Published in final edited form as:
IEEE Trans Biomed Eng. 2007 October ; 54(10): 1798–1813. doi:10.1109/TBME.2007.895104.

Segmentation of Brain MR Images Using a Charged Fluid Model

Herng-Hua Chang,
Biomedical Engineering IDP and Laboratory of Neuro Imaging (LONI), University of California at Los Angeles, UCLA Radiology Mail Stop 172115, Los Angeles, CA 90095-1721 USA

Daniel J. Valentino,
Biomedical Engineering IDP, Laboratory of Neuro Imaging (LONI), and Division of Interventional Neuro Radiology (DINR), University of California at Los Angeles, Los Angeles, CA 90095-1721 USA and with the Radlink, Inc., Redondo Beach, CA 90278-1103 USA

Gary R. Duckwiler, and
Division of Interventional Neuro Radiology (DINR), University of California at Los Angeles, Los Angeles, CA 90095-1721 USA

Arthur W. Toga
Laboratory of Neuro Imaging (LONI), Department of Neurology, University of California at Los Angeles, Los Angeles, CA 90095-1721 USA

Herng-Hua Chang: emwave@ucla.edu; Daniel J. Valentino: dvalentino@mednet.ucla.edu

Abstract

In this paper, we developed a new deformable model, the charged fluid model (CFM), that uses the simulation of a charged fluid to segment anatomic structures in magnetic resonance (MR) images of the brain. Conceptually, the charged fluid behaves like a liquid such that it flows through and around different obstacles. The simulation evolves in two steps governed by Poisson's equation. The first step distributes the elements of the charged fluid within the propagating interface until an electrostatic equilibrium is achieved. The second step advances the propagating front of the charged fluid such that it deforms into a new shape in response to the image gradient. This approach required no prior knowledge of anatomic structures, required the use of only one parameter, and provided subpixel precision in the region of interest. We demonstrated the performance of this new algorithm in the segmentation of anatomic structures on simulated and real brain MR images of different subjects. The CFM was compared to the level-set-based methods [Caselles *et al.* (1993) and Malladi *et al.* (1995)] in segmenting difficult objects in a variety of brain MR images. The experimental results in different types of MR images indicate that the CFM algorithm achieves good segmentation results and is of potential value in brain image processing applications.

Index Terms

Charged fluid model (CFM); deformable models; electrostatic equilibrium; fast Fourier transform (FFT); finite-size particles (FSP); magnetic resonance imaging (MRI); Poisson's equation; segmentation

Correspondence to: Herng-Hua Chang, emwave@ucla.edu.

I. Introduction

Image segmentation is the partition of an image into several regions of interest such that the contents of each region have similar characteristics. The segmentation of anatomic structures in the brain plays a crucial role in neuroimaging analyses. The complexity of human brain structure mandates the use of computerized approaches derived from computer vision, image analysis, and applied mathematics fields to extract brain data. Successful numerical algorithms in segmenting anatomic structures in neuroimages can help researchers, physicians, and neurosurgeons to investigate and diagnose the structure and function of the brain in both health and disease. However, extracting the ventricle, the brain, and brain tumors in magnetic resonance (MR) images is often highly challenging due to the convoluted shape, blurred boundaries, inhomogeneous intensity distribution, background noise, and low intensity contrast between adjacent brain tissues.

This has motivated the need for segmentation techniques that are robust in applications involving a broad range of anatomic structure, disease, and imaging type [3], [4]. A number of segmentation methods have been proposed to extract specific brain structures, including threshold-, region-, and statistics-based methods, deformable models, atlas-guided techniques, and knowledge-based approaches [3]–[5]. One of the most popular and successful approaches has been the deformable model due to its ability to accurately recover the shape of biological structures in many segmentation applications [6], [7].

Deformable models involve the formulation of a propagating interface, which is a closed curve in 2-D or a closed surface in 3-D, that is moving under a speed function determined by local, global, and independent properties [8]. Given the initial position of a propagating interface and the corresponding speed function, deformable models track the evolution of the interface during the segmentation process. Existing deformable models can be divided into two broad categories: parametric and geometric. Parametric deformable models, originating from the active contour model introduced by Kass *et al.* [9], explicitly represent the interface as parameterized contours in a Lagrangian framework. Active contour models use an energy-minimizing spline that is guided by internal and external energies in such a way that the spline is deformed by geometric shape forces and influenced by image forces. By optimizing the weights used in the internal energy and choosing the proper image forces (e.g., lines or edges), one can use active contour models to evolve the curve toward the boundary of objects being segmented. Subsequently, Cohen and Cohen [10] proposed a 3-D deformable model that uses finite-element methods to deform a balloon-like surface. Xu and Prince [11] introduced an external force called gradient vector flow (GVF) to guide the active contour into boundary concavities and achieve a larger capture range. McInerney and Terzopoulos [12] proposed a new deformable model called topology adaptive snakes (T-snakes) to address the topology limitations of standard snakes by introducing an affine cell image decomposition reparameterization mechanism to notably improve automation.

With the introduction of the level set numerical algorithm developed by Osher and Sethian [13], geometric deformable models enable automatic topological changes without using special tracking procedures. Level set methods use an Eulerian approach to implicitly model the propagating interface using a level set function φ, whose zero-level set always corresponds to the position of the interface [8]. The evolution of this propagating interface is governed by a partial differential equation in one higher dimension. The level set function can be constructed with high accuracy in space and time. The position of the zero-level set is evolved using a speed function that consists of a constant term and a curvature deformation in its normal direction [13]. This speed function can then be incorporated with image-based stopping terms (e.g., image gradient forces) for segmentation purposes. Caselles *et al.* [1]

and Malladi *et al.* [2] proposed geometric deformable contours with an image gradient stopping force (GDCIG) that satisfied the evolution

$$\frac{\partial \varphi}{\partial t} = g(I)(V_0 + \varepsilon\kappa)|\nabla\varphi| \tag{1}$$

where $V_0$and ε are constant weights, k is the level set curvature, ∇ is the gradient operator, and $g(I)$ is the stopping force based upon the image gradient given as

$$g(I) = \frac{1}{1 + |\nabla G_\sigma * I(x, y)|^p}$$

where G_σ is a Gaussian filter with standard deviation σ, $I(x, y)$ is a given 2-D image, and p = 1 or 2. In the previous equation, * represents convolution and $|\cdot|$is the modulus of the smoothed image gradients.

Kichenassamy *et al.* [14] and Yezzi *et al.* [15] added a doublet term $\nabla g \cdot \nabla\varphi$ to (1) to efficiently attract the evolving contour to the desired feature. Siddiqi *et al.* [16] subsequently modified the speed function by adding a term based upon the gradient flow derived from a weighted area energy functional so that the contour could more flexibly evolve toward the desired edges. Recently, Xu *et al.* [17] proposed a level-set-based segmentation method that uses an adaptive triangular mesh to achieve higher resolution at the interface. Gout *et al.* [18] proposed a segmentation approach that combines the idea of the geodesic active contour and interpolation of points in the Osher–Sethian level set framework to find a boundary contour from a finite set of given points.

Traditional level-set-based deformable methods require the contour to be symmetrically initialized somewhere near the center with respect to the boundary of interest [7], [8]. Therefore, it is not easy to successfully segment irregularly shaped structures in medical images. For example, Fig. 1 illustrates that it is difficult to choose an appropriate stopping factor to achieve satisfactory results based upon the image gradient force in (1). Moreover, the stopping criteria in many deformable models (e.g., [14], [16], [17], and [19]) are often determined empirically or are poorly defined. As a consequence, it is difficult to extend and modify these algorithms for practical brain image segmentation problems.

In this paper, we propose a new deformable model, the charged fluid model (CFM), that uses a system of charged elements on a propagating interface to guide the evolution of the contour for brain image segmentation. This electrostatic charged system is governed by Poisson's equation that has only one driving force resulting from the repulsion between fluid elements. In our approach, the electric force is numerically calculated using the finite-size particle (FSP) method implemented via the fast Fourier transform (FFT) algorithm. Two alternate procedures are developed to guide the evolution of the CFM in such a way that the charged fluid behaves like a liquid flowing through and around different obstacles. Our approach can automatically handle topological changes at the interface, in response to the geometry of objects, and provide subpixel precision for the area and length of the segmented region.

The remainder of this paper is organized as follows. In Section II, we describe the theory and system of the CFM algorithm embedded in an electrostatic model governed by Poisson's equation. In Section III, we describe the numerical techniques used to perform the

CFM, including the FSP technique implemented via the FFT algorithm for efficient electric potential computation. In Section IV, we show the evaluation of our approach on simulated, T1-weighted, T2-weighted and proton density (PD)-weighted brain MR images, and compare the segmentation results to the GDCIG method. Finally, in Section V, we discuss the intrinsic properties and essential characteristics of the CFM algorithm and summarize the contributions of this paper.

II. Charged Fluid Model

Biomedical researchers have used physics-based particle systems to investigate and analyze biological models for many years. For example, Wang *et al.* [20], [21] proposed an electrical- field-based method to unravel and trace the convoluted colon by distributing charges along the central colon path. In their model, curved cross sections were defined by the electrical force lines. Now, suppose that a number of charged particles with like sign are initialized inside a region of interest (ROI) in an image for segmentation. The particles will keep advancing outward due to the repelling electric force until they encounter a balancing inward force related to features in the image (e.g., the gradient). However, it is complicated to organize and guide the particles toward the boundary of interest such that the final contour corresponding to the particle positions can accurately and correctly represent the ROI. This will become quite obvious when dealing with noisy images.

It is essential to introduce new approaches with reasonable computation complexity to perform image segmentation using a system of charged particles. Suppose that the charged particles (indicated by the solid dots in Fig. 2) are confined inside an isolated conductor that models a closed propagating interface as a curve (in 2-D) or a surface (in 3-D), such that the particles can only move within the interface until an electrostatic equilibrium is achieved. Following the equilibrium properties (see Table I), the particles accumulate where there is a locally greater curvature in the equilibrium state as shown in Fig. 2. With these new concepts, we will develop our electrostatic deformable model, which we refer to as a charged fluid system.

Each fluid element (the large circles in Fig. 2) has its own charge as if it were calculated by interpolating the charges of the covered particles (the solid dots in Fig. 2). At equilibrium, the electric forces ($\mathbf{F}_{\text{ele}}$ in Fig. 2) are perpendicular to the contour and their magnitudes are proportional to the charges at the corresponding positions. The charged fluid behaves like a liquid such that it flows through and around obstacles as well as deforms in response to external forces $\mathbf{F}_{\text{ext}}$ (e.g., the image gradient) as illustrated in Fig. 2. In Sections II-A–II-C, we describe the theory behind the charged fluid and the evolution procedures that constitute the foundations of our segmentation algorithm, the CFM. We use the notation (x, y) to represent Cartesian coordinates in the spatial domain, (m, n) to represent discrete coordinates in the spatial domain, and (u, v) to represent the corresponding discrete coordinates in the Fourier domain throughout the paper.

A. Charge Density

Let us first describe the behavior of a single fluid element with charge q. Suppose that a fluid element i with zero momentum is advanced a distance d along the direction of an electric force $\mathbf{F}_i$ whenever we advance it. Then, based upon Newton's lawof motion, the distance d is only a function of the electric force F_i, the mass m, and the time interval Δt. In addition, the trivial ratio of q to m can be set to 1.0 and the distance d can be expressed as

$$d=\frac{F_i}{2m}\Delta t^2=\frac{1}{2}E_i\Delta t^2 \tag{2}$$

where E_i is the electric field corresponding to F_i.

Now, assume the fluid element is advanced a distance (d_x, d_y) to position (x_i, y_i), as illustrated in Fig. 3. For computational convenience and efficiency, we approximate the charge density by summing the charges on grids. We do this by interpolating the charge in Fig. 3(a) to the nearest grid point (NGP) and its 4-neighbors using the subtracted dipole scheme (SUDS) [22] associated with the FSP technique [23], [24], which is described in Section III-A.

B. Evolution

Once the charge distribution of an electrostatic system is known, the electric potential Φ can be calculated through Poisson's equation

$$\nabla^2\Phi(\mathbf{r})=-4\pi\rho(\mathbf{r}) \tag{3}$$

where $\rho(\mathbf{r})$ is the charge density. The corresponding electric field **E** can then be computed in terms of the scalar potential Φ

$$\mathbf{E}(\mathbf{r})=-\nabla\Phi(\mathbf{r}). \tag{4}$$

Note that the magnitude of **E** is proportional to $1/r$ in 2-D and $1/r^2$ in 3-D. Therefore, the motion of each fluid element in the electrostatic system for each time step can be updated using (2)–(4). In order to well organize each individual fluid element on the propagating interface of a deformable model, we develop a two-stage evolution algorithm for image segmentation. The first procedure, *charge distribution*, enables the CFM to flow within the propagating interface until a specified electrostatic equilibrium is achieved. The second procedure, *front deformation*, deforms the propagating front into a new shape in response to the electric potential in equilibrium and the image potential, which is related to the image gradient. Those two procedures are repeated until the propagating front resides on the boundary of objects being segmented.

1) Charge Distribution—To advance fluid elements with given charges, we first have to solve the electric potential using (3). In reality, the electrostatic potential varies in response to the positions of elements if the total charge of the system is conservative. This will not be a problem when using a single charged fluid to perform curve evolution since fluid elements keep advancing outward due to the repelling force. However, when using multiple charged fluids, the one having the stronger electric potential dominates the behavior of the overall system. This can dramatically influence the contours of other charged fluids by, for example, repelling the fluid elements in the weaker charged fluids, which will prevent them from reaching the desired object boundary.

One way to solve this problem is to normalize the electric potential for each charged fluid through Poisson's equation. Therefore, we define the normalized electric potential $\widehat{\Phi}^j_{\text{ele}}$ for the charged fluid j as

$$\widehat{\Phi}_{\text{ele}}^{j}=\frac{\Phi_{\text{ele}}^{j}}{\overline{\Phi}^{j}}\Phi_0$$

where Φ_0 is an arbitrary positive constant and $\overline{\Phi}^j$ is the mean electric potential in the charged fluid j. Note that $\overline{\Phi}^j = \Phi_0$ at the beginning of the evolution since there is no potential at all. The corresponding normalized charge density is defined as

$$\widehat{\rho}^{j}=\frac{\rho^{j}}{\overline{\Phi}^{j}}\Phi_0. \tag{5}$$

Therefore, the overall system is governed by the modified Poisson's equation

$$\nabla^2\widehat{\Phi}_{\text{ele}}=\nabla^2\sum_j\widehat{\Phi}_{\text{ele}}^{j}=-4\pi\sum_j\widehat{\rho}^{j}=-4\pi\widehat{\rho} \tag{6}$$

where $\hat{\Phi}_{\text{ele}}$ is the normalized electric potential and $\hat{\rho}$ is the normalized charge density of the overall system at each time step. The numerical solution of Poisson's equation in (6) can be rapidly computed using the FSP method implemented via the FFT algorithm, which is described in Section III-B. During this procedure, the electric field $\mathbf{E}_{\text{ele}}$ is directly computed using the normalized electric potential

$$\mathbf{E}_{\text{ele}}=-\nabla\widehat{\Phi}_{\text{ele}}. \tag{7}$$

Once the electric potential is obtained, the electric field on each fluid element is numerically computed using the central difference approximation of (7).

In order not to cross over possible edges of interest, we further restrict the motion of the fluid element having the maximum electric field in the system such that it is advanced the half grid spacing (h/2) in the electric field direction. The time interval for each fluid element can then be obtained from (2)

$$\Delta t^2=\frac{h}{E_{\max}}=\frac{h}{\left|\sqrt{E_m^2+E_n^2}\right|_{\max}} \tag{8}$$

where $E_{\max}$ is the magnitude of the maximum electric field on the propagating interface of the system for each iteration and E_m and E_n are the components of $E_{\max}$ in the m- and n-axes, respectively. One advantage of this approach is that there is no explicit time interval setting for advancing fluid elements. There is no need to adjust the time interval for specific segmentation problems. It can also reduce the number of parameters in the CFM algorithm.

Substituting (8) into (2), the distances d_x^i and d_y^i for each fluid element i can be easily obtained

$$d_x^i = \frac{E_m^i}{E_{\max}} \frac{h}{2}$$
$$d_y^i = \frac{E_n^i}{E_{\max}} \frac{h}{2} \quad (9)$$

where E_m^i and E_n^i are the electric field components of element i in the m- and n-axes, respectively. Equation (9) implies that varied time intervals related to the maximum electric field in the system are implicitly used to compute the distance that each fluid element advances. Once the distances are obtained, we can use the SUDS technique described in Section III-A to interpolate the charge to the five discrete neighbors as shown in Fig. 3.

Electrostatic equilibrium is defined as a state of zero net flow of electric charge (see Table I). In order for the simulation to converge, we define the condition of electrostatic equilibrium such that a small amount of charge flow is still permitted. This is the case when the following inequality is satisfied:

$$\frac{\Delta Q_{\text{total}}}{Q_{\text{total}}} \leq \gamma \quad (10)$$

where Q_{total} is the total charge of the overall system for each iteration, ΔQ_{total} is the net flowing charge in total, and $\gamma > 0$ determines the degree of electrostatic equilibrium. In our experience, the selection of $0.01 \leq \gamma \leq 0.1$ is suggested because a smaller value of γ can retard the convergence speed without significantly improving the quality of equilibrium, and it is possible for the procedure to oscillate if a very tiny value of γ is used. Note that the system conserves charge throughout the charge distribution procedure, i.e., Q_{total} is the same for each iteration during this process.

At the beginning of this procedure, a uniform charge distribution over the fluid elements is placed on the 2–pixel-wide propagating interface that is obtained in the other procedure, as illustrated in Fig. 4(a). The fluid elements are repeatedly advanced inside the charged fluid until the overall system converges to an equilibrium state that satisfies (10). Note that if any of the 4-neighbors is not within the propagating interface, its charge is interpolated to other neighboring positions based upon the corresponding weights (see Fig. 3).

One of the interesting properties of conductors in electrostatic equilibrium (see Table I) is that the electric field is perpendicular to the curve in 2-D or surface in 3-D as illustrated in Fig. 4(b). Note that the system reaches an electrostatic equilibrium charge distribution, which is related to the equilibrium quality γ and to the geometry of the propagating interface. Also note that the contour remains the same shape. Another important property of an isolated conductor in equilibrium is that any net charge resides entirely on its curve or surface. After the system reaches electrostatic equilibrium in (10), a curve corresponding to a 1–pixel-wide propagating front (contour) is defined such that only fluid elements on that contour are preserved. We quickly generate the 1–pixel-wide front of the CFM using the boundary element detection method described in Section III-C.

2) Front Deformation—The front deformation procedure allows the CFM to interact with the image data such that the1–pixel-wide propagating front deforms the shape in response to the gradient of an image. We define the image gradient potential as

$$\Phi_{\text{img}}(m,n)=\beta\frac{|\nabla G_\sigma(m,n)*I(m,n)|}{|\nabla G_\sigma * I|_{\max}}\Phi_0 \tag{11}$$

where β is a weighting factor ($\beta \geq 0$) to adjust the image gradient potential, $|\cdot|$ is the modulus of the smoothed image gradients, and $|\cdot|_{\max}$ is the maximum modulus in the computation domain. The smoothing of the image is usually performed by convolution with a 3 × 3 or 5 × 5 Gaussian filter kernel.

The image potential Φ_{img} in (11) is incorporated into our electrostatic model by defining the effective potential Φ_{eff}

$$\Phi_{\text{eff}}(m,n) \equiv \widehat{\Phi}_{\text{equ}}(m,n) - \Phi_{\text{img}}(m,n)$$

where $\hat{\Phi}_{\text{equ}}$ is the normalized electric potential in electrostatic equilibrium, which corresponds to $\hat{\Phi}_{\text{ele}}$ when equilibrium is achieved in the charge distribution procedure. In order to deform the propagating front in response to the effective potential, the corresponding effective field $\mathbf{E}_{\text{eff}}$ is defined

$$\mathbf{E}_{\text{eff}}=\mathbf{E}_{\text{equ}}+\mathbf{E}_{\text{img}}=-\nabla\Phi_{\text{eff}}=\nabla\Phi_{\text{img}}-\nabla\widehat{\Phi}_{\text{equ}} \tag{12}$$

where $\mathbf{E}_{\text{equ}}$ is the electric field in equilibrium corresponding to $\hat{\Phi}_{\text{equ}}$ and $\mathbf{E}_{\text{img}}$ is the image field corresponding to Φ_{img}. The effective field components are numerically computed using the central difference approximation.

When the gradient of the image potential is insignificant, i.e., in a relatively homogenous region, the front of the CFM deforms approximately in its normal direction. However, when the gradient of the image potential is larger than that of the electric potential, the directions of the effective fields at those positions are changed dramatically as shown in Fig. 5(a). The charged fluid relies on the salient image gradient to change the direction of $\mathbf{E}_{\text{eff}}$ in (12) such that the fluid elements can be stopped at boundaries of interest. The front deformation is executed on each fluid element by locating binary positions corresponding to the four adjacent grid points based upon the effective field direction as illustrated in Fig. 6. Therefore, we can generate the 2–pixel-wide binary interface by applying this binary localization method to each fluid element as shown in Fig. 5(b). Note that, after this procedure, the propagating interface of the CFM evolves into a different shape in response to the effective field.

C. Subpixel Precision—Most existing deformable model methods provide subpixel precision for the area and length of the ROI. The CFM can also provide subpixel precision, however, the fluid elements are constrained on a grid during the evolution. After the evolution is terminated and the object of interest is detected, the effective fields of all fluid elements are approximately oriented inward based upon (12) as illustrated in Fig. 7(a). However, the effective fields on the rest of the interface [the circles in Fig. 7(a)] are approximately oriented outward. Therefore, the true boundary of the ROI is somewhere inside the final propagating interface; this is similar to the localization of the zero-level set function. We locate the boundary with subpixel precision by advancing the fluid element a real number distance computed by substituting the effective fields into (9). Now, the fluid elements in Fig. 7(b) are located on real number points and the precision required for a

specific application is obtained by using the appropriate interpolation techniques [25], [26] to calculate the area and length of the ROI.

III. Numerical Implementation

A. SUDS

Systems of charged particles have been widely studied in the physics community for decades [24]. Computer simulation and modeling methods have been extensively used to investigate the behavior of such systems [27], [28]. One computationally efficient approach to calculate the charge density is to arrange the particles to the grid points via the FSP method [23], [24]. Consider a collection of FSPs in 2-D. The charge density ρ can be expressed as

$$\rho(x,y)=\sum_i q_i S(|\mathbf{r}-\mathbf{r}_i|) \tag{13}$$

where q_i and $\mathbf{r}_i = (x_i, y_i)$ are the charge and location of the ith particle as shown in Fig. 3, and $S(r)$ is the shape factor giving the way a particle charge is distributed about its center. The shape factor is not required to be isotropic or symmetric but it usually is [23]. In this paper, we assume that the shape factor is real and isotropic as [24], [29]

$$\begin{aligned} S(r) &\equiv \frac{1}{2\pi}\exp\left(-\frac{r^2}{2}\right) \\ &= \frac{1}{2\pi}\exp -\frac{1}{2}[(x-x_i)^2+(y-y_i)^2]. \end{aligned} \tag{14}$$

We now perform a multipole expansion of ρ in (13) with respect to the NGP location using Taylor series and truncate the expansion at the dipole term [22], [28]. This gives us a dipole expansion approximation and replaces the sum over particles by a sum over grid points as [22], [24]

$$\rho(x,y) \cong \sum_g \sum_{i\in g} q_i \left[S(x-x_g, y-y_g) - (x_i - x_g) \times S_x(x-x_g, y-y_g) - (y_i - y_g) \times S_y(x-x_g, y-y_g) \right]$$

where (x_g, y_g) is the NGP location. Finally, let us approximate the derivatives of using the central difference method over the adjacent grids

$$\rho(x,y) \cong \sum_g S(x-x_g, y-y_g) \times \left\{ Q_{\mathrm{NGP}}(x_g, y_g) + \frac{1}{2h}[D(x_{g+1}) - D(x_{g-1}) + D(y_{g+1}) - D(y_{g-1})] \right\} \tag{15}$$

where Q_{NGP} is the monopole charge contribution

$$Q_{\mathrm{NGP}} = \sum_{i\in g} q_i$$

and $D(\xi_g)$ is the dipole charge contributions

$$D(\xi_g)=\sum_{i\in g} q_i(\xi_i-\xi_g).$$

Using this approximation based upon (15), we can obtain the charge density by summing the charge over the corresponding five neighboring grids on each fluid element.

B. Electric Potential Computation

Having charges distributed on a uniformly spaced grid enables rapid numerical solution of Poisson's equation [see (6)] by using the FSP model [23]. For a detailed description of electrostatic particle models, please refer to [24], [27], and [28]. The FSP method reduces collision effects between particles and avoids singularities in the use of Coulomb's law when the distance is zero, while retaining long-distance charge interactions. We now have a system of charged particles having charge $Q(m, n)$ on grid (m, n) in a discrete spatial domain. To numerically solve Poisson's equation, we first compute the discrete Fourier transform (DFT) of the charge $Q(m, n)$

$$Q(u,v)\equiv\frac{1}{L_mL_n}\sum_{m=0}^{L_m-1}\sum_{n=0}^{L_n-1}Q(m,n)\times\exp\left[\frac{-2\pi ium}{L_m}\right]\exp\left[\frac{-2\pi ivn}{L_n}\right] \tag{16}$$

where L_m and L_n are the lengths along the m- and n-axes, respectively. Substituting (14) into (13) and taking the Fourier transform of it, we can obtain the relationship between ρ and Q in the Fourier domain. Then, applying the same technique to (6) to relate ρ to Φ in the Fourier domain, we can finally establish the relationship between Φ and Q in the Fourier domain through a simple arithmetic function.

Now, taking the inverse Fourier transform of Φ and assuming that the electric potential Φ is evaluated only on grid points and interpolated between them, we can facilitate the computation and obtain the discrete potential $\Phi(m, n)$ through Poisson's equation [29], [30]

$$\Phi(m,n)=\sum_{u=0}^{L_m-1}\sum_{v=0}^{L_n-1}{}'\frac{Q(u,v)}{\pi\left(\frac{u^2}{L_m^2}+\frac{v^2}{L_n^2}\right)}\times\exp\left[-2\pi^2\left(\frac{u^2}{L_m^2}+\frac{v^2}{L_n^2}\right)\right]\times\exp\left[\frac{2\pi ium}{L_m}\right]\exp\left[\frac{2\pi ivn}{L_n}\right] \tag{17}$$

where the prime represents that $u = v = 0$ is excluded from the sum. The DFT pair in (16) and (17) can then be rapidly computed via the FFT algorithm provided that $L_m = 2^s$and $L_n = 2^t$ with s and t positive integers.

C. Boundary Element Detection

The 2–pixel-wide interface was refined to a 1–pixel-wide front by boundary element detection using a boolean array corresponding to the image dimension. An initial boolean value was assigned to each corresponding pixel based upon the following rules: *true* if it was inside the initial contour and *false* if it was outside. *True* pixels were reset to the position of the new fluid element during curve evolution. An examination was performed at each time step by checking the boolean values of the 3 × 3 neighboring positions of each element on the 2–pixel-wide interface. If the boolean value of any of the eight neighbors was *false*, the fluid element was treated as a boundary element, which constitutes the 1–pixel-wide front. If the boolean values of all neighbors were *true*, the fluid element was treated as an inner

element and discarded. Using this simple technique, the 1–pixel-wide front was quickly constructed and the fluid elements were connected by 4-connectivity as illustrated in Fig. 5(a). Note that the topological changes of fronts for multiple charged fluids were also handled during this procedure.

D. Mean Electric Field

One problem of using a pure electrostatic model for the CFM is that the magnitude of the electric field on each fluid element varies greatly when the geometry of the contour is irregularly shaped. This is due to the fact that the magnitude of the electric field is proportional to the corresponding local charge density (see Table I). When segmenting noisy images using the CFM, it is possible that some weak fluid elements that have a relatively small electric field magnitude will be confined by inner obstacles (e.g., the ventricle in Fig. 8) during the evolution based upon (12). As a consequence, the contour of the CFM becomes more ragged and those inner fluid elements can dramatically retard the convergence speed of the overall system. We addressed this problem by using the mean magnitude of the electric fields in the charged fluid system. The magnitude of $\mathbf{E}_{\text{equ}}$ in (12) was modified as

$$|\mathbf{E}_{\text{equ}}| = \max(|\mathbf{E}_{\text{equ}}|, \langle|\mathbf{E}_{\text{equ}}|\rangle) \tag{18}$$

where $|\cdot|$ is the magnitude of $\mathbf{E}_{\text{equ}}$, $\langle \cdot \rangle$ is the mean magnitude of $\mathbf{E}_{\text{equ}}$ on all fluid elements for each charged fluid, and $\max(\cdot, \cdot)$ is the greater of the two values. Therefore, the electric strength of weak fluid elements was increased such that the magnitude of the overall electric field in the charged fluid systemwas uniform, which makes the CFM more robust in segmenting noisy images.

E. Segmentation Algorithm

There are two effective parameters in the CFM algorithm: γ in (10) and β in (11). The performance of the algorithm using different values of parameter γ was similar provided that it was between 0.01 and 0.1 as suggested in Section II-B1. We therefore set the value of γ to 3% for all segmentation tasks, which leaves one parameter (β). The setting of β is discussed in Section IV. The algorithm was terminated when the number of the fluid elements on the 1–pixel-wide front [see Fig. 5(a)] remained equivalent for two consecutive steps, i.e., there was no deformation in the charged fluid shape after one more iteration. After the evolution was terminated, an ROI extraction procedure was performed on the entire image using a standard contour tracing algorithm [31]. The pseudocode for the CFM algorithm is summarized in Algorithm 1, which consists of two core algorithms corresponding to the charge distribution procedure and the front deformation procedure, respectively.

F. Computational Complexity

The charge distribution procedure (Algorithm 2) dominated the overall computational cost of the charged fluid algorithm. Using an FFT-based FSP algorithm changed the computational complexity from approximately $O(N^2)$, with N equal to the number of particles, to $O(M^2 \log M)$, where M is the length of the square that is used for the electric potential computation provided that $L_m = L_n = M$ [see (16) and (17)]. Most parametric deformable models have complexity $O(m)$, with m equal to the number of nodes. Since the level set framework added one extra dimension to the problem [13], early deformable models have complexity $O(n^3)$ in 3-D with n equal to the number of grid points in the spatial direction [8]. The more efficient narrowband implementation technique has computational complexity $O(kn^2)$, with k equal to the number of cells in the narrowband. Nevertheless,

unlike most existing level-set-based methods that require the contour to be located at floating points throughout the process, we limit the contour on the lattice during the evolution and terminate the algorithm with one extra iteration. In addition, there is no intensive initialization procedure to establish the signed distance function using our approach.

IV. Results

To demonstrate the CFM algorithm, we used both simulated images as well as real brain data from several subjects. The data from the first four subjects was acquired from the medical image database in the Division of Interventional Neuro Radiology, Department of Radiology, University of California at Los Angeles (UCLA) under an approved Human Research Subject Protection Protocol. The size of these images is 256 × 256 with spatial resolution 0.94 × 0.94 mm. Subjects 5 and 6 were taken from the BrainWeb simulator repository [32] and the Internet Brain Segmentation Repository (IBSR) [33], respectively. The Laboratory of Neuro Imaging (LONI) Debabeler [34] was used to manage image data and to convert file formats. The raw data of the images was directly used to test our approach without any preprocessing. For visualization, the dynamic range of the images was compressed into 256 gray levels.

The results of using the CFM algorithm to segment brain MR images were compared to the results obtained using the GDCIG method. Since there were no clear stopping criteria for the GDCIG algorithm, the results were obtained after a steady state was observed by inspection, unless stated otherwise. Some trivial constants in our approach were set as follows: $\Phi_0 = 10\,000$ in (11) and $h = 1$ in (9). The value of β is usually set close to unity, however, if the position of the maximum gradient is outside the ROI, then a larger value of β is required. A normalized image gradient map can be used to facilitate the procedure of finding an appropriate value of β for the ROI. To evaluate the CFM algorithm, we used five different performance measures, three of which are widely used performance metrics, i.e., the Jaccard coefficient k_j, the Dice coefficient k_d, and the sensitivity measure k_s. We defined two new performance measures: conformity k_c and particularity k_p. As shown in the Appendix, the k_c has a wider range of index score and provides more insight into the performance compared to the k_j and k_d. While the k_p has similar function with the well known specificity metric, it offers a more sensitive measure. The performance measures for the first four subjects were computed based upon the manual segmentation results by experts who have good knowledge in Anatomy and Radiographics at UCLA. The software[1] was developed in Java using the UCLA jViewBox [35] for image input/output (I/O), display, and manipulation. All experiments were executed on a Pentium M1.6-GHz machine running the Windows XP operating system.

A. Initial Position and Capture Range

The CFM algorithm is accomplished via an initial contour to start the evolution process. The position and size of the initial contour are crucial to the repeatability of segmentation. We first investigated the effect of placing the initial contours of the CFM algorithm at different locations in a brain MR image from subject 1. Three different contours with the same parameter setting of $\beta = 1.2$ were initialized at the anterior, middle, and posterior part of the brain as shown in Fig. 8. Similar brain segmentation results were obtained by the CFM compared to the results by the GDCIG. The computation times were approximately 34 s for Fig. 8(d), 16 s for Fig. 8(h), and 37 s for Fig. 8(l).

[1]The executable files and source code can be accessed at http://www.loni.ucla.edu/

We studied the robustness of the CFM to initialization with different contours. Square contours with sizes equal to 2 × 2, 4 × 4, 16 × 16, and 64 × 64 as well as circular contours with diameters equal to 8, 16, 32, and 64 were automatically created at the center of PD-weighted MR images from subject 1. Close segmentation results were obtained, $k_c = 98.94 \pm 0.01\%$ for squares and $k_c = 98.96 \pm 0.03\%$ for circles, using the same $\beta = 0.5$ to segment the brain as illustrated in Fig. 9. The computation time for each experiment was approximately the same (13–14 s) regardless of initial contour geometry and size.

B. Sensitivity Analysis of Parameter β

We also investigated the ability of the CFM to detect anatomic structures in brain MR images using different values of the parameter β. We repeated the experiment in Fig. 9 using two other values of β. The statistical analyses of the performance measures are presented in Table II, which shows very close segmentation results using three different values of β. Fig. 10 illustrates the performance of the CFM in segmenting the ventricle with blurred boundaries in T1-weighted MR images from subject 2 using different values of β. The contour leaked through weak boundaries using small values of β until a larger value was used. Note that the contour was influenced by the image gradients for $\beta = 20.0$ as shown in Fig. 10(d), which worsened the conformity score. This is because an overweighted value of β enhances the noise in the image based on the image gradient potential in (11). The computation time was approximately 1 s.

C. Noise Sensitivity of Simulated Brain Data

The noise sensitivity experiment was performed on T2-weighted simulated brain data of subject 5 from the BrainWeb with 1%–9% noise levels and no spatial inhomogeneity. Fig. 11 shows the brain segmentation results with similar conformity scores on slice 95 of subject 5 across varying noise levels. A constant value of $\beta = 0.8$ was used for all experiments. The average computation times were approximately 16.8, 18.6, 22.6, 26.2, and 36.5 s for noise levels 1%, 3%, 5%, 7%, and 9%, respectively. The overall performance measures are summarized in Table III.

D. Segmentation of the Brain

This section demonstrates the use of the CFM to separate brain from nonbrain tissue (known as *skull stripping*) on real brain MR image data. Fig. 12 shows the comparison of the GDCIG and CFM methods in the extraction of the brain in different T2-weighted MR images from subjects 1 (upper row) and 3 (lower row). It was not easy to use the GDCIG to successfully extract the brain due to the convoluted shape of the structures. A slightly stronger stopping force limited the propagation of the contour as shown in Fig. 12(b) and (f), while a slightly weaker one resulted in leakage as shown in Fig. 12(c) and (g). Compared to the GDCIG method, the CFM using $\beta = 0.6$ accurately extracted the brain boundaries as illustrated in Fig. 12(d) with $k_c = 98.66\%$ and Fig. 12(h) with $k_c = 98.99\%$. The average computation time was approximately 16 s.

We also validated the CFM to segment the brain in a large number of consecutive MR images from subjects 1 and 3. Each subject consists of eight PD-weighted and eight T2-weighted brain MR images with an equal slice distance 3 mm. Without any prior knowledge being used, the CFM successfully segmented the PD-weighted images using $\beta = 0.6 \sim 1.5$ and T2-weighted images using $\beta = 0.4 \sim 0.8$ that had similar performance scores as presented in Table III.

E. Segmentation of Brain Tumors

An important neuroimaging application is the volumetric measurement of brain tumors (e.g., in response to therapy), which requires the segmentation of the tumor boundary [36]. The CFM was evaluated to segment over eight brain tumor images of subjects 3 and 4 as well as 12 brain MR images of subject 6 from the IBSR with tumor 536. Figs. 13 and 14 show the segmentation results of the CFM algorithm as compared to the GDCIG method.

With the embedded curvature constraint force [see (1)], the contours of the GDCIG tended to be smooth, without entirely reaching the deep concavities of the tumors as shown in Figs. 13(b) and (e) and 14(a)–(c). The results obtained using the CFM were better than those obtained using the GDCIG in that the contour faithfully deformed in response to the shape of the tumors as illustrated in Figs. 13(c) and (f) and 14(g)–(i). Finally, Fig. 15 shows the use of two simultaneous charged fluids to segment the tumor and surrounding tissue from subject 4 (upper row), and the irregularly shaped tumor from subject 6 (lower row). Note that the ROI consists of regions with blurred boundaries and a large variation of gray level. The processing times were approximately 1 s for Figs. 13(f), 14(g)–(i), and 15(d), 4 s for Fig. 13(c), and 8 s for Fig. 15(b).

V. Discussion and Conclusion

We described a new deformable model, the CFM algorithm, that is based upon the theory of electrostatics. We validated the ability of this algorithm in segmenting anatomic structures in brain images without requiring prior knowledge of the underlying brain anatomy. Our approach is conceptually straightforward, using a two-stage evolution procedure as described in Section II-B. The topological changes of the contours are handled by the boundary element detection technique described in Section III-C. The spirit of our approach is to rapidly advance the contour toward the boundary of the ROI (pixel by pixel) during the evolution, and then refine the results to the desired precision. We evaluated different initial contour positions with the same value of β to segment the brain in T2-weighted MR images as illustrated in Fig. 8. The segmentation results were quite close with less sensitivity to the initial contour positions compared to the GDCIG method.

The segmentation results of the CFM were closely matched using different geometries and sizes of initial contours as illustrated in Fig. 9. The computation times were approximately the same regardless of the size of initial contours. This was because the electric potential computation using the FFT algorithm dominated the overall computational cost of the CFM algorithm as described in Section III-F. As a consequence, the computation time of using a short length of the FFT corresponding to a small contour was insignificant compared to that of using a larger length of the FFT corresponding to a bigger contour (see Section III-B). It is thus the size of the ROI rather than that of initial contours that dominates the computation time of the CFM algorithm.

We studied the sensitivity of the CFM to segment the ventricle and brain in MR images using different values of the parameter β in Section IV-B. A broad range of β can be used to achieve similar segmentation results. Noise sensitivity was also evaluated using a variety of simulated brain data across different noise levels. The CFM was robust to noise in segmenting the brain with close performance measure scores as summarized in Table III. This is due to the two-stage evolution procedures described in Section II-B. The fluid elements continuously advance outward due to the repelling force during the evolution. Therefore, the fluid elements can flow through and around inner obstacles (e.g., noise) to successfully extract the brain. This unique property of the CFM is advantageous in segmenting objects in images containing high-intensity noise.

The results of the CFM were compared to those of the GDCIG method in a variety of practical MR image processing applications. The CFM algorithm performed better than the GDCIG method in segmenting anatomic structures on the brain MR image data presented in this paper. The GDCIG tended to smooth the resulting contours (e.g., Figs. 13 and 14), in a way that was analogous to filling an elastic balloon. The CFM tended to better match the shape of the ROI in a way that was analogous to the behavior of a liquid in a container.

The GDCIG method used an edge-based stopping force to slow the propagating curve as it approached an image gradient [see (1)]. In practice, the image gradient stopping force is small but nonzero (since the image intensity values are always finite), making it difficult to choose the appropriate stopping factor required to achieve accurate segmentation. This is illustrated in Figs. 13 and 14, where the boundaries of the tumors were not accurately captured. In addition, a stronger stopping force limited curve evolution as illustrated in Fig. 12(b) and (f), while a weaker one resulted in leakage as illustrated in Fig. 12(c) and (g). The GDCIG was also sensitive to the initial contour positions as illustrated in Fig. 8.

With only one effective parameter, the CFM algorithm is easier to manipulate. The CFM also uses the image gradient force to confine the contour inside the ROI [see (11)]. However, the effective field that is used to guide curve evolution is the vector sum of the electric field and the gradient of the image potential [see (12)]. The fluid element changes the advancement direction when it encounters an image gradient that is regarded as the object boundary as described in Section II-C and illustrated in Fig. 7(a). Therefore, the initial contour does not have to be placed close to or symmetrically with respect to the boundary (although symmetric initialization could save computation time) as illustrated in Fig. 8.

One limitation of the CFM algorithm is that it must be initialized inside the ROI, but it is not necessary to place the initial contour at the center of the region. If the contour is initialized across the boundary of an object, the fluid elements that are not at the boundary will advance outward since they have no salient image gradient force. This can make the entire contour move across the object after further evolution. For example, we used two charged fluid contours initialized across the tumors in Fig. 15(a), and the contours passed the tumor boundary and merged to one that was partly stopped by the surrounding tissue.

In developing the CFM, we addressed two key problems: the simulation of a charged fluid using an electrostatic model and the propagation of the interface in deformable models. The challenge was to find an optimal simulation with sufficient accuracy at the lowest computational cost. We used the electrostatic plasma models that were originally developed for particle simulation, as described in Sections III-A and III-B, to develop the CFM embedded in a deformable model for front propagation. Unlike other charged particle systems, we changed the property and structure of charged particles in such a way that the charged fluid behaves like a liquid that flows through and around objects. We established the correlation of the intrinsic properties between the deformable model and the electrostatic system such that the explicit parameters (curvature and normal direction) in a deformable model are implicitly related to the electrostatic equilibrium properties (see Table I).

In summary, we demonstrated the ability of the CFM algorithm to capture a variety of anatomic structures in brain MR images. Our results illustrate that this new approach requires only one parameter (β) for robust brain MR image segmentation problems such as the extraction of brain tumors and skull stripping as compared with the GDCIG method. In particular, our approach is useful in segmenting regions with blurred boundaries and large variations in intensity without requiring prior knowledge as illustrated in Section IV.

Some properties of the CFM for brain image segmentation are as follows: 1) no computation of curvature, velocity, or acceleration terms is required to advance the fluid elements, 2) there is no time interval setting, 3) only one effective parameter setting is needed, 4) topological changes of the propagating interface are handled automatically, and 5) the CFM can provide subpixel precision.

Further research is required to improve the electric potential computation via more efficient numerical techniques. We could also introduce other attractive forces, or we could include image regional forces to guide the CFM contour toward the ROI. Additional work is needed to investigate the current CFM algorithm for interactive segmentation tools. Another interesting application is that, since fluid element charges vary in response to the geometry of objects, this feature can be used as a guide for spatial transformation in image registration across different types of medical imaging modalities. We believe that the CFM is of potential value in a variety of brain image processing applications requiring semiautomatic and fully automatic procedures.

Acknowledgments

This work was supported by the NIH/NIMH under Research Grant R01 MH071940, the NIH/NCRR under Resource Grant P41 RR013642, and the National Institutes of Health through the NIH Roadmap for Medical Research under Grant U54 RR021813 entitled Center for Computational Biology (CCB).

References

1. Caselles V, Catte F, Coll T, Dibos F. A geometric model for active contours in image processing. Numer Math. 1993; 66:1–31.
2. Malladi R, Sethian JA, Vemuri BC. Shape modeling with front propagation: A level set approach. IEEE Trans Pattern Anal Mach Intell. Feb; 1995 17(2):158–175.
3. Bankman, IN. Handbook of Medical Imaging. San Diego, CA: Academic; 2000.
4. Udupa, JK.; Herman, GT. 3D Imaging in Medicine. 2. Boca Raton, FL: CRC; 2000.
5. Dhawan, AP. Medical Image Analysis. New York: Wiley; 2003.
6. McInerney T, Terzopoulos D. Deformable models in medical image analysis: A survey. Med Image Anal. 1996; 1(2):91–108. [PubMed: 9873923]
7. Osher, S.; Paragios, N. Geometric Level Set Methods in Imaging, Vision, and Graphics. New York: Springer-Verlag; 2003.
8. Osher, S.; Fedkiw, R. Level Set Methods and Dynamic Implicit Surfaces. New York: Springer-Verlag; 2003.
9. Kass M, Witkin A, Terzopoulos D. Snakes: Active contour models. Int J Comput Vis. 1988; 01(04): 321–331.
10. Cohen LD, Cohen I. Finite-element methods for active contour models and balloons for 2- D and 3- D images. IEEE Trans Pattern Anal Mach Intell. Nov; 1993 15(11):1131–1147.
11. Xu C, Prince JL. Snakes, shapes, and gradient vector flow. IEEE Trans Image Process. Mar; 1998 7(3):359–369. [PubMed: 18276256]
12. McInerney T, Terzopoulos D. T-snakes: Topology adaptive snakes. Med Image Anal. 2000; 4(2): 73–91. [PubMed: 10972323]
13. Osher S, Sethian JA. Fronts propagating with curvature dependent speed: Algorithms based on Hamiltons-Jacobi formulations. J Comput Phys. 1988; 79:12–49.
14. Kichenassamy, S.; Kumar, A.; Olver, P.; Tannenbaum, A.; Yezzi, A. Gradient flows and geometric active contour models. IEEE Proc. Int. Conf. Comput. Vis.; 1995. p. 810-815.
15. Yezzi A, Kichenassamy S, Kumar A, Olver P, Tannenbaum A. Ageometric snake model for segmentation of medical imagery. IEEE Trans Med Imag. Apr; 1997 16(2):199–209.
16. Siddiqi K, Lauziere YB, Tannenbaum A, Zucker SW. Area and length minimizing flows for shape segmentation. IEEE Trans Image Process. Mar; 1998 7(3):433–443. [PubMed: 18276263]

17. Xu M, Thompson PM, Toga AW. An adaptive level set segmentation on a triangulated mesh. IEEE Trans Med Imag. Feb; 2004 23(2):191–201.
18. Gout C, Guyader CL, Vese L. Segmentation under geometrical conditions using geodesic active contours and interpolation using level set methods. Numer Algorithms. 2005; 39(1–3):155–173.
19. Han X, Xu C, Prince JL. A topology preserving level set method for geometric deformable models. IEEE Trans Pattern Anal Mach Intell. Jun; 2003 25(6):755–768.
20. Wang G, McFarland EG, Brown BP, Vannier MW. Gi tract unraveling with curved cross sections. IEEE Trans Med Imag. Apr; 1998 17(2):318–322.
21. Wang, G.; Dave, SB.; Brown, BP.; Zhang, Z.; McFarland, EG.; Haller, JW.; Vannier, MW. Colon unraveling based on electrical field: Recent progress and further work. Proc. SPIE Conf. Med. Imag.; May 1999; p. 125-132.
22. Kruer WL, Dawson JM, Rosen B. The dipole expansion method for plasma simulation. J Comput Phys. 1973; 13:114–129.
23. Langdon AB, Birdsall CK. Theory of plasma simulation using finite-size particles. Phys Fluids. 1970; 13(8):2115–2122.
24. Dawson JM. Particle simulation of plasmas. Rev Mod Phys. 1983; 55(2):403–447.
25. Kisworo M, Venkatesh S, West G. Modeling edges at subpixel accuracy using the local energy approach. IEEE Trans Pattern Anal Mach Intell. Apr; 1994 16(4):405–410.
26. Jensen K, Anastassiou D. Subpixel edge localization and the interpolation of still images. IEEE Trans Image Process. Mar; 1995 4(3):285–295. [PubMed: 18289979]
27. Birdsall, CK.; Langdon, AB. Plasma Physics via Computer Simulation. New York: McGraw-Hill; 1985.
28. Tajima, T. Computational Plasma Physics: With Applications to Fusion and Astrophysics. Reading, MA: Addison-Wesley; 1989.
29. Neu, SC. PhD dissertation. Dept. Phys., Univ. California; Los Angeles, CA: 1997. Computer simulation of a non-neutral plasma column.
30. Decyk VK, Dawson JM. Computer model for bounded plasma. J Comput Phys. 1979; 30:407–427.
31. Pavlidis, T. Algorithms for Graphics and Image Processing. Rockville, MD: Comput. Sci. Press; 1982.
32. BrainWeb: Simulated Brain Database [Online]. Available: http://www.bic.mni.mcgill.ca/brainweb/
33. Internet Brain Segmentation Repository (IBSR) [Online]. Available: http://www.cma.mgh.harvard.edu/ibsr/
34. Neu SC, Valentino DJ, Toga AW. The LONI Debabeler: A mediator for neuroimaging software. NeuroImage. 2005; 24(4):1170–1179. [PubMed: 15670695]
35. Valentino, DJ.; Ma, K.; Wei, CC.; Neu, SC. A portable framework for medical imaging display: The jViewbox. Proc. Annu. Meeting Radiol. Sci. North Amer.; 2002. p. 891-891.
36. Haney SM, Thompson PM, Cloughesy TF, Alger JR, Toga AW. Tracking tumor growth rates in patients with malignant gliomas: A test of two algorithms. Amer J Neuroradiol. 2001; 22(1):73–82. [PubMed: 11158891]
37. Shattuck DW, Sandor-Leahy SR, Schaper KA, Rottenberg DA, Leahy RM. Magnetic resonance image tissue classification using a partial volume model. NeuroImage. 2001; 13:856–876. [PubMed: 11304082]
38. Cheng, DK. Field and Wave Electromagnetics; Reading, MA: Addison-Wesley; 1989.

Biographies

Herng-Hua Chang received the B.S. degree in mechanical engineering from National Taiwan University, Taiwan, in 1996, the M.S. degree in biomedical engineering from National Yang-Ming University, Taiwan, in 1998, and the Ph.D. degree in biomedical engineering from University of California at Los Angeles (UCLA), in 2006.

From 1998 to 1999, he was a Research Assistant at the Department of Radiology, Veterans General Hospital, Taipei, Taiwan. From 1999 to 2000, he was a Teaching Assistant in Mechanical Engineering at National Taiwan University. Currently, he is a Postdoctoral Scholar at the Laboratory of Neuro Imaging (LONI), UCLA, where he is researching and developing image processing algorithms for neurology and radiology applications. His research interests include bioinformatics, pattern recognition, medical image analysis, stereotactic radiosurgery, and biomechanics.

Daniel J. Valentino received the B.S. degree in chemistry from the University of California, Irvine and the Ph.D. degree in biomedical physics from the University of California at Los Angeles (UCLA), in 1991. His Ph.D. dissertation focused on computational techniques for mapping relationships between brain structure and function.

He was formerly the Chief of Imaging Informatics for UCLA Healthcare, and on the founding faculty of the Laboratory of Neuro Imaging (LONI) and the Center for Computational Biology (CCB) at UCLA. He is currently an Adjunct Associate Professor of Biomedical Physics in the Division of Interventional Neuro Radiology (DINR), UCLA, and the Chief Scientist at Radlink, Inc., Redondo Beach, CA. His research interests include x-ray computed radiography, variational methods for image restoration and segmentation, and cerebrovascular hemodynamics simulation.

Gary R. Duckwiler received the B.S. degree in biochemistry from Claremont Men's College, Claremont, CA, in 1983 and the M.D. degree from the University of California at Los Angeles (UCLA), in 1989.

He completed his radiology residency and Neuroradiology fellowship at UCLA, where he is currently a Professor of Radiology in the Division of Interventional Neuroradiology (DINR). His research interests include the design and development of cerebrovascular navigation techniques and therapeutic devices, and the simulation of cerebrovascular hemodynamics for endovascular treatment planning.

Arthur W. Toga received the B.S. degree in psychology and the Ph.D. degree in neuroscience from Washington University in St. Louis, St.Louis, MO.

While at Washington University in St. Louis, he developed pioneering techniques for the 3-D visualization and atlasing of the brain. He founded the Laboratory of Neuro Imaging (LONI), University of California at Los Angeles (UCLA), and is the Principal Investigator for the NIH-sponsored Center for Computational Biology (CCB), UCLA. His research focus is on imaging-based neuroscience studies, including neuroimage processing and analysis, brain visualization and brain atlasing.

APPENDIX. PERFORMANCE MEASURES

The CFM algorithm was evaluated by several performance measures, two of which are widely used global performance metrics. The Jaccard coefficient k_j measures the ratio of the intersection area of two sets (Ω_1 and Ω_2) divided by the area of their union, while the Dice coefficient k_d computes the ratio of the intersection divided by the sum of each individual area [37]

$$\begin{aligned}\kappa_j &= \frac{|\Omega_1 \cap \Omega_2|}{|\Omega_1 \cup \Omega_2|} \times 100\% \\ \kappa_d &= \frac{2|\Omega_1 \cap \Omega_2|}{|\Omega_1| + |\Omega_2|} \times 100\% = \frac{2\kappa_j}{\kappa_j + 1}.\end{aligned} \tag{19}$$

We also defined a global measure of conformity k_c to measure the similarity of two sets using the relationship

$$\kappa_c \equiv \left(1 - \frac{F_P + F_N}{T_P}\right) \times 100\%$$

where F_P represents false positives, F_N represents false negatives, and T_P represents true positives of the segmentation results. Note that k_j is related to k_c through the following:

$$\kappa_j = \frac{T_P}{T_P + F_P + F_N} = \frac{1}{2 - \kappa_c}. \tag{20}$$

Fig. 16 illustrates the relationship between k_d, k_j, and k_c based upon (19) and (20) in the domain of to k_c = 0% to 100%, which shows that k_c is always smaller than the other two indexes (except at the apex where they are equal). The three metrics agree when the two sets are perfectly matched, but the conformity measure k_c has a wider range of index scores, which provides more insight into the performance of segmentation algorithms. For reference, k_c = 0% corresponds to k_j = 50% and k_d = 66.67%, respectively. We are also interested in how much of the ROI is being excluded or included, respectively. The sensitivity measure k_s is used for measuring how many pixels in the ROI are correctly segmented as

$$\kappa_s = \left(\frac{T_p}{T_P + F_N} \right) \times 100\%.$$

While the specificity metric $T_N/(T_N + F_P)$ is sensitive to true negatives T_N, we defined a metric, particularity k_p, for measuring how many pixels outside the ROI are included as given in the following:

$$\kappa_p \equiv \left(1 - \frac{F_p}{T_P + F_N} \right) \times 100\%.$$

The major disadvantage of using the specificity metric for evaluating segmentation performance is that one can obtain different scores when measuring the same target in various images with different dimensions due to T_N. The manual delineation is treated as a reference for the measurement of the segmentation performance. Note that when the segmentation mask entirely includes the referenced one, i.e., $F_N = 0$, we can obtain that k_s = 100% and $k_p = k_c$. Also note that when the segmentation mask is inside the referenced one, i.e., $F_P = 0$, we can obtain that k_p = 100% and $k_s = k_j$.

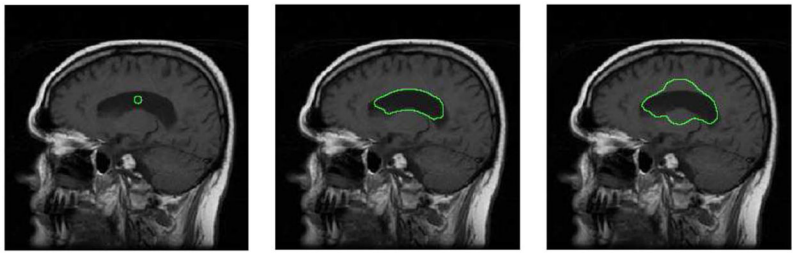

Fig. 1.

Difficulty of using the GDCIG method in segmenting the ventricle in T1-weighted MR images. (a) Initial contour. (b) Using a slightly stronger stopping force, the contour of the GDCIG was confined inside the ventricle. (c) Using a slightly weaker stopping force made the contour stretch toward both ends of the ventricle, but it also leaked through the boundary.

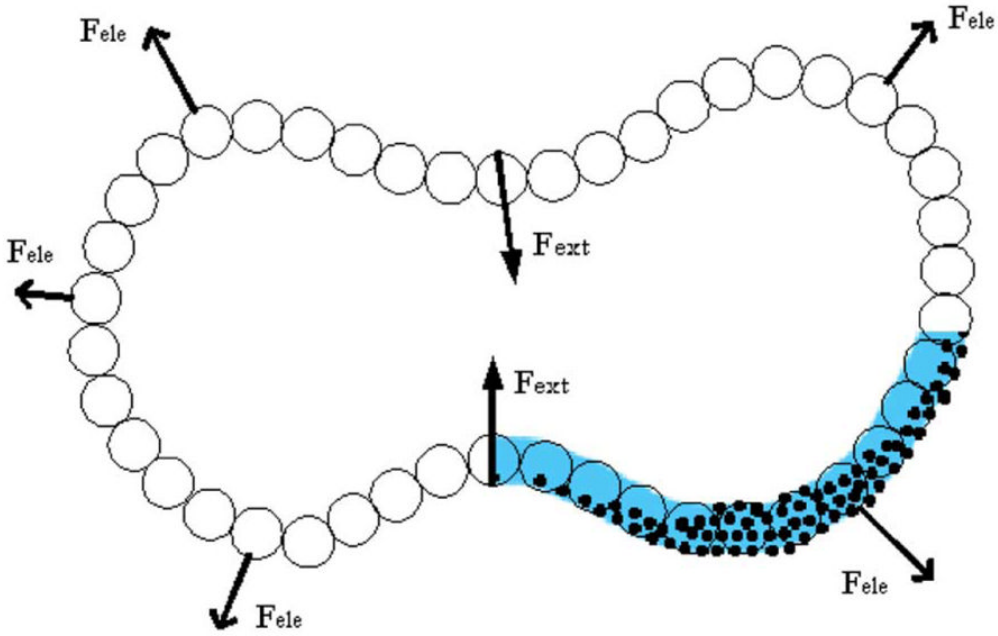

Fig. 2.

Schematic illustrating the concept of a charged fluid. A charged fluid conceptually consists of charged elements (the large circles), each of which exerts a repelling electric force upon the others. The fluid elements, as if they were consisted of different numbers of charged particles (the solid dots), are connected to one another by 4-connectivity when they advance. The charged fluid, behaving like a liquid, can be influenced by internal electric forces $\mathbf{F}_{ele}$ of repulsion as well as external forces $\mathbf{F}_{ext}$ from the image data.

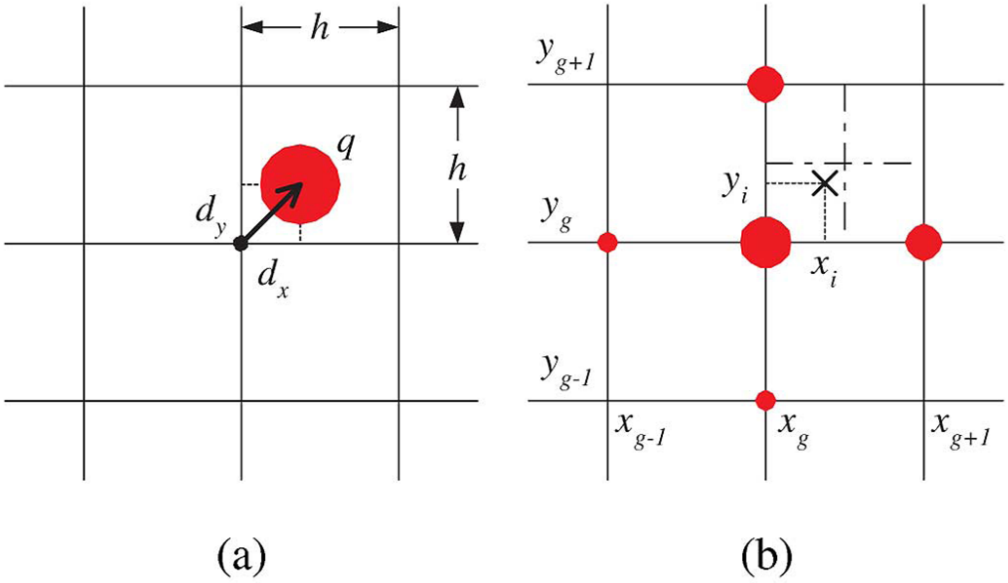

Fig. 3.

Charge interpolation using the SUDS technique. (a) Fluid element with charge q is advanced a distance (d_x, d_y). (b) Charge of the fluid element at (x_i, y_i) is interpolated to the NGP location and its 4-neighbors with different charge, if any.

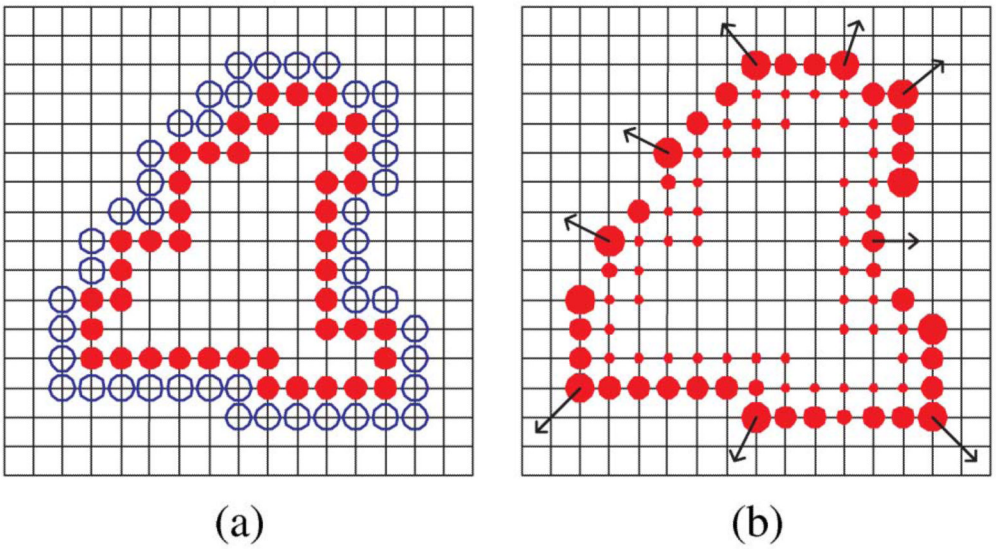

Fig. 4.

Charge distribution procedure. (a) At the beginning of this procedure, a uniform charge distribution is applied to the fluid elements (the red solid dots). They are only allowed to share charge within the 2–pixel-wide propagating interface that is obtained from the front deformation procedure (see Fig. 5). Note that the empty charge positions on the interface are represented by the blue hollow circles. (b) System reaches the equilibrium charge distribution and the electric fields (the arrows) on the elements are approximately perpendicular to the contour. The 1–pixel-wide front (not shown) is then obtained by using the boundary element detection technique as described in Section III-C. Note that the CFM in this procedure is a pure electrostatic system without the influence of external forces and the change of contour shapes.

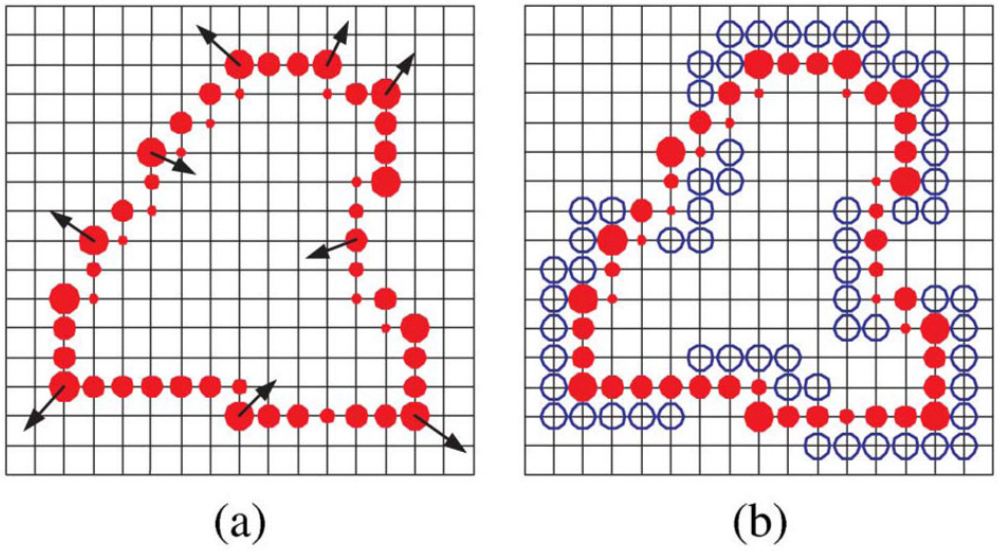

Fig. 5.

Front deformation procedure. (a) After the charge distribution procedure, the fluid elements are on the 1–pixel-wide front by 4-connectivity. Note that the tiny inner charges in Fig. 4(b) are discarded after the boundary element detection procedure. The effective fields (the arrows) are computed based upon the electric field in equilibrium and the gradient of the image potential. Some of the effective fields are in very different directions compared to the electric fields at the corresponding positions in Fig. 4(b). (b) New 2–pixel-wide propagating interface is obtained by locating the four adjacent grid points according to the effective field directions in (a) based upon Fig. 6 for all elements. Note that, compared to Fig. 4(a), the propagating interface evolves into a different shape in response to the effective fields in (a).

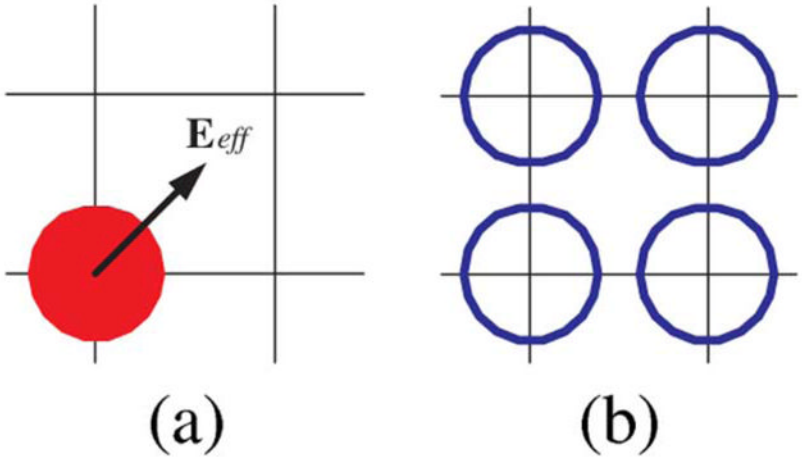

Fig. 6.

Schematic illustrating the localization of the 2–pixel-wide binary interface on an individual fluid element. (a) Effective field $\mathbf{E}_{\text{eff}}$ on a fluid element (the red solid dot). (b) Four adjacent grid points (the blue hollow circles) of the element are generated according to the effective field direction in (a) and denoted as a part of the 2–pixel-wide propagating interface.

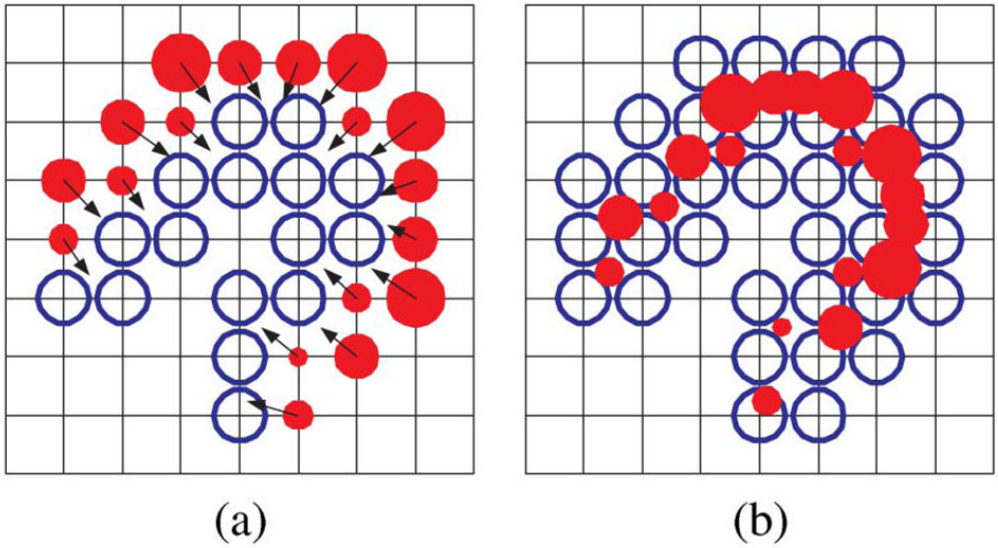

Fig. 7.
Schematic illustrating the subpixel precision computation. (a) Effective fields on the fluid elements (the red solid dots) are approximately oriented inward after the evolution is terminated. (b) Subpixel precision for the area and length of the ROI can be calculated by advancing the fluid elements a real number distance based upon the effective fields in (a) using (9).

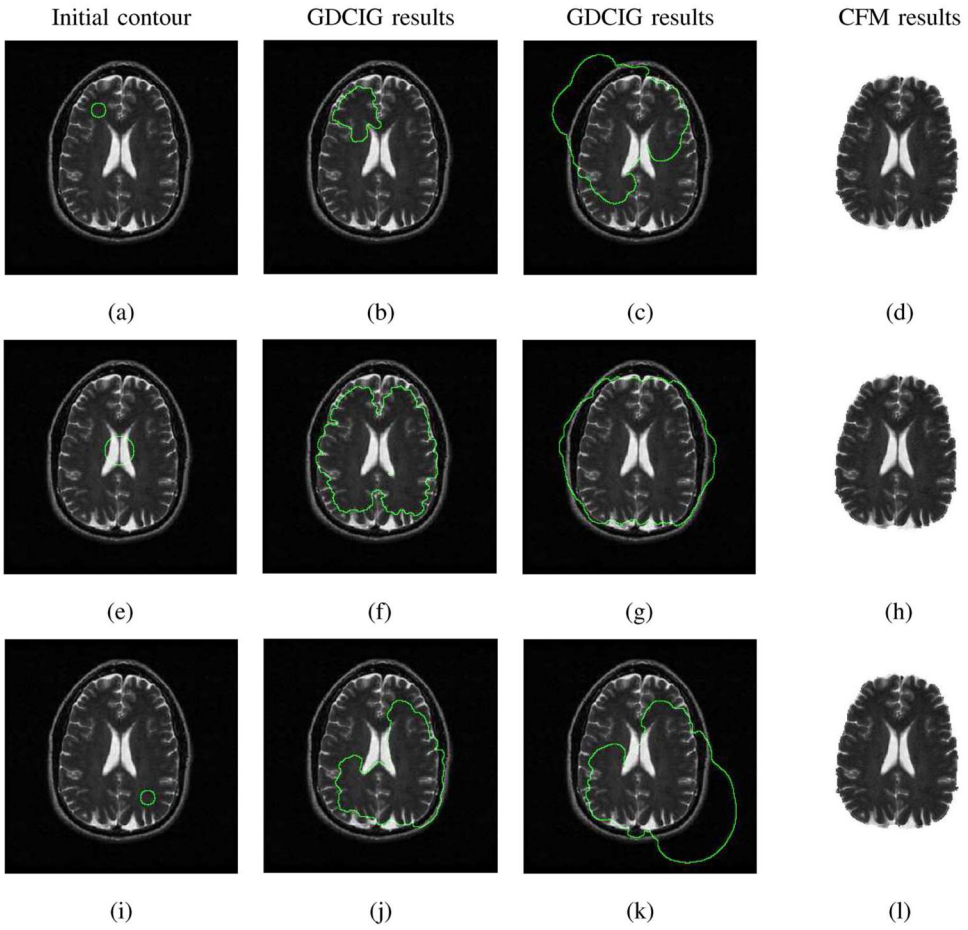

Fig. 8.

Effect of initial contour positions in segmenting T2-weighted MR images using the GDCIG and CFM methods. The figures in the first column show the initial contours respectively located at the (a) anterior, (e) middle, and (i) posterior region of the brain. The figures in the middle two columns show the GDCIG results and those in the last column the CFM results. Using a slightly stronger stopping force, the GDCIG contours were confined inside as shown in (b), (f), and (j). Using a slightly weaker stopping force, the GDCIG contours leaked over the brain as shown in (c), (g), and (k). An appropriate stopping factor was difficult to choose for the GDCIG to achieve successful results. (d), (h), and (l) Segmentation results of the CFM using the same $\beta = 1.2$ with conformity $k_c = 97.45\%$, $k_c = 97.45\%$, and $k_c = 97.56\%$, respectively. The computation times were approximately (d) 34 s, (h) 16 s, and (l) 37 s.

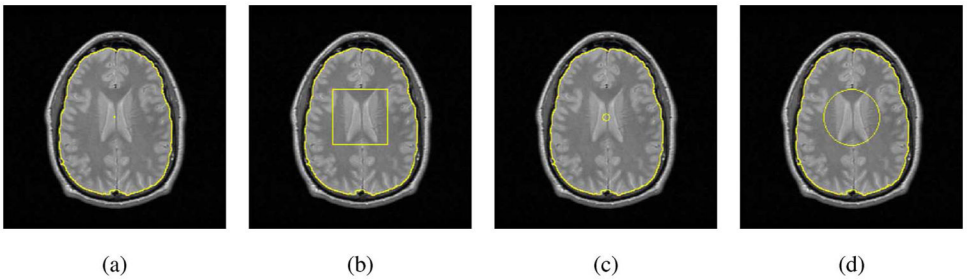

Fig. 9.

Experimental results on the effect of initial contours with different shapes and dimensions in segmenting the brain in PD-weighted MR images. All initial contours were automatically created at the center of the image with the same $\beta = 0.5$. The CFM achieved $k_c = 98.94 \pm 0.01\%$ segmentation results using square contours with sizes equal to 2×2 (a), 4×4 (not shown), 16×16 (not shown), and 64×64 (b), respectively. Close segmentation results with $k_c = 98.96 \pm 0.03\%$ were also obtained using circular contours with diameters equal to 8 (c), 16 (not shown), 32 (not shown), and 64 (d), respectively. Note that the CFM algorithm required approximately the same computation time for all experiments (13–14 s). The overall performance measures are summarized in Table II.

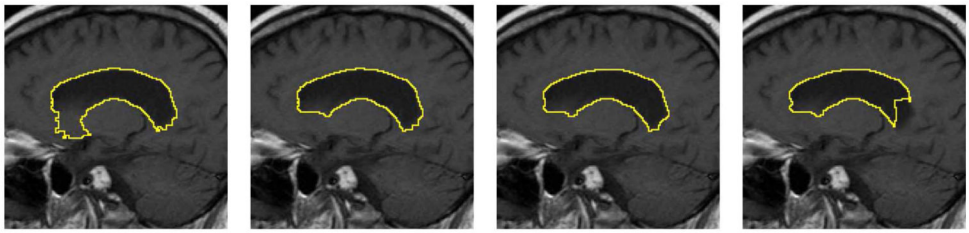

Fig. 10.

Sensitivity analysis of parameter β in segmenting the ventricle with blurred boundaries in T1-weighted MR images. (a) Contour leaked through weak boundaries when using a lower value of $\beta = 12.0$. (b) Leakage stopped when using $\beta = 13.0$ ($k_c = 91.65\%$). (c) Segmentation result with $k_c = 92.21\%$ using $\beta = 19.0$. (d) Result ($k_c = 81.55\%$) started to deteriorate when using a slightly higher value of $\beta = 20.0$. The computation time was approximately 1 s.

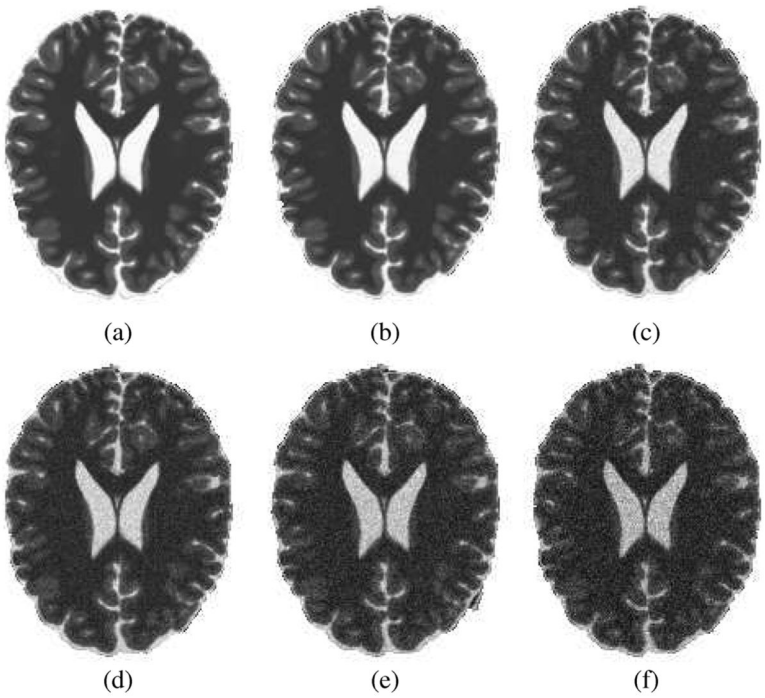

Fig. 11.

Noise analysis for the segmentation of slice 95 from the BrainWeb simulator with different noise levels using the same $\beta = 0.8$. (a) Ground truth. (b) Result with $k_c = 98.87\%$ for 1% noise. (c) Result with $k_c = 98.78\%$ for 3% noise. (d) Result with $k_c = 98.75\%$ for 5% noise. (e) Result with $k_c = 98.41\%$ for 7% noise. (f) Result with $k_c = 98.76\%$ for 9% noise.

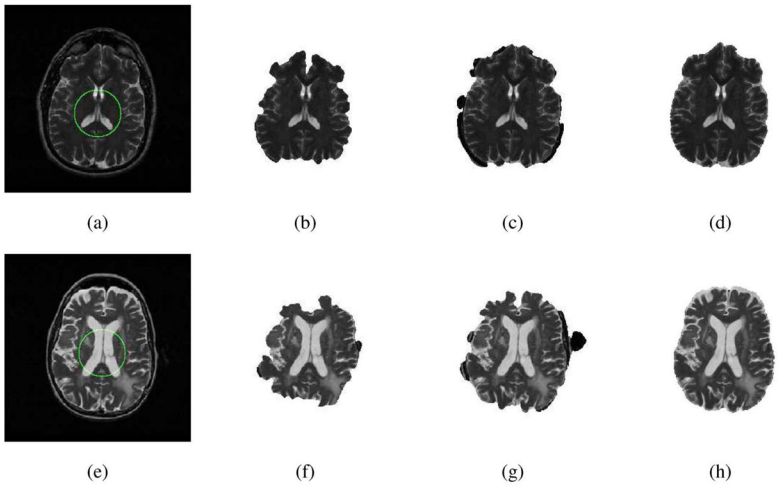

Fig. 12.

Comparison of the GDCIG and CFM methods in segmenting the brain in T2-weighted MR images of subjects 1 (upper row) and 3 (lower row). (a) and (e) Initial contour. (b) and (f) Results using the GDCIG with k_c = 81.53% and k_c = 65.99%. (c) and (g) Contour of the GDCIG leaked through the boundaries using a weaker stopping force. (d) and (h) Segmentation results of the CFM using the same β = 0.6 with k_c = 98.66% and k_c = 98.89%. The processing time was approximately 16 s.

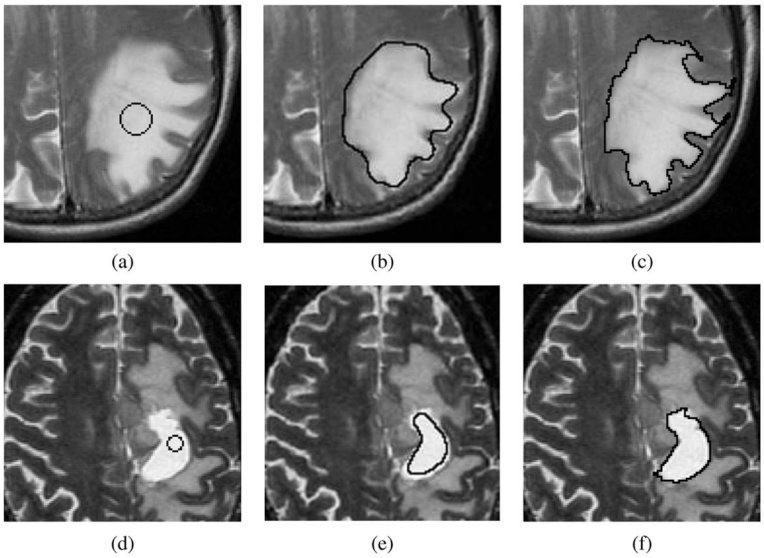

Fig. 13.

Comparison of the brain tumor segmentation results in T2-weighted MR images of subjects 3 (upper row) and 4 (lower row) using the GDCIG and CFM methods. (a) and (d) Initial contour. (b) and (e) Results of using the GDCIG with k_c = 82.87% and k_c = 49.09%. (c) and (f) Segmentation results of the CFM using the same β = 6.0 with k_c = 87.42% and k_c = 75.09%. The processing times were approximately (c) 4 s and (f) 1 s.

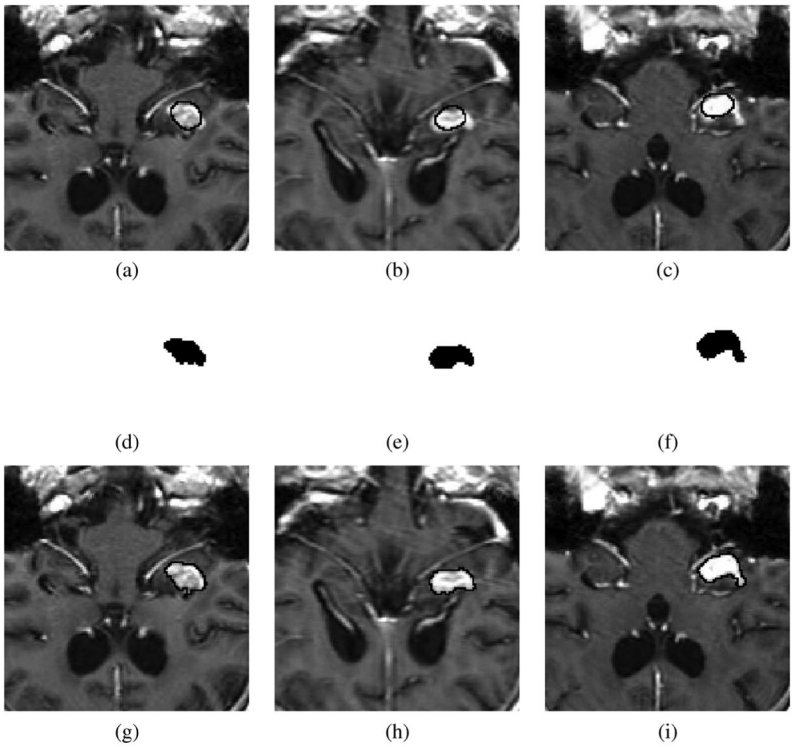

Fig. 14.

Comparison of the GDCIG and CFM methods for the segmentation of tumor 536 from the IBSR. Top row: the GDCIG results with (a) k_c = 54.48%, (b) k_c = 22.13%, and (c) k_c = 25.17%. Middle row: the manual segmentation masks with (d) 536_47_26, (e) 536_68_24, and (f) 536_88_28. Bottom row: the CFM results with (g) k_c = 84.39% (β = 6.0), (h) k_c = 86.03% (β = 6.0), and (i) k_c = 85.54% (β = 8.0). The processing time was approximately 1 s.

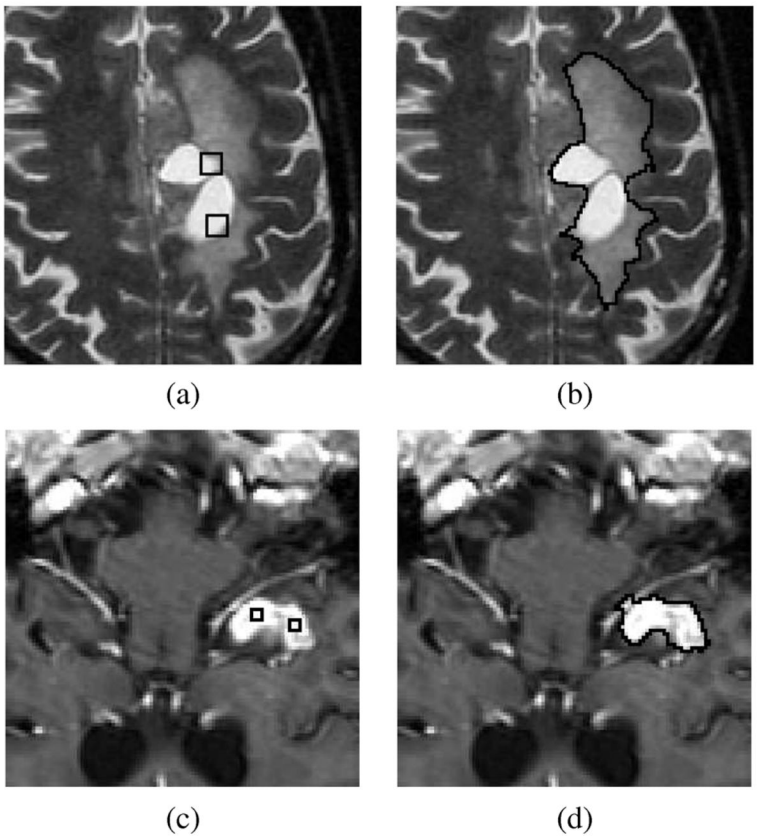

Fig. 15.

Segmentation of difficult structures in brain MR images using two initial contours of the CFM with $\beta = 8.0$. Upper row: segmentation of the brain tumors and surrounding tissue in a T2-weighted MR image of subject 4 with $k_c = 81.62\%$. Lower row: segmentation of the irregularly shaped tumor 536_88_26 from the IBSR with $k_c = 70.52\%$. The processing times were approximately (b) 8 s and (d) 1 s.

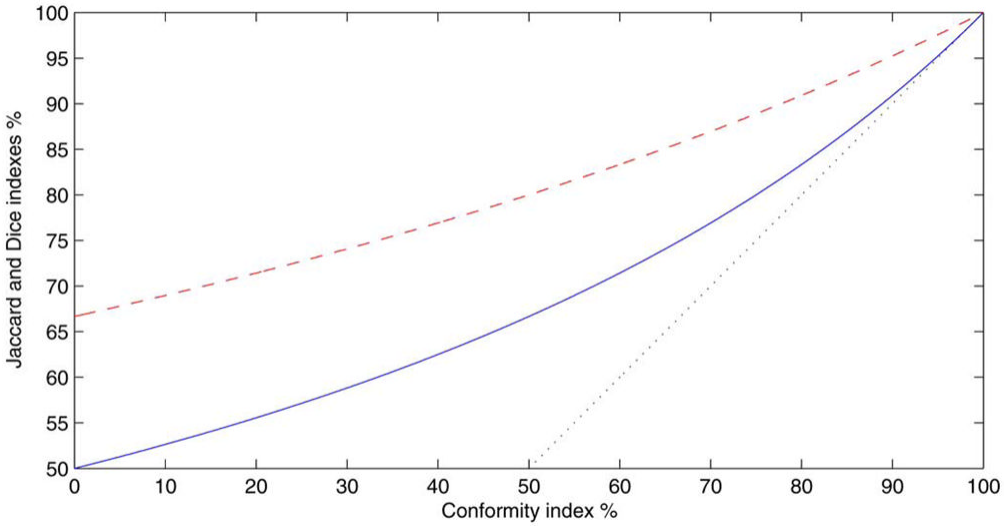

Fig. 16.

Comparison of the Jaccard k_j (the blue solid curve) and Dice k_d (the red dashed curve) coefficients with respect to conformity k_c in the domain between 0% and 100%. Conformity k_c, which provides a wider range of index scores, is always smaller than the other two coefficients k_j and k_d (except at 100%). For reference, k_c = 0% corresponds to k_j = 50% and k_d = 66.67%, respectively.

TABLE I

Unique Characteristics of Charged Conductors in Electrostatic Equilibrium [38]. Electrostatic Equilibrium Means That There Is No Net Flow of Electric Charge or No Electric Current

	Characteristics of charged conductors in equilibrium
1	The electric field anywhere inside a conductor is zero in electrostatic equilibrium.
2	Any net charge on an isolated conductor resides entirely on its surface.
3	The electric field just outside the surface of an isolated conductor is perpendicular to the surface and has a magnitude equal to $\frac{\sigma}{\varepsilon_0}$, where σ is the local surface charge density at that point.
4	On an irregularly shaped conductor, the surface charge density σ and hence the electric field just outside is greatest where the curvature is largest.
5	Every point on the surface of a conductor in equilibrium is at the same potential (the surface is an equipotential).

TABLE II

Statistical Analyses of the Performance Measures for the Experiments in Fig. 9 Using Three Different Values of β

Parameter β	k_c	k_s	k_p
0.5	98.95 ± 0.02%	99.38 ± 0.01%	99.58 ± 0.01%
1.0	98.73 ± 0.04%	98.98 ± 0.04%	99.76 ± 0.02%
1.5	98.48 ± 0.03%	98.68 ± 0.03%	99.81 ± 0.00%

k_C = Conformity, k_S = Sensitivity, k_P = Particularity.

TABLE III

Statistical Analyses of the Performance Measures in Segmenting the Brain on Different Types and Subjects Using the CFM

Image and Subject	k_c	k_s	k_p
8 PD-weighted MR images, Subject 1	98.78 ± 0.19%	99.14 ± 0.22%	99.64 ± 0.08%
8 PD-weighted MR images, Subject 3	98.69 ± 0.21%	99.09 ± 0.22%	99.62 ± 0.06%
8 T2-weighted MR images, Subject 1	98.47 ± 0.15%	99.27 ± 0.17%	99.22 ± 0.11%
8 T2-weighted MR images, Subject 3	98.61 ± 0.19%	99.15 ± 0.25%	99.47 ± 0.14%
8 1%-noise T2 MR images, Subject 5	98.55 ± 0.34%	99.63 ± 0.06%	98.92 ± 0.31%
8 3%-noise T2 MR images, Subject 5	98.28 ± 0.29%	99.43 ± 0.38%	98.86 ± 0.33%
8 5%-noise T2 MR images, Subject 5	98.35 ± 0.17%	99.41 ± 0.27%	98.94 ± 0.25%
8 7%-noise T2 MR images, Subject 5	98.50 ± 0.22%	99.58 ± 0.12%	98.92 ± 0.23%
8 9%-noise T2 MR images, Subject 5	98.34 ± 0.40%	99.39 ± 0.47%	98.96 ± 0.18%

k_C = Conformity, k_S = Sensitivity, k_P = Particularity.

Algorithm 1

Charged fluid

1 parameter setting of in (11)

2 image potential computation using (11)

3 **repeat**(i)

- **a.** uniform charge distribution over fluid elements
- **b.** **repeat**(j)

  Algorithm 2
- **c.** **until**(j) electrostatic equilibrium is achieved by setting γ = 3% in (10)
- **d.** 1–pixel-wide front construction using the boundary element detection method
- **e.** Algorithm 3
- **f.** mean potential computation and charge normalization using (5)

4 **until**(i) no deformation in the charged fluid shape

5 ROI extraction

6 subpixel precision calculation, if desired

Algorithm 2

Charge distribution procedure

1 forward FFT computation of the charge array based upon (16)

2 inverse FFT computation of the potential array based upon (17)

3 electric field computation using (7)

4 advance distance computation using (9)

5 charge density computation using the SUDS based upon (15) and Fig. 3

Algorithm 3

Front deformation procedure

1	mean electric field compensation using (18)
2	effective field computation using (12)
3	2–pixel-wide interface localization based upon Fig. 6