Title
Image segmentation using a charged fluid method

Permalink
https://escholarship.org/uc/item/0hk8s3vz

Journal
Journal of Electronic Imaging, 15(2)

ISSN
1017-9909

Authors
Chang, H H
Valentino, Daniel J

Publication Date
2006-04-01

Peer reviewed

Image segmentation using a charged fluid method

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

Daniel J. Valentino
University of California at Los Angeles
Biomedical Engineering IDP
and Laboratory of Neuro Imaging (LONI)
and Division of Interventional Neuro Radiology
Radiology Mail Stop 172115
Los Angeles, California 90095-1721
E-mail: dvalentino@mednet.ucla.edu

Abstract. *A new image segmentation algorithm that uses the simulation of a charged fluid is developed. Conceptually, a charged fluid consists of charged elements, each of which exerts a repelling electric force on the others. The charged fluid behaves like a liquid such that it flows through and around different obstacles. The boundary of the segmented object is determined by the image gradient, which is modeled as potential wells that stop the propagating front. The simulation is evolved in two steps that are governed by Poisson's equation. The first step distributes the elements of the charged fluid along 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 equilibrium electric potential and the image potential. The procedure is repeated until the propagating front resides on the boundary of objects being segmented. The electric potential of the simulated system is rapidly calculated using the finite-size particle (FSP) method implemented via the fast Fourier transform (FFT) algorithm. Experimental results using phantom images, photographic pictures, and medical images demonstrate the utility of this new algorithm in a wide variety of image processing applications. © 2006 SPIE and IS&T.* [DOI: 10.1117/1.2199555]

1 Introduction

Image segmentation is a process of partitioning images into several regions of interest such that each region has similar gray-level, color, and texture characteristics[1,2] A wide variety of image segmentation methods have been proposed, but few techniques achieve satisfactory results in a broad range of applications. Segmentation methods for monochrome images can be classified into several categories including pixel-based, edge-based, region-based, matching approaches and deformable models.[1–4] Deformable models involve the formulation of a propagating interface, which is a closed curve in two dimensions or a closed surface in three dimensions, that is moving under a speed function determined by local, global, and independent properties.[5] 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 mathematically divided into two broad classes: parametric and geometric.

Parametric deformable models, originating from the active contour model (also known as snakes) introduced by Kass *et al.*,[6] 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[7] extended the original snake such that the curve behaves like a balloon and obtained more stable results.

With the introduction of the level set numerical algorithm developed by Osher and Sethian,[8] geometric deformable models enable automatic topological changes without using special tracking procedures.[9] The principle underlying level set methods is to adopt a 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.[5] 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.[8] Image stopping terms, such as regional and gradient forces, are incorporated into this speed function for segmentation purposes.

Parametric and geometric deformable models are both

Paper 05133R received Jul. 11, 2005; revised manuscript received Nov. 17, 2005; accepted for publication Dec. 9, 2005; published online May 2, 2006.

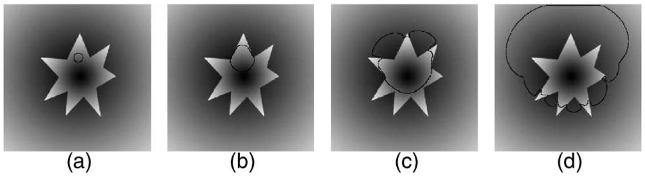

Fig. 1 Difficulty of using a level set based deformable model with edge-based stopping terms in segmenting an object with sharp corners and cusps: (a) the initial contour and (b) to (d) the evolution of the contour that crossed over the boundary.

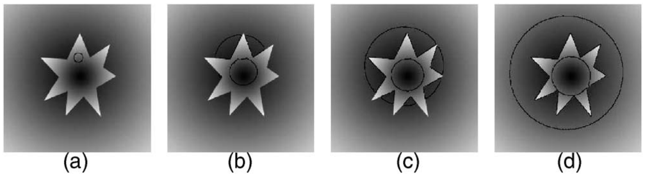

Fig. 2 Difficulty of using the active contour without edges in segmenting an object with a wide intensity distribution that is similar to the background: (a) the initial contour, (b) to (c) the evolution of the contour, and (d) the darker part of the object is excluded and the brighter part of the background is included as the segmentation results.

curvature-dependent moving contours. Theoretically, geometric deformable models offer several important advantages over parametric deformable models. First, the intrinsic geometric properties of the interface, such as the unit normal vector and curvature, can be easily and accurately computed from the level set function.[5,10] Second, the topological merging and breaking of the propagating front is handled naturally in level set based methods.[10] Third, level set based deformable models can easily capture sharp corners and cusps.[11] Fourth, the resulting contours in level set based methods do not automatically contain self-intersections.[10] Finally, it is straightforward to implement level set based deformable models in higher dimensions.[5,11] However, geometric deformable models based on the level set techniques are inherently more computationally expensive than parametric deformable models.

A drawback of many deformable models is the initial placement of the contour. Parametric deformable models require the initial contour to be placed somewhere near the target.[6] Similarly, many level set based deformable methods must symmetrically initialize the contour somewhere around the center with respect to the boundary of interest.[11] For example, Fig. 1 demonstrates the evolution of a classical level set based deformable model that uses edge-based stopping terms in segmenting an object with sharp corners and cusps. The contour was initialized close to the upper part of the object and it crossed over the boundary because the stopping factor is small but nonzero.

One way to make the Osher-Sethian level set approach more robust and flexible for practical segmentation applications is to solve the problem directly, as many researchers have attempted.[11] For example, Malladi and Sethian[12] added an attractive force to attract the interface toward the object boundary. This extra term denotes the projection of an attractive force vector on the interface normal. Caselles *et al.*[9] developed a particular active contour model called geodesic active contours, which establishes a connection between energy and curve evolution via the level set framework. Siddiqi *et al.*[13] modified the speed function with an extra term based on the gradient flow derived from a weighted area energy functional. This approach introduces another term into the partial differential equation to balance the speed functions such that the propagating interface can more flexibly evolve toward the desired position. Recently, Gout *et al.*[14] 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. Guyader *et al.*[15] used the Osher-Sethian level set to evolve an explicit function, while minimizing the energy.

Moreover, some of the proposed methods adopt a hybrid system in the form of mathematical and statistical models embedded in the deformable model to guide the segmentation process. For example, Chan and Vese[16] and Tsai *et al.*[17] independently proposed similar hybrid deformable models based on level set methods and Mumford-Shah segmentation techniques.[18] The active contour without edges[16] can detect objects whose boundaries are not necessarily defined by the gradient of images, as in the classical active contour models, and provides robust contour initialization. Subsequently, Vese and Chan[19] extended this model by proposing a multiphase level set framework to segment images with more than two regions. However, if the object has a wide intensity distribution that is similar to the background, this approach cannot correctly capture the target object and some part of the background is incorrectly captured, as shown in Fig. 2.

Another common problem in level set based methods originates from the time interval setting for the numerical computation of time-dependent partial differential equations. An experimental time interval is required such that the position of the propagating interface is advanced in such a way that the evolution of the front can be optimized while capturing the boundary of interest. Precise segmentation can be obtained using a relatively smaller time interval, but the speed of evolution is relatively slow. A large time interval setting can accelerate the advancement, but the deformable contour can miss object boundaries leading to poor results.

An alternative approach for evolving deformable models is to use particle systems. Reeves[20] introduced the idea of using the simulation of a physical system of particles to model objects such as fire, clouds, and water. In this model, particles move under the influence of external forces and constraints without interacting with each other. Numerical simulation techniques using particles are described in detail by Hockney and Eastwood.[21] More recent particle systems use a simulation of molecular dynamics governed by the Lennard-Jones function to add links between the particles to guide the evolution of a deformable model.[22,23] In these elastic particle systems, intensive numerical computation is required to obtain particle attributes such as position, velocity, force, torque, and orientation.

Recently, Jalba *et al.*[24] proposed a new physics-based, electrodynamics model that uses charged particles. Using their charged-particle model (CPM), positive free particles are attracted by fixed negative charges that correspond to each pixel of the edge map of an input image. Each free particle with equal charge moves under the influence of three forces: an internal Coulombic force that is due to the interaction between the particles, an external Lorentz force that is due to the electric field generated by the fixed charges, and a damping force that drives the system to a stable equilibrium state. Among the advantages of this powerful approach are flexible contour initialization and the ability to segment multiple objects in an image. However, an empirical selection of a number of parameters is required for the CPM to converge. A curve (or surface) is reconstructed from the unorganized particles, which might result in gaps in the recovered contour (or surface).

In this paper, we introduce a new physics-based deformable model, the charged fluid method (CFM), by proposing a two-stage evolution framework for image segmentation. The first step distributes the elements of the charged fluid along the propagating interface until an electrostatic equilibrium is achieved. During this procedure, the charge of the fluid elements is interpolated into the neighboring discrete points such that the fluid elements are constrained to move on the lattice. In our approach, the electric force is numerically calculated using the finite-size particle (FSP) method implemented via the fast Fourier transform (FFT) algorithm. After this procedure, the electric force is approximately normal to the propagating contour and can be used to guide curve evolution. The second step advances the propagating front of the charged fluid such that it deforms into a new shape in response to the equilibrium electric force and the image force, which is computed using the smoothed image gradient. The purpose of this procedure is to enable the charged fluid to detect object boundaries that change the shape of the propagating curve. The two-stage evolution is repeated until the propagating front resides on the boundary of objects being segmented. The CFM can provide subpixel precision for the area and length of the segmented region. It requires a contour initialization step to start the algorithm, but we will show that it is less critical than in many existing deformable models. This approach has only one effective parameter, and does not require curvature computation and time interval setting. Last, the CFM automatically handles topological changes at the interface and evolves from an initial contour to the boundary of interest in a variety of images.

The paper is organized as follows. Section 2 introduces the theory of electrostatic systems and describes the FSP method for efficient electric potential computation implemented via the FFT algorithm. Section 3 describes the evolution procedures and the numerical techniques to implement the CFM algorithm. Section 4 demonstrates a variety of potential applications and the validation of our approach on phantom, photographic, and medical images. Finally, in Sec. 5, we discuss the intrinsic properties and essential characteristics of the CFM as well as the difficulties and disadvantages of using the CFM to perform image segmentation.

2 Electrostatic Systems

Systems of charged particles have been broadly studied in the physics community for decades.[25] Computer simulation and modeling methods have been widely used to investigate the behavior of such systems, and are well described in Refs. 25 and 26. Here, we restrict the discussion by considering one of the electrostatic models in two dimensions for simplicity. In an electrostatic system, the force on a particle i due to all other particles is given by Coulomb's law:

$$\mathbf{F}_i = q_i \sum_{\substack{j=1 \\ i\neq j}}^{N} \mathbf{E}_{ij} \propto q_i \sum_{\substack{j=1 \\ i\neq j}}^{N} \frac{q_j(\mathbf{r}_i - \mathbf{r}_j)}{|\mathbf{r}_i - \mathbf{r}_j|^2}, \tag{1}$$

where q is the charge of particle i, N is the total number of particles, and $\mathbf{E}_{ij}$ is the electric field from particle j at position $\mathbf{r}_j$ to particle i at position $\mathbf{r}_i$. The ensemble number of the arithmetic operations required to compute the electric force is[25] approximately of the order of $10\times N^2$. It is impractical to compute the electric force on every particle directly using Eq. (1) for a large number of particles.

Another important consideration of electrostatic models is the collision between particles. The electric force in Eq. (1) gets larger when two particles approach each other, which increases the collision rate. One would like to reduce these collisional effects to some extent such that the model represents a small portion of real plasma electrons or ions.[25] Langdon and Birdsall[27] believe a partial answer is to smooth the interaction at short distances by modifying the particle structure, because it is the long-range behavior of the interaction that governs most plasma phenomena. The new particles have a spread-out charge distribution and move freely through one another; this is the basis of the FSP method.[25,27]

2.1 *FSPs*

The objective of the FSP method is to reduce the collisional effects between particles and avoid the singularities in the use of point particles with Eq. (1), while retaining the long-distance behavior. The charge density of an FSP (or cloud) whose center is at the origin is now changed to $qS(r)$, where q is the charge, and $S(r)$ is the shape factor used to modify the structure of the particle. The shape factor is not required to be isotropic or symmetric but it usually is.[27] In this paper, we assume that the shape factor is real and isotropic as[25,28]

$$S(r) \equiv \frac{1}{2\pi} \exp\left(-\frac{r^2}{2}\right),$$

where $S(r)$ is the Gaussian shape factor in terms of distance r and its integration over the entire space is normalized to unity through the factor $1/2\pi$. The size of the particles is usually set equal to the size of a grid spacing, and their charge is interpolated into the adjacent four, five, or nine grid points with different interpolation techniques.[25]

Rather than calculating the electric force using Coulomb's law in Eq. (1), we can compute it in terms of the electric field. The electric force $\mathbf{F}_i$ of a point charge q_i at position $\mathbf{r}_i$ is related to the given electric field $\mathbf{E}$ at the corresponding position:

$$\mathbf{F}_i = q_i \mathbf{E}(\mathbf{r}_i),$$

where the electric field $\mathbf{E}$ can be computed in terms of a scalar potential Φ:

$$\mathbf{E}(r) = -\nabla\Phi(\mathbf{r}), \tag{2}$$

and the electric potential can be computed from Poisson's equation

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

where $\rho(\mathbf{r})$ is the charge density. Once the charge density in an electrostatic system is known, we can calculate the electrostatic field by using Eqs. (2) and (3). A computationally efficient method to calculate the electric potential through Poisson's equation using the FFT algorithm is described in the next section. We have adopted (x,y) representing Cartesian coordinates in the spatial domain, (m,n) representing discrete coordinates in the spatial domain, and (u,v) representing the corresponding discrete coordinates in the Fourier domain throughout the paper.

2.2 *Electric Potential Computation*

Suppose that a system of charged particles having charge $Q(m,n)$ on grid (m,n) is initialized in a discrete spatial domain, the charge density due to the charged particles can be approximated as

$$\rho(x,y) \cong \sum_{m=1}^{L_m}\sum_{n=1}^{L_n} Q(m,n)S[|\mathbf{r}-\mathbf{r}(m-n)|], \tag{4}$$

where L_m and L_n are the lengths along the m and n axes, respectively. Now, taking the Fourier transform of Eq. (4), we can obtain the relationship between ρ and Q in the Fourier domain. We then apply the same technique to Eq. (3), and the electric potential in the Fourier domain can be expressed as a simple arithmetic function in terms of the charge density. The electric potential,[25,26,28] which is the solution of Poisson's equation, can then be obtained by combining the transformed functions and inverting the potential in the Fourier domain as

$$\Phi(x,y) = \sum_{u=-\infty}^{\infty}\sum_{v=-\infty}^{\infty}{}' \frac{Q(u,v)}{\pi(u^2/L_x^2 + v^2/L_y^2)} \exp\left[-2\pi^2\left(\frac{u^2}{L_x^2} + \frac{v^2}{L_y^2}\right)\right]\exp\left(\frac{2\pi iux}{L_x}\right)\exp\left(\frac{2\pi ivy}{L_y}\right), \tag{5}$$

where the prime represents the fact that $u=v=0$ is excluded from the sum, and L_x and L_y are the computation ranges in the x and y directions, respectively. Here, we have assumed that the system is doubly periodic such that the potential Φ and its normal directive are periodic at the boundaries. In Eq. (5), $Q(u,v)$ is the discrete Fourier transform (DFT) of the charge $Q(m,n)$

$$Q(u,v) \equiv \frac{1}{L_m L_n}\sum_{m=0}^{L_m-1}\sum_{n=0}^{L_n-1} Q(m,n)\exp\left(\frac{-2\pi ium}{L_m}\right) \times\exp\left(\frac{-2\pi ivn}{L_n}\right). \tag{6}$$

Obviously, solving Poisson's equation directly using Eq. (5) is impractical due to the infinite sum. If the potential in Eq. (5) is evaluated only on grid points and interpolated between them, we can facilitate the computation and obtain the discrete potential $\Phi(m,n)$:

$$\Phi(m,n) = \sum_{u=0}^{L_m-1}\sum_{v=0}^{L_n-1}{}' \frac{Q(u,v)}{\pi(u^2/L_m^2 + v^2/L_n^2)} \exp\left[-2\pi^2\left(\frac{u^2}{L_m^2} + \frac{v^2}{L_n^2}\right)\right]\exp\left(\frac{2\pi ium}{L_m}\right)\exp\left(\frac{2\pi ivn}{L_n}\right). \tag{7}$$

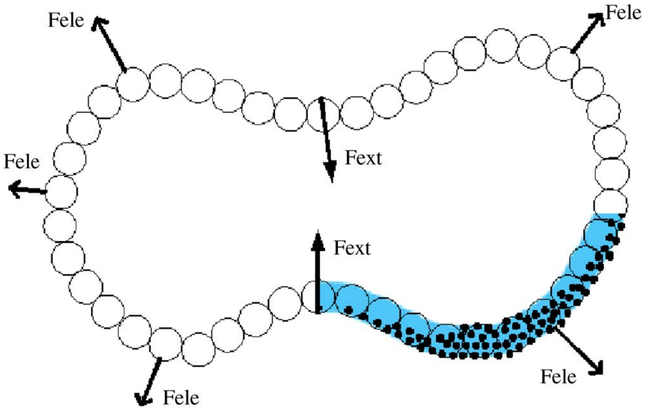

Fig. 3 Concept of a charged fluid. A charged fluid conceptually consists of charged elements (the large circles), each of which exerts a repelling electric force on the others. The fluid elements, as if they were consisted of different amounts of charged particles (the solid dots), are connected to one another by 8-connectivity when they advance. The charged fluid, behaving like a liquid, can be influenced by internal electric forces $\mathbf{F}_{\text{ele}}$ of repulsion as well as external forces $\mathbf{F}_{\text{ext}}$ from the image data.

The DFT pair in Eqs. (6) and (7) can then be rapidly computed via the FFT algorithm provided that $L_m=2^s$ and $L_n=2^t$, where s and t are positive integers. In the following section, we describe a new deformable model that makes use of the numerical techniques of this electrostatic model to perform the electric potential computation.

3 Charged Fluid Method

If a system of charged particles is initialized within a small region, the particles will gradually move outward due to the repelling electric forces between one another. Now, suppose that a system of charged particles is initialized inside a region of interest (ROI) in an image. The particles will keep advancing outward 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.

Now, suppose that the charged particles (indicated by the solid dots in Fig. 3) are confined inside an isolated conductor that models the closed propagating interface as a curve in two dimensions or a surface in three dimensions, such that the particles can move only on the interface until an electrostatic equilibrium is achieved. The particles accumulate where there is a locally greater curvature in the equilibrium state (see Table 1), as shown in Fig. 3. To make use of such a charged system for image segmentation, we

Table 1 The essential characteristics of charged conductors in electrostatic equilibrium. Charged conductors that have reached electrostatic equilibrium, which means that there is no net flow of electric charge or no electric current, share a variety of unusual characteristics.[29,30]

	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 of σ/ϵ_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 electrostatic equilibrium is at the same potential (the surface is an equipotential).

propose a charged fluid system in such a way that each fluid element (the large circles in Fig. 3) is treated as a cloud in the FSP model and has its own charge as if it was calculated by interpolating the charges of the covered particles (the solid dots in Fig. 3). The electric forces ($\mathbf{F}_{\text{ele}}$ in Fig. 3) 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. 3. Having described these basic concepts of the charged fluid, we now describe the evolution procedures and the numerical techniques used to implement the algorithm.

3.1 *Charge Interpolation*

We begin by describing 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 on Newton's law of 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

$$d = \frac{F_i}{2m}\Delta t^2 = \frac{1}{2}E_i \Delta t^2, \tag{8}$$

where E_i is the electric field corresponding to F_i.

Now, assume that one arbitrary fluid element with charge Q is advanced to position (d_x, d_y) with respect to the origin, as illustrated in Fig. 4(a). Since fluid elements are constrained on grids, one simple way to achieve this is to linearly interpolate the charge Q into adjacent discrete points. Considering the interpolation accuracy as well as the computational cost, we use the SUDS technique, which is the accelerated version[31] of the DEM, to interpolate the charge to the nearest grid point and its 4-neighbors while achieving the conservation of charge, as shown in Fig. 4(b).

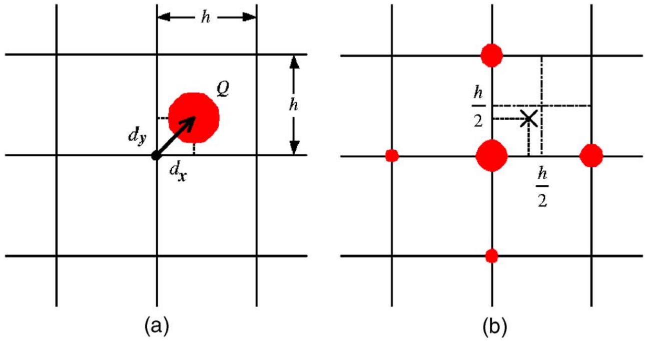

Fig. 4 Charge interpolation using the subtracted dipole scheme (SUDS) technique, which is the accelerated version of the dipole expansion method (DEM): (a) the fluid element with charge Q is advanced to position (d_x, d_y) and (b) the fluid element in (a) is linearly interpolated to the nearest grid point and its 4-neighbors with different charges, if any.

3.2 *Evolution*

The evolution of the CFM consists of two different procedures. The first procedure, charge distribution, enables the CFM to flow along 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.

3.2.1 *Charge distribution*

One practical problem of using the electrostatic system described in Sec. 2 to advance a group of charged particles is that the electric potential in the system varies when the positions of the particles change. 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 $\hat{\Phi}^j_{\text{ele}}$ for the charged fluid j as

$$\hat{\Phi}^j_{\text{ele}} = \frac{\Phi^j_{\text{ele}}}{\bar{\Phi}^j}\Phi_0,$$

where Φ_0 is an arbitrary positive constant, and $\bar{\Phi}^j$ is the mean electric potential in the charged fluid j. The corresponding normalized charge density is defined as

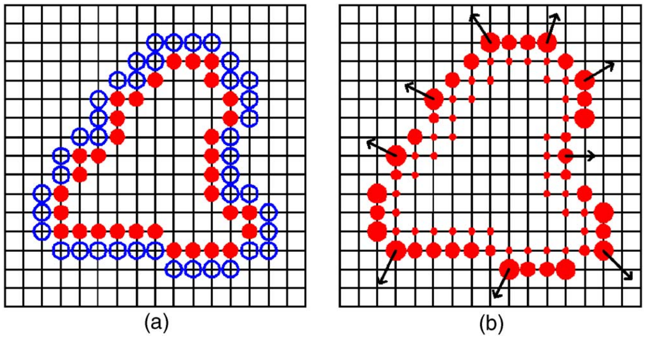

Fig. 5 Charge distribution procedure. (a) At beginning of this procedure, a uniform charge distribution is applied to the fluid elements (the red solid dots). They are only allowed to share charge on the 2-pixel-wide propagating interface that is obtained from the front deformation procedure (Fig. 6 in Sec. 3.2.2). Note that the empty charge positions on the interface are represented by the blue hollow circles. (b) The 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 a standard contour tracing method. Note that the CFM in this procedure is a pure electrostatic system without the influence of external forces.

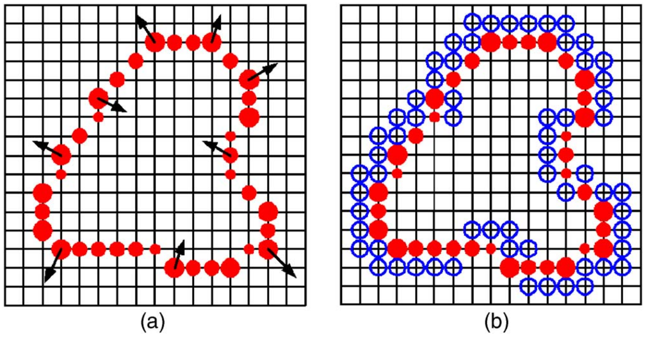

Fig. 6 Front deformation procedure. (a) After the charge distribution procedure, the fluid elements are on the 1-pixel-wide front by 8-connectivity. Note that the tiny inner charges in Fig. 5(b) are discarded after the contour tracing procedure. The effective fields (the arrows) are computed based on 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. 5(b). (b) The new 2-pixel-wide propagating interface is obtained by locating the four adjacent grid points according to the effective field directions in (a) based on Fig. 7 for all elements. Note that, compared to Fig. 5(a), the propagating interface evolves into a different shape in response to the effective fields in (a).

$$\hat{\rho}^j = \frac{\rho^j}{\bar{\Phi}^j}\Phi_0. \tag{9}$$

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

$$\nabla^2\hat{\Phi}_{\mathrm{ele}} = \nabla^2\sum_j \hat{\Phi}^j_{\mathrm{ele}} = -4\pi\sum_j \hat{\rho}^j = -4\pi\hat{\rho}, \tag{10}$$

where $\hat{\Phi}_{\mathrm{ele}}$ is the normalized electric potential, and $\hat{\rho}$ is the normalized charge density of the overall system at each time step. During this procedure, the electric field $\mathbf{E}_{\mathrm{ele}}$ is directly computed using the normalized electric potential:

$$\mathbf{E}_{\mathrm{ele}} = -\nabla\hat{\Phi}_{\mathrm{ele}}, \tag{11}$$

where $\hat{\Phi}_{\mathrm{ele}}$ in Eqs. (10) and (11) is computed using a computationally efficient technique, as described in Sec. 2. Once the discrete potential is obtained, the discrete electric field on each fluid element is directly computed using the central difference approximation of Eq. (11).

To advance the fluid elements within the propagating interface using the interpolation technique described in Sec. 3.1, we further restrict the motion of the fluid element having the maximum electric field in the system such that it is advanced by the half grid spacing $h/2$ in the electric field direction, then the time interval for each fluid element can be obtained from Eq. (8):

$$\Delta t^2 = \frac{h}{E_{\max}} = \frac{h}{|(E_m^2 + E_n^2)^{1/2}|_{\max}}, \tag{12}$$

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 $E_{\max}$ on the m and n axes, respectively. The advantage of this approach is that there is no time interval setting for advancing fluid elements. Substituting Eq. (12) into Eq. (8), the distances d_x^i and d_y^i for each fluid element i can be easily obtained:

$$d_x^i = \frac{E_m^i}{E_{\max}}\frac{h}{2},$$

$$d_y^i = \frac{E_n^i}{E_{\max}}\frac{h}{2}, \tag{13}$$

where E_m^i and E_n^i are the electric field components of element i on the m and n axes, respectively. Equation (13) implies that the distance by which each fluid element advances is related to the maximum electric field in the system without the influence of the time interval setting. Once the distances are obtained, we can use the SUDS technique described in Sec. 3.1 to interpolate the charge to the five discrete neighbors. Note that if any of the 4-neighbors is not on the propagating interface, its charge is preserved to

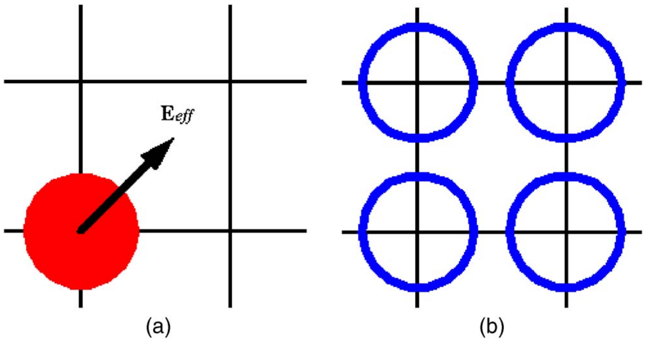

Fig. 7 Locating the 2-pixel-wide binary interface on an individual fluid element: (a) the effective field $\mathbf{E}_{\mathrm{eff}}$ on a fluid element (the red solid dot) and (b) the four adjacent grid points (the blue hollow circles) of the element are generated according to the effective field direction in (a) and denoted as a part of the 2-pixel-wide propagating interface.

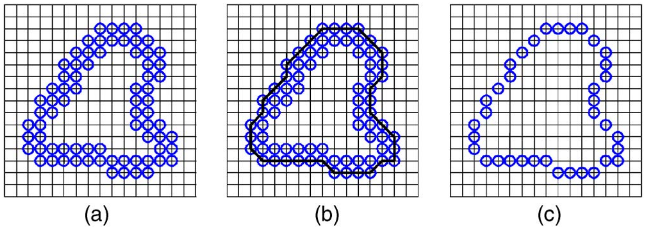

Fig. 8 Generation of the 1-pixel-wide front by refining the 2-pixel-wide interface using a standard contour tracing algorithm: (a) the location of the 2-pixel-wide interface on a binary array, (b) applying the contour tracing algorithm to refine the interface, and (c) the location of the 1-pixel-wide front.

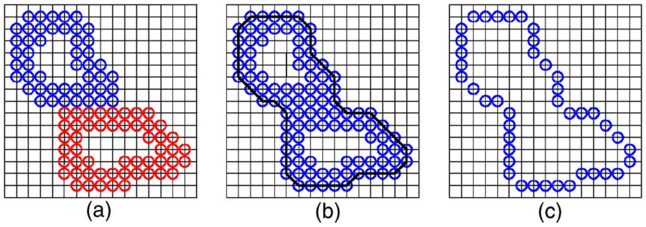

Fig. 9 Generation of the 1-pixel-wide front for two merged charged fluid contours: (a) the location of the 2-pixel-wide interfaces on a binary array, (b) applying the contour tracing algorithm to refine the interface, and (c) the location of the 1-pixel-wide front.

the old position, which is the center of the five neighboring grids (see Fig. 4).

Electrostatic equilibrium is defined as a state of zero net flow of electric charge (see Table 1). 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:

$$\frac{\Delta Q_{\text{total}}}{Q_{\text{total}}} \leqslant \gamma, \tag{14}$$

where Q_{total} is the total charge of the overall system for each iteration, ΔQ_{total} is the net flowing charge in total, and γ determines the degree of electrostatic equilibrium. Note that the system conserves charge throughout the charge distribution procedure, i.e., Q_{total} is the same for each iteration during this process.

In our approach, a uniform charge distribution over the fluid elements is initially placed on the 2-pixel-wide propagating interface that is obtained from the other procedure, as illustrated in Fig. 5(a). The fluid elements are repeatedly advanced inside the propagating interface, which is treated as the curve in two dimensions or the surface in three dimensions of an isolated conductor, until the overall system converges to an equilibrium state that satisfies Eq. (14). One of the interesting properties of conductors in electrostatic equilibrium (see Table 1) is that the electric field is perpendicular to the curve or surface as shown in Fig. 5(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. Another important property of an isolated conductor in equilibrium is that any net charge resides entirely on its curve (in two dimensions) or surface (in three dimensions). For computer simulation, the challenge is to define the curve or surface thickness such that the charged fluid can reasonably approximate a real system. After the system reaches the specified electrostatic equilibrium with Eq. (14), a more strict curve or surface corresponding to a 1-pixel-wide propagating front (contour) is defined such that only fluid elements on that contour are preserved, which is described in Sec. 3.3.

3.2.2 *Front deformation*

After the charge distribution procedure, the charged fluid is still an isolated system without the interaction with external forces. This procedure enables the CFM to interact with the image data such that the 1-pixel-wide propagating front deforms the shape in response to the gradient of an image. We define the image gradient potential using the following equation:

$$\Phi_{\text{img}}(m,n) = \beta \frac{|\nabla G_{\sigma}(m,n) * I(m,n)|}{|\nabla G_{\sigma} * I|_{\max}} \Phi_0, \tag{15}$$

where β is a weighting factor ($\beta \leqslant 0$) to adjust the image gradient potential, ∇ is the gradient operator, and $G_{\sigma}(m,n)$ is a 2-D Gaussian function with standard deviation σ. In Eq. (15), $*$ represents convolution, $|\cdot|$ is the modulus of the smoothed image gradients, and $|\cdot|_{\max}$ is the maximum modulus in the computation domain. The selection of values for β is discussed in Sec. 4.

The image potential Φ_{img} in Eq. (15) is further incorporated into our electrostatic model by defining the effective potential Φ_{eff}

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

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. To deform the propagating front in response to the effective potential, the corresponding effective field $\mathbf{E}_{\text{eff}}$ is defined

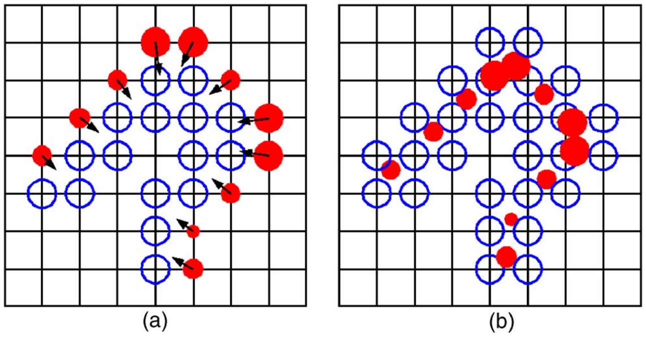

Fig. 10 Subpixel precision computation: (a) the effective fields on the fluid elements (the red solid dots) are approximately oriented inward after the evolution is terminated and (b) the subpixel precision for the area and length of the ROI can be calculated by advancing the fluid elements a real number distance based on the effective fields in (a) using Eq. (13).

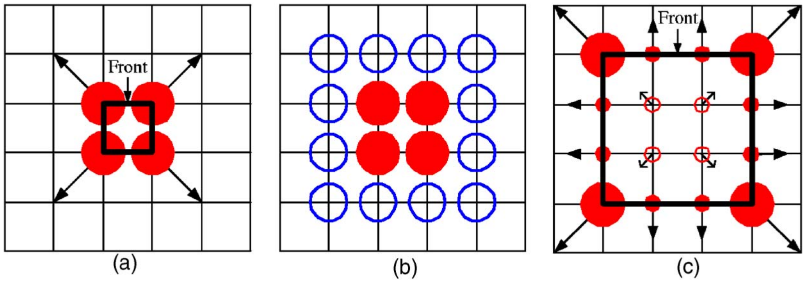

Fig. 11 Front propagation using a single charged fluid contour in a free 2-D space: (a) the initial front of the charged fluid consists of four fluid elements, (b) the 2-pixel-wide propagating interface is obtained by locating the four adjacent grid points according to the electric field directions in (a) based on Fig. 7 for all elements, and (c) a new propagating front (1 pixel wide) is obtained by the standard contour tracing technique after achieving the specified electrostatic equilibrium.

$$\mathbf{E}_{\text{eff}} = -\nabla\Phi_{\text{eff}} = -\nabla(\hat{\Phi}_{\text{equ}} + \Phi_{\text{img}}), \tag{16}$$

where the discrete effective fields can be numerically computed using the central difference approximation. When the gradient of the image potential is insignificant, i.e., in a relatively homogeneous 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. 6(a). The charged fluid relies on the salient image gradient to change the direction of $\mathbf{E}_{\text{eff}}$ in Eq. (16) such that the fluid elements are confined inside the ROI. The front deformation is executed on each fluid element by locating binary positions corresponding to the four adjacent grid points based on the effective field direction, as illustrated in Fig. 7. Therefore, we can generate the 2-pixel-wide binary interface by applying this binary localization method to each fluid element, as shown in Fig. 6(b). Note that, after this procedure, the propagating interface of the CFM evolves into a different shape in response to the effective field.

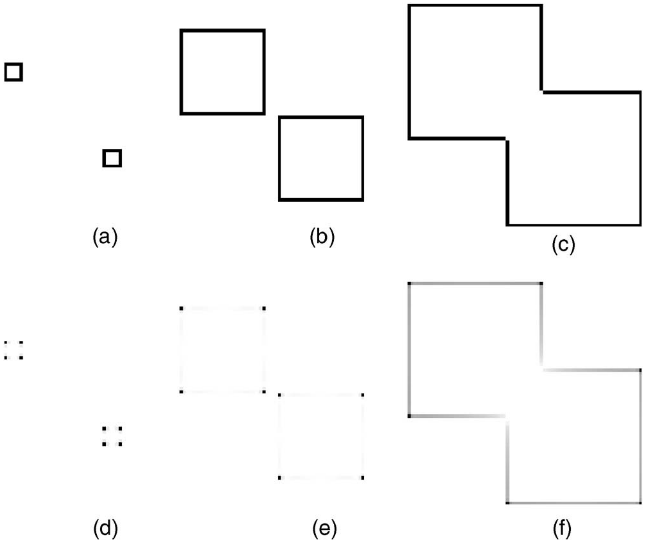

Fig. 12 Front propagation using two charged fluid contours in a free 2-D space (obtained from the computer simulation results): (a) initialization of two charged fluids using square contours, which are treated as independent systems when they are away without touching each other; (b) the geometry of both contours retains the same (square) while they are evolving and approaching; (c) the two charged fluids merged to one that continues to evolve; and (d) to (f) the inverted charge density plots of (a) to (c), respectively.

3.3 *Front Construction*

As described in Sec. 3.2, the 2-pixel-wide interface [see Fig. 5(b)] is refined to the 1-pixel-wide front [see Fig. 6(a)] for curve evolution. This is achieved by creating a binary array that indicates the positions [the circles in Fig. 8(a)] of the 2-pixel-wide interface. A standard contour tracing algorithm[32] starting from the maximum charge element on the 2-pixel-wide interface is then used for refinement, as shown in Fig. 8(b). Note that the algorithm aims to trace the outer positions of the interface. Finally, in Fig. 8(c), the location of the 1-pixel-wide front of the CFM is generated such that the fluid elements are 8-connected. Figure 9 illustrates the generation of the 1-pixel-wide front for two merged charged fluid contours. Note that the interfaces in Fig. 9(a) are drawn with different colors only for better visualization and understanding. After the contour tracing procedure, the inner fluid elements are discarded and a new 1-pixel-wide front is generated that represents only one charged fluid contour, as shown in Fig. 9(c). Therefore, the topological changes of fronts for multiple charged fluids are handled in this procedure. Note that the CFM does not allow for contour splitting due to the contour tracing techniques being used.

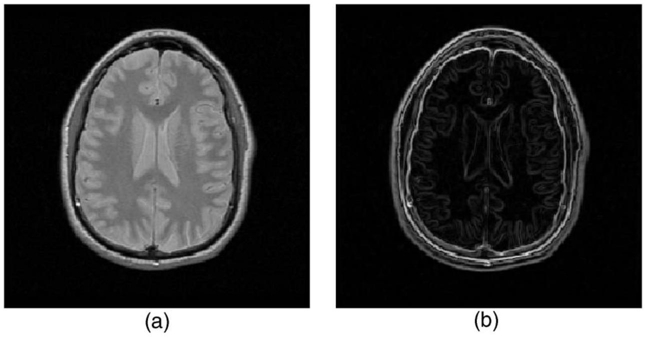

Fig. 13 Example of the normalized image gradient map using a 3 ×3 Gaussian filter: (a) the given image and (b) the normalized smoothed image gradient map.

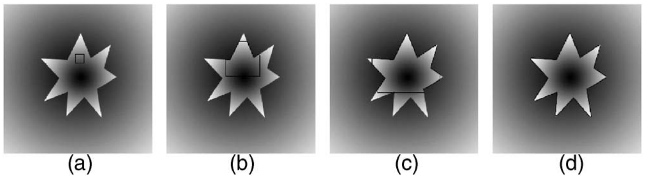

Fig. 14 Experimental results in segmenting an object with sharp corners and cusps using $\beta=-1.2$ in a 256×256 phantom image. Note that the object has a wide intensity distribution that is similar to the background. (a) The initial contour was purposely placed close to the upper part of the object. (b) to (c) The evolution of the contour. (d) The segmentation results with $\kappa=94.8\%$. The processing time was approximately 20 s.

3.4 *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 grids 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, as shown in Fig. 10(a). However, the effective fields on the rest of the final 2-pixel-wide interface [the circles in Fig. 10(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 can locate the boundary with subpixel precision by advancing the fluid element a real number distance, which is computed by substituting the effective fields into Eq. (13). Now, the fluid elements in Fig. 10(b) are located on real number points and the required precision of the specific application can be obtained by using the appropriate interpolation techniques[33–35] to calculate the area and length of the ROI.

3.5 *Initialization and Front Propagation*

Segmentation methods can be classified into manual, semiautomatic, and automatic schemes based on the level of interaction with users. Initialization of segmentation algorithms is one of the most important steps for a fully automatic procedure. The initialization presented in this section leads to a semiautomatic segmentation. Theoretically, any kind of geometrical shape can be used for the initial contour. In our approach, an initial square contour was interactively placed inside an ROI by the end-user for its simple construction. The front propagation of the CFM in a free 2-D space using single and multiple square contours is described in the following.

3.5.1 *Front propagation using a single charged fluid contour*

Figure 11(a) shows that there is one CFM initialized in a free plane. The four fluid elements having equal charge

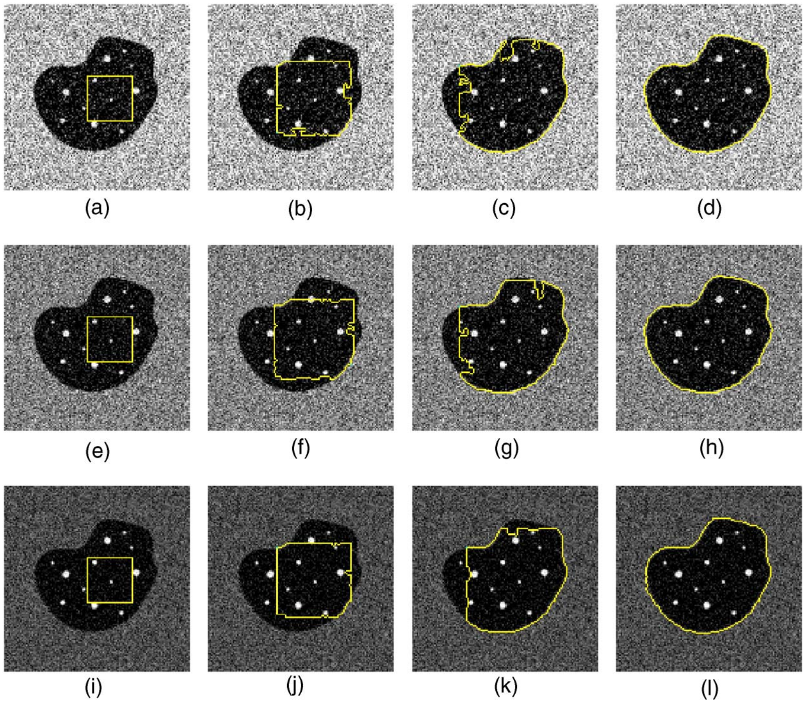

Fig. 15 Experimental results using three simulated, irregularly shaped phantom images (128×128) with high-intensity structural noise embedded in the object and Gaussian background noise. The three images are of 30% black background with 20% Gaussian noise, 50 with 15%, and 70 with 10% in (a) to (d), (e) to (h), and (i) to (l), respectively. (a), (e), and (i) One initial contour was manually placed to segment the object. Figures in the middle two columns are the evolution of the corresponding charged fluid that flowed around the high-intensity structural noise and through the lower background noise. (d), (h), and (l) The CFM overcame the structural and background noise, and successfully segmented the region of interest with conformity $\kappa=96.8$, $\kappa=96.7$, and $\kappa=98.9\%$, respectively. The processing time for all experiments was approximately 4 s, while using slightly different values of $\beta=-1.2$ for (a), $\beta=-1.4$ for (e), and $\beta=-1.6$ for (i).

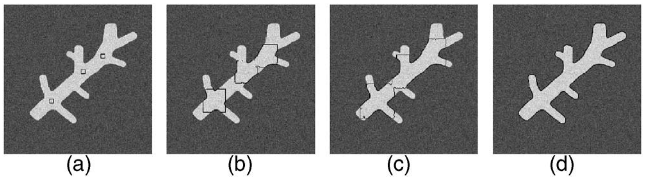

Fig. 16 Segmenting a simulated vascular-shaped object in a 256 × 256 image with 5% Gaussian background noise using multiple charged fluids: (a) three charged fluid contours with $\beta=-1.2$ manually placed inside the simulated structure; (b) the charged fluids then advanced toward the boundary of the simulated structure and two of them were merged while evolving; (c) the remaining two charged fluids were further merged while advancing toward the boundary of interest, and (d) the whole boundary of the ROI was segmented with $\kappa=95.0\%$ and the approximate processing time was 18 s.

determine the position of the initial propagating front. The propagating interface (2 pixels wide) is then computed based on the directions of the effective fields in the front deformation procedure, as illustrated in Fig. 11(b). During the charge distribution procedure, the fluid elements are only allowed to share charges at those positions. When the specified electrostatic equilibrium condition is achieved, the new propagating front (1 pixel wide) can be obtained using the contour tracing technique described in Sec. 3.3. Note that the charge distribution of the system is usually not uniform [see Fig. 11(c)]. This is due to the fact that the electric field is greater where the curvature is larger for an irregularly shaped conductor in electrostatic equilibrium (see Table 1).

3.5.2 *Front propagation using multiple charged fluid contours*

One of the advantages of the charged fluid method is that we can automatically handle the topological changes of multiple propagating fronts, as described in Sec. 3.3. The front propagation of multiple charged fluids is illustrated in Fig. 12(a), where two charged fluids are initialized in a free plane [Fig. 12(d) shows the corresponding charge density]. The charged fluids are treated as independent systems when they are not touching each other. Moreover, the square contours retain the same geometry while they are evolving in a free space, as shown in Fig. 12(b), and the corresponding charge density is shown in Fig. 12(e). However, the fluid elements from one charged fluid not only interact with themselves but also with those from the other fluid when they are close. The two propagating fronts continue to evolve and are merged when they contact each other. Figure 12(c) shows the merged propagating front of the CFM after further evolution, and the corresponding charge density is shown in Fig. 12(f).

3.6 *Segmentation Algorithm and Implementation*

The pseudocode of the overall charged fluid algorithm is given in Algorithm 1, which consists of two core algorithms corresponding to the charge distribution procedure and the front deformation procedure, respectively. The algorithm is terminated when the number of the fluid elements on the 1-pixel-wide front (see Figs. 8 and 9) remains equivalent for two consecutive steps, i.e., there is no deformation in the charged fluid shape after one more iteration. The software was developed[36] in Java using the UCLA jViewBox for image I/O, display and manipulation.

Algorithm 1. Charged fluid

```
1. parameter setting of $\beta$ in Eq. (15) and $\gamma$ in Eq. (14)
2. image potential computation using Eq. (15)
3. repeat(i)
    (a) uniform charge distribution over fluid elements
    (b) repeat (j)
            Algorithm 2
    (c) until(j) electrostatic equilibrium is achieved
        based on Eq. (14)
    (d) 1-pixel-wide front construction described in
        Sec. 3.3
    (e) Algorithm 3
    (f) mean potential computation and charge normalization using Eq. (9).
4. until(i) no deformation in the charged fluid shape
5. subpixel precision calculation, if desired
```

Algorithm 2. Charge distribution procedure

```
1. forward FFT computation of the charge array based
   on Eq. (6)
2. inverse FFT computation of the potential array based
   on Eq. (7)
3. electric field computation using Eq. (11)
4. advance distance computation using Eq. (13)
5. charge interpolation using the SUDS based on Fig. 4
```

Algorithm 3. Front deformation procedure

```
1. effective field computation using Eq. (16)
2. 2-pixel-wide interface localization based on Fig. 7
```

The charge distribution procedure (Algorithm 2) dominates the overall computational cost of the charged fluid algorithm. Using an FFT-based FSP algorithm, we change 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 $L_m=L_n=M$ [see Eqs. (6) and (7)]. The iterations of Algorithm 2 in loop j of Algorithm 1 are approximately 5 for $\gamma=5\%$ and 10 for $\gamma=1\%$. Similarly, the computational complexity of the CPM method is dominated by the calculation of the Coulombic force $O(N \log N)$, with N equal to the number of particles.[24] Since level set based deformable model methods have added one extra dimension to the problem, its complexity is $O(n^3)$ in three dimensions with n equal to the number of grid points in the spatial direction.[5] The narrow band efficient implementation technique has reduced the computational complexity to $O(kn^2)$, with k equal to the number of cells in the narrow band.[5] In the following section, we discuss the settings of the parameters in our approach: β and γ. Some trivial constants are given as follows: $\Phi_0=10{,}000$ in Eq. (15) and $h=1$ in Eq. (13).

4 Results

We applied the CFM algorithm to segment simulated phantom images, photographic pictures as well as medical im-

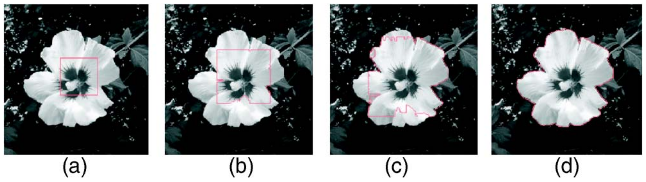

Fig. 17 Experimental results in capturing the flower in a digital picture (256 × 256): (a) one initial contour with $\beta = -1.1$ manually placed inside the flower; (b) the fluid elements at the black inner region boundary were trapped and became the inner part of the front, while the contour was evolving outward due to the repulsive force between fluid elements; (c) the inner fluid elements were discarded when the outer fluid elements were merged using the contour tracing technique; and (d) the contour finally located at the boundary of the flower with $\kappa = 96.4\%$ and the processing time was approximately 24 s.

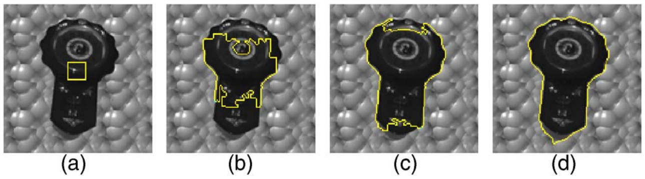

Fig. 18 Experimental results in capturing the clock in a digital picture (128 × 128) with texture background: (a) One initial contour with $\beta = -1.8$ manually placed inside the clock; (b) and (c) the CFM overcame the inner structures and evolved toward the boundary of the clock during the segmentation process; (d) finally, the clock was segmented including the shadow with $\kappa = 91.9\%$ and the approximate processing time was 9 s.

ages. In all experiments, we used 7% for the value of γ, which leads to only one remaining parameter β in many applications. The value of β is related to the position of the maximum gradient magnitude of an image [see Eq. (15)]. It is usually set close to unity (absolute value), however, if the maximum gradient position is not in the ROI, 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 as illustrated in Fig. 13. To quantitatively evaluate our approach, we define an overall pixel-based measure, conformity κ, which includes measures of undersegmentation and oversegmentation

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

where F_P represents false positives, F_N represents false negatives, and T_P represents true positives of the segmentation results. All measures were based upon manual segmentation results for each experiment that was executed on a Pentium M 1.6 GHz machine with a Windows XP operating system.

4.1 *Experiments on Simulated Phantoms*

We begin by demonstrating the ability of the CFM in segmenting sharp corners and cusps in a 256 × 256 phantom image, as shown in Fig. 14. Note that the object and the background have a similar intensity distribution, which is difficult to handle for intensity-based segmentation approaches. Figure 14(a) shows the initial contour with $\beta = -1.2$ that is purposely placed close to the upper part of the object. The object was segmented with conformity $\kappa = 94.8\%$, as shown in Fig. 14(d), which cost approximately 20 s.

One of the major advantages of the CFM is that it can be applied to segment irregularly shaped objects with some high-intensity structural noise and lower background noise. Examples are illustrated in Fig. 15 using 128 × 128 noisy phantom images. An initial contour in Fig. 15(a) was manually placed to segment the irregularly shaped object with different kinds of noise artifacts. Figures 15(b) and 15(c) show the evolution of the charged fluid flowing through the 20% Gaussian background noise and around the high-intensity structural noise. Although the front propagation was influenced by the structural and background noise, the CFM successfully segmented the boundary of the irregularly shaped object, as shown in Fig. 15(d). We also demonstrate our deformable model working on two other images with the same geometry but less intensity contrast and noise in Figs. 15(e)–15(h) and Figs. 15(i)–15(l), respectively. Slightly different values of β [−1.2 for Fig. 15(a), −1.4 for Fig. 15(e), and −1.6 for Fig. 15(i)] were used in segmenting the three images due to the different intensity contrast. The scores of the conformity κ are 96.8% for Fig. 15(d), 96.7% for Fig. 15(h), and 98.9% for Fig. 15(l), respectively. The approximate processing time for all experiments in Fig. 15 was 4 s.

Figure 16 illustrates the ability to capture deep concavities and cusps in a simulated vascular-shaped object, as well as the merging behavior of multiple charged fluids. Three charged fluid contours with $\beta = -1.2$ were manually placed inside the simulated vascular structure in a 256 × 256 image with 5% Gaussian background noise, as shown in Fig. 16(a). We show in Fig. 16(b) that the charged fluids were advanced toward the boundary of the simulated structure and two of them were merged while evolving. The remaining two charged fluids were further merged while advancing toward the boundary of interest, as shown in Fig. 16(c). Finally, in Fig. 16(d), the whole boundary of the ROI was segmented with $\kappa = 95.0\%$, and the approximate processing time was 18 s.

4.2 *Experiments on Photographic Pictures of Real Scenes*

The segmentation of objects in photographic pictures was also evaluated. Figure 17 illustrates the ability of the CFM to segment the flower in a 256 × 256 digital picture. The position of the initial contour is shown in Fig. 17(a). Note that the contour crossed some part of the black inner region. The fluid elements at the black region boundary were trapped and became the inner part of the front while the contour was evolving outward due to the repulsive force between fluid elements, as shown in Fig. 17(b). They were discarded when the outer fluid elements were merged using the contour tracing technique, as illustrated in Fig. 17(c). Finally, in Fig. 17(d), the contour located at the boundary of the flower was obtained. The approximate processing time was 24 s. The values of β and κ for this experiment were −1.1 and 96.4%, respectively.

A charged fluid with $\beta = -1.8$ was manually initialized in a 128 × 128 photographic picture superimposed by a texture background, as shown in Fig. 18(a). The CFM contour

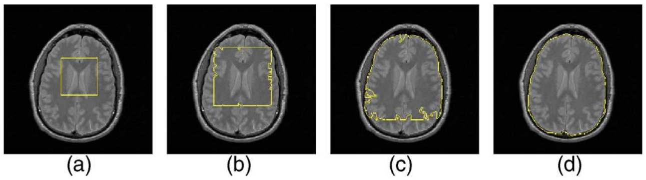

Fig. 19 Segmenting the brain in an MR image (256×256): (a) one charged fluid with $\beta=-1.0$ manually placed inside the brain, (b) and (c) the charged fluid flowed through the inner structures and evolved toward the boundary of the brain, and (d) the segmentation results with $\kappa=97.1\%$ that took approximately 24 s.

then advanced toward the boundary of the clock, as shown in Figs. 18(b) and 18(c). Since the intensity of the clock shadow and the clock itself were somewhat close, the final segmentation results in Fig. 18(d) were slightly poor with conformity $\kappa=91.9\%$. The processing time was approximately 9 s.

4.3 *Experiments on Magnetic Resonance (MR) Images*

In the previous sections, we illustrated how the charged fluid successfully handled various common noise artifacts. Here, we demonstrate the use of the CFM in the segmentation of medical images. Figure 19(a) shows an initial contour ($\beta=-1.0$) that was manually placed inside the brain in an MR image (256×256). Figures 19(b) and 19(c) show that the charged fluid flowed through the inner structures and evolved toward the boundary of the brain. Figure 19(d) shows the segmentation results with conformity $\kappa=97.1\%$ and the approximate processing time was 24 s. We also investigated the effect of the weight β in Eq. (15) to the segmentation results in an MR image. All experiments used a charged fluid with the same initial contour and position [see Fig. 20(a)], but different β values to extract the tumor with some blurred boundaries in an MR image (256×256). The CFM contours leaked from where there are relatively weaker boundary gradients when using relatively lower (absolute) values of β. A typical leakage result using $\beta=-3.0$ is illustrated in Fig. 20(b). The contour leakage stopped when we increased the (absolute) value of β to -4.5, which achieved a $\kappa=90.9\%$ result, as shown in Fig. 20(c). The segmentation results are approximately the same until an extremely high (absolute) value of $\beta=-15.0$ is used to segment the tumor, as shown in Fig. 20(d). Multiple charged fluids can also be used to simultaneously segment objects in images. For example, Fig. 21(a) shows two charged fluids with $\beta=-27.0$ that were initialized inside a ventricle with blurred boundaries in an MR image (256×256). The charged fluids evolved toward the boundary of the ventricle, as illustrated in Figs. 21(b) and 21(c). Finally, Fig. 21(d) shows the individual charged fluid merged at the boundary of the ventricle with $\kappa_c=86.4\%$. The entire procedure took approximately 2 s.

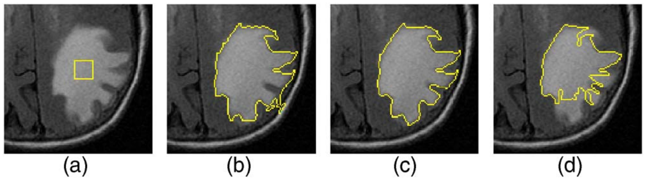

Fig. 20 Influence of different values of β in Eq. (15) in the segmentation of the tumor in an MR image (256×256) using the CFM: (a) the initial contour of the CFM for all experiments, (b) an example of the contour leaking from relatively weaker boundaries using $\beta=-3.0$, (c) an example ($\kappa=90.9\%$) of the CFM in segmenting the tumor using $\beta=-4.5$, and (d) the segmentation results using an extremely high (absolute) value of $\beta=-15.0$. Note that the segmentation results are approximately the same using different values of β ranging from -4.5 to -15.0.

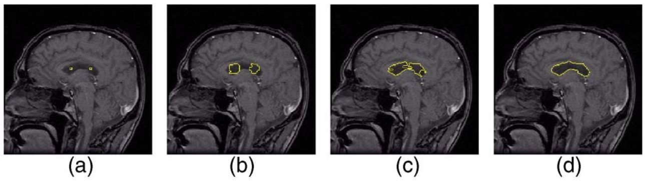

Fig. 21 Applying two simultaneous charged fluids to segment the ventricle in an MR image (256×256): (a) two charged fluids with $\beta=-27.0$ initialized in the ventricle, (b) to (c) the charged fluids evolved and flowed toward the boundary of the ventricle, and (d) the segmentation results with $\kappa_c=86.4\%$ that took approximately 2 s.

4.4 *Experiments on Computed Tomography (CT) Images*

We also applied the CFM to segment large objects in CT images (512×512). For example, Fig. 22 shows the initial contours (inner squares) as well as the corresponding results in the segmentation of the lungs in two different chest CT images using the charged fluid technique. The values of β are respectively -1.4 for Fig. 22(a) and -1.5 for Fig. 22(b). The approximate processing times taken to segment Fig. 22(a) with $\kappa=96.3\%$ and Fig. 22(b) with $\kappa=97.7\%$ were 57 and 47 s, respectively. Two different charged fluid contours with the same $\beta=-3.0$ were applied to segment the brain in multiple adjacent CT slices as illustrated in Figs. 23(a) and 23(b). While the conformity values are high with $\kappa=97.0\%$ for Fig. 23(a) and $\kappa=98.7\%$ for Fig. 23(b), the processing time was expensive (approximately 2 min 26 s). Finally, in Fig. 23(c), we demonstrate a failed example of using the CFM to segment an object with broken boundary gradients in a CT image. We manually initialized

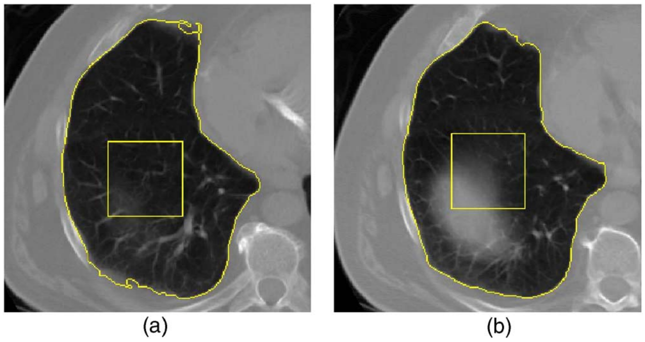

Fig. 22 Segmentation results of the lung in chest CT images (512×512) with conformities (a) $\kappa=96.3\%$ and (b) $\kappa=97.7\%$. The charged fluids with $\beta=-1.4$ and $\beta=-1.5$ evolved from the inner squares toward the boundary of the lung that took approximately 57 s for (a) and 47 s for (b), respectively.

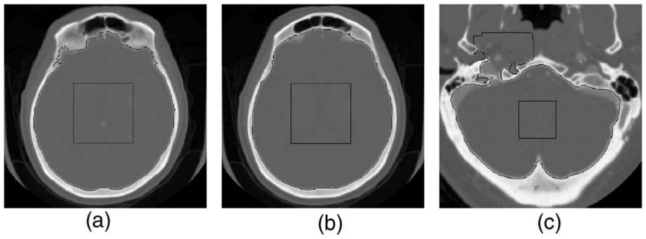

Fig. 23 Examples of segmenting the brain in CT images (512 × 512). The inner squares represent the initial contours. (a) and (b) The segmentation results [κ=97.0% for (a) and κ=98.7% for (b)] using the CFM (β=−3.0) in extracting the brain in multiple adjacent CT slices, which took approximately 2 min 26 s. (c) A failed example of using the CFM (β=−1.0) in the segmentation of the brain in a CT image. The contour leaked through the broken boundaries because there are no significant image gradient forces to confine the fluid elements at those positions.

a charged fluid contour [the inner square in Fig. 23(c)] with β=−1.0 inside the brain. While most fluid elements were confined at the boundary of the ROI, few of the fluid elements leaked through the broken boundaries and kept moving outward.

5 Discussion

We described a new, physics-based deformable model using a simulated charged fluid, and demonstrated the ability and behavior of this algorithm in segmenting a variety of photographic and medical images. The CFM is capable of capturing objects with sharp corners and cusps that have similar intensity distribution to the background, as shown in Fig. 14 (see also Figs. 1 and 2). Figures 15, 17, and 18 illustrate the ability of the algorithm to flow through and around structural and background noise to successfully segment objects of interest. A characteristic of this approach is that the fluid elements flow around inner obstacles because there is no charge inside an isolated conductor in electrostatic equilibrium (see Table 1).

We demonstrated the characteristics and performance of the charged fluid method on a variety of images. For example, Figs. 16 and 21 respectively illustrate the use of multiple charged fluids to capture deep concavities and cusps in a simulated vascular structure, and to extract the ventricle in an MR image. Figure 19 illustrates the segmentation of the brain in an MR image. The boundary gradients of the inner structures are weaker and somewhat discontinuous, so the charged fluid flowed around and through the inner structures and successfully segmented the brain. Finally, Figs. 22 and 23 illustrate the segmentation of the lung in chest CT images and the brain in adjacent CT images.

Many high-level methods use a variety of mathematical techniques with or without prior knowledge to achieve the goal of accurate and automatic segmentation. Recently, McInerney and Terzopoulos[37] proposed a new deformable model called topology adaptive snakes (T-snakes) to improve the topological inflexibility of standard snakes.[6] They introduced an affine cell image decomposition (ACID) reparameterization mechanism to the conventional snake model such that it is relatively insensitive to the initial placement and notably improves automation. The topological transformations of T-snakes were handled by introducing the entropy condition from the level set framework.[8] Although the algorithm performs well, the modifications are complicated and the number of experimental parameters has been increased. On the other hand, using the CFM algorithm is conceptually straightforward due to the evolution procedures described in Sec. 3.2. Moreover, there is no time interval assignment and only one parameter β setting in our approach. The setting of β is related to the maximum image gradient position as well as the gradient distribution of the ROI, as demonstrated in Sec. 4. We investigated the influence of β in the segmentation results of an MR image, which demonstrated similar conformity measures with a flexible range of β from −4.5 to −15.0 (see Fig. 20).

Table 2 The intrinsic properties of the CFM.

	Intrinsic Properties
1	In a free space, the propagating front is advanced in the normal direction corresponding to the direction of the electric field.
2	There is only a repelling interaction between fluid elements.
3	The propagating front is always on a grid during the evolution.
4	There is no time interval setting.
5	There is no curvature computation.
6	The CFM uses the FSP model implemented via the FFT algorithm to facilitate the electric potential computation through Poisson's equation.
7	An initial square contour in a free 2-D space retains the same geometry while it is evolving.

Most existing deformable model methods require the contour to be located at floating positions throughout the process such that intrinsic properties such as the curvature, the normal direction, and the internal forces can be more accurately calculated from the geometry of the contour. Compared to those deformable models, the advancement of the CFM is optimized in such a way that the actual positions of the contour are always on grids during the evolution. After the target is detected, the area and length of the ROI can be readily computed with subpixel precision. 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 precision as desired; this can reduce overall processing time. For example, the approximate processing times to obtain the results illustrated in Figs. 15, 18, and 19 were, respectively, 4, 9, and 24 s on a Pentium M 1.6 GHz machine with a Windows XP operating system. The intrinsic properties of the CFM and the essential characteristics for image segmentation are summarized in Tables 2 and 3. The disadvantages of this new deformable model (as summarized in Table 4) as well as the implementation difficulties are discussed next.

Table 3 The essential characteristics of the CFM for image segmentation.

	Essential Characteristics for Image Segmentation
1	The CFM interacts with the image data via the image gradient potential manner.
2	The topological change of the propagating front is automatically handled by tracing a 1-pixel-wide front on the 2-pixel-wide interface.
3	The charged fluid can flow through and around inner structural and background noise to segment the boundary of irregularly shaped objects.
4	The CFM can capture sharp corners and cusps.
5	The initial contour does not have to be placed at the center of the object of interest.
6	There is only one parameter β setting in most of the segmentation tasks.
7	The area and length of the ROI can be obtained with subpixel precision.

5.1 *Disadvantages of the CFM*

Objects with blurred boundaries or nonuniform boundary intensities have areas of weaker gradients along their boundaries. Since the CFM uses the image gradient technique as the image potential, it is possible for the CFM to leak through the weaker gradients of objects being segmented, as demonstrated in Fig. 20(b). One way to avoid leakage is to increase the weight (absolute value) of the image potential. This will not only increase the magnitude of the gradient, but also inevitably enhance the noise. This can sometimes generate unwanted noise barriers that confine fluid elements inside the ROI, leading to poor results [e.g., Fig. 20(d)]. The contour can also leak through broken image gradients, as shown in Fig. 23(c).

Suri[38] pointed out the weakness of methods that use edge-based techniques in the stopping criteria. They are noise sensitive and nonrobust when segmenting objects with blurred boundaries. He proposed a region-based level set method that improves edge-based approaches by introducing a fuzzy classifier.[38] Other methods using Bayesian classification have also been proposed to improve edge-based deformable models.[11] Another possible way to prevent leakage in the CFM is to introduce an attractive force among fluid elements such that the motion of the leaking fluid elements can be confined or stopped. All of these are interesting topics for future research.

The computation time of the CFM is problem-dependent due to the selection of the electrostatic equilibrium condition, which we have set to a constant value of $\gamma=7\%$. In our opinion, such a setting is adequate in most of the segmentation tasks in terms of the computational cost and accuracy. The average iteration number needed to complete the charge distribution procedure (Algorithm 2) is approximately 7, which retards the overall computational cost of our approach. Modified charge interpolation methods and more efficient electrostatic models could be investigated to improve the computational cost.

Table 4 The disadvantages of the CFM for image segmentation.

	Disadvantages for Image Segmentation
1	The CFM can leak through the weaker and broken gradients of objects being segmented.
2	The initial contours have to be placed entirely inside regions of interest.

5.2 *Implementation Difficulties*

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 simulation techniques described in Sec. 2 to develop the CFM embedded in a deformable model for front propagation. 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 1). 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. There is no velocity and acceleration computation using the CFM algorithm. Moreover, because each fluid element with variable charge is always advanced on grid points during the evolution, the number of fluid elements is automatically equal to the length of the contour (the number of pixels in the contour) using the contour tracing technique without extra heuristic rules to handle the particle number (see Sec. 3.3).

An important characteristic of electrostatic systems in equilibrium is that the electric field is stronger at locations where the curvature of the conducting surface is larger (see Table 1). Moreover, the magnitude of the electric field at the surface is proportional to the local charge density. A flat location with zero curvature will have relatively weaker charge, while a sharp point with a high degree of curvature will have a relatively larger charge. Suppose that there are two charges located on a flat surface, as shown in Fig. 24(a). Since the electrostatic forces are inversely proportional to the square of the distance, these two charges would tend to adjust their positions in such a way that the system energy is minimized. On a perfect sphere, the final distance between each adjacent charge would be equal.

However, on an irregularly shaped object, charges would tend to accumulate at positions of greater curvature. Consider the two charges on a concave surface with a sharper curvature shown in Fig. 24(b). These two charges repel

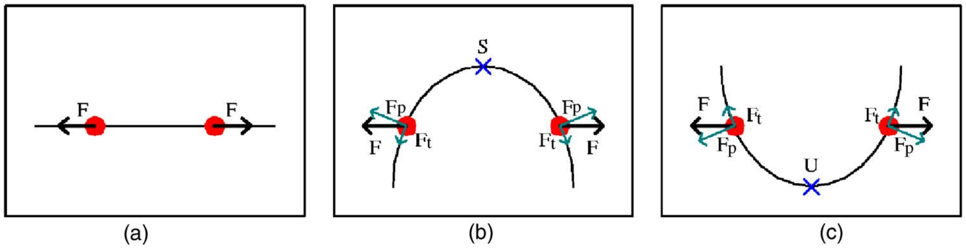

Fig. 24 Electric forces between two charges on different curved surfaces. (a) The electric force between two charges on a flat surface is parallel to the surface. (b) The electric force between two charges on a concave surface with high degree of curvature is primarily perpendicular to the surface. The two charges will eventually advance to the position *S*, where the charge density is locally higher. (c) The interaction of electric force between two charges on a convex surface is similar to (b), except that they are pushed away. The charge density at position *U* will be relatively weaker when the electrostatic equilibrium is achieved.

each other with a force along the line connecting their centers. Unlike the force in Fig. 24(a), which is primarily parallel to the surface, a majority of the repulsive force between the charges in Fig. 24(b) is perpendicular to the surface. It is the parallel components of the repelling forces to make charges move along the surface of a conductor. Since the component of the force parallel to the surface is considerably less, the two charges in Fig. 24(b) are pushed by other neighboring charges to the position *S*. On the other hand, two charges on a convex surface would be pushed away like the diagram shown in Fig. 24(c). Therefore, there are relatively weaker charges at the position *U*.

In image segmentations the propagating interface will be irregularly shaped. In a noisy region, the propagating interface is particularly rough, with many sharp points. As a consequence, the charges tend to accumulate at locations where the curvature is locally higher through the whole propagating interface. This leads to a nonuniform charge distribution. Such a nonuniform charge distribution along the interface of the charged fluid increases the complexity and difficulty of tracing the propagating front, which we addressed by tracing a 1-pixel-wide front from the 2-pixel-wide interface during curve evolution (see Secs. 3.2 and 3.3). Therefore, the propagating front can be rapidly and correctly reconstructed using a simple contour tracing technique.

5.3 *Summary*

We introduced the concept of using the simulation of a charged fluid to perform image segmentation. The algorithm was used to segment noisy and inhomogeneous objects with sharp corners and cusps in monochrome images. We illustrated a wide variety of applications of using the CFM in the segmentation of phantom, photographic, and medical images with the sole setting of parameter β, which we quantitatively evaluated by κ, the conformity measure.

This algorithm is limited to the segmentation of an object with a closed boundary, and it cannot simultaneously segment objects within that boundary. Another limitation is that the current algorithm must be initialized somewhere inside this boundary. It is not necessary to place the initial contour at the center of the object being segmented, but initialization near the center can save processing time. Lastly, the current algorithm does not allow splitting of the propagating front (although this could be implemented if needed). Our approach requires no curvature computation, and the area and length of the ROI can be obtained with subpixel precision. The advantages and properties of the CFM indicate that it is a promising image segmentation technique in a wide variety of image processing applications. The acceleration and automation of the CFM algorithm is currently under investigation. Additional work is required to investigate the segmentation of blurred images by incorporating image regional data or attractive forces, as described in Sec. 5.1.

Acknowledgments

This research was supported in part by National Institutes of Health (NIH) Grants NCRR P41 RR13642 and NIMH/NINDS P20 MH65166 to the Laboratory of Neuro Imaging. Additional support was provided by the NIH through the NIH Roadmap for Medical Research, Grant No. U54 RR021813, entitled Center for Computational Biology (CCB).

References

1. W. K. Pratt, *Digital Image Processing*, 2nd ed., Wiley, New York (1991).
2. M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*, PWS Publishing (1999).
3. I. N. Bankman, *Handbook of Medical Imaging*, Academic Press, San Diego (2000).
4. N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recogn.* **26**(09), 1277–1294 (1993).
5. J. A. Sethian, *Level Set Methods and Fast Marching Methods*, 2 ed., Cambridge Univ. Press, Cambridge (1999).
6. M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: active contour models," *Int. J. Comput. Vis.* **1**(04), 321–331 (1988).
7. L. D. Cohen, "On active contour models and balloons," *CVGIP: Image Understand.* **53**(2), 211–218 (1991).
8. S. Osher and J. A. Sethian, "Fronts propagating with curvature dependent speed: algorithms based on hamiltons-jacobi formulations," *J. Comput. Phys.* **79**, 12–49 (1988).
9. V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *Int. J. Comput. Vis.* **22**(1), 61–79 (1997).
10. X. Han, C. Xu, and J. L. Prince, "A topology preserving level set method for geometric deformable models," *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(6), 755–768 (2003).
11. J. S. Suri, K. Liu, S. Singh, S. N. Laxminarayan, X. Zeng, and L. Reden, "Shape recovery algorithms using level sets in 2-D/3-D medical imagery: a state-of-the-art review," *IEEE Trans. Inf. Technol. Biomed.* **6**(1), 8–28 (2002).
12. R. Malladi and J. A. Sethian, "An O(*N* LOG *N*) algorithm for shape modeling," *Proc. Natl. Acad. Sci. U.S.A.* **93**, 9389–9392 (1996).
13. K. Siddiqi, Y. B. Lauziere, A. Tannenbaum, and S. W. Zucker, "Area and length minimizing flows for shape segmentation," *IEEE Trans. Image Process.* **7**(3), 433–443 (1998).
14. C. Gout, C. L. Guyader, and L. Vese, "Segmentation under geometri-

cal conditions using geodesic active contours and interpolation using level set methods," *Numer. Algorithms* **39**(1–3), 155–173 (2005).
15. C. L. Guyader, D. Apprato, and C. Gout, "Using a level set approach for image segmentation under interpolation conditions," *Numer. Algorithms* **39**(1–3), 221–235 (2005).
16. T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.* **10**(2), 266–277 (2001).
17. A. Tsai, J. A. Yezzi, and A. S. Willsky, "Curve evolution implementation of the mumford-shah functional for image segmentation, denoising, interpolation, and magnification," *IEEE Trans. Image Process.* **10**(8), 1169–1186 (2001).
18. D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Commun. Pure Appl. Math.* **42**, 577–685 (1989).
19. L. A. Vese and T. F. Chan, "A multiphase level set framework for image segmentation using the mumford and shah model," *Int. J. Comput. Vis.* **50**(3), 271–293 (2002).
20. W. T. Reeves, "Particle systems—a technique for modeling a class of fuzzy objects," *ACM Trans. Graphics* **2**(2), 91–108 (1983).
21. R. W. Hockney and J. W. Eastwood, *Computer Simulation Using Particles*, Adam Hilger, UK (1988).
22. R. Szeliski, D. Tonnesen, and D. Terzopoulos, "Modeling surfaces of arbitrary topology with dynamic particles," in *IEEE Proc. CVPR*, pp. 82–87 (1993).
23. D. Stahl, N. Ezquerra, and G. Turk, "Bag-of-particles as a deformable model," in *IEEE Proc. TCVG*, pp. 141–150 (2002).
24. A. C. Jalba, M. H. Wilkinson, and J. B. Roerdink, "CPM: a deformable model for shape recovery and segmentation based on charged particles," *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(10), 1320–1335 (2004).
25. J. M. Dawson, "Particle simulation of plasmas," *Rev. Mod. Phys.* **55**(2), 403–447 (1983).
26. V. K. Decyk and J. M. Dawson, "Computer model for bounded plasma," *J. Comput. Phys.* **30**, 407–427 (1979).
27. A. B. Langdon and C. K. Birdsall, "Theory of plasma simulation using finite-size particles," *Phys. Fluids* **13**(8), 2115–2122 (1970).
28. S. C. Neu, "Computer simulation of a non-neutral plasma column," PhD thesis, UCLA (1997).
29. S. V. Marshall and G. G. Skitek, *Electromagnetic Concepts and Applications*, Prentice-Hall, Englewood Cliffs, NJ (1987).
30. D. K. Cheng, *Field and Wave Electromagnetics*, 2 ed., Addison-Wesley, Reading, MA (1989).
31. W. L. Kruer, J. M. Dawson, and B. Rosen, "The dipole expansion method for plasma simulation," *J. Comput. Phys.* **13**, 114–129 (1973).
32. T. Pavlidis, *Algorithms for Graphics and Image Processing*, Computer Science Press, (1982).
33. A. J. Tabatabai and O. R. Mitchell, "Edge location to subpixel values in digital imagery," *IEEE Trans. Pattern Anal. Mach. Intell.* **6**(2), 188–201 (1984).
34. M. Kisworo, S. Venkatesh, and G. West, "Modeling edges at subpixel accuracy using the local energy approach," *IEEE Trans. Pattern Anal. Mach. Intell.* **16**(4), 405–410 (1994).
35. K. Jensen and D. Anastassiou, "Subpixel edge localization and the interpolation of still images," *IEEE Trans. Image Process.* **4**(3), 285–295 (1995).
36. D. J. Valentino, K. Ma, C. C. Wei, and S. C. Neu, "A portable framework for medical imaging display: the jViewbox," in *Proc. Annual Meeting Radiological Sci. North Am.*, p. 891 (2002).
37. T. McInerney and D. Terzopoulos, "T-snakes: topology adaptive snakes," *Med. Image Anal* **4**(2), 73–91 (2000).
38. J. S. Suri, "Two-dimensional fast magnetic resonance brain segmentation," *IEEE Eng. Med. Biol. Mag.* **20**(4), 84–95 (2001).

Herng-Hua Chang received his BS degree in mechanical engineering in 1996 from National Taiwan University and his MS degree in biomedical engineering in 1998 from National Yang-Ming University, Taiwan. He is currently a PhD candidate in the Biomedical Engineering Interdepartmental Program and a graduate student researcher with the Laboratory of Neuro Imaging (LONI) at the University of California, Los Angeles, where he is researching and developing numerical techniques for the segmentation and analysis of biomedical images.

Daniel J. Valentino received his BS degree in chemistry and his PhD degree in biomedical physics. He is currently an associate professor of radiology, biomedical physics and biomedical engineering at the University of California, Los Angeles (UCLA) and a member of the Laboratory of Neuro Imaging (LONI) and the Division of Interventional Neuro Radiology (DINR). His current research interests include x-ray imaging physics and digital image detectors, brain image segmentation and modeling, and numerical simulation of cerebrovascular hemodynamics and biomechanics. He is a member of the American College of Radiology (ACR) and the SPIE.