Deformable Models with Arbitrary Initializations

Huaizhong Zhang, *Member, IEEE,* and Xianghua Xie, *Member, IEEE,*

***Abstract*—Existing image gradient based active contour approaches are largely based on fitting the evolving contours according to edge strength, i.e. gradient magnitude. For making full use of the directional information of the image gradient vectors, we propose a novel gradient convolution field (GCF), which can be conveniently obtained by convolving the image gradient vector field with an inverse distance kernel. The divergence of this vector flow is effectively a region indication function and can be as an evolving velocity in active contour models. Based on this region-like geometric property, a novel deformable model is proposed. Subsequently, we show its explicit relationship to the most recent MAC model and present an efficient vector field nonlinear diffusion to refine the GCF field efficiently, which performs much better than the scalar diffusion approach in the MAC model. Then, a global minimizer of the proposed deformable model is formulated using convex relaxation. By combining with the Gaussian filtering for reinitialization free, our global solution scheme can perform comparable to region based global minimizers. Due to GCF being automatically adaptive in higher dimensions, the proposed deformable model is naturally extended to 3D and both of 2D and 3D versions are implemented using the level set method. In addition, we implement an intrinsically regularized level set scheme, which allows the level set to evolve without re-initialization and concurrently has the ability to develop new contours away from existing ones. We reveal that this regularized level set scheme is functionally analogous to the proposed global minimizer. Superior performance has been achieved compared to the state-of-the-art edge based approaches.**

***Index Terms*—Deformable models, MAC, nonlinear diffusion, global minimization, level set method.**

I. Introduction

CONVENTIONAL image gradient based active contour models, such as [1], [2], rely on contour fittings using local intensity discontinuity and generally have difficulties in dealing with boundary concavities, weak edges and image noise. In particular, this local optimization approach is highly initialization dependent and prone to local minima [3]. There has been a multitude of works on the improvement of these gradient based approaches. Most of them are mainly based on edge strength or magnitude of image gradient. The directional information of image gradient has been explored only to a limited extent. On the other hand, gradient direction or orientation contains important information in describing object boundaries. For example, its spatial coherence can effectively suggest whether or not an edge is present. Strong gradients with large directional variation are more likely caused by noise than object boundaries. Image gradient vectors with small magnitude but strong spatial conformity is in fact a good indication of weak edges, which can be crucial to preserve in object segmentation using active contours In [4], Xu and Prince proposed to use a vector diffusion equation that diffuses the gradient of an edge map to increase the contour capture range and also to reduce noise interference. Kimmel in [5] proposed a region and edge combined approach, where an alignment measure is used to optimize the orientation of the contour with respect to the image gradients. In [6], Vasilevskiy and Siddiqi proposed an active contour model specifically designed to segment elongated, thin structures with consistent contrast, such as blood vessels in angiograms, by incorporating the direction of an appropriate vector field in defining the flux maximization geometric flow. However, various adversities such as convergence and locality issues prevent these approaches from dealing with the mentioned problems efficiently [7], [8].

Recently, there have been a few research works on physics-inspired deformable modeling. In [9], a charged-particle model (CPM) based on electrostatics was applied to localize object boundaries by assigning opposite charges to edge pixels and free particles so that the particles are pulled towards edges while repelling each other. Although this approach does not suffer from those convergence issues, the fact that particles on weak edges may be attracted to nearby strong edges often causes broken contours to be formed. In [10], the authors adapted the CPM model into an active contour model and showed subsequent improvements. However, the dominant force is static and its dynamic behavior due to repulsion force can be difficult to predict. Similar work based on electrostatics has been reported in [11].

In [8], instead of assigning fixed charges, the authors allow the charges flow through the edges and then leads to generating a magnetic field. The active contour, carrying similar flow of charges, is attracted towards the edges under this magnetic influence. The direction of the currents, flows of charges, running through object boundary is estimated based on edge orientation, which is obtained by a 90° rotation in the image plane of the normalized image gradient vectors $(\hat{I}_x, \hat{I}_y)$, where I denotes an image. The image plane is considered as a 2D plane in a 3D space whose origin coincides with the origin of the image coordinates. Additionally, the third dimension of this 3D space is considered perpendicular to the image plane. Thus, the object boundary current direction, $\mathbf{O}(\mathbf{x})$, is estimated as:

$$\mathbf{O}(\mathbf{x}) = (-1)^{\lambda}(-\hat{I}_y(\mathbf{x}), \hat{I}_x(\mathbf{x}), 0), \tag{1}$$

where $\mathbf{x}$ denotes a point in the image domain, $\lambda = 1$ gives an anti-clockwise rotation in the image coordinates, and $\lambda = 2$ provides a clockwise rotation. Since the active contour is embedded in a signed distance function while using the level set representation, the direction of current for the contour, denoted as $\boldsymbol{\Upsilon}$, is similarly obtained by rotating the gradient vector $\nabla\Phi$ of the level set function Φ. Let $f(\mathbf{x})$ be the magnitude of image gradient or edge strength; usually, the

H. Zhang and X. Xie are with the Department of Computer Science Department, Swansea University,Swansea SA2 8PP, United Kingdom e-mail:

magnitude of boundary current is considered proportional to edge strength. The magnetic field $\mathbf{B}(\mathbf{x})$ generated by gradient vectors at each pixel position is computed as:

$$\mathbf{B}(\mathbf{x}) = \frac{\mu_0}{4\pi} \sum_{\mathbf{s} \neq \mathbf{x}} f(\mathbf{s}) \mathbf{O}(\mathbf{s}) \times \frac{\hat{\mathbf{R}}_{\mathbf{xs}}}{R^2_{\mathbf{xs}}}, \tag{2}$$

where μ_0 is the permeability constant, $\mathbf{s}$ denotes an edge pixel position, $\hat{\mathbf{R}}_{\mathbf{xs}}$ denotes a 3D unit vector from $\mathbf{x}$ to $\mathbf{s}$ in the image plane, and $R_{\mathbf{xs}}$ is the distance between them. The active contour is assigned with unit magnitude of electric current. The force imposed on it is derived as:

$$\mathbf{F}_m(\mathbf{x}) \propto \boldsymbol{\Upsilon}(\mathbf{x}) \times \mathbf{B}(\mathbf{x}). \tag{3}$$

The magnetostatic active contour (MAC) model is then formulated as:

$$C_t = \alpha g(\mathbf{x}) \kappa \hat{\mathbf{N}} + (1 - \alpha)(\mathbf{F}_m(\mathbf{x}) \cdot \hat{\mathbf{N}}) \hat{\mathbf{N}}, \tag{4}$$

where $g = 1/(1 + f)$, κ denotes the curvature, and $\hat{\mathbf{N}}$ is inward unit normal. Note, $\mathbf{F}_m$ lies in the image domain and its third element equals zero, which can be ignored. The MAC model showed significant improvements on convergence issues, e.g. reaching deep concavities, and in handling weak edges and broken boundaries [8]. Derived from a different physical phenomenon, the long range interaction force based on the elastic interaction between line defects in solids in the model proposed by Xiang *et al.* [12] is similar to the magnetic force used in the MAC model.

However, the presence of image noise will inevitably disturb the gradient vectors and cause irregularities in the estimated edge orientation $\mathbf{O}$, which will in turn affect the magnetic field computation (see (2)). The authors in [8] proposed a scalar field diffusion to refine the magnetic field. However, this approach is problematic since there is a great risk of positive and negative magnetic flux canceling out each other which hampers the desired diffusion from strong edges to propagate into noise dominant regions. In addition, the denoising strategy proposed in [12] requires either multiple trial segmentations or independent contour shrinking.

In this paper, a novel deformable model is proposed by making use of information from both magnitude and orientation of image gradient, which can deal with the difficulties that the traditional methods are hampered. The main contributions of this paper are summarized as:

- We propose to use a novel gradient convolution field to help extracting image feature, which can be considered as a region indication function and has explicit relationship to the MAC model.
- A deformable model is derived using the divergence of the proposed vector field as the static external force, which can naturally extended to higher dimensional spaces. The 3D deformable model is numerically implemented by applying the level set method.
- A tensor diffusion scheme is proposed to diffuse the vector field in order to improve the performance of the deformable model, which The edges are better preserved and image noise can be more efficiently removed, even though they may appear far from strong edges.
- An intrinsic level set regularization scheme is proposed for the derived deformable model, which is motivated from smoothing the transition between different level sets, so that contours can be arbitrary initialized and still be able to localize objects, which is functionally analogous to the proposed global minimizer.
- The convex relaxation strategy is used to formulate a new energy functional in terms of the derived deformable model and thus the global optimization scheme is achieved. The numerical implementation is realized using the level set method while the isotropic Gaussian filtering is employed to maintain the level set function as an signed distance function for avoiding the reinitialization issue.

In experiments, various synthetic and real world images are used to examine and validate the effectiveness of the proposed method both in 2D and 3D, where two state-of-the-art global minimizers are employed to perform the comparison.

II. Proposed Method

Inspired by physics-based methods, e.g. [9], [13], in Section II-A we derive a gradient convolution field by convolving the image gradient field with an inverse distance kernel function. Unlike those approaches, our convolution is carried out on the spatial components of the image gradient field, instead of its magnitude alone, so that both the magnitude and directional information are utilized. Based on the proposed vector field, a deformable model is derived, which is easily extended to higher dimensions and is intrinsically associated with the MAC model. In Section II-B, for dealing with noise interference more efficiently, a nonlinear diffusion method is proposed to refine the gradient convolution field by considering the directional role in diffusion process. This is fundamentally different from [8] where post-processing is adopted. For improving the performance of the derived deformable model, a regularized level set method and its implementation are presented in Section II-C, which allows perturbations away from existing contours so as to create new contours. Furthermore, a global minimizer of the derived deformable model is presented in Section II-D using convex relaxation so that this new global formulation can successfully eliminate the locality issue and produce the initialization-independent result. For avoiding the re-initialization, an isotropic Gaussian filtering is used to regularize the level set function in the numerical implementation of the derived global solution scheme. In addition, functionally, it is revealed that the regularized level set method in Section II-C is analogous to the proposed global minimization scheme. Finally, in Section II-E, we extend the derived deformable model to high dimensional spaces and provide the level-set solution scheme for the 3D case.

A. Image gradient convolution field and its derived deformable model

In [13], Li and Acton used a vector field convolution of the image edge map as an external force to attract the active contour towards image boundaries. The vector field kernel $\mathbf{k}(\mathbf{x})$ consists of radial symmetric vectors pointing towards the center of the kernel, and is given as $\mathbf{k}(\mathbf{x}) = m(r)\,\hat{\mathbf{r}}$, where $m(r)$ is the magnitude, $r = |\mathbf{x}|$ is the distance from the kernel origin, and $\hat{\mathbf{r}} = -\mathbf{x}/r$ is the unit vector pointing to the origin. The force derived from this vector field convolution at $\mathbf{x}$ can then be written as:

$$\mathbf{F}_v(\mathbf{x}) = f(\mathbf{x}) * \mathbf{k}(\mathbf{x}). \quad (5)$$

The magnitude $m(\mathbf{x})$ of the vector field kernel should be a decreasing positive function of distance from the origin, and can be defined as $m(r) = (r+\varepsilon)^{-\zeta}$, where ε is a small constant to avoid division by zero and ζ controls the speed of spatial degradation. When choosing $\varepsilon = 0$ and $\zeta = 2$, the force between edge pixels and active contour conforms to Newton's law of universal gravitation in physics [13], and (5) can be in effect computed as:

$$\mathbf{F}_v(\mathbf{x}) = \sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s}) \frac{\hat{\mathbf{R}}_{\mathbf{xs}}}{R^2_{\mathbf{xs}}}. \quad (6)$$

This external force is in fact equivalent to the fundamental force interaction used in [9] where the electrostatic theory is applied.

However, none of these methods takes into account the edge orientation or gradient direction in deriving the external force fields. To this end, we propose a similar kernel convolution process but it takes place on the image gradient vector field, instead of the edge map which only contains its magnitude information. Let $\nabla_i I = f\hat{I}_x$ and $\nabla_j I = f\hat{I}_y$ denote the two components of the image gradient ∇I in the image coordinates (i, j), respectively, i.e. $\nabla I = (\nabla_i I, \nabla_j I)^T$. We carry out the convolution computation on both components. Note, since we are not going to treat the convolution results directly as force field to evolve the active contour as in [9], [13], a scalar kernel function is used. It simply takes the form of $k(\mathbf{x}) = m(\mathbf{x})$. Moreover, we choose the magnitude function m as an inverse of distance, i.e. $m(r) = 1/r^{\zeta}$ with $\zeta = 1$, instead of inverse of squared distance. However, we show later that since we further compute the spatial derivatives of the convolution results, the spatial decay is actually raised to power of two, i.e. $\zeta = 2$. Thus, the result of this convolution process can be expressed as:

$$\begin{cases} E_i(\mathbf{x}) = \nabla_i I * k(\mathbf{x}) = \sum_{\mathbf{s}\neq\mathbf{x}} \frac{\nabla_i I(\mathbf{s})}{R_{\mathbf{xs}}} = \sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s}) \frac{\hat{I}_x(\mathbf{s})}{R_{\mathbf{xs}}}, \\ E_j(\mathbf{x}) = \nabla_j I * k(\mathbf{x}) = \sum_{\mathbf{s}\neq\mathbf{x}} \frac{\nabla_j I(\mathbf{s})}{R_{\mathbf{xs}}} = \sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s}) \frac{\hat{I}_y(\mathbf{s})}{R_{\mathbf{xs}}}, \end{cases} \quad (7)$$

where $\mathbf{E} = (E_i, E_j)$ denotes the resulting gradient convolution field (named as GCF). Due to the smoothing effect when applying the kernel function, the original image gradient vectors have extended their influence from immediate vicinity of edge pixels to much larger neighborhood. In fact, the computation in (7) is across the whole image domain. It takes similar form to [13], but with more extensive spatial smoothing. We may directly use this vector field GCF as an external force to evolve the active contours. However, this will lead us to those similar convergence problems that GVF/GGVF [4], vector kernel convolution [13] and many other vector field based methods have suffered from as mentioned earlier. Instead, we show its fundamental relationship to the MAC model [8] and formulate a new deformable model with this vector field. We also demonstrate later how we can further refine this gradient convolution field in order to achieve more robust segmentation.

Using the relationship between the magnetic field $\mathbf{B}$ and the GCF $\mathbf{E}(\mathbf{x})$ (see the inference in Appendix (A)), the magnetic field shown in (2) can be indirectly calculated:

$$\mathbf{B}(\mathbf{x}) = \frac{\mu_0}{4\pi}(0, 0, B), \quad B = \nabla \cdot \mathbf{E}(\mathbf{x}) = \nabla \cdot (E_i, E_j) \quad (8)$$

where B is the divergence of GCF and actually is the third component of the magnetic field $\mathbf{B}$ and the external force in the MAC model (4). In addition, the B value can be geometrically interpreted as the change rate of the GCF along the normal direction. Thus, the GCF $\mathbf{E} = (E_i, E_j)$ presents its explicit relationship to the magnetic field in the MAC model by imposing the divergence operator on it as shown in (8).

Accordingly, the divergence of GCF B is generated from geometric interactions between the object boundary and the active contour, which can be as the evolution velocity (external force) for attracting the curve to the boundary. Thus, the shape of the contour can be formed by the normal component of this velocity $(-B)$ and the evolution equation can be described as:

$$C_t = -B(\mathbf{x})\hat{\mathbf{N}}. \quad (9)$$

where $\hat{\mathbf{N}}$ is the unit inward normal.

For contour smoothing, the curvature flow is added to (9). Thus, a novel deformable model, just based on the proposed GCF without any assumption, can be formulated as follows.

$$C_t = \alpha g(\mathbf{x})\kappa\hat{\mathbf{N}} - (1-\alpha)B(\mathbf{x})\hat{\mathbf{N}}. \quad (10)$$

Furthermore, due to the relationship between the evolution velocity B and the GCF, (8) allows us to indirectly refine the B in (10) by diffusing each component of $\mathbf{E} = (E_i, E_j)$, instead of postprocessing B as proposed in [8] which, as we show in the next section, is problematic. As expected, in the subsection II-E we can naturally extend the proposed vector field GCF to 3-D and (10) is then readily applied to the 3D case.

B. Gradient convolution field diffusion

Here, we present a nonlinear diffusion method to refine the vector field GCF in order to enhance object boundaries and reduce noise interference. We first examine the role of magnetic field $\mathbf{B}$ in the MAC framework and the limitation of magnetic field post-processing in [8], which then leads to the proposed diffusion scheme in terms of GCF.

Let us incorporate (3) into (4) and re-write (4) as:

$$C_t = \alpha g(\mathbf{x})\kappa\hat{\mathbf{N}} + (1-\alpha)\left(\mathbf{B}(\mathbf{x}) \cdot \mathbf{M}\right)\hat{\mathbf{N}}, \quad (11)$$

where $\mathbf{M} = (\hat{\mathbf{N}}, 0) \times \boldsymbol{\Upsilon}$ is a binormal unit vector (this forms the image plane), which is computed as the cross product of level set normal and its tangent vector, and it is perpendicular to the image plane. The external force in the second term is thus a projection of the magnetic field $\mathbf{B}$ onto the image plane, i.e. B in (8). A positive projection will force the contour to expand and a negative projection will shrink the contour, which acts in a similar way as what a region indication function does in a region based approach, however, this is derived from the edge based assumption.

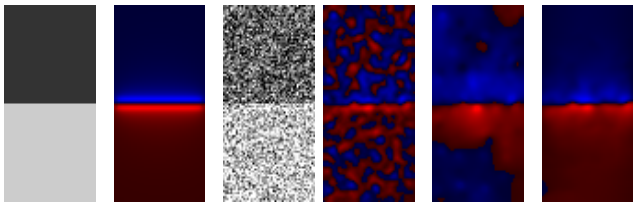
This fundamentally explains why the MAC model does not suffer from those convergence issues related to other vector field based methods, such as GVF [4]. It is also evidently clear that $\mathbf{B}$ plays the dominant role in the MAC segmentation. From (7) and (8), we can see that the success in finding meaningful $\mathbf{B}$ largely depends on the conformity of image gradients, not only in their magnitude but also their orientation. The presence of image noise will inevitably disrupt

the gradient orientation, as well as magnitude, which creates fluctuations in the magnetic field $\mathbf{B}$. It is also worth noting that regions away from strong edge pixels suffer more from image noise due to spatial decay of image gradient interactions.

As derived in in (8), $\mathbf{B} = \frac{\mu_0}{4\pi}(0, 0, B)$. Thus, the authors in [8] argued diffusing the scalar field B in the image domain was sufficient:

$$\frac{\partial}{\partial t}\mathcal{B}(\mathbf{x}) = p(B(\mathbf{x}))\nabla\cdot\nabla\mathcal{B}(\mathbf{x}) - q(B(\mathbf{x}))(\mathcal{B}(\mathbf{x}) - B(\mathbf{x})), \quad (12)$$

where the initial state $\mathcal{B}(0, \mathbf{x})$ is set to $B(\mathbf{x})$, and functions p and q are given as: $p(B(\mathbf{x})) = e^{-\frac{|B(\mathbf{x})|f(\mathbf{x})}{K}}$, $q(B(\mathbf{x})) = 1 - p(B(\mathbf{x}))$, where K is the diffusion parameter. Weighting the flux density magnitude with f in the diffusion term, p, attempts to ensure there is as little diffusion as possible at the object boundaries, while homogeneous areas will generally have small flux density due to a lack of support from edges, resulting in substantial diffusion.



(a) (b) (c) (d) (e) (f)

Fig. 1: (a): A test image contains a horizontal edge; (b): Its magnetic field visualization; Brightness corresponds to magnitude. Blue color indicates positive flux, and red indicates negative. (c): A noisy image; (d): Initial magnetic field; (e): Diffusion result using [8]; (f): magnetic field computed from our diffusion result.

However, this diffusion has an unintended effect of canceling out positive and negative magnetic field on strong edges, which are supposed to be preserved. Let us consider a simple horizontal edge as shown in Fig. 1a. From Fig. 1b, we can see that the center of the edge has magnetic field close to zero. The magnetic field is perpendicular to the image domain. Image gradient vectors on opposite side of an edge produce opposing magnetic field. This means that the preserving term q is ineffective as cancelation of positive and negative magnetic flux inevitably occurs around the edge. It creates a dilemma when strong diffusion is needed to overcome noise interference further away from edges. A larger diffusion will introduce more diffusion at strong edges, causing them diminishing at a faster speed due to cancelation which restrains their ability to influence regions further away from edges. A smaller diffusion, however, may not meet the desired diffusion effect. Fig. 1c-1e gives an example when noise interference is strong. The diffusion in [8] is no longer effective.

For our proposed GCF $\mathbf{E}$, as described in (7), the direction of the vectors $\mathbf{E}$ is along image gradient direction and the divergence of $\mathbf{E}$ correlates to the edge strength. In addition, (8) presents a clear quantitative relationship between the external force B in the deformable model (10) and the GCF $\mathbf{E}$. Thus, instead of post-processing B, we propose a nonlinear diffusion of $\mathbf{E}$ before computing its divergence for obtaining the external force B. Nonlinear diffusion, eg. [14], has been found useful in a range of applications, such as image denoising [15] and vector flow visualization [16]. Consider a class of nonlinear parabolic differential equation of the following kind:

$$\frac{\partial}{\partial t}u - \nabla\cdot(\mathbf{D}(\nabla u)\nabla u) = \mathcal{F}(u_0), \quad (13)$$

where $u(t, \mathbf{x})$ is the diffused version, t can be considered as the "scale parameter", $\mathbf{D} = \begin{pmatrix} a & b \\ b & c \end{pmatrix}$ is the diffusion tensor (a positive definite symmetric matrix), $\mathcal{F}$ can be considered as a penalty function which forces the diffusion result to conform to certain criteria, and $u_0(\mathbf{x}) = u(0, \mathbf{x})$ denotes the initial state. In our case, $\mathbf{E} = (E_i, E_j)$ is a vector filed. Thus, the nonlinear diffusion takes the following coupled form:

$$\begin{cases} \frac{\partial}{\partial t}\mathcal{E}_i - \nabla\cdot(\mathbf{D}(\nabla\mathcal{E}_i, \nabla\mathcal{E}_j)\nabla\mathcal{E}_i) = \mathcal{F}(E_i), \\ \frac{\partial}{\partial t}\mathcal{E}_j - \nabla\cdot(\mathbf{D}(\nabla\mathcal{E}_i, \nabla\mathcal{E}_j)\nabla\mathcal{E}_j) = \mathcal{F}(E_j), \end{cases} \quad (14)$$

where $\mathcal{E}_i(0, \mathbf{x}) = E_i(\mathbf{x})$, $\mathcal{E}_j(0, \mathbf{x}) = E_j(\mathbf{x})$. The diffusion tensor can be decomposed into two orthogonal components, one of which is parallel to the local vector field and the other is perpendicular to the local vector. The orientation of a vector in the $\mathbf{E}$ field can be denoted as $(-\sin\theta, \cos\theta)$ and its orthogonal unit vector can be obtained as $(\cos\theta, \sin\theta)$. Obviously, the mapping $\mathbf{R}$ from image coordinates to this orthogonal coordinates is computed as $\mathbf{R} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}$. Thus, (14) can be re-written as:

$$\frac{\partial}{\partial t}\boldsymbol{\mathcal{E}} - \nabla\cdot\left(\mathbf{R}^T\begin{pmatrix} \omega & 0 \\ 0 & \gamma \end{pmatrix}\mathbf{R}\nabla\boldsymbol{\mathcal{E}}\right) = \mathcal{F}(\mathbf{E}), \quad (15)$$

where ω is the diffusion function in the direction of the vector field $\mathbf{E}$ and γ denotes the diffusion function orthogonal to the field. Note the divergence and gradient operations are applied to each spatial component of $\boldsymbol{\mathcal{E}}$ separately.

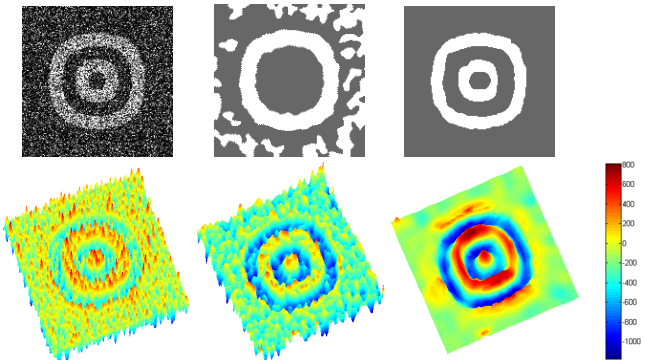


Fig. 2: In the first row, left: the noisy image (Im); middle: result using the method in [8]; right: result using the proposed method. In the second row, left: the magnetic field directly computed from Im; middle: diffused magnetic field using Laplacian diffusion in [8]; right: diffused magnetic field using proposed diffusion method.

A linear diffusion can be applied in the direction perpendicular to $\mathbf{E}$ that is along the edge direction. However, considering the fact that we aim to have larger diffusion where field vectors have smaller magnitude and preserve large vectors that are spatially consistent, we select the weighting function $\omega(\mathcal{E}(\mathbf{x})) = e^{-\frac{|\mathcal{E}(\mathbf{x})|^3}{K'}}$, where K' is the parameter controlling the amount of diffusion and $|\mathcal{E}| \in [0, 1]$. However, the magnitude of field vectors is raised to third order, since the vector field spatially varies over several orders of magnitude. Additionally, image noise introduces relatively larger

(often isolated) $\mathbf{E}$ vectors. This nonlinear transformation thus introduces more effective diffusion.

The diffusion along local vector field (gradient direction) plays a critical role in propagating the field vectors from strong edges to regions further away from them, which may be dominated by image noise. Similar diffusion function can be adopted, e.g. let $\gamma = \omega$. Alternatively, if we wish to increase the diffusion in this gradient direction, we can first compute the magnitude of the directional gradient, $|\nabla(|\boldsymbol{\mathcal{E}}|) \cdot \hat{v}_{\perp}|$, as a measure of degradation along this gradient direction, $\hat{v}_{\perp}$, and then similarly define the diffusion function γ as:

$$\gamma = e^{|\nabla(|\boldsymbol{\mathcal{E}}|)\cdot\hat{v}_{\perp}|}. \tag{16}$$

The conformity function $\mathcal{F}$ on the right hand side of (15) can be defined as $\mathcal{F}(\mathbf{E}(\mathbf{x})) = \mathbf{E}(\mathbf{x}) - \boldsymbol{\mathcal{E}}(\mathbf{x})$ so that the solution to (15) has a nontrivial steady state and it eliminates the problem of choosing a stopping time [17]. Moreover, we can add a weight to this term so that for regions that require substantial diffusion, i.e. where ω is large, there is less constraint on conformity. Thus, $\mathcal{F}$ can be defined as:

$$\mathcal{F}(\mathbf{E}(\mathbf{x})) = (1 - \omega(\mathbf{E}(\mathbf{x})))(\mathbf{E}(\mathbf{x}) - \boldsymbol{\mathcal{E}}(\mathbf{x})). \tag{17}$$

The divergence of this nonlinear diffused vector field $\mathbf{E}$, i.e. the generated external force, provides a better attraction force of object boundary to the curve, particularly in the presence of image noise and/or textures. Fig. 1 shows a comparison of the proposed diffusion against that in [8] on a noise corrupted image. In Fig. 2, the original magnetic field, the diffused magnetic fields using Laplacian and the tensor diffusion are visualized in the second row respectively. The first row shows the noisy image, the result using the method in [8] and the result using the proposed method in this paper.

C. Level set implementation with a regularized scheme

Note the first term in (10) is effectively a weighted length minimization flow and the second term can be viewed as a weighted area minimization flow. Thus, the contour evolution can be realized by minimizing the following energy functional:

$$\mathfrak{E} = \alpha \int_{\Omega} g(\mathbf{x})|\nabla\mathcal{H}(\Phi)|d\mathbf{x} + (1-\alpha)\int_{\Omega} \tilde{B}(\mathbf{x})\mathcal{H}(\Phi)d\mathbf{x}, \tag{18}$$

where $\tilde{B}$ is the divergence of the diffused GCF ($\tilde{B} = \nabla \cdot \boldsymbol{\mathcal{E}}$ and $\boldsymbol{\mathcal{E}}$ is found by solving (15)), and $\mathcal{H}(.)$ is the Heaviside function. Minimization of this energy functional is possible via gradient descent using variational calculus:

$$\frac{\partial}{\partial t}\Phi = \alpha g \nabla \cdot \left(\frac{\nabla\Phi}{|\nabla\Phi|}\right)\delta(\Phi) - (1-\alpha)\tilde{B}\delta(\Phi), \tag{19}$$

where $\delta(.)$ is the Dirac delta function. As in [18], we choose $\delta_{\epsilon}(z) = \epsilon/(\pi(\epsilon^2 + z^2))$, with $\epsilon = 1$ so that its discretization has a support larger than zero which permits the perturbation away from zero level set.

However, without conventional level set periodic re-initialization level sets close to zero level update much faster than others due to the use of the delta function. This results in steep gradient around zero level set, which causes irregularities, and can slow down the contour evolution as more levels are pushed away from zero level set. Here, we impose intrinsic regularization by explicitly smoothing the level set surface using anisotropic diffusion. Consider the following Perona-Malik [15] type diffusion equation:

$$\frac{\partial}{\partial t}\Phi = \nabla \cdot (c(\Phi, t)\nabla\Phi), \tag{20}$$

where $c(.)$ denotes a diffusion function. Conventionally, level set function is initialized as a signed distance field, which has the characteristic of unit gradient magnitude [19], i.e. $|\nabla\Phi| = 1$. Since we are only interested in reducing steep slopes and flat level set surfaces away from zero level are encouraged in order to develop new contours, we hence follow [20] and define the diffusion function as $c = \mathcal{H}(|\nabla\Phi| - 1)$. Thus, the level set diffusion can be formulated as:

$$\frac{\partial}{\partial t}\Phi = \nabla \cdot (\mathcal{H}(|\nabla\Phi| - 1)\nabla\Phi)\,. \tag{21}$$

The diffusion is largely inactive when $|\nabla\Phi| <= 1$ and actively takes place when $|\nabla\Phi| > 1$. The regularized Heaviside function $\mathcal{H}_{\epsilon} = \frac{1}{2}(1 + \frac{2}{\pi}\tan^{-1}(\frac{z}{\epsilon}))$, can be used to ensure a smooth transition between different level sets. It is different from [19] where level set is forced to conform to unit gradient magnitude across the domain, which will prevent it from developing new components. This diffusion predominantly takes place around zero level set, where steep surface most likely appears. It effectively minimize steep surfaces and also smoothes the contour. Thus, we can replace the curvature term and re-write the level set formulation as:

$$\frac{\partial}{\partial t}\Phi = \alpha\nabla \cdot (\mathcal{H}_{\epsilon}(|\nabla\Phi| - 1)\nabla\Phi) - (1-\alpha)\tilde{B}\delta_{\epsilon}(\Phi), \tag{22}$$

This intrinsic regularization allows the level set function to evolve without reinitialization and to efficiently develop more sophisticated topological changes.

In the following subsection, it can be revealed that this regularized version of the derived deformable model is actually an analogue of a global minimizer with a smoothing scheme of preserving an sign distance function.

D. Global minimizer with convex relaxation

As presented in [8], MAC or the derived deformable model (10) demonstrates superior performance in initialization invariancy and convergence ability by applying a novel external force in terms of global pixel interactions (produced from the assumed magnetic field). However, the energy minimization scheme of the deformable model (10) is homogeneous of degree one in the level set function (weighted area minimization flow) and then the relevant problem is non-convex, which cannot guarantee its gradient descent to a stationary state. As a consequence, this deformable model might converge to local minima and lead to its performance typically depending on initializations although it alleviates this limitation to some extent in comparison with other gradient based active contour models. Recent advances in optimization theory [21] allow to cast the respective problem in terms of convex functionals [22] so that its global minimizer can be found and thus the independency of the initialization is taken for granted. In the following, we infer how the global minimizer of the derived deformable model can be achieved by following the similar inference fashion in [22].

In the deformable model (10), without loss of generality, we use a length minimization flow (total variation flow) instead

of the previous weighted one in (18). In addition, the external force term can be viewed as a weighted area minimization flow. The contour evolution can therefore be described by minimizing the following energy functional:

$$\mathfrak{E}_1 = \alpha \int_\Omega |\nabla \mathcal{H}(\Phi)| \mathrm{d}\mathbf{x} + (1-\alpha) \int_\Omega B(\mathbf{x}) \mathcal{H}(\Phi) \mathrm{d}\mathbf{x}, \quad (23)$$

where B is the divergence of GCS and $\mathcal{H}(.)$ is the Heaviside function. Minimization of this energy functional is carried out using gradient descent:

$$\frac{\partial}{\partial t}\Phi = \alpha \nabla \cdot \left(\frac{\nabla \Phi}{|\nabla \Phi|} \right) - (1-\alpha) B \delta(\Phi), \quad (24)$$

where $\delta(.) = \mathcal{H}^{'}$ is the Dirac delta function. In fact, this gradient descent equation has the same stationary solutions with the following one:

$$\frac{\partial}{\partial t}\Phi = \alpha \nabla \cdot \left(\frac{\nabla \Phi}{|\nabla \Phi|} \right) - (1-\alpha) B, \quad (25)$$

where the delta function is simply omitted. Conversely, this equation can be derived from the following energy:

$$\mathfrak{E}_2 = \alpha \int_\Omega |\nabla (\Phi(\mathbf{x}))| \mathrm{d}\mathbf{x} + (1-\alpha) \int_\Omega B(\mathbf{x}) \Phi(\mathbf{x}) \mathrm{d}\mathbf{x}. \quad (26)$$

Consequently, the minimizer of the energy functional (23) can be obtained by minimizing the functional (26). Unfortunately, this energy functional will merely correspond to local minima due to the following reason: for a long curve evolution, the level set function could go to $+\infty$ when it is positive, and to $-\infty$ when it is negative. For fixing this issue, we restrict Φ in a closed set $[0, 1]$ for the functional minimization. Thus, this can convert the energy functional (26) to a convex functional as the following theorem states:

Theorem 1: The global minimizer for the energy functional (23) can be obtained by solving the following convex minimization:

$$min_{0 \le u \le 1} \{ \mathfrak{E}_3 := \alpha \int_\Omega |\nabla u| \mathrm{d}\mathbf{x} + (1-\alpha) \int_\Omega B(\mathbf{x}) u \mathrm{d}\mathbf{x} \} \quad (27)$$

and then setting $\Sigma = \{x : u(x) \ge \mu\}$ for a.e. $\mu \in [0, 1]$. The proof of Theorem 1 is presented in Appendix (B).

Following Theorem 1, the global minimizer of the deformable model (10) can be found by solving (27). In [23], the projected gradient flow is proposed to carry out this minimization scheme. We apply this solution scheme in the minimization functional (27). Thus, the global minimizer is obtained as the solution of the projected gradient flow:

$$\frac{\partial u}{\partial t} = \begin{cases} max\{\alpha div(\frac{\nabla u}{|\nabla u|}) - (1-\alpha)B, 0\} & if \quad u = 0, \\ min\{\alpha div(\frac{\nabla u}{|\nabla u|}) - (1-\alpha)B, 0\} & if \quad u = 1, \\ \alpha div(\frac{\nabla u}{|\nabla u|}) - (1-\alpha)B & else. \end{cases} \quad (28)$$

where u is normalized to keep $0 \le u \le 1$ during the curve evolution.

In numerical implementation, the curvature term can be calculated using the following equation with the forward-backward method.

$$div(\frac{\nabla u}{|\nabla u|}) = D_x^-(\frac{D_x^+ u}{\sqrt{(D_x^+ u)^2 + (D_y^+ u)^2 + \epsilon}}) + D_y^-(\frac{D_y^+ u}{\sqrt{(D_x^+ u)^2 + (D_y^+ u)^2 + \epsilon}}), \quad (29)$$

where D^- is the backward difference, D^+ is the forward one and ϵ is a small constant.

For numerical reasons, the level set function needs to keep the property of a desired signed distance function (SDF) during the curve evolution. Usually, this can be achieved with the reinitialization by solving a Hamilton-Jacobian equation [24]. In our case, we take another strategy that an isotropic Gaussian filtering of the level set function (LSF) is used to constrain the LSF to be close to an SDF in the evolution, i.e. $u_\sigma = u * G_\sigma$, which the solution scheme is similar to the method proposed in [25]. This regularizing scheme of the LSF is based on two considerations. One is that the curvature term is degraded as a Laplacian function when the SDF condition ($|\nabla u| = 1$) is satisfied. Thus, it is feasible that a Gaussian filtering is used to be instead of the Laplacian operation in (28) due to their equivalence [15]. The other is that the Gaussian filtering can help smooth the level set function while keeping it with the SDF property during the evolution. Consequently, the curvature term in (28) can be actually omitted because of utilizing the Gaussian filtering in the algorithm implementation. Note that the σ in the Gaussian kernel is given 3 in our experiments. Thus, along with this LSF regularization, (28) can be calculated in the evolution as follows:

$$\frac{\partial u}{\partial t} = \begin{cases} max\{-(1-\alpha)B, 0\} & if \quad u = 0, \\ min\{-(1-\alpha)B, 0\} & if \quad u = 1, \\ -(1-\alpha)B & else. \end{cases} \quad (30)$$

In addition, the regularized solution scheme (22) in Subsection II(C) in effect is functionally analogous to this global minimizer of the MAC scheme, which the LSF regularization is achieved by employing a Heaviside function to keep the SDF of the LSF in the context of local minimization that helps enhancing the capture range as well. This can be addressed in two aspects of the solution scheme. On the one hand, with the regularizing of the Heaviside function, the right side of (22) is approximately to be the value as obtained in (30); on the other hand, the first term of (22) becomes a Laplacian diffusion of the LSF when the SDF condition is not followed ($|\nabla u| > 1$), which is actually the similar way used in the solution (30).

E. Active surfaces using GCF

In the higher dimensional case, the GCF can be obtained by extending the definition in (7) with an appropriate kernel. For n dimensions, the kernel $k(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}$ is defined as: $k(\mathbf{x}) = 1/r^\zeta$ with $\zeta = n - 1$, $r = |x|$ is the distance from the origin. Given the image gradient $\nabla I : \mathbb{R}^n \to \mathbb{R}^n$ generated from the n-D image I, the GCF is calculated as follows:

$$\mathbf{E} = (E_{i_1}, ..., E_{i_n}) = (\nabla_{i_1} I * k(\mathbf{x}), ..., \nabla_{i_n} I * k(\mathbf{x})). \quad (31)$$

where $(i_1, ..., i_n)$ are the image volume coordinates.

For 3-D case, accordingly, $\mathbf{E} = (E_i, E_j, E_k)$ is actually obtained with the following equations:

$$\begin{cases} E_i(\mathbf{x}) = \nabla_i I * k(\mathbf{x}) = \sum_{\mathbf{s} \ne \mathbf{x}} \frac{\nabla_i I(\mathbf{s})}{R^2_{\mathbf{xs}}} = \sum_{\mathbf{s} \ne \mathbf{x}} f(\mathbf{s}) \frac{\hat{I}_x(\mathbf{s})}{R^2_{\mathbf{xs}}}, \\ E_j(\mathbf{x}) = \nabla_j I * k(\mathbf{x}) = \sum_{\mathbf{s} \ne \mathbf{x}} \frac{\nabla_j I(\mathbf{s})}{R^2_{\mathbf{xs}}} = \sum_{\mathbf{s} \ne \mathbf{x}} f(\mathbf{s}) \frac{\hat{I}_y(\mathbf{s})}{R^2_{\mathbf{xs}}}, \\ E_k(\mathbf{x}) = \nabla_k I * k(\mathbf{x}) = \sum_{\mathbf{s} \ne \mathbf{x}} \frac{\nabla_k I(\mathbf{s})}{R^2_{\mathbf{xs}}} = \sum_{\mathbf{s} \ne \mathbf{x}} f(\mathbf{s}) \frac{\hat{I}_z(\mathbf{s})}{R^2_{\mathbf{xs}}}. \end{cases} \quad (32)$$

Then, by generalizing (8) to the 3-D case, the change rate of the GCF along the surface normal direction B in 3-D is given as:

$$B = \nabla \cdot \mathbf{E}(\mathbf{x}) = \nabla \cdot (E_i, E_j, E_k). \quad (33)$$

Thus, by integrating (33) with (30), a level set solution of surface evolution is naturally formulated for the 3D extension of the deformable model (10).

In [26], a 3-D deformable model using the geometric potential force (GPF) is successfully proposed, which is obtained by computing the total potential field $G(\mathbf{x})$. Theoretically, the derived B in (33) is actually equivalent to this $G(\mathbf{x})$ that the proof is presented in Appendix (C).

Due to the GCF being orthogonal to the surface, the tangential plane with respect to GCF is actually parallel to the surface. For coping with noise/artefact interferences in image volumes, we just need to do the smoothing work along the GCF. Thus, we employ the common used isotropic diffusion/Laplacian for refining the GCF E. In accordance with (32), the refined GCF can be obtained by solving the following Euler equations.

$$\begin{cases} \frac{\partial}{\partial t}\mathcal{E}_i = p(E_i)\Delta\mathcal{E}_i - q(E_i)(\mathcal{E}_i - E_i), \\ \frac{\partial}{\partial t}\mathcal{E}_j = p(E_j)\Delta\mathcal{E}_j - q(E_j)(\mathcal{E}_j - E_j), \\ \frac{\partial}{\partial t}\mathcal{E}_k = p(E_k)\Delta\mathcal{E}_k - q(E_k)(\mathcal{E}_k - E_k), \end{cases} \tag{34}$$

where $\mathcal{E}_i(0,\mathbf{x}) = E_i(\mathbf{x})$, $\mathcal{E}_j(0,\mathbf{x}) = E_j(\mathbf{x})$, $\mathcal{E}_k(0,\mathbf{x}) = E_k(\mathbf{x})$, and $p(y)$ and $q(y)$ are given as:

$$p(y) = exp(-|y|f/K), q(y) = 1 - p(y)), \tag{35}$$

where $f = |\nabla I|$ and K is a constant.

III. Experimental Results

The proposed method has been tested on various synthetic and real images. To better exhibit the effectiveness and efficiency of our method, two state-of-the-art global minimizers of active contour based techniques are employed to make the comparison quantitatively and qualitatively, which their derived convex formulations are achieved by introducing a L^1 fidelity term instead of the L^2 one in the energy functional that is originally inspired by the global minimizer of the Chan-Vese method (GMCV) in [22]. In addition, these solution schemes benefit much from the unification of the complementary methods of the GAC and the Chan-Vese model and thus significantly outperform GMCV [27], [28]. Conveniently, the global minimizers based on the convex theory are named the global convex segmentation (GCS). Due to using various numerical solution schemes in the implementation of GCS, their overall performances are quite different in practice. One is to apply a dual formulation of the TV-norm for fast minimization in the solution scheme [27] (DF-GCS); the other [28] is to use a Split Bregman approach [29], [30] for providing a much more efficient solver (SB-GCS) and overcoming the drawbacks of the previous global regularized scheme [22], [27] such as penalty issue and time-consuming. The algorithms are implemented using MATLAB code and performed on an Intel Xeon 2×2.50 GHz processors with 32 GB RAM.

A. Parameter Setting

Our algorithm was run with the following parameter settings. For the computation of GCF in (7) and (32), the kernel radius is set between $N/2$ and N (N is the smallest coordinate of the image size) according to the situation. For solving the tensor diffusion (15), the parameter K' is set to 0.5. For the level-set implementation in (22), (28) and (11), the balancing parameter α is uniformly set to 0.1. In addition, for the Gaussian filter of keeping the level-set function SDF, the standard variation is given between 2 and 5 according to image modalities.

For obtaining the best performance of SB-GCS and DF-GCS, we use a series of fidelity values for selection in the SB-GCS case and set the most appropriate parameters as suggested in [27] for DF-GCS in accordance with image situations. Furthermore, in some cases, the relevant parameters are carefully adjusted for better adaptive to actual modalities such as $beta = 1/255^2$ used in the images with poor contrast rather than the suggested $beta = 50/255^2$ for SB-GCS.

B. Computational Cost

In 2D, the main computation comes from the calculation of the gradient convolution field GCF and the diffusion process. For the calculation of the GCF ((7) and (8)), it is performed by convolving the image gradient ($N \times N$) with an inverse distance kernel ($(2R+1) \times (2R+1)$)). By using fast Fourier transform (FFT) in the frequency domain, the calculation can be efficiently performed. The computational order are then estimated as $O((N+2R)^2 log(N+2R)^2) = O((N+2R)^2 log(N+2R))$ [31]. Due to the typical values of the kernel radius R being range from $N/2$ to N, the complexity of this part in the proposed method is $O(N^2 logN)$. For the tensor diffusion, the numerical implementation can be performed using the additive operation scheme (AOS) and its computational complexity is $O(N^2)$ [14]. Therefore, the proposed method is totally computational of the order $O(N^2 logN)$. Similarly, the computational complexity in 3D is $O(N^3 logN)$.

C. Arbitrary Initialization

As a global minimizer, our method does actually not depend on any initialization. To demonstrate this ability and its performance, the images with various conditions are used in Fig. 3. The original images are illustrated in the first row. The intermediate results are presented in the second row, the fourth row and the sixth row for the proposed, SB-GCS and DF-GCS respectively. The converged results are illustrated in the third (the proposed), fifth (SB-GCS) and seventh (DF-GCS) rows. Although both SB-GCS and DF-GCS meet difficulties in these cases, our method can detect the object boundaries quite well such as the small obscure objects in "Fish" and the bottom ends in "Vessel". It is worth noting that the proposed method also exhibits good convergence ability that all examples are converged within 10 runs.

For examining the initialization invariancy of the proposed method, one vessel example of angiogram imaging is used in Fig. 4 that the proposed method can consistently converge to the boundaries with various initializations. Row 2 and Row 3 in Fig. 4 illustrate the intermediate results of the first run and the second run. The converged results are presented in Row 4.

We have to mention that the proposed method is not intended for texture segmentation as it is based on the assumption that intensity discontinuities reasonably collocate with object boundaries, as in any other edge based approaches. On the other hand, the advantages out method achieved is significant and can be considered as a potential alternative to regions based approach. The "fish" example in Fig. 3 demonstrates the advantages of our method in detecting multiple objects with unknown topologies, since it does not require any specific initialization. It is practically impossible for other vector field based methods, such as GGVF where careful initialization is necessary.

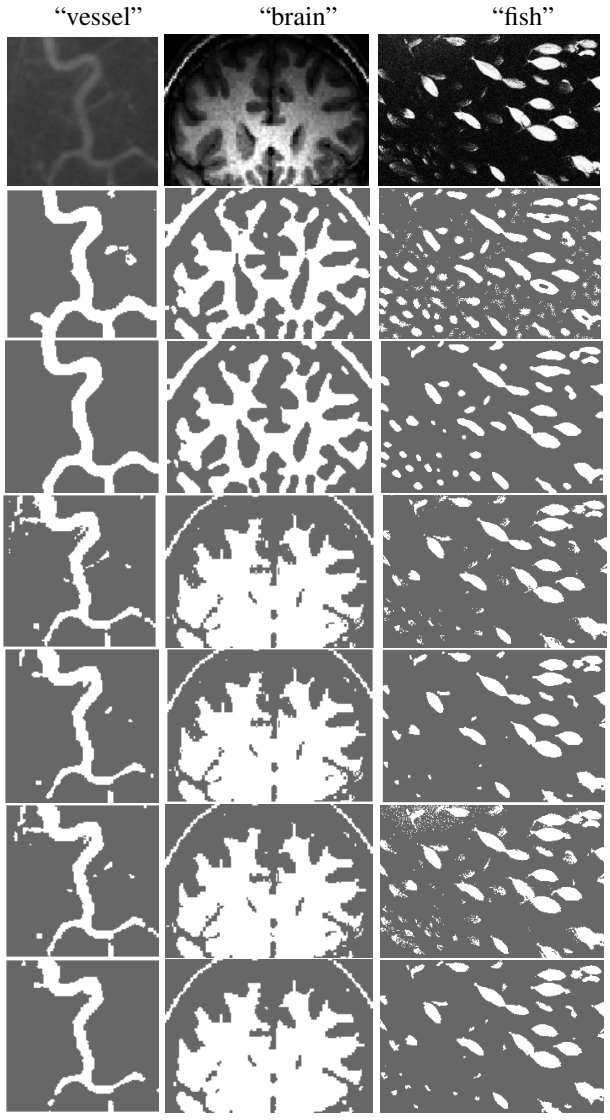


Fig. 3: Results of the proposed method, SB-GCS and DF-GCS without initialization. Original images (first row), the intermediate results of the proposed (second row, the 1^{th} run for all), SB-GCS (fourth row, the fidelity values are 1, e-2 and e-3 respectively) and DF-GCS (sixth row, the 10^{th}, 1^{th} and 1^{th} iteration respectively), and the final results for the proposed (third row), SB-GCS (fifth row, the fidelity values are e-2, e-1 and e-4 respectively) and DF-GCS (seventh row, the 50^{th}, 8^{th} and 10^{th} iteration respectively).

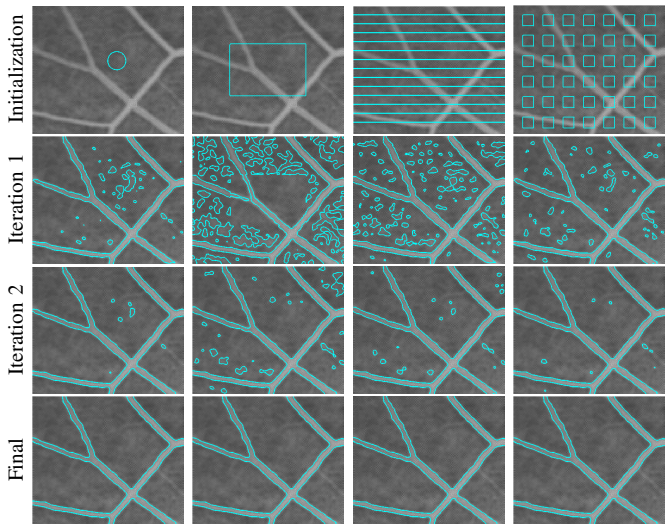


Fig. 4: Results of the proposed method with various initializations. Row 1 is the initialization, Row 2 is the intermediate results of the first iteration, Row 3 is the results of the second iteration and Row four is the the converged results.

D. Sensitivity of Weak Boundaries and Poor Contrast

Due to a global representation of object boundaries using image gradient, the proposed method can effectively deal with the difficulties in the images with incomplete edge information, which the performance of the traditional methods are restricted as discussed in [8]. In addition, the traditional global minimizer also meets problem to recover shapes in weak edge/poor contrast situations. In the "fish" case of Fig. 3, the proposed method can extract the shapes of the small obscure "fish" but DF-GCS is limited to be capable of capturing these boundaries.

In Fig. 5, the "bear" is hidden behind the grasses and the boundary of its bottom is then faintly discernible. We apply the proposed method with various initializations to this case of weak boundaries. The second and third columns show the intermediate results. The final results in the fourth column illustrate that the weak boundaries can be recovered pretty well even though the initializations are greatly varied. However, the performances of both DF-GCS and SB-GCS have been seriously interfered by the insufficient boundary information as shown in Fig. 7.

In the situation of poor contrast imaging such as blood vessel images, image quality is significantly degraded due to the inherent nature of the acquisition process for example "vessel" in Fig. 3. The traditional methods usually suffer the problem of blurring features that seriously disturbs the boundary extraction and leads to spurious edges. In Fig. 6, the proposed method is applied to three images. One is a telangiectasia image with weak boundaries. The others are two retinal images with poor contrast. Two intermediate results are presented in the second and third columns. The final results show that our method can extract the main features properly and handle this difficult situation robustly due to integrating a region-like method (MAC) and a direction-based diffusion scheme (tensor diffusion). In contrast, the results in Fig. 7 show that the performance of both SB-GCS and DF-GCS is seriously affected by incomplete boundary information and

poor contrast even though the running parameters are carefully used.

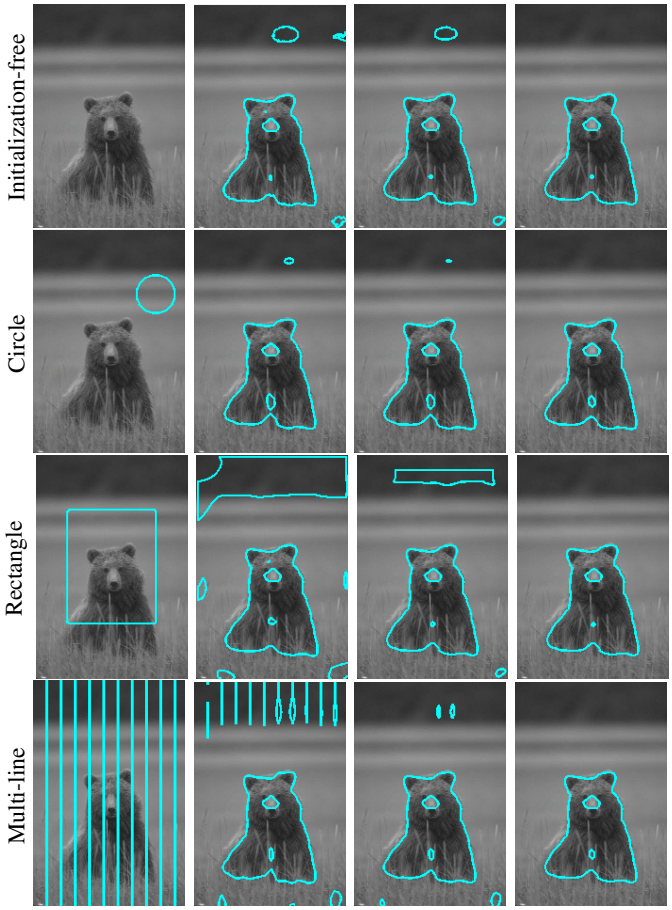


Fig. 5: Results using the proposed method for weak boundaries. The rows are corresponding to different initializations: free, circle, rectangle and multi-line respectively.

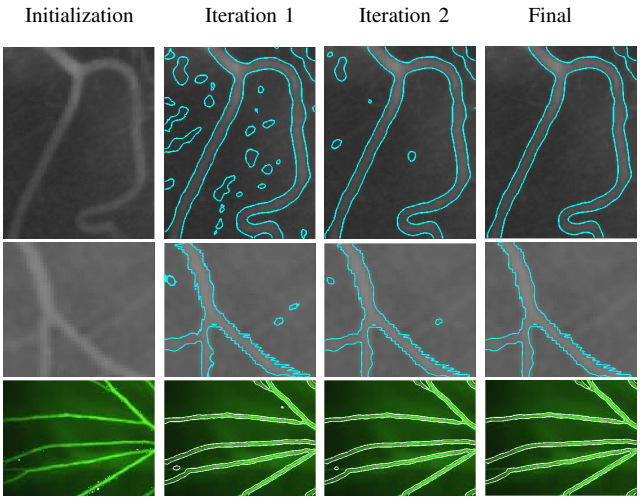


Fig. 6: Three examples of the retinal images using the proposed method for weak boundaries and poor contrast. First column shows the initialization images. Two intermediate results (the 1^{th} and 2^{th} iterations) are presented in the second and third columns. Final results are illustrated in the fourth column.

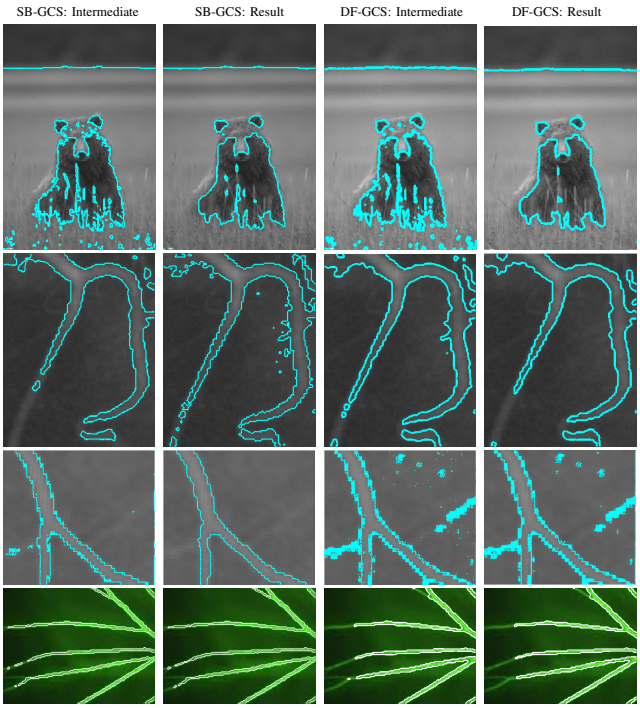


Fig. 7: Results applying SB-GCS and DF-GCS in the images used in Fig. 5 and Fig. 6. The first and second columns are the results for SB-GCS. The intermediate results correspond to the fidelity values e-2, e-1, e-2 and e-2 for each image. The final results are with the fidelity values e-3, e-3, e-1, e-3 and e-1 respectively. The third and fourth columns are the results for DF-GCS. The intermediate ones are from the 5^{th}, 20^{th}, 1^{th} and 20^{th} iterations. The converged ones are obtained from 50^{th}, 50^{th}, 20^{th} and 50^{th} iterations.

E. Sensitivity of Complex Geometries and Topologies

Usually, the gradient-based methods have limited capture range for complex geometries such as deeper concavities because they inevitably depend on boundary gradient [8]. On the contrary, thanks to global pixel interactions and a global minimization scheme, the proposed method can perform superior in extracting complex object geometris and topologies. Two typical images are used for demonstrating this capture ability of the proposed method in Fig. 8. We apply the proposed method with various initializations (Rows (a) and (d)) to these images. The intermediate results are presented in Rows (b) and (e). The final results in Rows (c) and (f) show that the proposed method can accurately capture the boundaries including the deeper concavities and complex topologies such as the tail and neck parts of "tiger" and the different objects of "shape".

For comparison, we apply SB-GCS and DF-GCS to the "tiger" and "shape" images. Fig. 9 presents the intermediate and converged results for the proposed method (Rows (a) and (d)), SB-GCS (Rows (b) and (e)) and DF-GCS (Rows (c) and (f)) respectively. It is shown that both SB-GCS and DF-GCS cannot neatly extract the object structures in the images although the parameters and iteration process have been finely applied. For examples, SB-GCS gets lost in the tail of "tiger" and some parts (spoon, button and crayon) of "shape", and DF-GCS produces the artefacts in "tiger" and misses several parts of "shape". In contrast, the proposed method can detect both geometrical and topological structures well although a small

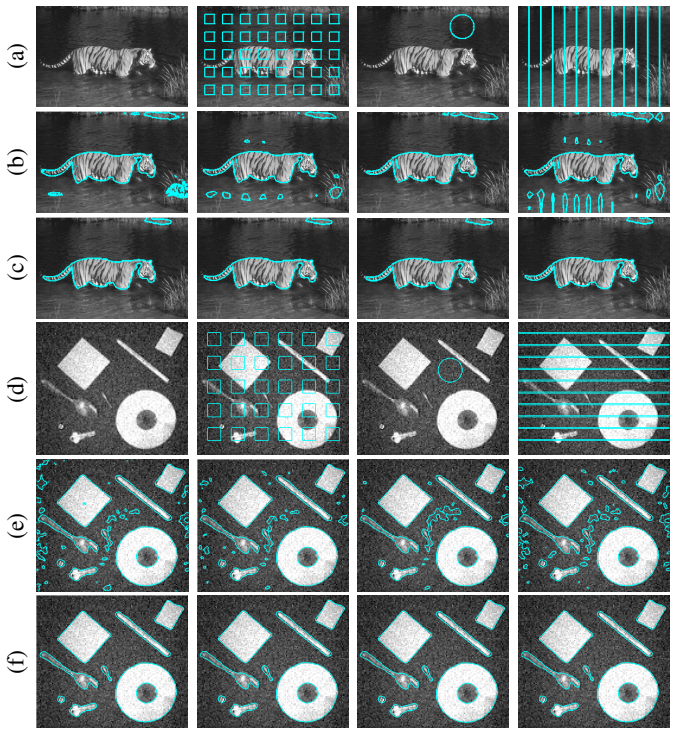


Fig. 8: Examples using the proposed method for complex geometries and topologies. Row (a): "tiger" image using various initializations, row (b): intermediate results (first run), row (c): final results, row(d): "shape" image with the Gaussian noise of SD=0.01 using various initializations, row (e): intermediate results (first run), row (f): final results.

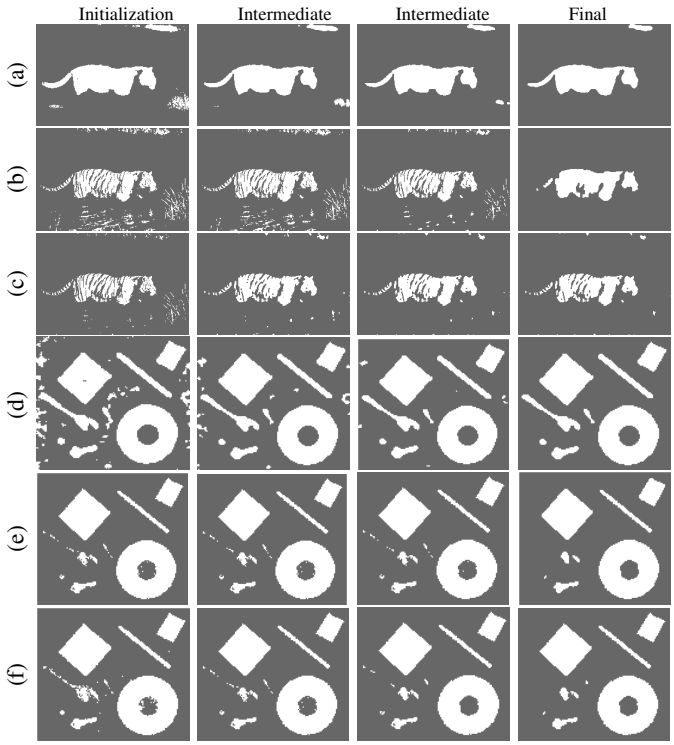


Fig. 9: Results using the proposed method, SB-GCS and DF-GCS for the "tiger" (a-c) and "shape" (d-f) images in Fig. 8. Row (a): Results of the proposed method with the 1^{th}, 2^{th}, 5^{th} and 8^{th} (converged) iterations, row (b): Results of SB-GCS with the fidelity values: e-1, e-2, e-3 and e-4 (best), row (c): Results for DF-GCS with the 1^{th}, 20^{th}, 50^{th} and 100^{th} (converged) iterations; row (d): Results of the proposed method with the 1^{th}, 2^{th}, 3^{th} and 6^{th} (converged) iterations, row (e): Results of SB-GCS with the fidelity values: e-1, e-2, e-3 and e-4 (best), row (f): Results for DF-GCS with the 1^{th}, 3^{th}, 5^{th} and 20^{th} (converged) iterations.

part of the spoon is not captured due to noise interference.

In Fig. 10, another example is presented for demonstrating the ability of capturing complex geometries with our method. Rows (a-e) illustrate that our method performs robustly with various initializations and consistently converges to the object boundaries (Final results in Column 4). However, the results in Rows (f-g) show that both SB-GCS and DF-GCS meet problems while capturing the complex geometries in the image. For fair comparison, several results using different parameter settings and iterations for SB-GCS and DF-GCS are presented in Rows (f-g) where the ones in the fourth column are the best.

F. Noise and Artefact Sensitivity

Noise interference is a critical factor to affect the capture range of object boundaries using the edge-based method. In the proposed method, a GCF nonlinear diffusion scheme is used to enhance the ability of dealing with noise interference whilst the global pixel interactions are in fact to play a region-like role in boundary finding. Our approach can therefore exhibit its comparable performance to the sophisticated region-based methods such as SB-GCS and DF-GCS. In the previous examples, Fig. 1 and Fig. 2 show that the proposed nonlinear diffusion used in our method performs significantly better than the Laplacian diffusion used in [8].

For substantially demonstrating the ability of our method in dealing with noise interference, five synthetic images with various noise conditions are firstly used in Fig. 11. The resulting images in each column correspond to a given iteration so as to show the curve evolution whilst the converged results are presented in the sixth column. It can be seen that our method can successfully extract the object boundaries although serious noise and artefacts lead to very low contrast or insufficient boundary information in each case. However, Fig. 12 shows that the performances of SB-GCS and DF-GCS are more or less affected by the noise in these cases. Particularly in the fourth and fifth cases, DF-GCS is seriously affected by the noise interference and get lost in detecting the boundaries. Meanwhile, SB-GCS is incapable of detecting the rectangle in the middle of the image in the fourth case. In addition, both SB-GCS and DF-GCS do not neatly obtain the object boundaries in the first and second cases. For the concentric circle case in the first row, the quantitative analysis is also provided in Fig. 20 (the case of SD=1).

Another four examples of real images are illustrated in Fig. 13. The images used are from the blood vessel imaging of angiogram that are severely contaminated by noise and artefacts. The intermediate results of the first five consecutive iterations are presented in the first five columns. The converged results (the tenth run for all cases) are shown in the last column. In these examples, we can see that the proposed method demonstrates the capability of extracting the shapes of the objects and the ability of fast convergence. On the other hand, for comparative reason, Fig. 14 presents the results of applying SB-GCS, DF-GCS and our method to the same images. We can see that our method outperforms the others obviously in all cases while DF-GCS completely gets lost in the third and fourth cases and SB-GCS meets great difficulties in dealing with noise interferences.

In medical image modalities, various artefacts may occur due to acquisition processes, for examples scattering noise, partial volume effect and motion artefacts. Furthermore, there usually have several main intensity classes reflecting different components in tissues such as muscles, bones and cartilage in CT images, how to properly identify the objects that we prefer to is a great challenge in deformable models? For instance in Fig. 15, we need to recover the bone information (the parts in white) surrounded by muscle tissues where Slices 25 and 101 are from a CT-knee volume. Fig. 15 shows that our method can lead the curve to trespassing the soft tissue area and reaching the bone boundaries successfully. The curve evolution is presented by showing the results of the $1^{th} to 7^{th}$ iterations and the converged in Columns 2-9. However, to some extent, both SB-GCS and DF-GCS meet respective difficulties as shown in Fig. 17. Particularly, SB-GCS is merely to approach the soft tissues' boundaries rather than the bones' ones.

Three further examples from different medical volume scans are used in Fig. 16. In the first row, our method is applied to a foot X-ray scan slice and the bone's boundaries are successfully detected which the situation is similar to the examples in Fig. 15. In the second row, a slice of a lobster CT scan is used in that there is heavy artefacts in the background; the evolving process shows that the curve can trespass the artefacts and reach the object boundaries with our method. In the third row, a slice of a tooth CT scan is used and the iterative results demonstrate that our method can overcome the interference of the artefacts and detect the boundaries of the ROIs properly. In contrast, Fig. 17 illustrates that the performances of both SB-GCS and DF-GCS have been seriously affected by the artefacts in these images, which lead to extracting wrong object boundaries. Particularly in SB-GCS, the curve can only reach the boundaries of the softe tissues or artefacts in the first three cases although the parameters have been finely adjusted. In the last example of the tooth slice, SB-GCS and DF-GCS can roughly detect the object boundaries after the careful adjustment of the parameters. However, these methods will meet problems when they are extended to 3D case due to a large of slices used and the circumstances of slices being changeable. For illustrating the issues in 3D, Fig. 18 shows the results of 6 consecutive images after using the 3D version of SB-GCS and DF-GCS, which are obviously interfered by the artefacts and lead to inaccurate or even wrong surface reconstruction as shown in Figs. 30 and 31. On the other hand, the proposed method can perform robustly in surface reconstruction even though a large image sequence is involved as shown in the last row of Fig. 18 (the recovered surface shown in Fig. 27). Another example of a slice sequence is presented in Fig. 19 that the resulting images come from the surface reconstruction of a foot scan with our method and the other two. Comparing to SB-GCS and DF-GCS, our method can detect the boundaries of the foot's bone pretty

better while the others cannot perform properly (the recovered surfaces shown in Fig. 27, Fig. 30 and Fig. 31). Accordingly, in 3D reconstruction, our method can effectively deal with the artefacts and naturally recover the structural information from a large slice sequence.

For quantitatively demonstrating the performance of the proposed method in noisy images, we apply our method, SB-GCS and DF-GCS to a set of generated noisy images with various Gaussian noise levels (zero mean, the standard deviations (SD) ranging from 0.1 to 1 respectively). The noise-free image is the same as the concentric circle image used in Fig. 11 (the first case). Figs. 20 (a-c) present the correct rates of these three methods recovering the boundaries in the background, the foreground and the whole image respectively. As shown, our method does outperform DF-GCS in all the cases except the case of SD=0.1 in the foreground (Fig. 20(b)). Comparing with SB-GCS, Fig. 20(a) illustrates that the performance of our method is better in the background. For the performance in the foreground, Fig. 20(b) shows that our method works comparative to SB-GCS while the two methods alternatively present the advantages in different SD ranges. Consequently, the overall performance of the proposed method is sure to win out over that of both SB-GCS and DF-GCS as shown in Fig. 20(c). Additionally, it is worth noting that the performance of DF-GCS is significantly inferior to the other two methods in these noisy cases.

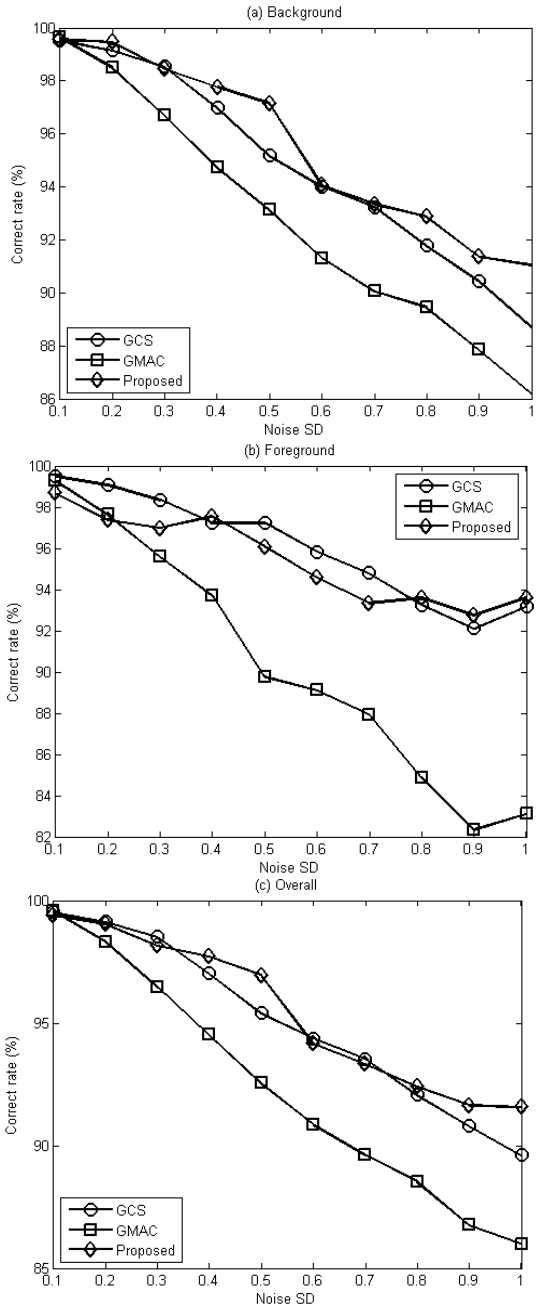


Fig. 20: Correction rates of object detection using the proposed method, SB-GCS and DF-GCS in the concentric circle images with the noise levels of SD ranging from 0.1 to 1. (a) Results in Background, (b) Results in Foreground and (c) Overall results

G. Active Surfaces

As derived in Section II(E), our GCF scheme can naturally extended from 2D to higher dimensional space. In 3D, the following experiments are performed using the obtained level set formulation with (33) and (28), and a Laplacian diffusion scheme eq. (34). The volume image data is courtesy of the Volume Library (http://www9.informatik.uni-erlangen.de) and the Volvis medical datasets (www.volvis.org).

Firstly, two scan volumes and their generated noisy datasets are used for surface reconstruction. Fig. 21 shows the results using the proposed method in a CT Fuel Injection scan and its generated noisy datasets. Row (a) is with the original data; rows (b) and (c) correspond to the respective noisy datasets. First column illustrates the slice example in the sagittal, coronal and transverse planes. Columns 2-4 present the initial surface and two intermediate results. The final results in the fifth column show that the proposed method can recover the structures properly and deal with noise efficiently. The detected object contours in Slices 24-28 are presented in Fig. 22. Another surface example for dealing with noise interference is shown in Fig. 23 and Fig. 24, which the image data is acquired for Daisy Pollen Grain using a confocal laser scanning microscopy. For comparison, Fig. 25 illustrates the surface reconstruction using SB-GCS and DF-GCS in the Fuel and Daisy data. We can see that both of SB-GCS and DF-GCS have been seriously interfered by noise and even they cannot recover the object shape properly in the noise-free case of Fuel Injection due to low contrast. In addition, the performance of these two methods in the noisy slices are presented in Fig. 26 which correspond to the reconstructed surfaces in Fig. 25.

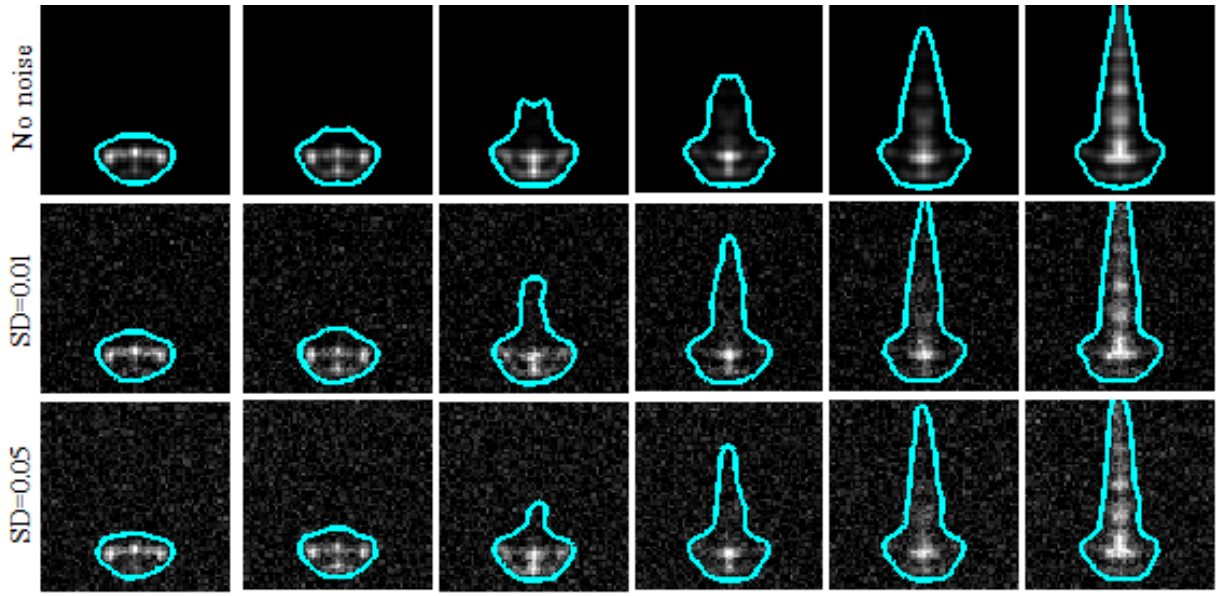


Fig. 22: Detected contours in Slices 24-28. Row 1: the original data; row 2: the data with the Gaussian noise of sd=0.01; row 3: the data with the Gaussian noise of sd=0.05.

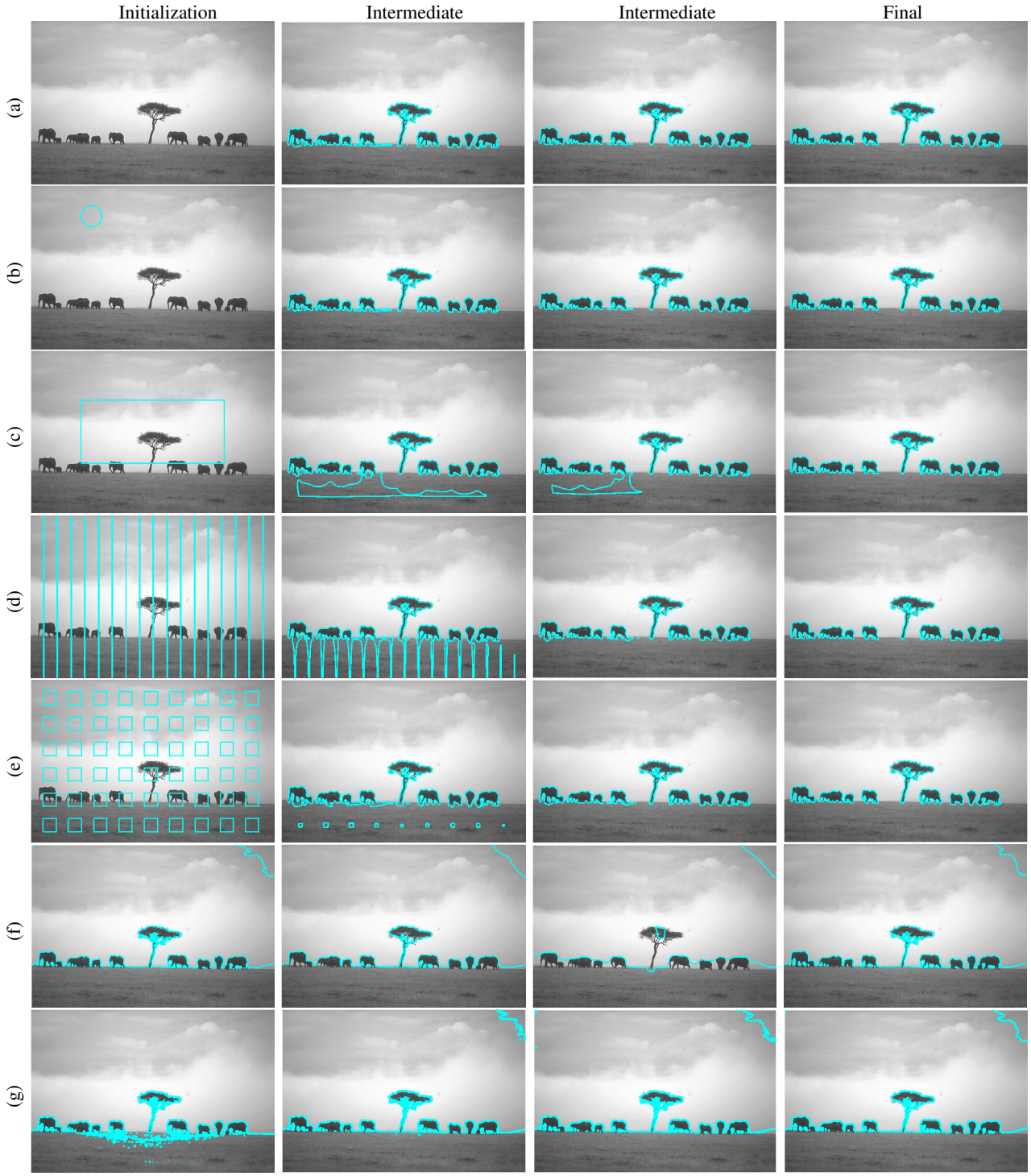


Fig. 10: Results using the proposed method, SB-GCS and DF-GCS for the "elephant". Rows (a-e) are the results using our method with various initializations, which Column 1 is the initialization, Columns 2-3 show the intermediate results of the 1^{th} and 2^{th} iterations, Column 4 is the converged result. Row (f): Results of SB-GCS with the fidelity values: e-2, e-4, e-5 and e-3 (best), row (g): Results for DF-GCS with the 1^{th}, 2^{th}, 3^{th} and 10^{th} (converged) iterations.

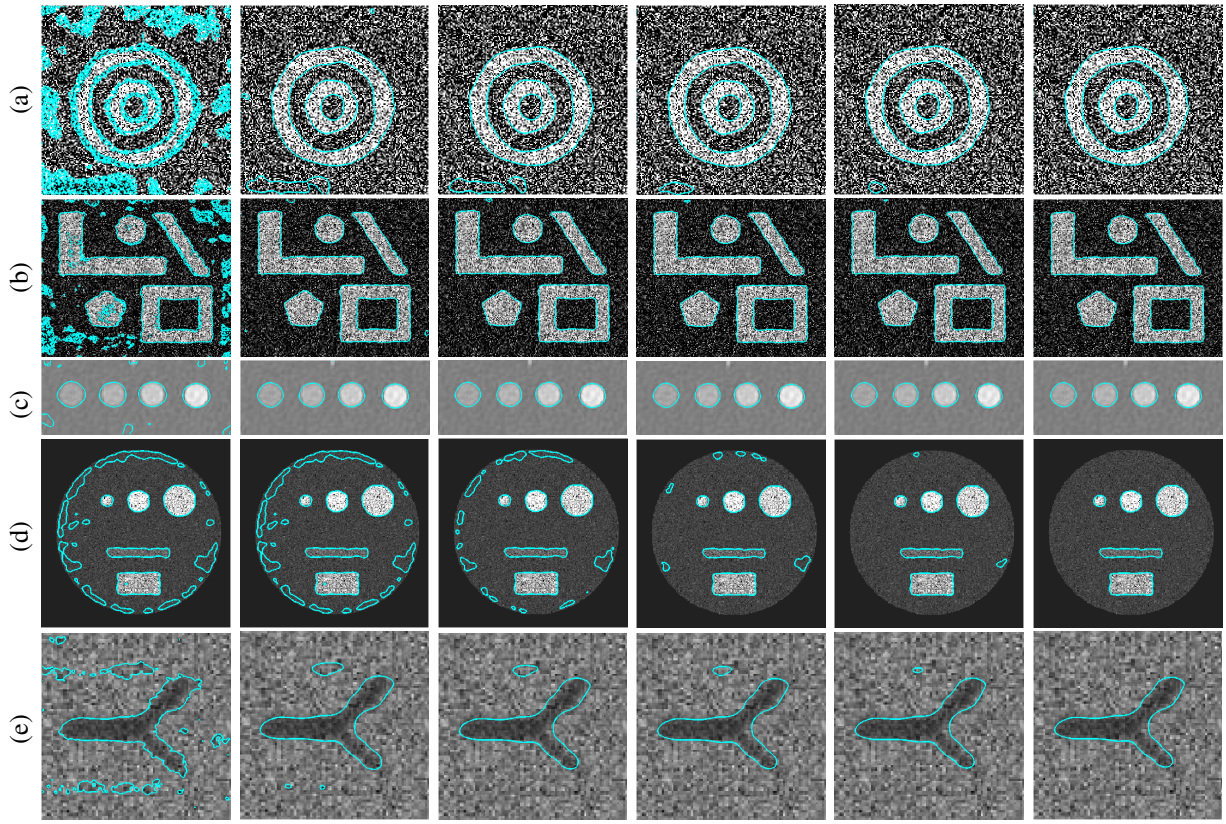


Fig. 11: Examples of the proposed method applying in various synthetic noisy images. Each Column shows the intermediate and the converged results. Row (a) corresponds to the $1^{th}, 2^{th}, 3^{th}, 6^{th}, 9^{th}$ and 10^{th} iterations; Row (b) corresponds to the $1^{th}, 2^{th}, 3^{th}, 4^{th}, 5^{th}$ and 6^{th} iterations; Row (c) corresponds to the $1^{th}, 2^{th}, 3^{th}, 4^{th}, 5^{th}$ and 6^{th} iterations; Row (d) corresponds to the $1^{th}, 2^{th}, 3^{th}, 5^{th}, 8^{th}$ and 10^{th} iterations; Row (e) corresponds to the $1^{th}, 2^{th}, 3^{th}, 4^{th}, 4^{th}$ and 6^{th} iterations.

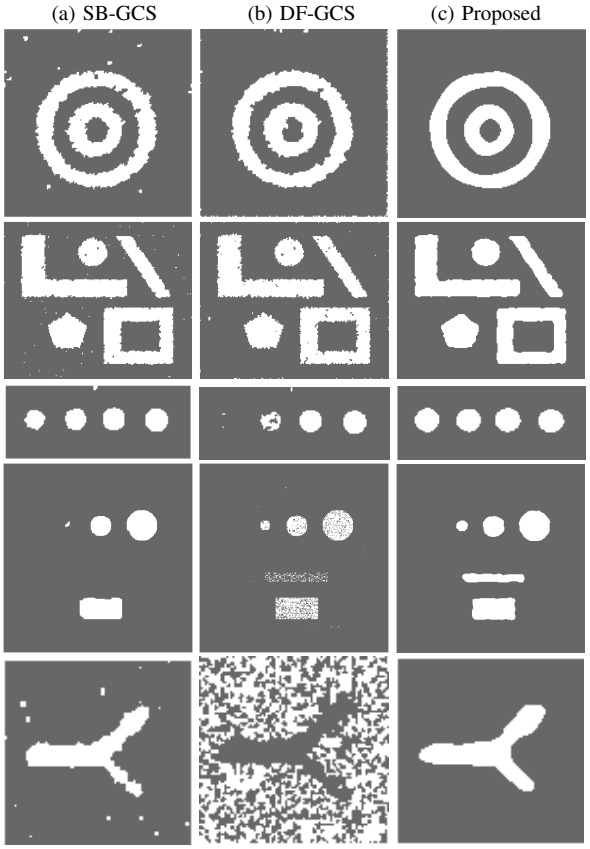


Fig. 12: Comparison results for the images in Fig. 11. Column (a): Results using SB-GCS ; Column (b): Results using DF-GCS ; Column (c): Results using the proposed method . For the best performance, from top within each column: we use the fidelity values e-5, e-4, e-3, e-5 and e-4 respectively for SB-GCS, iterations:100, 50, 50, 50, 50 respectively for DF-GCS's convergence.

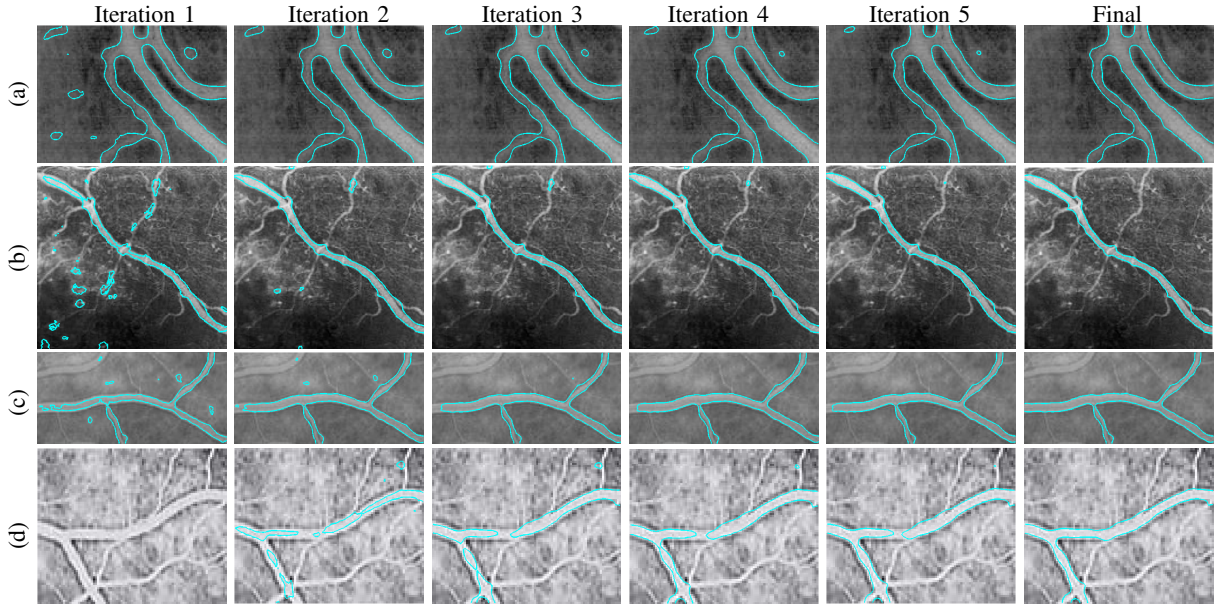


Fig. 13: Examples of the proposed method applying in angiogram images. Each Column shows the results after the given iteration from the first run to the fifth run and the converged one. Rows (a-d) are for the real images with various noise and artefact interferences.

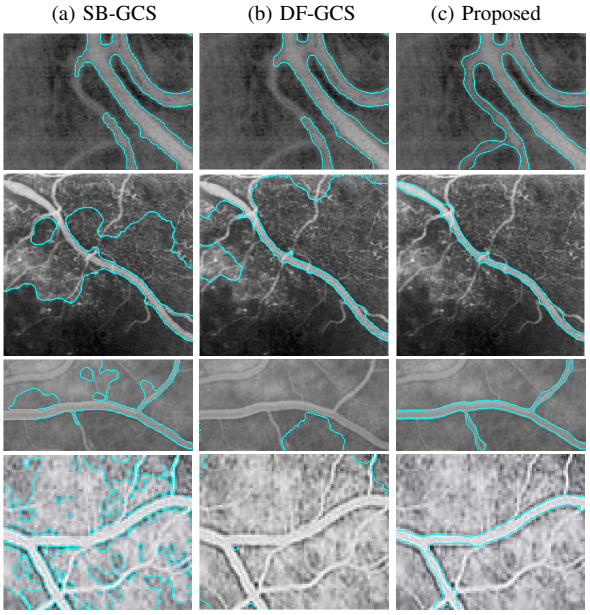


Fig. 14: Comparison results for the images in Fig. 13. Column (a): Results using SB-GCS ; Column (b): Results using DF-GCS ; Column (c): Results using the proposed method . For the best performance, from top within each column: we use the fidelity values e-3, e-4, e-3 and e-1 respectively for SB-GCS, iterations:60, 80, 80, 50 respectively for DF-GCS's convergence.

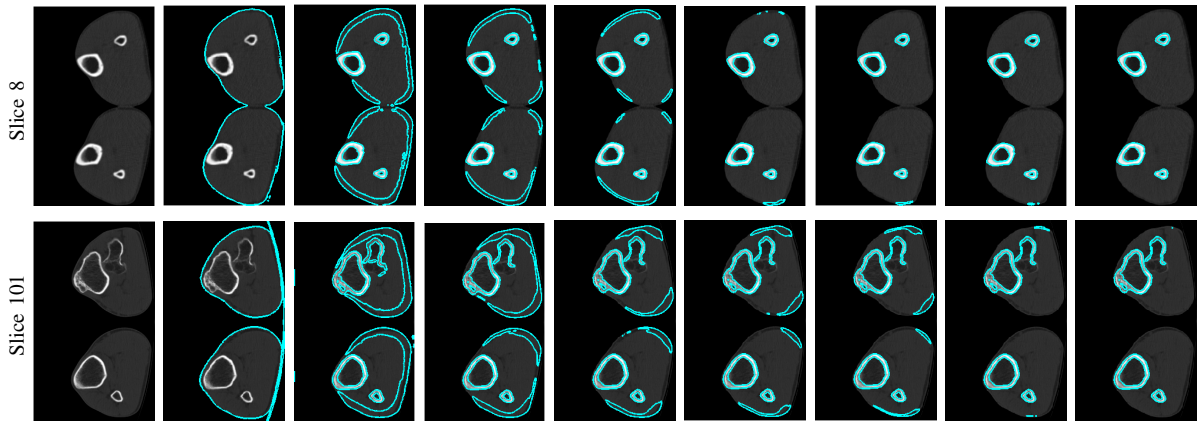


Fig. 15: Results using the proposed method for two images in a knee CT scan (379*229*305). Row (a): Slice 8, row (b): Slice 101. Column 1 to column 9: original image, the 1^{th} iteration to the 7^{th} iteration, final result (the 10^{th} iteration)

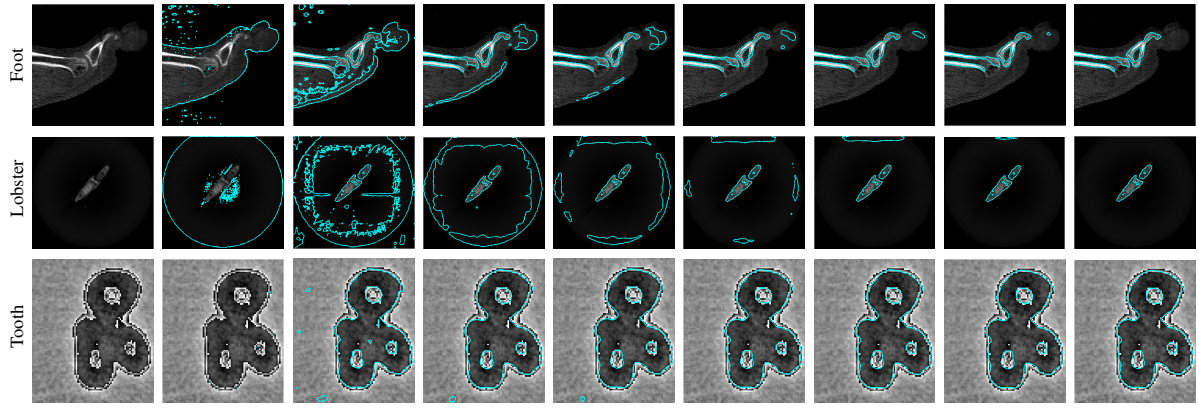


Fig. 16: Three examples using the proposed method. Row 1: Slice 105 in a foot X-ray scan (256*256*256), row 2: Slice 48 in a lobster CT scan (301*324*56), row 3: Slice 71 in a tooth micro CT scan (92*78*161). Column 1 to column 9: original image, iteration 1 to iteration 7, final result (iteration 10).

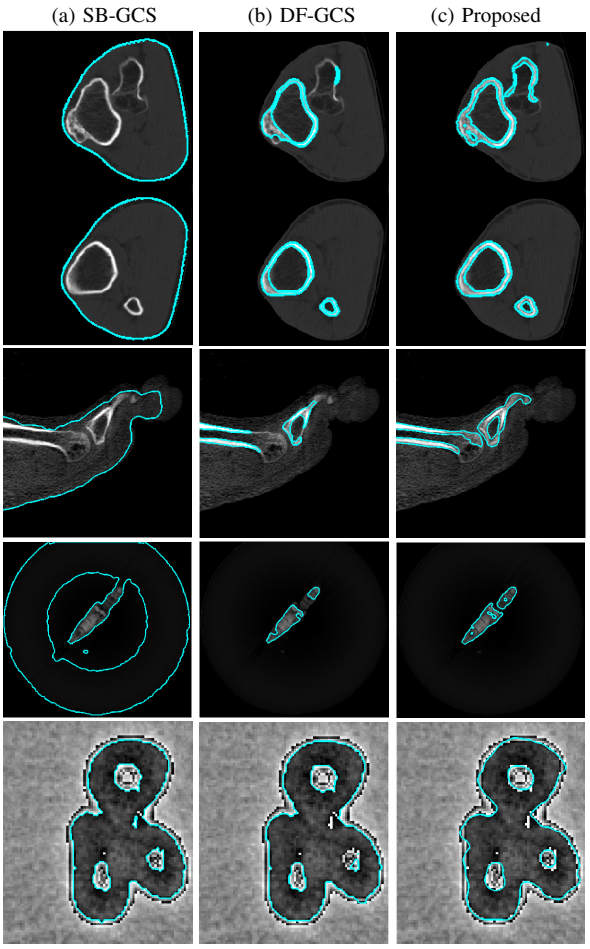


Fig. 17: Comparison results for the real images in Fig. 15 (Fig. 101) and Fig. 16. Column (a): Results using SB-GCS; Column (b): Results using DF-GCS; Column (c): Results using our method. For the best performance, from top within each column: we use the fidelity values e-3,e-4,e-2 and e-4 respectively for SB-GCS, iterations:50, 20, 20, 50 respectively for DF-GCS's convergence.

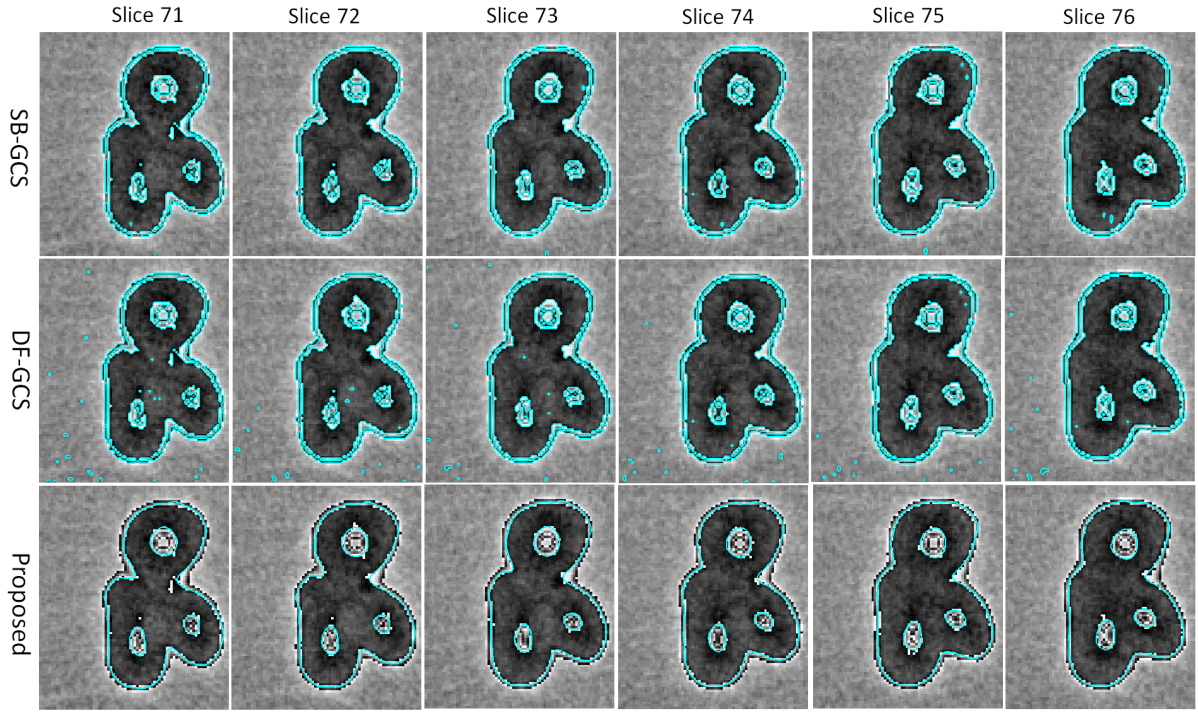


Fig. 18: Results for Slice 71-Slice 76 after applying the 3D version of SB-GCS, DF-GCS and the proposed method in the tooth volume.

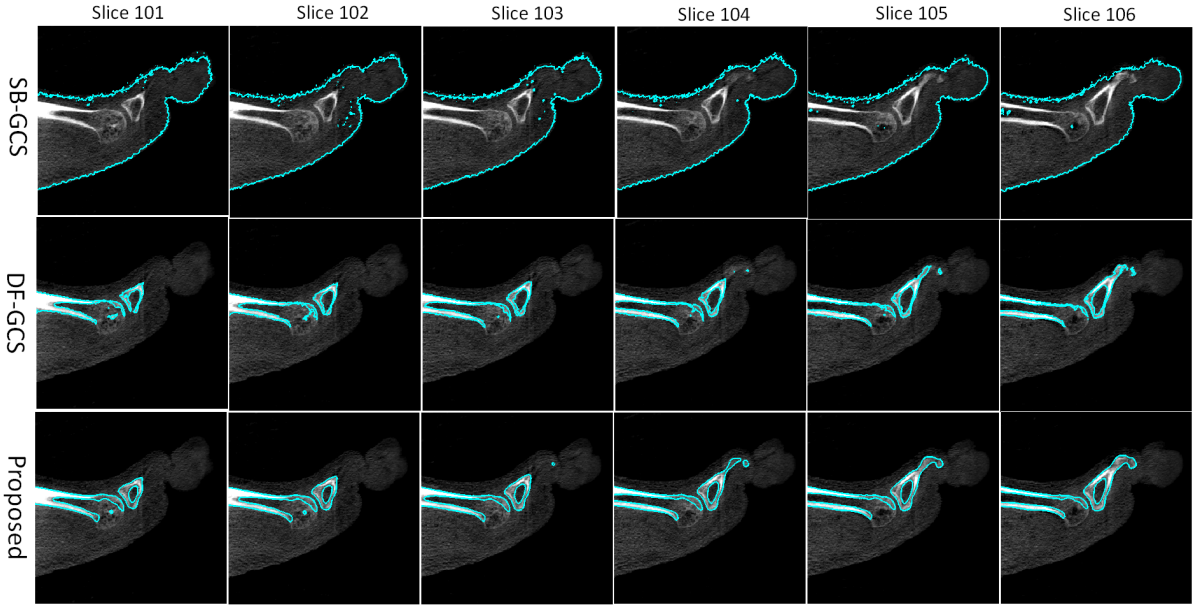


Fig. 19: Results for Slice 101-Slice 106 after applying the 3D version of SB-GCS, DF-GCS and the proposed method in the foot volume.

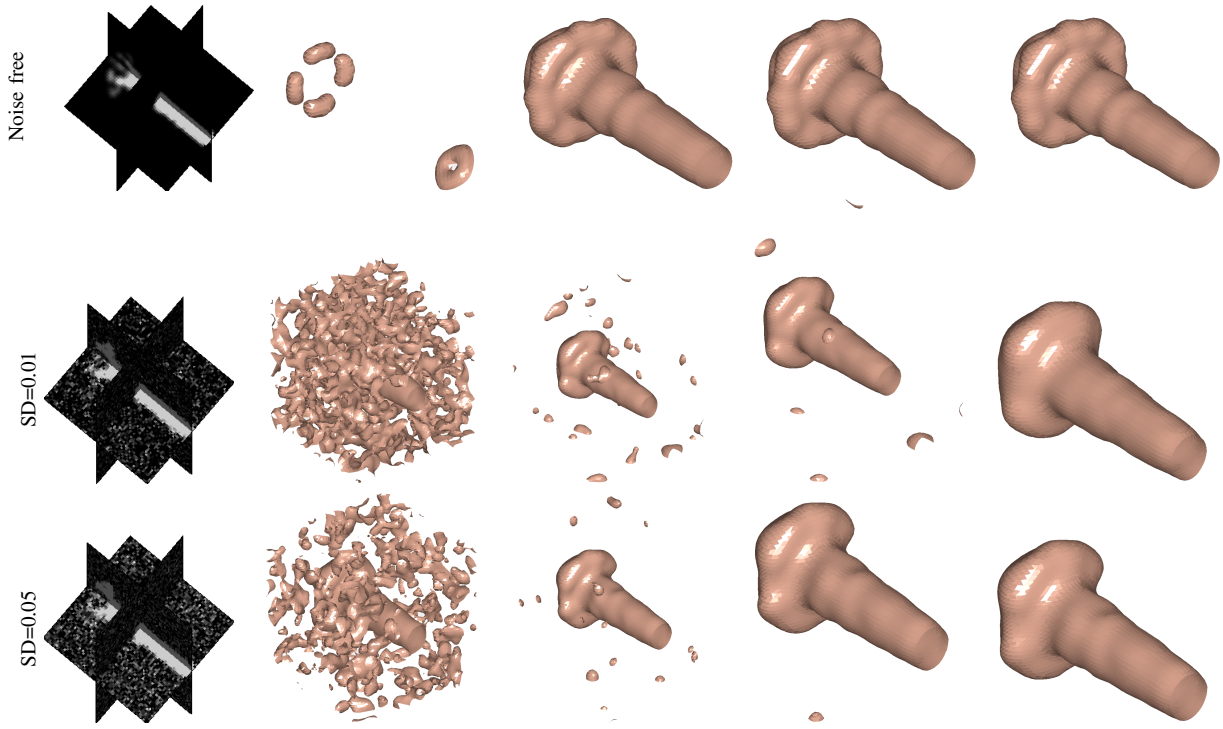


Fig. 21: Results using the proposed method with the fuel injection data (64*64*64). First row is with the original dataset. The datasets in the second and third rows are with the Gaussian noise (SD=0.01, 0.05). The first column shows the example slices on the sagittal, coronal and transverse planes. The second column presents the initial surface. The final column is the converged result. The other two columns show the intermediate results. Row (a): Slice example (32,32,32), the 2^{th}, 3^{th} and 5^{th} iterations, and final; row (b): Slice example (32,32,32), the 2^{th}, 3^{th} and 4^{th} iterations, and final; row (c): Slice example (32,32,32), the 2^{th}, 3^{th} and 4^{th} iterations, and final.

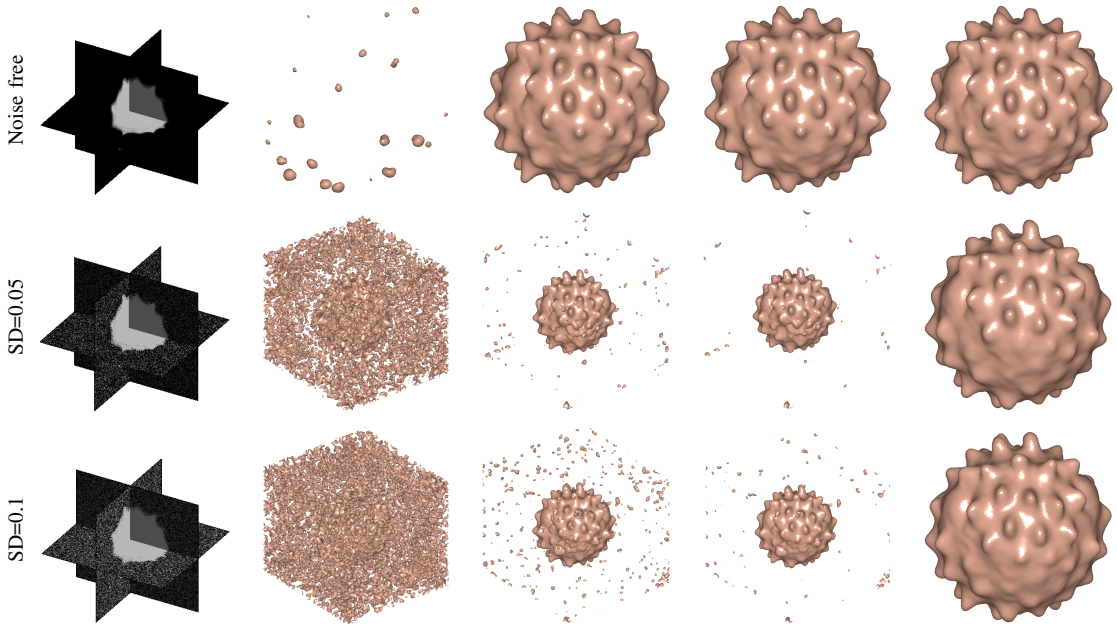


Fig. 23: Results using the proposed method in the Daisy Pollen Grain data (192*180*168). First row is with the original dataset. Second and third rows are with the noisy datasets (sd=0.05, 0.1). The first column shows the example slices of sagittal, coronal and transverse directions. The second column presents the initial surface of the data; the third and fourth columns are the intermediate results; the final column is the converged result. Row (a): 1^{th}, 2^{th}, 3^{th} and final; row (b): 1^{th}, 2^{th}, 3^{th} and final; row (c): 1^{th}, 2^{th}, 3^{th} and final.

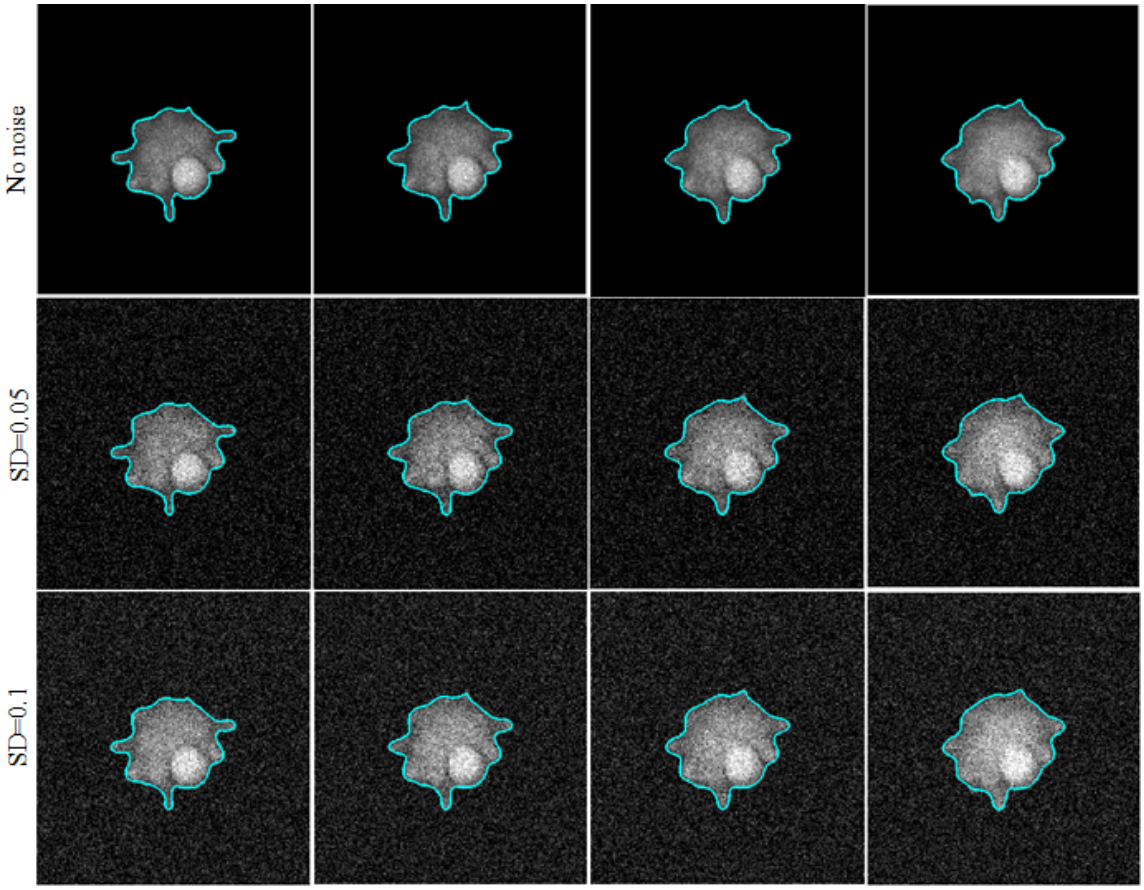


Fig. 24: Detected contours in Slices 101-104 of the Daisy volume. Row (a): the original data; row (b): the data with the Gaussian noise of sd=0.05; row (c) the data with the Gaussian noise of sd=0.1.

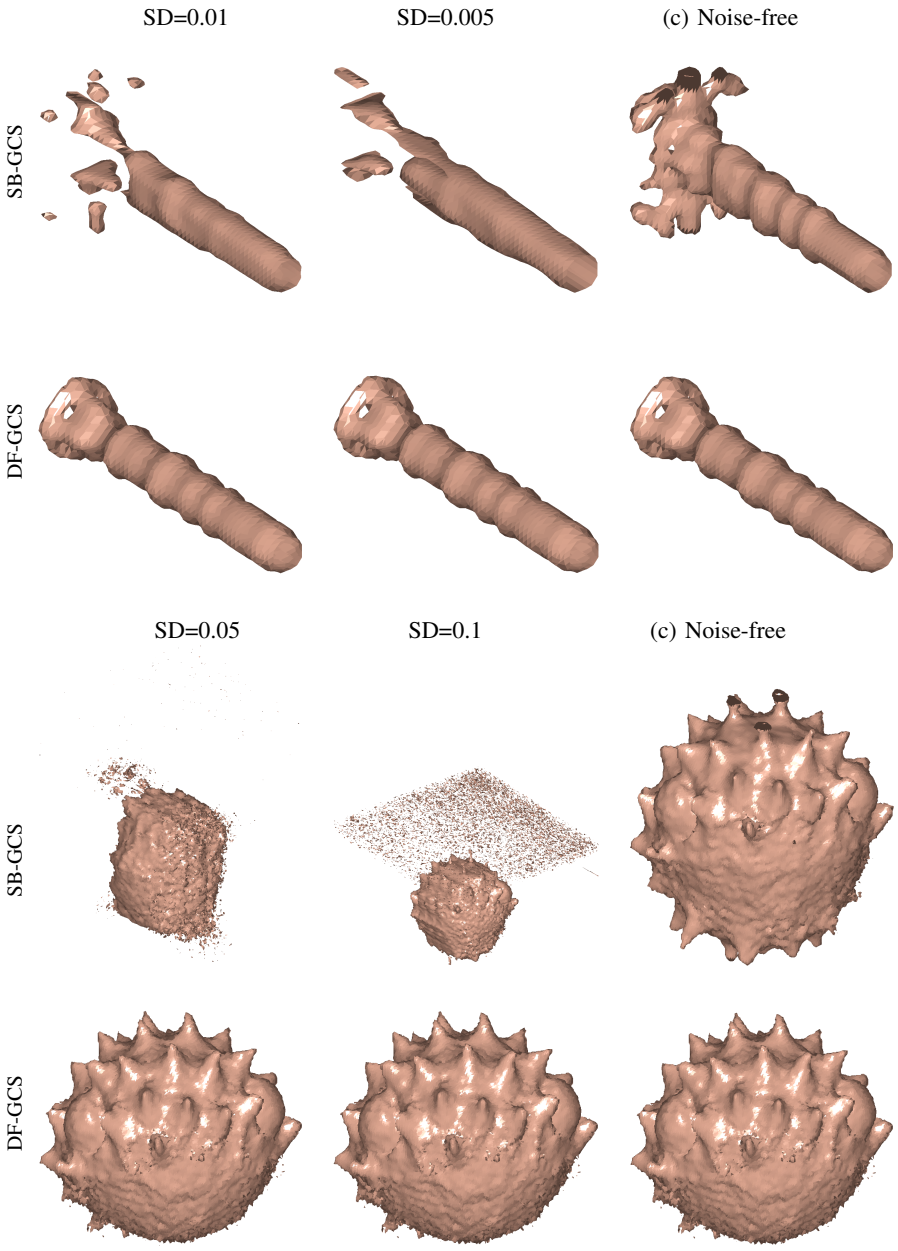


Fig. 25: Results using SB-GCS and DF-GCS for the Daisy and Fuel volumes. Column 1: the data with the Gaussian noise of sd=0.01 (Fuel Injection) and sd=0.05 (Daisy); Column 2: the data with the Gaussian noise of sd=0.05 (Fuel Injection) and sd=0.1 (Daisy); Column 3: Noise free.

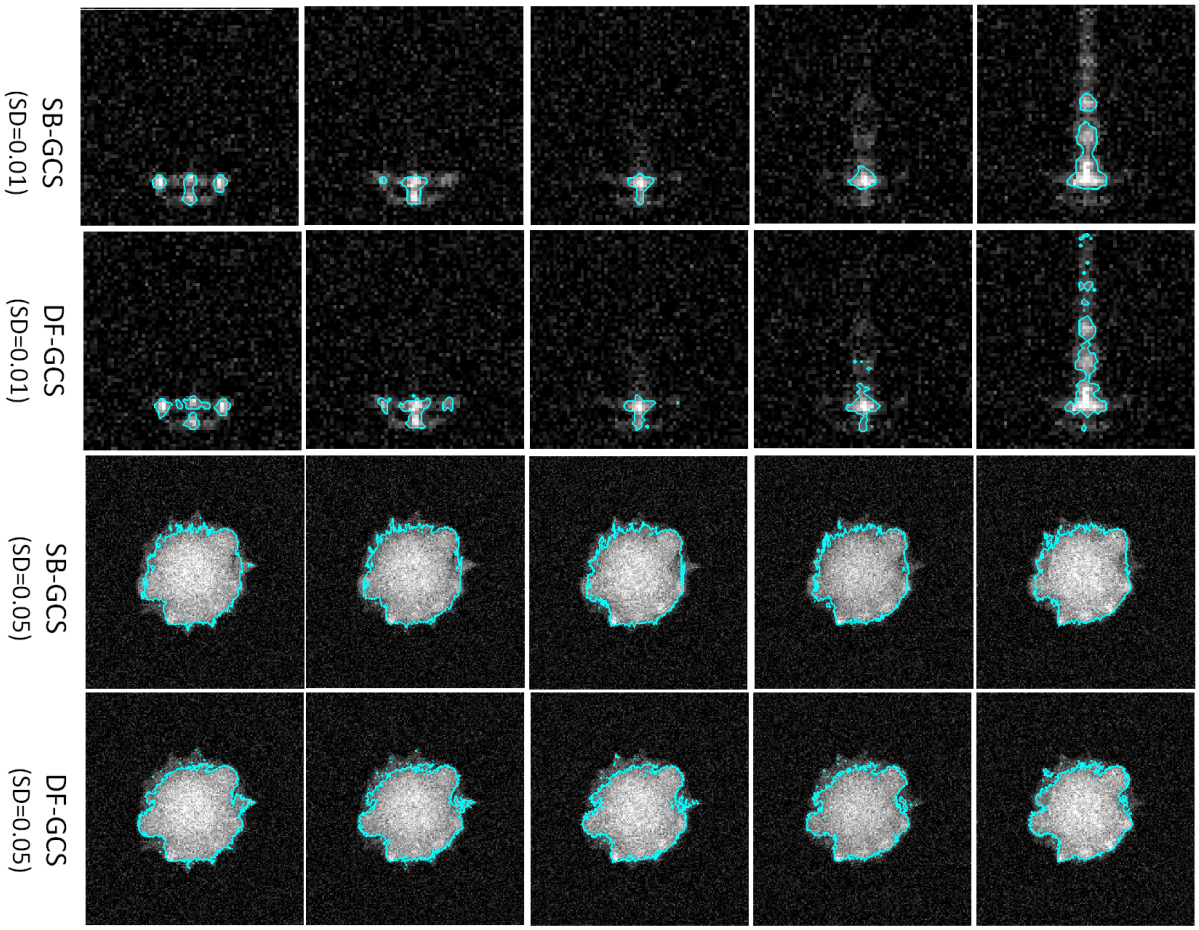


Fig. 26: Extracted boundaries using SB-GCS and DF-GCS in a serial of slices of the Daisy and Fuel volumes. Rows 1 and 2: Slices 24-28 with the Gaussian noise of sd=0.01 (Fuel Injection); Rows 3 and 4: Slices 101-104 with the Gaussian noise of sd=0.05 (Daisy).

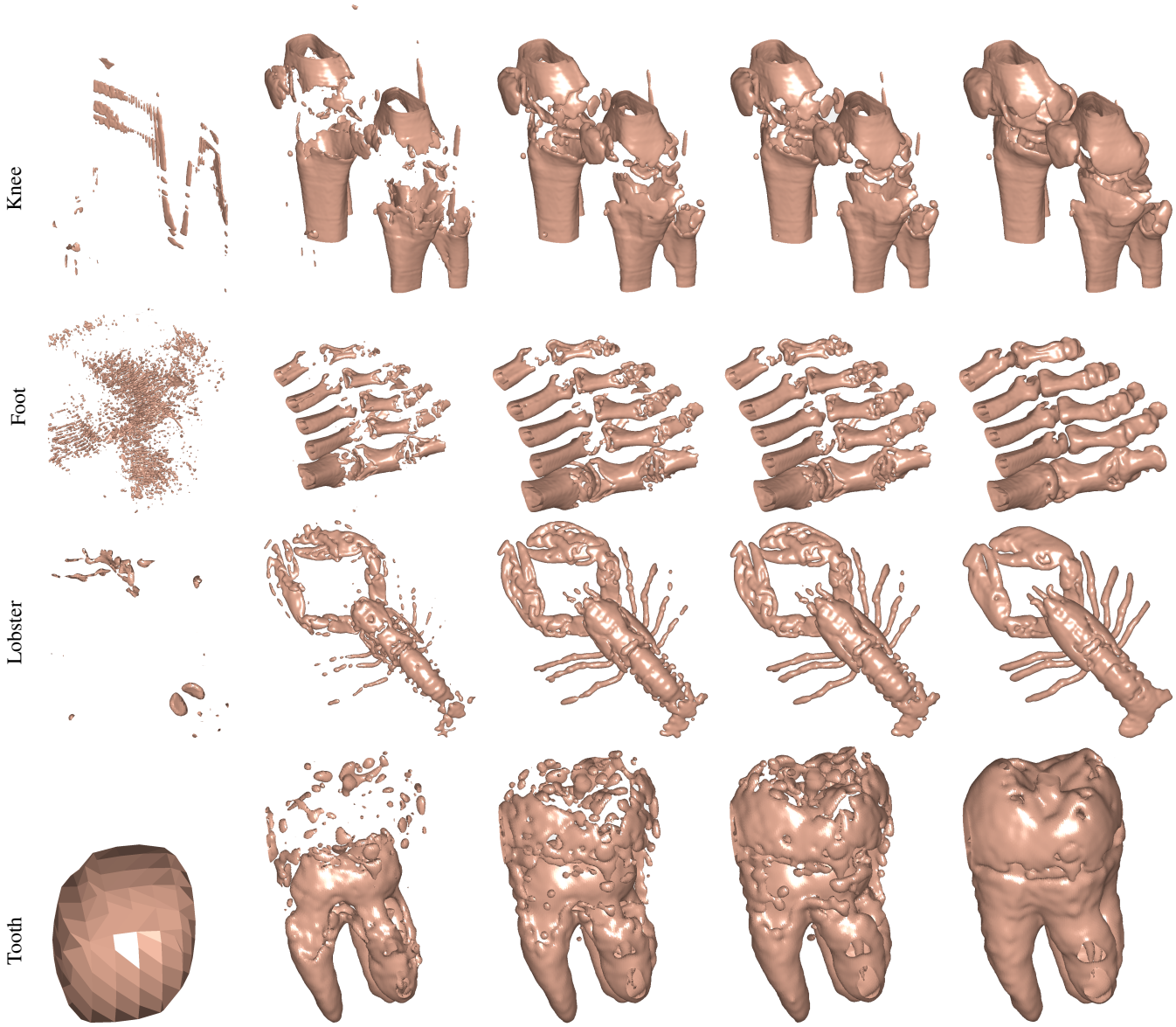


Fig. 27: Results using the proposed method in the biomedical datasets of the Knee (379*229*305), Foot (256*256*256), Lobster (301*324*56) and Tooth (92*78*161). The final column is the converged result. The first four columns are the intermediate results. Row 1: 1^{th}, 2^{th}, 7^{th}, 9^{th} and 25^{th} iterations; row 2: 1^{th}, 2^{th}, 3^{th}, 4^{th} and 25^{th} iterations; row 3: 1^{th}, 2^{th}, 4^{th}, 5^{th} and 15^{th} iterations; and row 4: 1^{th}, 2^{th}, 3^{th}, 4^{th} and 30^{th}.

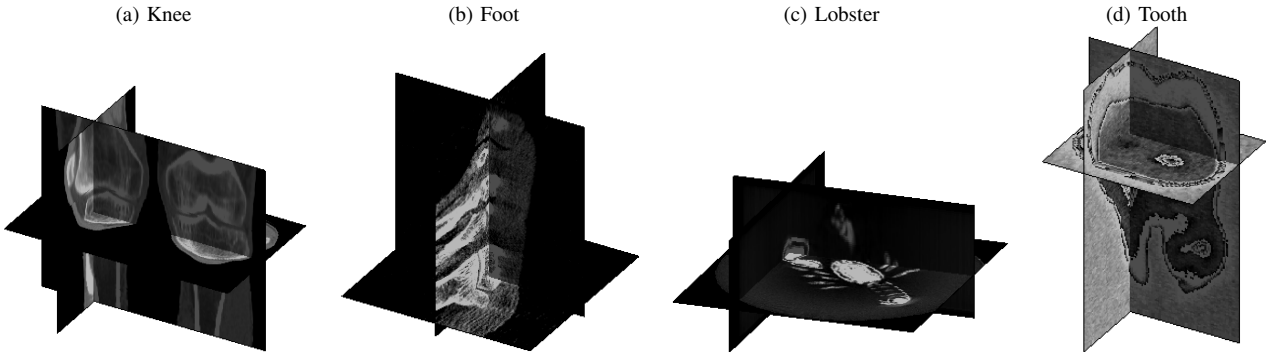


Fig. 28: Sagittal, coronal and transverse slices of an example in the volume data. (a) Knee: slice 90, slice 110 and slice 152; (b) Foot: slice 128, slice 128 and slice 18; (c) Lobster: slice 90, slice 200 and slice 28; (d) Tooth: slice 26, slice 50 and slice 100.

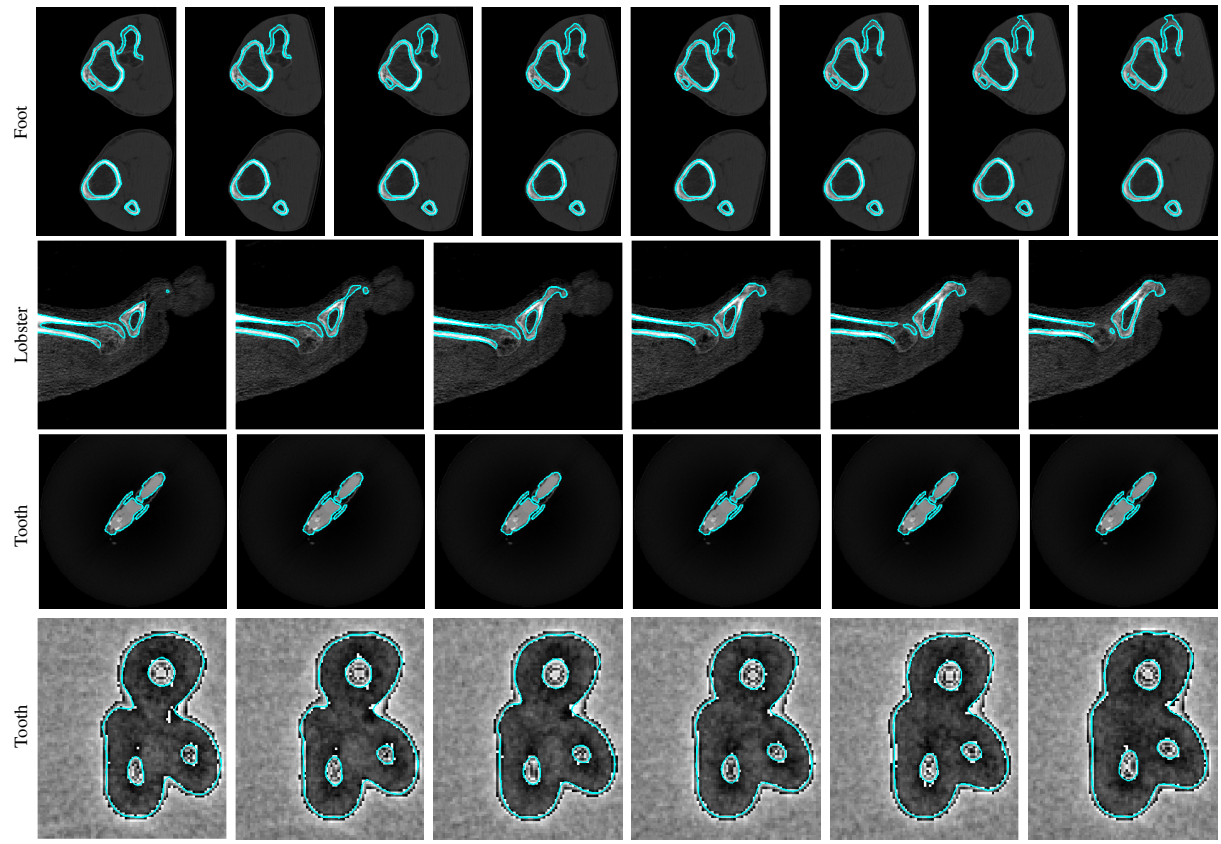


Fig. 29: Extracted boundaries in a serial of slices using the proposed method in the Knee, Foot, Lobster and Tooth datasets. Row 1: Slices 101-108 in the Knee; row 2: Slices 103-108 in the Foot; row 3: Slices 43-48 in the Lobster; and row 4: Slices 71-76 in the Tooth.

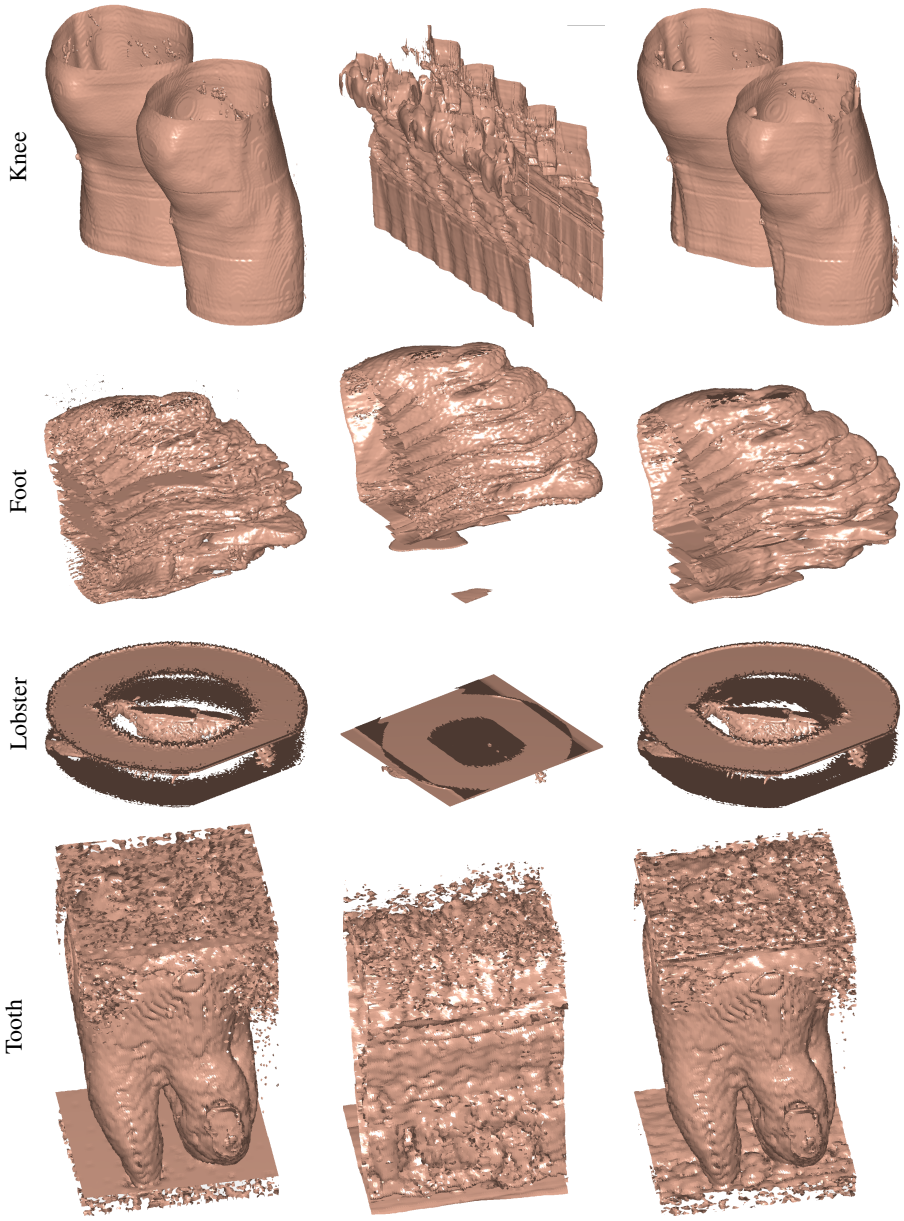


Fig. 30: Results using SB-GCS in the Knee, Foot, Lobster and Tooth datasets. Row 1: fidelity values: e-2,e-4,e-3 (final); row 2: fidelity values: e-3, e-5,e-4 (final) ; row 3: fidelity values: e-1,e-3, e-2 (final); and row 4: fidelity values: e-2, e-4, e-3 (final).

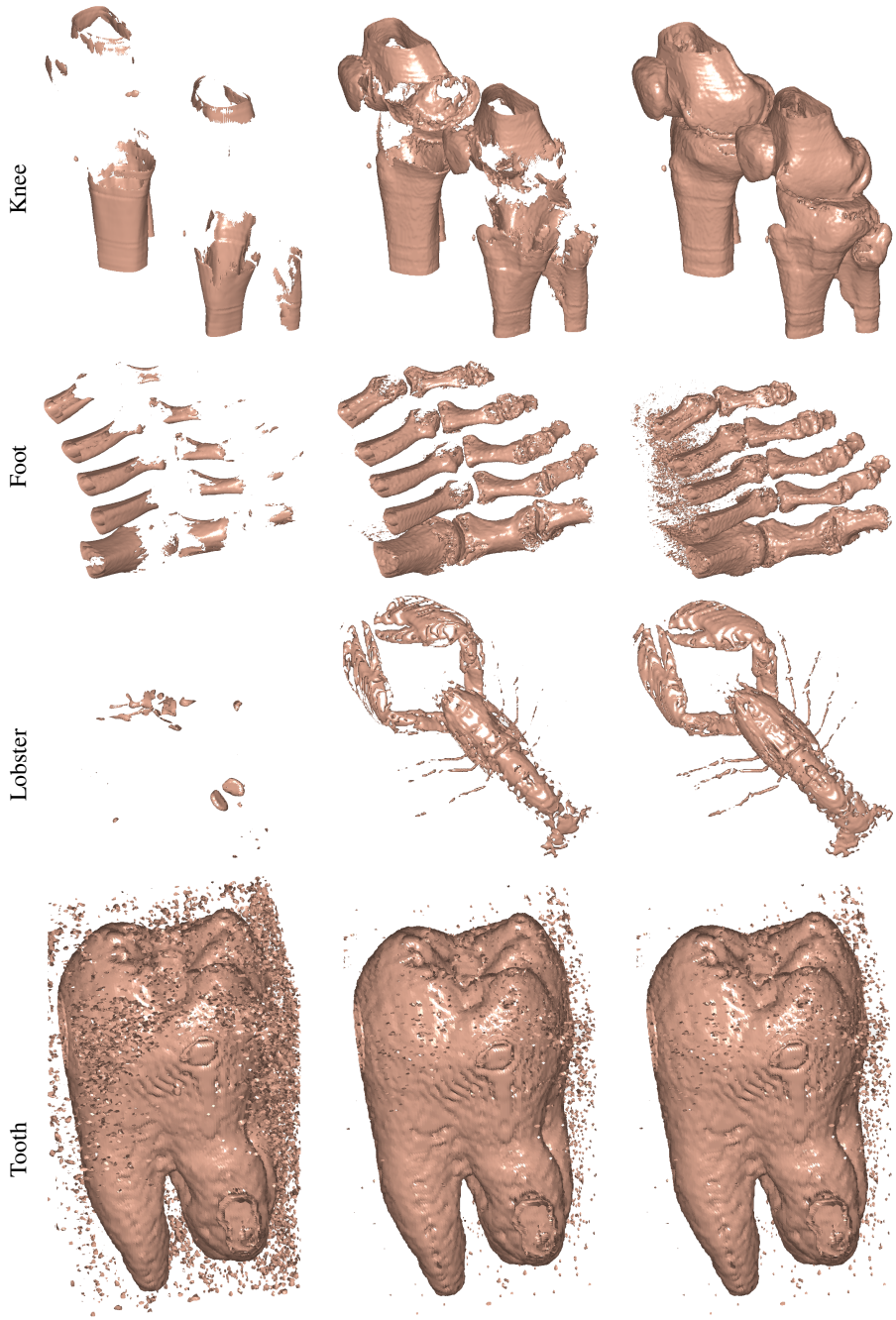


Fig. 31: Results using DF-GCS in the Knee, Foot, Lobster and Tooth datasets. Row 1: Iteration 1, iteration 2, final (Iteration 20); row 2: Iteration 1, iteration 2, final (Iteration 5); row 3: Iteration 1, iteration 2, final (Iteration 20); and row 4: Iteration 1, iteration 5, final (Iteration 10).

For further demonstrating its ability in surface reconstruction, the proposed method is applied to several biomedical real datasets. In Fig. 27, four rows correspond to the datasets from Knee, Foot, Lobster and Tooth scans respectively while each row includes four intermediate results and the converged one. These recovered surfaces show that our method is capable of capturing the complex structures of the objects in biomedical scan volumes. Furthermore, we illustrate the sagittal, coronal and transverse slices of an example for each dataset in Fig. 28 that present the spatial view of the datasets. For showing the performance of our method in a single slice, Fig. 29 presents the extracted boundaries in a series of slices for each example after surface reconstruction. From the presentation of Fig. 27, Fig. 28 and Fig. 29, it can be clearly seen that our method recovers the object surfaces and spatial structures very well even though the noise and artefacts are seriously in some cases. However, as discussed in the previous section, Fig. 30 and Fig. 31 show that both SB-GCS and DF-GCS to some extent are frustrated to recover the surface information in these cases because they are quite sensitive to low contrast and artefacts. In particular, SB-GCS meets great difficulties when object boundaries are surrounded by artefacts as presented in Fig. 17.

IV. Conclusion

We proposed a novel gradient convolution field GCF, whose divergence can be effectively used for active contouring. The proposed deformable model based on GCF can be naturally extended to 3D. Furthermore, a global solution scheme of the deformable model is presented using convex relaxation. For dealing with image noise interference and possible texture variations, a nonlinear diffusion method is proposed to diffuse GCF. Compared to post-processing based approach used in the MAC model, the proposed approach is much more robust and efficient. In addition, the implementation of an intrinsic level set regularization allowed the deformable model perform in a similar way to the global solution scheme. Experimental results in both 2D and 3D have demonstrated superior performance compared to the state-of-the-art methods with the global minimizer.

Future work will go to explore in two aspects. One is to apply our solution scheme in more medical image modalities and present the real application with its robustness and effectiveness of structural reconstruction in practice. Another possible improvement is that the current diffusion scheme in 3D is resorted to the isotropic diffusion in accordance with the geometric property of GCF in 3D. Although the current diffusion theory has not referred to this topic, is it feasible to extend the nonlinear diffusion to 3D so that the diffusion scheme is more efficient and effective? This is a promising way to do more investigation in the proposed framework.

Appendix A
The relationship between the magnetic field $\mathbf{B}$ and the gradient convolution field GCF $\mathbf{E}$

The magnetic field $\mathbf{B}$ has the relationship with its magnetic vector potential $\mathbf{A}(\mathbf{x})$ as follows:

$$\mathbf{B}(\mathbf{x}) = \nabla \times \mathbf{A}(\mathbf{x}), \quad \mathbf{A}(\mathbf{x}) = \frac{\mu_0}{4\pi} \sum_{\mathbf{s} \neq \mathbf{x}} f(\mathbf{s}) \frac{\mathbf{O}(\mathbf{s})}{R_{\mathbf{xs}}}. \tag{36}$$

The magnetic vector potential in the image plane can be similarly decomposed into two orthogonal terms in the image coordinates and zero component in the coordinate perpendicular to the image plane, i.e. $(A_i(\mathbf{x}), A_j(\mathbf{x}), 0)$:

$$A_i(\mathbf{x}) = \frac{\mu_0}{4\pi} \sum_{\mathbf{s} \neq \mathbf{x}} f(\mathbf{s}) \frac{-\hat{I}_y(\mathbf{s})}{R_{\mathbf{xs}}}, \; A_j(\mathbf{x}) = \frac{\mu_0}{4\pi} \sum_{\mathbf{s} \neq \mathbf{x}} f(\mathbf{s}) \frac{\hat{I}_x(\mathbf{s})}{R_{\mathbf{xs}}}, \tag{37}$$

where we consider $\lambda = 1$ (see (1)). It does not make any theoretical difference if $\lambda = 2$ is used, which simply leads to $\mathbf{B}(\mathbf{x}, \lambda = 2) = -\mathbf{B}(\mathbf{x}, \lambda = 1)$. Obviously, $E_i = A_j$ and $E_j = -A_i$. Then, the following can be derived:

$$\begin{aligned} \mathbf{B}(\mathbf{x}) &= \nabla \times \mathbf{A}(\mathbf{x}) \\ &= (0, 0, \frac{\partial A_j}{\partial x} - \frac{\partial A_i}{\partial y}) \\ &= (0, 0, \frac{\partial E_i}{\partial x} + \frac{\partial E_j}{\partial y}) \\ &= (0, 0, \nabla \cdot \mathbf{E}) \end{aligned} \tag{38}$$

Thus, we have: $B = \nabla \cdot \mathbf{E}$

Appendix B
Proof of Theorem 1

Based on the coarea formula, since $u \in [0, 1]$, we have the first term in (27):

$$\int_\Omega |\nabla(u)| d\mathbf{x} = \int_0^1 Per(\{x : u(\mathbf{x}) > \mu\}; \Omega) \mathrm{d}\mu, \tag{39}$$

where Ω is image domain.

The second term in (27) stands for the weighted area term and then we can process as follows:

$$\begin{aligned} \int_\Omega (\mathbf{B}(\mathbf{x}))\, u(\mathbf{x}) \mathrm{d}\mathbf{x} &= \int_\Omega \mathbf{B}(\mathbf{x}) \int_0^1 \mathit{1}_{[0, u(\mathbf{x})]}(\mu) \mathrm{d}\mathbf{x} \mathrm{d}\mu \\ &= \int_0^1 \int_\Omega \mathbf{B}(\mathbf{x}) \mathit{1}_{[0, u(\mathbf{x})]} \mathrm{d}\mathbf{x} \mathrm{d}\mu \\ &= \int_0^1 \int_{\Omega \cap \{\mathbf{x} : u(\mathbf{x}) > \mu\}} \mathbf{B}(\mathbf{x}) \mathrm{d}\mathbf{x} \mathrm{d}\mu \\ &= \int_0^1 \int_{\Sigma(\mathbf{x})} \mathbf{B}(\mathbf{x}) \mathrm{d}\mathbf{x} \mathrm{d}\mu, \end{aligned} \tag{40}$$

where $\mathit{1}_{\mathbf{S}}$ is the characteristic function of a set $\mathbf{S}$.

Thus, we have:

$$\begin{aligned} \mathcal{E}_1 &= \alpha \int_\Omega |\nabla u| \mathrm{d}\mathbf{x} + (1 - \alpha) \int_\Omega (\mathbf{B}(\mathbf{x}))\, u(\mathbf{x}) \mathrm{d}\mathbf{x} \\ &= \alpha \int_0^1 Per(\Sigma(\mu); \Omega) \mathrm{d}\mu + (1 - \alpha) \int_0^1 \int_{\Sigma(\mu)} \mathbf{B}(\mathbf{x}) \mathrm{d}\mathbf{x} \mathrm{d}\mu \\ &= \int_0^1 \mathcal{E}(\Sigma(\mu)) \mathrm{d}\mu. \end{aligned} \tag{41}$$

It agrees that if $u(\mathbf{x})$ is a minimizer of the convex problem (27), then for a.e. $\mu \in [0, 1]$, the set $\Sigma(\mu)$ has to be a minimizer of the energy functional $\mathcal{E}$ (23).

Note that the energy functional (23) can be expressed:

$$\mathcal{E} = \alpha \int_\Omega |\nabla \mathcal{H}(\Phi)| \mathrm{d}\mathbf{x} + (1 - \alpha) \int_{\Sigma_0} (\mathbf{B}(\mathbf{x}))\, \mathcal{H}(\Phi) \mathrm{d}\mathbf{x} \tag{42}$$

$$= \alpha Per(\Sigma_0; \Omega) \mathrm{d}\mu + (1 - \alpha) \int_{\Sigma_0} \mathbf{B}(\mathbf{x}) \mathrm{d}\mathbf{x} \mathrm{d}\mu, \tag{43}$$

where $\Sigma_0 = \{x : u(x) \geq 0\}$.

APPENDIX C
EQUIVALENCE BETWEEN THE PROPOSED SURFACE MODEL AND GPF

In the following, we prove that the total potential field G in the GPF model is equal to the potential field B in the proposed model.

Let $\mathbf{x}$ and $\mathbf{x}'$ denote the position variables, $R_{\mathbf{x}\mathbf{x}'} = |\mathbf{x}-\mathbf{x}'|$ is their distance and v' is the unit volume element in image domain Ω.

$$\begin{aligned} E_i(\mathbf{x}) &= \nabla_i I * k(\mathbf{x}) \\ &= \iiint_{\Omega} \nabla_i I(\mathbf{x}') k(\mathbf{x}-\mathbf{x}') \mathrm{d}\mathbf{v}' \\ &= \iiint_{\Omega} \frac{\nabla_i I(\mathbf{x}')}{R_{\mathbf{x}\mathbf{x}'}(\mathbf{x}-\mathbf{x}')} \mathrm{d}\mathbf{v}' \\ &= \iiint_{\Omega} \frac{\nabla_i I(\mathbf{x}')}{\sqrt{(x-x')^2+(y-y')^2+(z-z')^2}} \mathrm{d}\mathbf{v}' \end{aligned} \tag{44}$$

$$\begin{aligned} \frac{\partial E_i(\mathbf{x})}{\partial x} &= \iiint_{\Omega} \frac{\partial \sqrt{(x-x')^2+(y-y')^2+(z-z')^2}}{\partial x} \nabla_i I(\mathbf{x}') \mathrm{d}\mathbf{v}' \\ &= -\iiint_{\Omega} (x-x')((x-x')^2+(y-y')^2+(z-z')^2)^{-\frac{3}{2}} \nabla_i I(\mathbf{x}') \mathrm{d}\mathbf{v}' \\ &= -\iiint_{\Omega} \frac{(x-x')\nabla_i I(\mathbf{x}')}{R^3_{\mathbf{x}\mathbf{x}'}(\mathbf{x}-\mathbf{x}')} \mathrm{d}\mathbf{v}' \end{aligned} \tag{45}$$

In a similiar way, we get:

$$\frac{\partial E_j(\mathbf{x})}{\partial y} = -\iiint_{\Omega} \frac{(y-y')\nabla_j I(\mathbf{x}')}{R^3_{\mathbf{x}\mathbf{x}'}(\mathbf{x}-\mathbf{x}')} \mathrm{d}\mathbf{v}' \tag{46}$$

$$\frac{\partial E_k(\mathbf{x})}{\partial z} = -\iiint_{\Omega} \frac{(z-z')\nabla_k I(\mathbf{x}')}{R^3_{\mathbf{x}\mathbf{x}'}(\mathbf{x}-\mathbf{x}')} \mathrm{d}\mathbf{v}' \tag{47}$$

Thus, we have:

$$\begin{aligned} B &= \nabla \cdot E(\mathbf{x}) \\ &= \frac{\partial E_i}{\partial x} + \frac{\partial E_j}{\partial y} + \frac{\partial E_k}{\partial z} \\ &= -\iiint_{\Omega} \frac{(\mathbf{x}-\mathbf{x}')}{R^3_{\mathbf{x}\mathbf{x}'}(\mathbf{x}-\mathbf{x}')} \nabla_k I(\mathbf{x}') \mathrm{d}\mathbf{v}' \\ &= -\iiint_{\Omega} \nabla I(\mathbf{x}') \mathbf{K}_{\lambda}(\mathbf{x}-\mathbf{x}') \mathrm{d}\mathbf{v}' \\ &= -G, \end{aligned} \tag{48}$$

where $\mathbf{K}_{\lambda}(\cdot)$ is the vector kernel in the GPF model.

The magnitudes of both the potentials are the same and therefore the two surface models are equivalent because the derived force based on the potential can alternatively choose the inward/outward unit normal directions for surface evolution.

REFERENCES

[1] M. Kass, A. Witkin, and D. Terzopoulus, "Snakes: Active contour model," *IJCV*, vol. 1, no. 4, pp. 321–331, 1988.

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

[3] D. Cremers, M. Rousson, and R. Deriche, "A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape," *IJCV*, vol. 72, no. 2, pp. 195–215, 2007.

[4] C. Xu and J. Prince, "Generalized gradient vector flow external forces for active contours," *Signal Processing*, vol. 71, no. 2, pp. 131–139, 1998.

[5] R. Kimmel, "Fast edge integration," in *Geometric Level Set Methods in Imaging, Vision, and Graphics*. Springer, 2003, pp. 59–77.

[6] A. Vasilevskiy and K. Siddiqi, "Flux maximizing geometric flows," *T-PAMI*, vol. 24, no. 12, pp. 1565–1578, 2002.

[7] N. Paragios, O. Mellina-Gottardo, and V. Ramesh, "Gradient vector flow fast geometric active contours," *T-PAMI*, vol. 26, no. 3, pp. 402–407, 2004.

[8] X. Xie and M. Mirmehdi, "MAC: Magnetostatic active contour," *T-PAMI*, vol. 30, no. 4, pp. 632–646, 2008.

[9] A. Jalba, M. Wilkinson, and J. Roerdink, "CPM: A deformable model for shape recovery and segmentation based on charged particles," *T-PAMI*, vol. 26, pp. 1320–1335, 2004.

[10] R. Yang, M. Mirmehdi, and X. Xie, "A charged active contour based on electrostatics," in *ACIVS*, 2006, pp. 173–184.

[11] H. Chung and M. Chung, "External force of snake: virtual electric field," *ELL*, vol. 38, no. 24, pp. 1500–1502, 2002.

[12] Y. Xiang, A. Chung, and J. Ye, "An active contour model for image segmentation based on elastic interaction," *J. Comp. Phy.*, vol. 219, no. 1, pp. 455–476, 2006.

[13] B. Li and T. Acton, "Active contour external force using vector field convolution for image segmentation," *T-IP*, vol. 16, no. 8, pp. 2096–2106, 2007.

[14] J. Weickert, B. Romeny, and M. Viergever, "Efficient and reliable scheme for nonlinear diffusion filtering," *T-IP*, vol. 7, no. 3, pp. 398–410, 1998.

[15] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *T-PAMI*, vol. 12, no. 7, pp. 629–639, 1990.

[16] U. Diewald, T. Preuber, and M. Rumpf, "Anisotropic diffusion in vector field visualization on euclidean domains and surfaces," *T-VCG*, vol. 6, no. 2, pp. 139–149, 2000.

[17] K. Nordström, "Biased anisotropic diffusion - a unified regularization and diffusion approach to edge detection," in *ECCV*, 1990, pp. 18–27.

[18] T. Chan and L. Vese, "Active contours without edges," *T-IP*, vol. 10, no. 2, pp. 266–277, 2001.

[19] C. Li, C. Xu, C. Gui, and M. Fox, "Level set evolution without re-initialization: A new variational formulation," in *CVPR*, 2005, pp. 430–436.

[20] X. Xie, "Active contouring based on gradient vector interaction and constrained level set diffusion," *T-IP*, vol. 19, no. 1, pp. 154–164, 2010.

[21] G. Strang, "Maximal flow through a domain," *J. Math. Progam.*, vol. 26, pp. 123–143, 1983.

[22] T. Chan, S. Esedoglu, and M. Nikolova, "Algorithms for finding global minimizers of image segmentation and denoising models," *SIAM J. Appl. Math.*, vol. 66, no. 5, pp. 1632–1648, 2006.

[23] M. Burger and M. Hintermuller, "Projected gradient flows for bv/level set relaxation," *PAMM*, vol. 5, pp. 11–14, 2005.

[24] M. Sussman, P. Smereka, and S. Osher, "A level set approach for computing solutions to incompressible two-phase flow," *J. Comput. Phys.*, vol. 114, pp. 146–159, 1994.

[25] K. Zhang, L. Zhang, H. Song, and W. Zhou, "Active contours with selective local or global segmentation: A new formulation and level set method," *J. Imag. Visi. Comput.*, vol. 28, pp. 668–676, 2010.

[26] S. Yeo, "Geometrically induced force interaction for three-dimensional deformable models," *T-IP*, vol. 20, no. 5, pp. 1373–1387, 2011.

[27] X. Bresson, S. Esedoglu, P. vandergheynst, J. Thiran, and S. Osher, "Fast global minimization of the active contour/snake model," *J. Math. Imaging Vis.*, vol. 28, pp. 151–167, 2007.

[28] T. Goldstein, X. Bresson, and S. Osher, "Geometric applications of the split bregman method: Segmentation and surface reconstruction," *J. Sci. Comput.*, vol. 45, pp. 272–293, 2010.

[29] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM J. Imag. Sci.*, vol. 1, no. 3, pp. 248–272, 2008.

[30] T. Goldstein and S. Osher, "The split bregman method for l1 regularized problems." *UCLA CAM report*, vol. 08, no. 29, 2008.

[31] K. P. Schwarz, M. G. Sideris, and R. Forsberg, "The use of fft techniques in physical geodesy," *Geophys. J. Int.*, vol. 100, pp. 485–514, 1990.