$

$$\langle -\nabla L_1, \vec{T} \rangle = 1, \quad \text{with } L_1(\partial_1 R) = 0 \quad (5)$$

where $\langle \cdot, \cdot \rangle$ denotes the usual Euclidean inner product. The characteristics of (4) are (by design) equal to the correspondence trajectories. Therefore, the tangent field \vec{T} determines the direction of the characteristics of (4). Similarly, the negative of

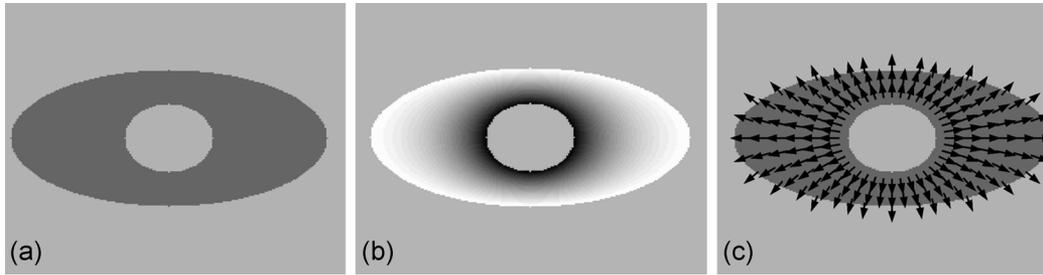


Fig. 2. Synthetic annular region between an ellipse and a circle. (a) Region. (b) Harmonic interpolant. (c) Tangent field.

the tangent field, $-\vec{T}$, determines the direction of the characteristics of (5). Since information flows in the forward direction (downwind), it is important to choose a differencing scheme in which the values of both L_0 and L_1 for a grid point depend upon the corresponding values of L_0 and L_1 in the backward direction (upwind) along the characteristic curve passing through the grid point. Specifically, if T_x , T_y , and T_z are the components of \vec{T} at the grid point (i, j, k) in R , and if the grids are assumed to have spacing $\Delta x = \Delta y = \Delta z = 1$, it is shown in [17] that the numerical finite difference approximations for $L_0[i, j, k]$ and $L_1[i, j, k]$ are shown in (6) and (7), at the bottom of the page, where

$$\begin{aligned} i \pm 1 &= \begin{cases} i + 1, & T_x > 0 \\ i - 1, & T_x < 0 \end{cases} \\ j \pm 1 &= \begin{cases} j + 1, & T_y > 0 \\ j - 1, & T_y < 0 \end{cases} \text{ and} \\ k \pm 1 &= \begin{cases} k + 1, & T_z > 0 \\ k - 1, & T_z < 0. \end{cases} \end{aligned} \quad (8)$$

Three iterative algorithms are proposed in [17] to solve (6) and (7). In all of them the initial values of L_0 and L_1 are set to 0 at all grid points so that values outside R serve as boundary conditions. Although all three methods yield the same solution, their convergence rates are different. The first algorithm proposed is the *iterative relaxation method*, which updates the values of L_0 and L_1 at points inside R using (6) and (7). It can use the Jacobi or the Gauss–Seidel procedure, depending on the way the updated values are used during each iteration. This method is the simplest to implement, but it is also the one with the slowest convergence rate.

The second one is called *ordered transversal* and is very similar to the “fast marching method” used to solve the Eikonal equation [25], [26]. Here, the idea is to solve the grid points in

the order they are reached by the correspondence trajectories, resulting in a very fast convergence rate. Besides speed, this algorithm has the advantage of converging automatically so it is not necessary to test for convergence.

Finally, the third algorithm proposed is called *cyclically alternating Gauss–Seidel*. It is not as fast as the order transversal, but it is easier to implement. It consists in cyclically alternating the order in which the grid points are updated.

III. EXTENSION OF THE EULERIAN APPROACH TO CORRESPONDENCES AND GRIDDING

Besides computing thickness, it is often useful to find the corresponding boundary points, that is, the points $\mathbf{x}_0 \in \partial_0 R$ and $\mathbf{x}_1 \in \partial_1 R$ such that the correspondence trajectory going from \mathbf{x}_0 to \mathbf{x}_1 passes through \mathbf{x} (see Fig. 1). Since the correspondence trajectories have the property that they do not intersect each other, there is only a pair of points, \mathbf{x}_0 and \mathbf{x}_1 , satisfying this condition. Having unique corresponding boundary points in both $\partial_0 R$ and $\partial_1 R$ for every grid point in R allows the generation of anatomically shaped discrete grids within the tissue region that, among other applications, can be used to subdivide the annular tissue, to create a mesh for finite element analysis, and to elaborate coordinates on which determined functions can be reported [18].

Since, for a given \mathbf{x} in R , the corresponding boundary points are just the intersections of the constructed trajectory with the inner and the outer boundaries of R , then they are readily computed using the Lagrangian approach. However, as we mentioned before, the Lagrangian approach is computationally intensive, especially when images are large or 3-D. That is why a preliminary work appeared in a conference proceedings [18], describing a method to compute correspondences within an Eulerian framework. We now elaborate on this extension.

$$L_0[i, j, k] = \frac{1 + |T_x|L_0[i \mp 1, j, k] + |T_y|L_0[i, j \mp 1, k] + |T_z|L_0[i, j, k \mp 1]}{|T_x| + |T_y| + |T_z|} \quad (6)$$

$$L_1[i, j, k] = \frac{1 + |T_x|L_1[i \pm 1, j, k] + |T_y|L_1[i, j \pm 1, k] + |T_z|L_1[i, j, k \pm 1]}{|T_x| + |T_y| + |T_z|} \quad (7)$$

A. Obtaining Correspondences

Let $\phi_0 : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $\phi_1 : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be defined as the correspondence functions that map $\mathbf{x} \in R$ to the inner and outer boundaries of R , respectively. One of the two conditions that both ϕ_0 and ϕ_1 must satisfy is that they must remain constant along the correspondence trajectories, implying that their directional derivatives vanish along the direction given by \vec{T} . For $n = 3$, we can expand the correspondence functions as follows: $\phi_0(\mathbf{x}) = (\phi_0^x(\mathbf{x}), \phi_0^y(\mathbf{x}), \phi_0^z(\mathbf{x}))$ and $\phi_1(\mathbf{x}) = (\phi_1^x(\mathbf{x}), \phi_1^y(\mathbf{x}), \phi_1^z(\mathbf{x}))$. Since all of these functions must remain constant along the direction given by \vec{T} , we have

$$\langle \nabla \phi_i^j, \vec{T} \rangle = 0, \quad \text{for } i = 0, 1 \text{ and } j = x, y, z \quad (9)$$

which can be rewritten in a more compact form as

$$(\nabla \phi_0) \vec{T} = (\nabla \phi_1) \vec{T} = \mathbf{0} \quad (10)$$

where $\nabla \phi_0$ and $\nabla \phi_1$ denote the Jacobian matrices of ϕ_0 and ϕ_1 , respectively. In addition, each correspondence function must map a point on its own boundary to itself as well, which yields another set of conditions, given by

$$\phi_0(\mathbf{x}) = \mathbf{x}, \quad \forall \mathbf{x} \in \partial_0 R \quad (11)$$

$$\phi_1(\mathbf{x}) = \mathbf{x}, \quad \forall \mathbf{x} \in \partial_1 R. \quad (12)$$

Boundary correspondences can be computed by solving (10) subject to the boundary conditions (11) and (12). In [18], the solution is found by using upwind schemes similar the ones used to solve (6) and (7) for computing the thickness of R . Specifically, for $\Delta x = \Delta y = \Delta z = 1$ the resulting finite difference equations are shown in (13) and (14), at the bottom of the page, where the terms $i \pm 1$, $j \pm 1$, and $k \pm 1$ are as defined in (8).

The same iterative procedures used to solve L_0 and L_1 can be used to solve for ϕ_0 and ϕ_1 above, with the only difference being in the initialization procedure. In [18], the initialization is done by making $\phi_0[i, j, k]$ and $\phi_1[i, j, k]$ equal to (i, j, k) at grid points outside R , which are next to the inner and outer boundaries, respectively.

B. Generating Anatomically Shaped Grids

Now that we have shown how to compute thickness and boundary correspondences in annular regions, we would like to be able to subdivide further these regions. A natural first step would be to locate a central curve or surface within the given region.

1) Central Curves and Surfaces: Since the harmonic function u is equal to 0 in $\partial_0 R$ and equal to 1 in $\partial_1 R$, an immediate way to define the central curve or surface of R is to say that it is the locus of points \mathbf{x} such that $u(\mathbf{x}) = 0.5$, which is halfway between the values of u on $\partial_0 R$ and $\partial_1 R$. The values of the harmonic function u generally do not change in the same proportion as does the arclength of the correspondence trajectories, which implies that the points \mathbf{x} such that $u(\mathbf{x}) = 0.5$ are generally not halfway between the boundaries themselves. A straightforward solution to this problem is to define the normalized length function \bar{L}_0 as follows:

$$\bar{L}_0 = \frac{L_0}{L_0 + L_1}.$$

This function is 0 on $\partial_0 R$ and 1 on $\partial_1 R$ and obtains the value 0.5 exactly halfway along its trajectory. The ‘‘central curve’’ or ‘‘harmonic medial axis’’ is defined as

$$\text{Central curve} = \{ \mathbf{x} \in R : \bar{L}_0(\mathbf{x}) = 0.5 \}.$$

Fig. 3 depicts the level sets of two annular regions arising from both the harmonic interpolant u [panels (a) and (b)] and the normalized length function \bar{L}_0 [panels (c) and (d)]. One advantage in using the level sets of u is that they, together with the correspondence trajectories, comprise an orthogonal curvilinear coordinate system within the region R . On the other hand, the level sets of \bar{L}_0 yield a more natural definition for central curves and appear to better capture the preserving the geometric features of the boundaries.

2) Intermediate Correspondences: Let $\phi_A, \phi_B, \phi_C, \dots$, be functions that map any point \mathbf{x} in R to a point in a curve or surface located between the inner and the outer boundary of R , such as the central curve or surface. Similar to ϕ_0 and ϕ_1 , these mappings will still use the correspondence trajectories. They will map any point \mathbf{x} to the point in the same correspondence trajectory that comprises a given function of the trajectory length. Consequently, they must remain constant along the trajectories implying that, as in (10) for ϕ_0 and ϕ_1 , the following systems of linear PDE’s must hold

$$(\nabla \phi_A) \vec{T} = (\nabla \phi_B) \vec{T} = (\nabla \phi_C) \vec{T} = \dots = \mathbf{0} \quad (15)$$

allowing us to compute them without having to explicitly trace any correspondence trajectory.

In order to define the mappings $\phi_A, \phi_B, \phi_C, \dots$, we can parameterize each correspondence trajectory from 0 to 1 and then select fixed parameter values A, B, C, \dots , such that $0 < A <$

$$\phi_0[i, j, k] = \frac{|T_x| \phi_0[i \mp 1, j, k] + |T_y| \phi_0[i, j \mp 1, k] + |T_z| \phi_0[i, j, k \mp 1]}{|T_x| + |T_y| + |T_z|} \quad (13)$$

$$\phi_1[i, j, k] = \frac{|T_x| \phi_1[i \pm 1, j, k] + |T_y| \phi_1[i, j \pm 1, k] + |T_z| \phi_1[i, j, k \pm 1]}{|T_x| + |T_y| + |T_z|} \quad (14)$$

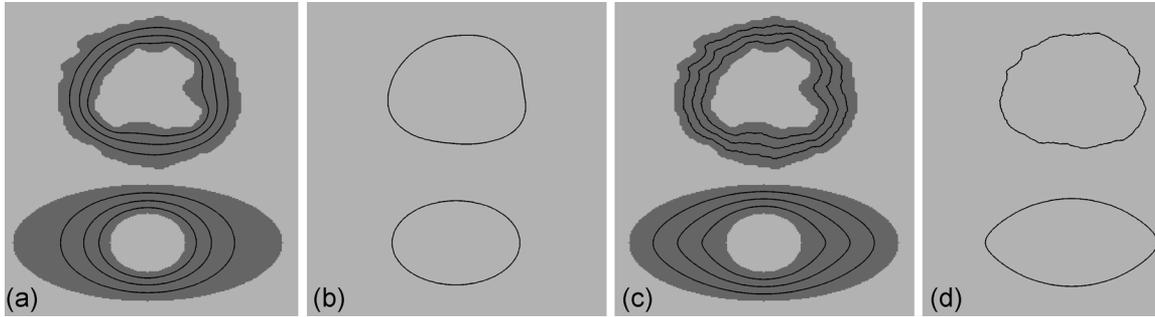


Fig. 3. Comparison of level sets and central curves. (a) Level sets of harmonic interpolant u . (b) Harmonic central curves (level set $u = 0.5$). (c) Level sets of normalized trajectory arclength \bar{L}_0 . (d) Normalized central curves (level set $\bar{L}_0 = 0.5$).

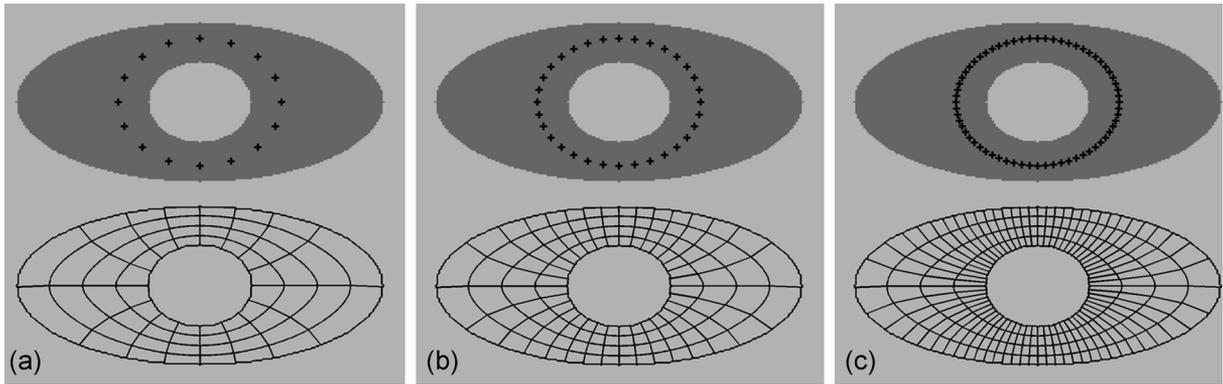


Fig. 4. (Top) Selected seed points and (bottom) the generated grids using the normalized arclength \bar{L}_0 . (a) 16, (b) 32, and (c) 64 seed points.

$B < C < \dots < 1$. This parameterization can be done by simply using the harmonic function u or the normalized length \bar{L}_0 (e.g., if we use the normalized length function \bar{L}_0 and $A = 0.5$, then ϕ_A will map any point \mathbf{x} in R to the central curve). The advantage of using \bar{L}_0 instead of u is that if the former is used, then an uniform spacing between A, B, C, \dots , will produce uniformly spaced points $\phi_A(\mathbf{x}), \phi_B(\mathbf{x}), \phi_C(\mathbf{x}), \dots$, along the correspondence trajectory that passes through \mathbf{x} .

Although the PDEs in (15) are exactly the same as those in (10), they are solved using different boundary conditions and different upwinding schemes. For example, let us suppose that we want to find ϕ_A and that we are using the normalized length function \bar{L}_0 in order to define the mappings. Then, the initial condition for (15) will be $\phi_A(\mathbf{x}) = \mathbf{x}$ for all \mathbf{x} such that $\bar{L}_0(\mathbf{x}) = A$. On the other hand, the upwinding scheme used will depend on whether the value of $\bar{L}_0(\mathbf{x})$ is greater or less than A . If $\phi_A(\mathbf{x}) < A$, then \vec{T} gives the upwind direction; whereas if $\phi_A(\mathbf{x}) > A$, then $-\vec{T}$ gives the upwind direction. As a result, the numerical update that should be used when solving iteratively for ϕ_A is (16), shown at the bottom of the page, where the terms $i \pm 1, j \pm 1$, and $k \pm 1$ are defined as in (8). This equation may be solved by the same iterative methods

used to compute thickness and correspondences. Initially, ϕ_A can be set up to be the identity map at all points. Then, (16) can be used iteratively to compute the desired mapping. We point out that the values of ϕ_A at points \mathbf{x} where $\bar{L}_0(\mathbf{x}) = A$ will be preserved due to the reversal in the flow of information.

3) *Generating a Grid*: Let us suppose that we already have the correspondence maps ϕ_0 and ϕ_1 and the intermediate correspondence maps $\phi_A, \phi_B, \phi_C, \dots$, for some numbers A, B, C, \dots . Now we select a set of n “seed points” $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_n$ located anywhere in R . Then, by construction, the points $\phi_0(\mathbf{x}_i), \phi_A(\mathbf{x}_i), \phi_B(\mathbf{x}_i), \phi_C(\mathbf{x}_i), \dots, \phi_1(\mathbf{x}_i)$, for $i = 1, \dots, n$, belong to the same correspondence trajectory and we can trace a smooth line passing through them. If we do this procedure to all seed points we will get a set of lines resembling corresponding trajectories. Finally, if we add the level sets of the function chosen corresponding to the values of $0, A, B, C, \dots, 1$, we will get a grid representation of R .

Fig. 4 shows examples of this gridding procedure for an elliptic annulus. We chose $A = 0.25, B = 0.50$, and $C = 0.75$ and the normalized length function in order to create the intermediate maps so that the points $\phi_A(\mathbf{x}_i), \phi_B(\mathbf{x}_i)$, and $\phi_C(\mathbf{x}_i)$ be uniformly spaced along the corresponding trajectory. For con-

$$\phi_A[i, j, k] = \begin{cases} \frac{|T_x|\phi_A[i\pm 1, j, k] + |T_y|\phi_A[i, j\pm 1, k] + |T_z|\phi_A[i, j, k\pm 1]}{|T_x| + |T_y| + |T_z|}, & \text{if } \bar{L}[i, j, k] < A \\ \frac{|T_x|\phi_A[i\mp 1, j, k] + |T_y|\phi_A[i, j\mp 1, k] + |T_z|\phi_A[i, j, k\mp 1]}{|T_x| + |T_y| + |T_z|}, & \text{otherwise} \end{cases} \quad (16)$$

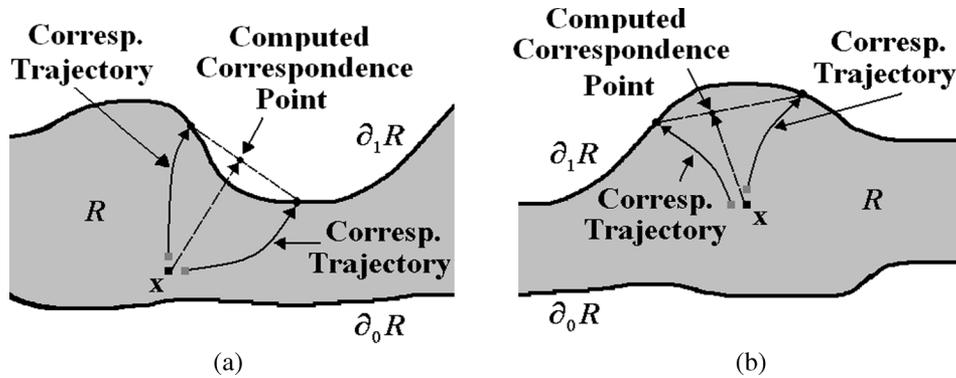


Fig. 5. Point mapped outside the annulus region due to (a) a concavity and to (b) a convexity.

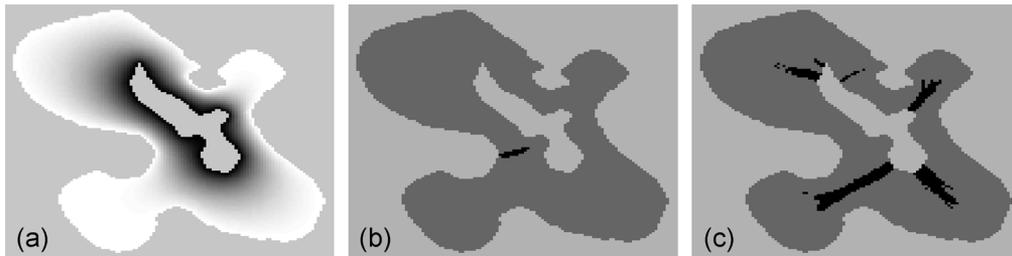


Fig. 6. Problem with the inner and outer boundary correspondence mappings for the Eulerian PDE approach. Regions in black denote points where numerically computed boundary correspondence “misses” the boundary by more than 1 pixel unit. (a) Harmonic interpolant. (b) Inner boundary map. (c) Outer boundary map.

venience, we picked the seed points by uniformly sampling a circumference contained within the annular region (16, 32, and 64 samples from left top to bottom).

C. Advantages and Disadvantages of the Eulerian Approach

The main advantage of the Eulerian approach for both thickness and correspondences is its computational speed—several times faster than the Lagrangian approach, as shown in [17], [18]. On the other hand, as it is going to be shown in Section V, its main disadvantage is that it does not yield the highly accurate results that the Lagrangian approach yields. This is mainly due to two factors. First, in the Eulerian PDE approach for thickness there is a loss of precision when setting up the boundary conditions for L_0 and L_1 . This is mostly the case because both the inner and the outer boundaries exist at an interpixel level and rarely contain a grid point. In the original Eulerian approach, the boundary conditions are established by setting to 0 the value of L_0 at the exterior grid points next to the inner boundary and setting to 0 the value of L_1 at the exterior grid points next to the outer boundary. When doing this, the algorithm implicitly assumes an outer boundary that is shifted a little bit outward and an inner boundary that is shifted a little bit inward. As a result, the computed thickness tends to be larger than what it should be. This is especially problematic when there are just a few grid points between the inner and the outer boundary. Similarly, in the Eulerian PDE approach for correspondences, the boundary conditions for ϕ_0 and ϕ_1 at grid points are set up to be equal to their coordinate positions, creating a similar undesirable effect as above.

We may encounter another problem when solving (13) and (14), as well. As illustrated in Fig. 5, it is possible to have the corresponding boundary points for grid point \mathbf{x} 's neighbors

lying far apart on the boundary. Careful examination of (13) and (14) reveals that the computed values of ϕ_0 and ϕ_1 at \mathbf{x} are convex combinations of ϕ_0 and ϕ_1 at grid points that are neighbors of \mathbf{x} . Therefore, it is possible for \mathbf{x} to be mapped either far outside the boundary Fig. 5(a) or far inside the boundary Fig. 5(b), depending on the boundary's curvature. This problem can be quite severe, especially for objects having highly detailed boundaries, as illustrated in Fig. 6. Fig. 6(b) shows a black region comprising points whose computed map ϕ_0 yields points that are mapped at least 1 pixel away from the inner contour. Similarly, Fig. 6(c) shows black regions comprising points whose computed map ϕ_1 yields points that are mapped and at least 1 pixel away from the outer contour. The maps computed this way do not have pixel accuracy, much less the sub-pixel that is sometimes required in applications.

IV. HYBRID EULERIAN-LAGRANGIAN APPROACH

It is logical to ask whether there is a way to naturally blend the Lagrangian and Eulerian PDE approaches so that the resulting method yields more accurate results than the Eulerian PDE approach, while requiring less computation time than the Lagrangian approach. Toward this end, we modify our previous Eulerian approach to obtain a new hybrid algorithm with prescribable accuracy and the minimal possible sacrifice in speed.

A. Boundary Conditions

The first step to increase the accuracy of the Eulerian PDE approach is to improve the boundary conditions of the PDEs involved. We do this by using the Lagrangian approach to compute the values of L_0 and ϕ_0 at the grid points of R located immediately next to the inner boundary. Similarly, we use the Lagrangian approach to compute the values of L_1 and ϕ_1 at

grid points immediately next to the outer boundary. Once we have computed these values, we use the Eulerian PDE approach to solve for L_0 , L_1 , ϕ_0 , and ϕ_1 at the remaining grid points. In doing so, not only we obtain more accurate values near the boundaries, but also we avoid propagating larger computational errors throughout the whole region R . Since these grid points are at most one grid away from the boundary, the explicit computation of their correspondence trajectories do not require extensive computations.

B. Obtaining Correspondences

Now that the initial conditions for the Eulerian approach have been improved, the next step is to guarantee that points be mapped as closely as desired to the boundaries. This can be done by making some changes to the order transversal algorithm proposed in [17]. In this algorithm, points are visited in the order that they are reached by the correspondence trajectories as they flow away from the known boundary. As a result, only one full sweep through the grid points in R is required to solve for L_0 and ϕ_0 followed by one other sweep, but in a different direction, for L_1 and ϕ_1 .

Let λ be a chosen tolerance constant. As we will explain shortly, λ will provide a means to control the precision of the proposed hybrid approach. In the 2-D case, when computing $\phi_\alpha[i, j]$ (where α could be either 0 or 1) the idea consists of obtaining first the distance between the initially calculated values of $\phi_\alpha[i \pm 1, j]$ and $\phi_\alpha[i, j \pm 1]$. If this distance is less than λ , then we can assume that their correspondence trajectories do not diverge a lot as they approach the boundary and that the error accumulation is small, resulting in a very good precision. If, however, the distance is equal to or larger than λ , then we will need to compute $\phi_\alpha[i, j]$ using another technique. One thing we can do is just use the Lagrangian approach and follow the correspondence trajectory for this particular grid point until we reach the corresponding boundary. By doing so, we are taking full advantage of the Lagrangian approach and getting an accurate value. However, a faster way would be to follow the correspondence trajectory until it reaches a horizontal or vertical line between two grid points that have already been solved such that the distance between the two boundary maps is less than the tolerance λ . If they are close enough, we use the linear interpolation implicit in the discretized Eulerian PDE approach to estimate the value of $\phi_\alpha[i, j]$ at that point (since ϕ_α is constant along the correspondence trajectories, this is the value of ϕ_α at the original grid point), otherwise we continue following the trajectory and doing the same procedure until we find two correspondences that are close enough according to the desired tolerance.

We stress out that in the proposed hybrid approach we use the Lagrangian approach just when needed (e.g., whenever the Euclidean distance between $\phi_\alpha[i \pm 1, j]$ and $\phi_\alpha[i, j \pm 1]$ is greater than or equal to λ); otherwise, we use the Eulerian PDE approach. Therefore, in general, we will not have to follow the whole correspondence trajectory. As a result, depending on the value of λ , the proposed procedure can give us results that are as accurate as the Lagrangian approach or as fast as the Eulerian PDE approach.

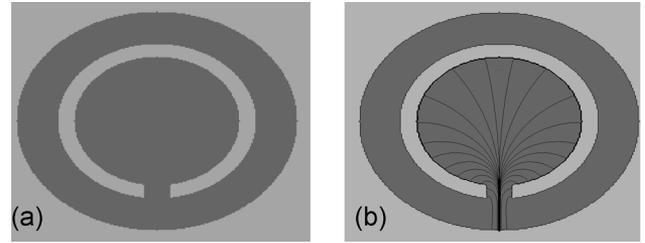


Fig. 7. Synthetic annulus. (a) Region R . (b) Some correspondence trajectories.

C. Computing Thickness

As can be seen from (6) and (7), the values of L_0 and L_1 are computed in a similar way to those of ϕ_0 and ϕ_1 . Consequently, it can be expected that the computed values for L_0 and L_1 will be better approximations to the real values whenever the Euclidean distances between ϕ_0 and ϕ_1 at the grid points involved in the estimations are less than λ than when they are greater than or equal to λ . Therefore, we can improve the precision of L_0 and L_1 if we compute them at the same time and in the same way as we compute ϕ_0 and ϕ_1 in the algorithm described above, taking into account that we have to add the arclength of the followed trajectory to the computed values.

D. Algorithm Description

The combined hybrid Eulerian–Lagrangian approach to compute L_0 and ϕ_0 is summarized as follows.

Algorithm (Hybrid Eulerian–Lagrangian Approach)

- Step 1) Initially tag all grid points in R as UNVISITED.
- Step 2) Use the Lagrangian approach to compute the values of L_0 and ϕ_0 at grid points in R adjacent to the boundary $\partial_0 R$ (grid points with at least 1 neighbor outside R) and re-tag them as SOLVED.
- Step 3) Use (6) and (13) to compute the values of L_0 and ϕ_0 at grid points in R next to the points already tagged as SOLVED, tag them as VISITED, and put them into a heap sorted by the values of u .
- Step 4) Grab the grid point from the top of the current heap of VISITED points (i.e., the grid point with the smallest value of u). Remove this point from the heap and tag it as SOLVED.
- Step 5) If the distance between the correspondence values of ϕ_0 at the neighboring grid points used in (6) is less than the desired tolerance λ , then compute ϕ_0 and L_0 using (6) and (13) and go to Step 7, else set the arclength variable δ to 0.
- Step 6) Follow the correspondence trajectory at the current grid point until it intersects the horizontal or vertical grid line between two grid points tagged as SOLVED and located 1 cell away from each other, record its arclength, and replace the value of δ with the sum of δ and the recorded arclength. If the distance between the values of ϕ_0 at these new grid points is greater than or equal to λ , then go to Step 6), else compute ϕ_0 using linear interpolation and assign the resulting value to ϕ_0 of the original

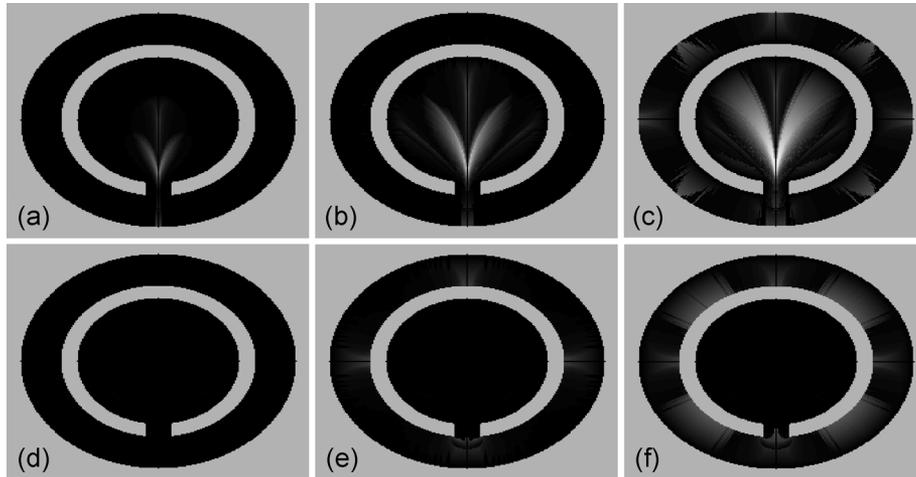


Fig. 8. Number of times, in terms of grid cells, that the trajectories are followed when using the Eulerian–Lagrangian approach for different values of λ . (a)–(c) Number of times the trajectories are followed when computing the inner contour correspondences for λ equal to 2, 1, and 0.5, respectively. (d)–(f) Number of times the trajectories are followed when computing the outer contour correspondences for λ equal to 2, 1, and 0.5, respectively. Whitest point equal to 128 grid cells.

grid point of Step 5). Compute L_0 using linear interpolation and add the value of δ to it. Assign this new computed value to L_0 of the original grid point of Step 5).

- Step 7) Update the values of L_0 and ϕ_0 using (6) and (13) for whichever neighbors of this grid point are not yet tagged as SOLVED. If any of these neighbors are currently tagged as UNVISITED, re-tag them as VISITED and add them to the current heap of VISITED grid points.
- Step 8) Stop if all points in R have been tagged SOLVED, else go to Step 4). ■

The algorithm for computing L_1 and ϕ_1 is almost the same as above with minimal differences: in Step 2), we use the Lagrangian approach to compute the values of L_1 and ϕ_1 at points in R next to the boundary $\partial_1 R$, and in Step 3) the heap should be sorted by the values of $1 - u$ instead of just u .

Fig. 7 depicts a synthetic region and some of its correspondence trajectories. Fig. 8 presents maps showing the number of times, in terms of grid cells, that the trajectories are followed for a particular 224×224 synthetic region. As the chosen tolerance λ gets smaller, more trajectories needed to be followed. We also observe that most regions in the maps are dark, meaning that good accuracy could be obtained by just using the Eulerian PDE approach and as a result none or few trajectories needed to be followed and so reducing the required computational time.

For the 3-D implementation of the hybrid Eulerian–Lagrangian approach, we observe that there are three grid points involved in each one of (6), (7), (13), and (14); consequently, we must compute the distances between the values of ϕ_0 (or ϕ_1) at each of these points and the others two (there are just three cases) and then compare the maximum of these distances with the desired tolerance λ . In addition, we must follow the trajectory until it intersects a square plane parallel to one of the three coordinate planes and formed by four neighbor grid points which are already solved.

Besides computing thickness and correspondences at the same time and in a very fast way, the hybrid Eulerian–Lagrangian approach has two more important advantages. First, it terminates automatically so we do not have to keep testing for convergence. And second, the desired tolerance λ gives us a way to control the accuracy of the computed values. If λ is very large, the hybrid Eulerian–Lagrangian approach will be nothing more than the Eulerian PDE approach with improved initial conditions, whereas if λ is 0, the hybrid Eulerian–Lagrangian approach will yield the same results as the Lagrangian approach.

Finally, we mention that the application of the hybrid Eulerian–Lagrangian approach to compute gridding is straightforward. First we need to find the initial boundary conditions using the Lagrangian approach and then we just adapt (16) to the hybrid Eulerian–Lagrangian approach above.

V. EXPERIMENTAL RESULTS

In this section, we compare the performance of the hybrid Eulerian–Lagrangian approach with those of the Eulerian and the Lagrangian approaches in terms of accuracy and computational speed for three experimental regions.

In the first part of the section, we briefly describe a novel numerical method [27], called harmonic embedding, used to compute a very accurate estimation of the harmonic interpolant u within an annular region R . We will use this method, together with the Lagrangian approach with as much precision as possible, to compute highly accurate estimates of thickness and correspondences to serve as a baseline for comparing the results of the three approaches on images for which the real values for thickness and correspondences are unknown in a simple closed form.

To measure the accuracy of the computed correspondences in the experiments we define the correspondence distance error as the average, over all grid points inside R , of the Euclidean distances between ϕ_0 and the most accurately measured corresponding inner boundary point and between ϕ_1 and the most accurately measured corresponding outer boundary point. Unless

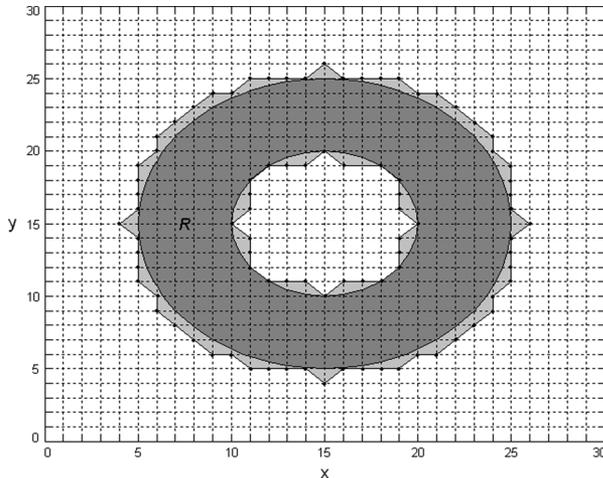


Fig. 9. Boundary set up problem for a region R between two concentric circumferences whose radii are slightly greater than 5 and 10. Since no grid point is located exactly on any of the boundaries, u is solved for the region formed by the light and dark grays instead of just the dark gray.

otherwise stated, the tolerance value λ is set to 1 for all experiments. We used the ordered traversal algorithm in our implementation of the Eulerian approach. All algorithms were implemented in C/C++ on a 2.52-GHz Pentium IV computer running Windows.

A. Pseudo Ground Truth: Harmonic Embedding

In this work, the definition of thickness and correspondences for an annular region R are based on the gradient flow of a harmonic interpolant u that is equal to 0 in the inner boundary $\partial_0 R$ and equal to 1 in the outer boundary $\partial_1 R$. Therefore, the more accurate the estimation of u is, the better the estimations of thickness and correspondences are going to be. When computing its values, the harmonic function u can be approximated by just assigning 1s to the grid points on or outside $\partial_1 R$ and 0s to the grid points on or inside $\partial_0 R$ and then solving iteratively for grid points inside R using any appropriate iterative method such as Gauss, Gauss–Seidel, and Multigrid. The obvious advantage of this method is that it is very easy to implement plus it is very fast. On the other hand, its main disadvantage is that the resulting harmonic function is most of the times different from 1 on the given outer boundary and different from 0 on the given inner boundary due to the fact that those boundaries are typically interpixel (see Fig. 9). As a result, the harmonic function computed in this manner is slightly different from the one we really wish to find. We may overcome this problem by using the technique proposed in [27], called harmonic embedding, which is summarized below.

Let $P_0 = \{p_{0,i}\}_{i=1,\dots,m}$ be the set of grid points in the inner boundary $\partial_0 R$ or just next to it, but not in R . Similarly, let $P_1 = \{p_{1,j}\}_{j=1,\dots,n}$ be the set of grid points in the outer boundary $\partial_1 R$ or just next to it, but not in R . In addition, let P be the set of all grid points located in R . We now define a set of basis functions $u_{0,i}$ and $u_{1,j}$ satisfying the following conditions:

$$\begin{cases} \Delta u_{i,j}(x,y) = 0, & \text{for } (x,y) \in P \\ u_{i,j}(x,y) = 1, & \text{for } (x,y) = p_{i,j} \\ u_{i,j}(x,y) = 0, & \text{for all other } P_0 \text{ and } P_1. \end{cases}$$

We observe that both $u_{0,i}$ and $u_{1,j}$ can be easily solved using any iterative procedure such as the ones previously mentioned. Now let u be a linear combination of these basis functions, i.e.,

$$u = \sum_{i=1}^m \alpha_{0,i} u_{0,i} + \sum_{j=1}^n \alpha_{1,j} u_{1,j}$$

where $\alpha_{i,j} \in \mathfrak{R}$. By construction, u is harmonic because it is a linear combination of harmonic functions. So, now the problem consists in finding the appropriate coefficients $\alpha_{i,j}$ so that $u(\partial_0 R) = 0$ and $u(\partial_1 R) = 1$. This is done by a gradient descent procedure described in [27].

To demonstrate the accuracy of the harmonic embedding technique, we applied it to the annulus between two concentric circles of radii 40 and 80 (all units in pixels) shown in Fig. 10, where the known closed form expression for u in this region, subject to the boundary conditions $u(\partial_0 R) = 0$ and $u(\partial_1 R) = 1$, is given by

$$u(x,y) = \frac{\log(\sqrt{x^2 + y^2}) - \log r_1}{\log r_2 - \log r_1}$$

where r_1 and r_2 are the radii of the inner and outer circles, respectively. The average relative error, over the 15,056 pixels inside R , of the computed harmonic function using harmonic embedding was just 0.055%, which is very small compared to the 7.54% we got without using harmonic embedding. This demonstrates the precision of this recent technique, giving us confidence in its results on the remaining experiments for which the closed form solution of u is unknown.

B. Circular Annulus

We tested the three different approaches on Fig. 10. Clearly, the thickness at any point in R is just the difference between the two radii, i.e., it is equal to 40 pixels. In addition, the corresponding boundary points for any \mathbf{x} in R are just the intersections of the line passing through \mathbf{x} and the center of the circles with the inner and outer circumferences. As a consequence, we can compare the computed thickness and correspondences with the exact values.

The average computed thickness was equal to 40.01, 40.05, and 40.83 for the Lagrangian, hybrid Eulerian–Lagrangian, and Eulerian approach, respectively, whereas the correspondence distance error was 0.10, 0.12, 0.41 pixels, respectively. As expected, the accuracy of the hybrid Eulerian–Lagrangian approach was much better than that of the Eulerian approach, but not better than that of the purely Lagrangian approach. However, the computational times tell a different story, with the Eulerian approach being the fastest at 0.69 s and the Lagrangian approach being the slowest at 4.39 s: the Eulerian–Lagrangian approach took just 1.11 s. Clearly, with the proposed hybrid approach we can get the best of both worlds: the precision of the Lagrangian approach with almost the same speed as the Eulerian approach. Fig. 10 shows the computed values of L_0 and L_1 using the hybrid Eulerian–Lagrangian approach. Also shown is the computed thickness, which values ranged from 39.98 to 40.15.

Applying the harmonic embedding technique together with the Lagrangian approach with a very fine resolution for (18) lead us to very accurate results. In particular, the computed length

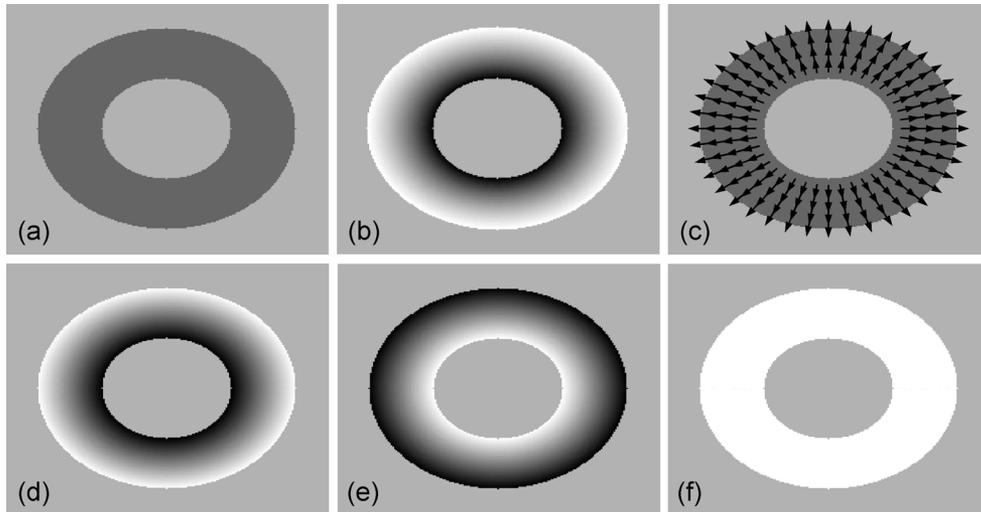


Fig. 10. Thickness computations for a synthetic annular region between two concentric circles of radii 40 and 80. (a) Circular annulus. (b) Harmonic interpolant. (c) Tangent field \vec{T} . (d) Length L_0 . (e) Length L_1 . (f) Thickness $(L_0 + L_1)$.

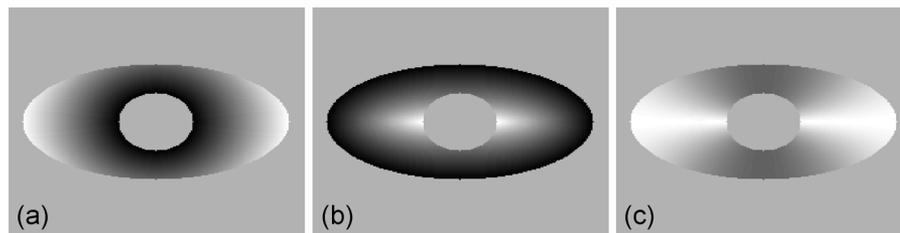


Fig. 11. Thickness computations for a synthetic annular region between an ellipse and a circle. (a) Length L_0 . (b) Length L_1 . (c) Thickness $(L_0 + L_1)$; the whiter the region, the thicker it is.

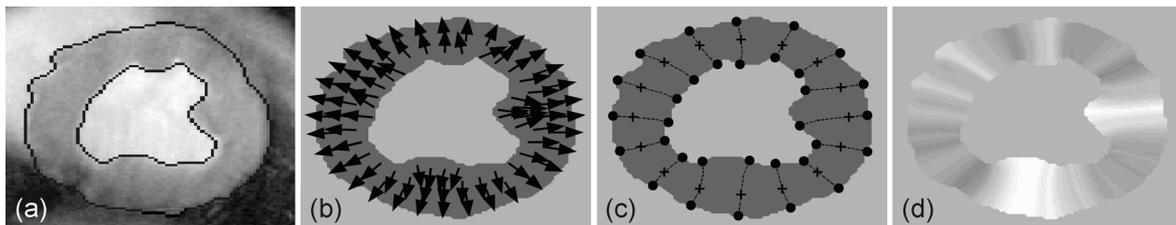


Fig. 12. Myocardial thickness from a short-axis MR image. (a) Endocardial and epicardial contours. (b) Tangent field. (c) Boundary correspondences for some selected points. (d) Thickness; the whiter the region, the thicker it is.

ranged from 3.98 to 40.02 with an average of 40.00. On the other hand, the correspondence distance error was of just 0.0065 pixel. These results supports the assumption that we are going to make for the next experiments in which we will use the results of this technique as a near-exact solution for baseline comparison.

C. Elliptic Annulus

We again tested the three approaches in another synthetic region, shown in Fig. 11. This time, the region R is the annulus between a circle of radius 25 and an ellipse with minor and major radii of 50 and 90. Comparing to what we assumed to be the exact solution, we got an average relative thickness error, over the 12 148 pixels inside R , of 0.078%, 0.26%, and 2.24% for the Lagrangian, hybrid Eulerian–Lagrangian, and Eulerian approach. Additionally, the correspondence distance errors were 0.13, 0.14, and 0.52 pixels, respectively. Finally, the computational times were 3.97, 0.91, and 0.58 s, respectively. Again, the

hybrid Eulerian–Lagrangian approach required much less time than the Lagrangian approach, while the accuracy of the latter was better than that of the former. Fig. 11 depicts the results from the hybrid Eulerian–Lagrangian approach.

D. Myocardium

In this final experiment, we applied all three methods to a 160×160 segmentation of the myocardium obtained from a short-axis MR image of the heart shown in Fig. 12(a). Fig. 12(b) shows a subsample of the tangent vector field computed from the solution of Laplace’s equation. Fig. 12(c) shows the correspondence trajectories for some selected points using the hybrid Eulerian–Lagrangian approach, evidencing the need to form curved correspondence trajectories in some parts of the annular region. Finally, Fig. 12(d) shows the computed thickness.

Fig. 13 shows generated grids using the harmonic interpolant u and the normalized length function \bar{L}_0 for three and four uni-

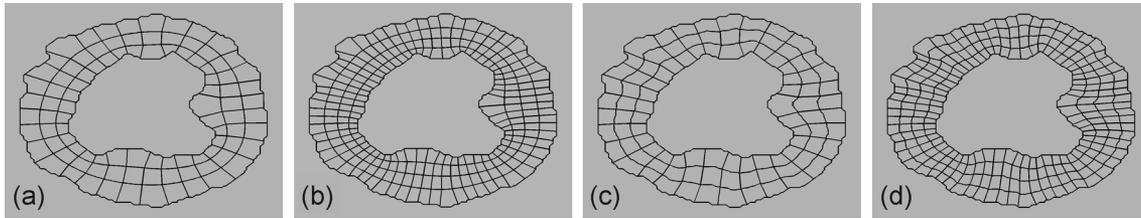


Fig. 13. (a), (b) Generated grids for uniformly spaced fixed parameters using the harmonic function u and (c), (d) the normalized arclength function \bar{L}_0 .

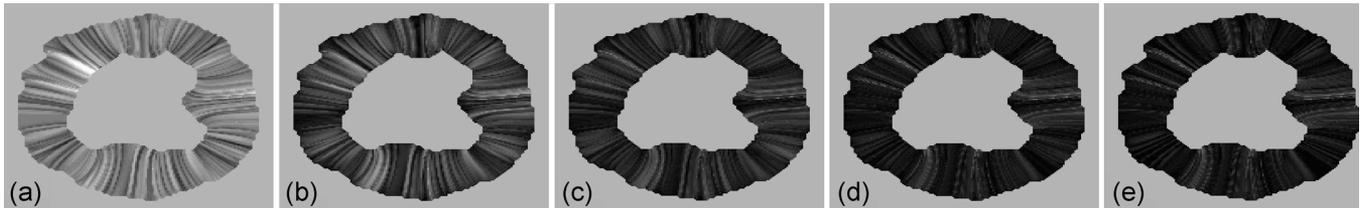


Fig. 14. Correspondence distance error for the different approaches. (a) Eulerian approach—average distance error equal to 0.87 pixel. (b) Hybrid approach with $\lambda = 2$ —0.42 pixel. (c) Hybrid approach with $\lambda = 1$ —0.27 pixel. (d) Hybrid approach with $\lambda = 0.5$ —0.18 pixel. (e) Lagrangian approach—0.16 pixel.

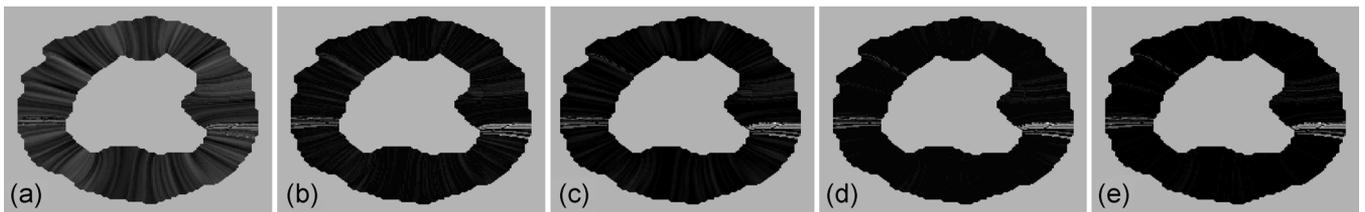


Fig. 15. Thickness relative error for the different approaches. (a) Eulerian approach—average relative error equal to 5.45%, computational time of 0.83 s. (b) Hybrid approach with $\lambda = 2$ —1.51%, 0.89 s. (c) Hybrid approach with $\lambda = 1$ —1.38%, 1.33 s. (d) Hybrid approach with $\lambda = 0.5$ —0.84%, 1.98 s. (e) Lagrangian approach—0.76%, 3.30 s.

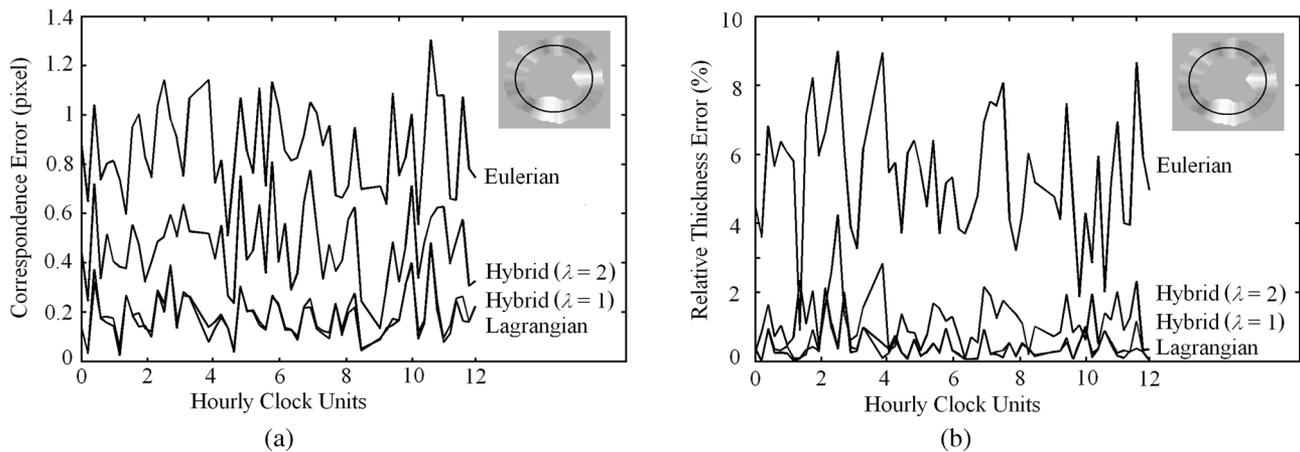


Fig. 16. Plots of (a) correspondence distance error and (b) relative thickness error around an inscribed circle (shown in the upper right-hand-side corners). The horizontal scales denote the location on the circle in “hourly clock units” (1:00, 2:00, etc.), while the vertical scales measure the different errors.

formly spaced parameters, and 32 and 64 seed points obtained by uniformly sampling a circle contained within the annular region. As it can be seen, using the normalized length function \bar{L}_0 produces grids that preserve more the geometric features of the boundaries.

Figs. 14 and 15 depict the correspondence distance errors and the thickness relative errors for the three approaches, including the hybrid Eulerian–Lagrangian approach with different values of tolerance λ . We note that for all cases the errors of the Eulerian are much larger than those of the other approaches.

We also observe that the performance of the proposed approach lies in-between those of the Eulerian and the Lagrangian approaches. Particularly, when $\lambda = 0.5$ the errors of the Lagrangian and hybrid Eulerian–Lagrangian approach are very similar, despite the fact that the hybrid approach required just about half the time required by the Lagrangian approach. Of course, this difference in computational time becomes much more relevant when working with 3-D images.

Fig. 16 shows a different visualization of the correspondence distance error and the relative thickness error as they are read

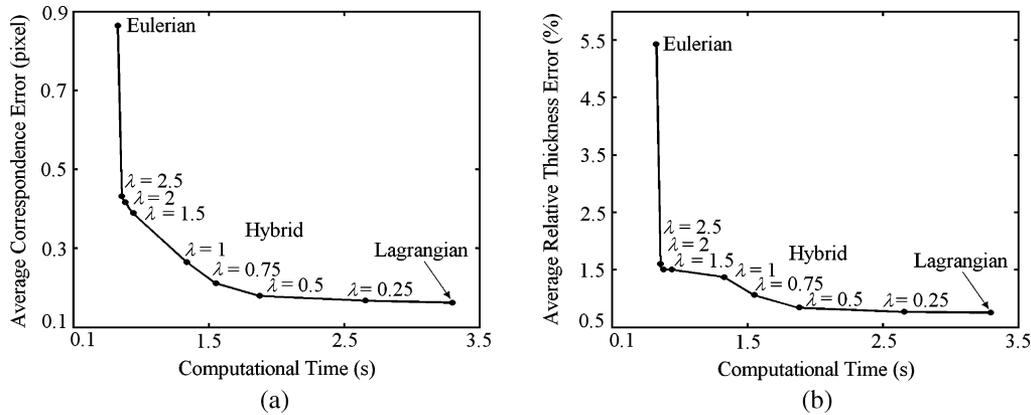


Fig. 17. Comparison of different approaches in terms of the average correspondence distance error, the average thickness relative error, and the computational time. As it can be seen, the performance of the hybrid lies in between that of the Eulerian and the Lagrangian approach.

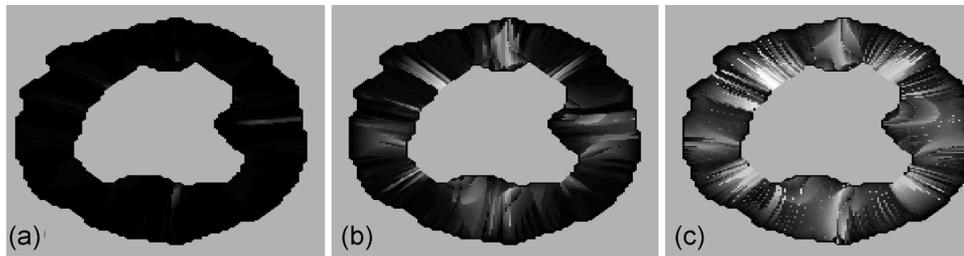


Fig. 18. Number of times the trajectories were followed, in terms of grid cells, when computing both the inner and the outer correspondences for the hybrid approach with different values for λ . (a) $\lambda = 2$. (b) $\lambda = 1$. (c) $\lambda = 0.5$. Whitest point equal to 94 grid cells.

along an inscribed circle (shown in the figure) and plotted in a clockwise manner starting at 12:00. As it can be seen, the hybrid Eulerian–Lagrangian approach produces more errors than the Lagrangian approach, but less than the Eulerian approach.

Two revealing plots are shown in Fig. 17 where the average errors are plotted against the computational times. We observe that while the Lagrangian approach is the most accurate but also the most expensive, the Eulerian approach is the least accurate but also the least expensive, with the hybrid Eulerian–Lagrangian approach lying in-between. If the chosen tolerance λ is very small, then the performance of the hybrid Eulerian–Lagrangian approach is very similar to that of the Lagrangian approach. However, when λ is relatively large, although the required computational time for hybrid approach is almost the same as that of the Eulerian approach, the errors are much larger in the Eulerian approach. The reason for this is that for a big λ the hybrid approach is equivalent to an Eulerian approach with improved boundary conditions and not just an Eulerian approach. This behavior shows up as a discontinuity in the plots of Fig. 17. Fig. 18 depicts maps showing the number of times the trajectories were followed when computing both the inner and the outer correspondences for the hybrid approach for different tolerance values λ . Finally, in Fig. 19 we observe that for $\lambda = 0$ the hybrid approach is 100% Lagrangian, whereas for λ greater than 2 it is almost 0% Lagrangian (i.e., 100% Eulerian).

VI. CONCLUSION

We have presented a new hybrid Eulerian–Lagrangian algorithm for computing thickness, correspondences, and gridding

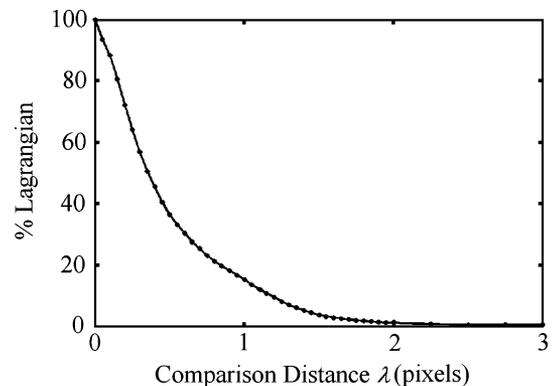


Fig. 19. Prominence of the Lagrangian role in the hybrid algorithm as a function of λ . (Ratio of number of cells visited while explicitly following trajectories by the hybrid approach compared to number of cells that would be visited by explicit trajectory following in the purely Lagrangian approach).

in annular regions. These types of regions are of particular interest in medical imaging since their analysis can be used to early detect certain diseases or estimate functional performance of some parts of the human body. The innovation of the new method lies in the intricate way the Eulerian PDE approach and the Lagrangian approach are combined. These two earlier methods are completely different from each other and the way they can be efficiently and usefully blended is not straightforward at all as is evident from the description of the hybrid algorithm proposed. The whole purpose of this work was to create a practical approach that people would be more likely to use in contrast to the pure Lagrangian and pure

Eulerian schemes previously published. The resulting technique possesses the best of both worlds, namely the speed of the Eulerian PDE approach and the accuracy of the Lagrangian approach, with the additional important (and practical) benefit of giving user precise control over the accuracy and maximum possible speed given that selected degree of accuracy. This makes the proposed method suitable for a much wider range of applications than either the Eulerian or Lagrangian approaches that are currently well known.

REFERENCES

- [1] D. MacDonald, N. Kabani, D. Avis, and A. C. Evans, "Automated 3-D extraction of inner and outer surfaces of cerebral cortex from MRI," *NeuroImage*, vol. 12, pp. 340–356, 2000.
- [2] C. Henery and T. Mayhew, "The cerebrum and cerebellum of the fixed human brain: efficient and unbiased estimates of volumes and cortical surface areas," *J. Anat.*, vol. 167, pp. 167–180, 1989.
- [3] G. Paxinos, *The Human Nervous System*. San Diego, CA: Academic, 1990.
- [4] C. Von Economo and G. Koskinas, *Die cytoarchitektonik der hirnrinde des erwachsenen*. Berlin, Germany: Springer, 1925.
- [5] J. Park, D. Metaxas, and L. Axel, "Analysis of left ventricular wall motion based on volumetric deformable models and MRI-SPAMM," *Med. Image Anal.*, vol. 1, no. 1, pp. 53–71, 1996.
- [6] A. H. Aletras, R. S. Balaban, and H. Wen, "High-resolution strain analysis of the human heart," *J. Magn. Res.*, vol. 140, pp. 41–57, 1999.
- [7] J. Lef *et al.*, "Morphometric analysis of temporal lobe in temporal lobe epilepsy," *Epilepsia*, vol. 7, pp. 727–736, 1998.
- [8] K. Double *et al.*, "Topography of brain atrophy during normal aging and Alzheimer's disease," *Neurobiol. Aging*, vol. 17, pp. 513–521, 1996.
- [9] Y. Grignon, C. Duyckaerts, M. Bennecib, and J. Hauw, "Cytoarchitectonic alterations in the supramarginal gyrus of late onset Alzheimer's disease," *Acta Neuropathol.*, vol. 4, pp. 395–406, 1998.
- [10] J. L. Tanabe *et al.*, "Tissue segmentation of the brain in Alzheimer disease," *Amer. J. Neurocardiol.*, vol. 18, pp. 115–123, 1997.
- [11] B. Fischl and A. M. Dale, "Measuring the thickness of the human cerebral cortex from magnetic resonance images," *Proc. Nat. Acad. Sci.*, vol. 97, no. 20, pp. 1050–1055, 2000.
- [12] E. Lambe, D. Katzman, D. Mikulis, S. Kennedy, and R. Zipursky, "Cerebral gray matter volume deficits after weight recovery from anorexia nervosa," *Ach. Gen. Psychiat.*, vol. 54, pp. 537–542, 1997.
- [13] M. E. DeBaakey and A. M. Gotto, Jr., *The New Living Heart*. Holbrook, MA: Adams Media, 1997.
- [14] S. Pizer, D. Eberly, D. Fritsch, and B. Morse, "Zoom-invariant vision of figural shape: the mathematics of cores," *Comput. Vis. Image Understand.*, vol. 69, no. 1, pp. 55–71, 1998.
- [15] N. R. Clark *et al.*, "Circumferential myocardial shortening in the normal human left ventricle," *Circ.*, vol. 84, pp. 67–74, 1991.
- [16] S. E. Jones, B. R. Buchbinder, and I. Aharon, "Three-dimensional mapping of the cortical thickness using Laplace's equation," *Human Brain Map.*, vol. 11, pp. 12–32, 2000.
- [17] A. J. Yezzi, Jr. and J. L. Prince, "An Eulerian PDE approach for computing tissue thickness," *IEEE Trans. Med. Imag.*, vol. 22, no. 10, pp. 1332–1339, Oct. 2003.
- [18] A. Yezzi and J. L. Prince, "A PDE approach for thickness, correspondence, and gridding of annular tissues," presented at the Eur. Conf. Comput. Vision, Copenhagen, Denmark, 2002.
- [19] M. I. Miller, A. B. Massie, J. T. Ratnanather, K. N. Botteron, and J. G. Csemansky, "Bayesian construction of geometrically based cortical thickness metrics," *NeuroImage*, vol. 12, pp. 676–687, 2000.
- [20] J. C. McEachen, II and J. S. Duncan, "Shape-based tracking of left ventricular wall motion," *IEEE Trans. Med. Imag.*, vol. 16, no. 3, pp. 270–283, Mar. 1997.
- [21] H. D. Tagare, "Deformable 2-D template matching using orthogonal curves," *IEEE Trans. Med. Imag.*, vol. 16, no. 2, pp. 108–117, Feb. 1997.
- [22] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical Recipes: The Art of Scientific Computing*. Cambridge, U.K.: Cambridge Univ. Press, 1988.
- [23] Y. Saad, *Iterative Methods for Sparse Linear Systems*. New York: PWS, 1996.
- [24] W. Briggs, *A Multigrid Tutorial*. Philadelphia, PA: SIAM, 1987.
- [25] J. Sethian, *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision, and Material Science*. Cambridge, U.K.: Cambridge Univ. Press, 1996.
- [26] J. Tsitsiklis, "Efficient algorithms for globally optimal trajectories," *IEEE Trans. Automat. Contr.*, vol. 40, no. 9, pp. 1528–1538, Sep. 1995.
- [27] A. Duci, A. Yezzi, S. K. Mitter, and S. Soatto, "Shape representation via harmonic embedding," in *Proc. Intl. Conf. Comput. Vis.*, Oct. 2003, vol. 1, pp. 656–663.

Kelvin R. Rocha (M'05), photograph and biography not available at the time of publication.

Anthony J. Yezzi, Jr. (M'99), photograph and biography not available at the time of publication.

Jerry L. Prince, photograph and biography not available at the time of publication.