Divergence of Gradient Convolution: Deformable Segmentation with Arbitrary Initializations

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

***Abstract*—In this paper, we propose a unified approach to deformable model based segmentation. The fundamental force field of the proposed method is based on computing the divergence of a gradient convolution field (GCF), which makes full use of directional information of the image gradient vectors and their interactions across image domain. However, instead of directly using such a vector field for deformable segmentation as in conventional approaches, we derive a more salient representation for contour evolution, and very importantly we demonstrate that this representation of image force field not only leads to global minimum through convex relaxation but also can achieve the same result using conventional gradient descent with an intrinsic regularization. Thus, the proposed method can handle arbitrary initializations. The proposed external force field for deformable segmentation has both edge-based properties in that GCF is computed from image gradients, and region-based attributes since its divergence can be treated as a region indication function. Moreover, nonlinear diffusion can be conveniently applied to GCF to improve its performance in dealing with noise interference. We also show the extension of GCF from 2-D to 3-D. In comparison to state-of-the-art deformable segmentation techniques, the proposed method shows greater flexibility in model initialization and optimization realization, as well as better performance toward noise interference and appearance variation.**

***Index Terms*—Deformable models, gradient convolution divergence, initialization invariance, global minimization, nonlinear diffusion, 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 have been a multitude of works on the improvement of these gradient based approaches. Often they are based on edge strength or magnitude of image gradient. However, gradient direction or orientation contains important information in describing object boundaries. For example, its spatial coherence can effectively suggest whether 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 are 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. However, it has convergence issues caused by saddle or stationary points in the vector field [5], [6], e.g. the active contour will not be able to propagate further once it becomes tangent to the underlying gradient vectors. Kimmel in [7] 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. This alignment is carried out locally and it may require careful tuning to balance various forces. In [8], 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. This active contour model, however, is not suitable for general image segmentation, where flux variation generally exists inside object regions.

Recently, there have been a number of 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 leads to broken contours. 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 [6], instead of assigning fixed charges, the authors allowed the charges flow through the edges which generates a magnetic field that can be used as an external force to drive deformable contours that carry similar flow of charges. The MAC model showed significant improvements on the convergence issues, e.g. reaching deep concavities, and in handling weak edges and broken boundaries [6]. 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. Yeo *et al.* [13] presented a generalized formulation of the MAC model to 3D. However, the presence of image noise will inevitably disturb the gradient vectors and cause irregularities in the estimated edge orientation, which in turn affects the magnetic field computation. The authors in [6] proposed a scalar field diffusion to refine the magnetic

H. Zhang and X. Xie are with the Department of Computer Science, Swansea University, Swansea SA2 8PP, United Kingdom. E-mail: x.xie@swansea.ac.uk

field. However, this approach is problematic since there is a great risk of positive and negative magnetic flux canceling out each other which prevents the desired diffusion at strong edges from propagating into noise dominant regions. The denoising strategy proposed in [12] requires either multiple trial segmentations or independent contour shrinking.

The solution schemes used in deformable model based segmentation are typically nonconvex and are thus prune to local minima. Recently, a number of schemes were proposed to formulate convex energy functional by regularizing model parameters and designing appropriate objective function that involves, for instance, prescribing the elasticity and rigidity of the deforming contours, e.g. [14]–[18]. In [14], based on the earlier work of [19], the authors proposed to use the upper level sets to formulate the functionals with weighted co-area formula and layer-cake formula [20]. The minimization of the functionals is carried out in a way such that the minimizer of the derived functionals can be subsequently interpreted into the minimizers of the original deformable model (such as the Chan-Vese model [21]) by using a thresholding scheme. Thus, a nonconvex deformable problem can be globally solved using a standard convex minimization scheme, i.e. L^1-norm convex energy minimization. Inspired by this work, several global minimization schemes have been proposed, notably [15]–[17] for deformable segmentation. These methods allowed the unification of the conventional edge based geodesic model and region based Chan-Vese model, and they have been shown outperforming the method presented in [14]. In the case of known piecewise constant values, the authors in [15], [16] proposed different numerical schemes that can solve the non-convex Chan-Vese problem globally, but their overall performances can vary substantially due to difference in numerical schemes, i.e. a dual formulation of the TV-norm for fast minimization is used in [15] (DFCV) and [16] (SBCV) applies the split Bregman method [22], [23] for overcoming the drawbacks, such as penalty issue and expensive computation, of the previous globally regularized schemes [14], [15] . On the other hand, in order to deal with the issue of unknown piecewise constant values, the authors in [17] presented a completely convex formulation (GCCV) that the non-convex Chan-Vese problem is solved in a higher-dimensional space by applying a TV-based vector-valued minimization scheme [16], [24]. This method however is computationally expensive.

In this paper, a novel deformable model is proposed based on a gradient convolution field that makes the full use of image gradient information. This vector field is unique in that its divergence can be used directly as an image feature to drive deformable models to their global solution using conventional gradient descent optimization with modified level set regularization. Segmentation result can be independent to its initialization. More formally, the energy functional of this model can be described by a total variation flow and a weighted area minimization flow. Thus, using convex relaxation, a global minimizer is proposed to solve the energy functional to achieve initialization independence. Furthermore, this vector field can be effectively enhanced using tensor diffusion to deal with noise interference. The model can be conveniently extended to higher dimensional space with substantially less computational overhead, compared to some other convex formulation, e.g. [17]. We compare our method against MAC and a number of state-of-the-art global minimizers, i.e. [15]–[17], on both synthetic and real world data in 2D and 3D.

The main contributions of this work can be summarized as:

- We propose to use the GCF and its divergence as an image feature, which can be considered as a region indicator and has explicit relationship to the MAC model.
- A deformable model is derived using the divergence of GCF as its external force. The 2D evolving curves can be naturally generalized to higher dimensional spaces, e.g. active surfaces in 3D.
- A tensor diffusion scheme is proposed to diffuse the GCF in order to improve the performance of the deformable model, where the edges are better preserved and image noise can be more efficiently removed even though they may appear far from strong edges.
- Even with conventional gradient descent minimization, the proposed method can achieve initialization invariance in practice by using intrinsic level set regularization.
- The global minimizer of the proposed deformable model is achieved by applying convex relaxation to formulate the energy functional, where the level set implementation can be readily extended from 2-D to 3-D with small computational overhead.
- An intrinsic level set regularization scheme is alternatively used for performing its numerical implementation in 2-D, which is originally motivated from smoothing the transition between different level sets [21], [25]. Moreover, we show it is in fact functionally analogous to the global minimizer of the proposed model.

II. Proposed Method

Briefly, in Section II-A we derive the gradient convolution field (GCF) by convolving the image gradient field with an inverse distance kernel function and propose a deformable model based on the divergence of GCF. A nonlinear diffusion method is presented in Section II-B in order to refine GCF. To demonstrate the uniqueness of the GCF, both gradient descent and convex relaxation techniques are developed to minimize the energy functional. As shown in Sections II-C and II-D, both approaches lead to global solutions. Finally, in Section II-E, we extend the model to high dimensional spaces and provide the level-set solution scheme for the 3D case.

A. GCF and its deformable model

In [26], 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:

$$\mathbf{F}_v(\mathbf{x}) = f(\mathbf{x}) * \mathbf{k}(\mathbf{x}). \tag{1}$$

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 [26], and (1) can be in effect computed as:

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

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

As demonstrated in [9], [10], [26], these inverse distance kernel based techniques performed much better than conventional edge based deformable models in, for instance, reaching concavities. However, none of these methods takes into account the edge orientation or gradient direction in deriving the external force fields. Moreover, the deformable models are directly based on these force fields and since they are static the models inevitably suffer from convergence and initialization issues that have been well documented in the literature [5], [6]. 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. We also show that instead of using the vector field directly its divergence is far more robust and versatile in deformable segmentation.

Let I denote an image, $\nabla_i I = f\hat{I}_x$ 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], [26], 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:

$$\begin{cases} E_i(\mathbf{x}) = \nabla_i I * k(\mathbf{x}) = \sum_{\mathbf{s}\neq\mathbf{x}} \frac{\nabla_i I(\mathbf{s})}{R_{\mathbf{xs}}} = \sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s}) \frac{\hat{I}_x(\mathbf{s})}{R_{\mathbf{xs}}}, \\ E_j(\mathbf{x}) = \nabla_j I * k(\mathbf{x}) = \sum_{\mathbf{s}\neq\mathbf{x}} \frac{\nabla_j I(\mathbf{s})}{R_{\mathbf{xs}}} = \sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s}) \frac{\hat{I}_y(\mathbf{s})}{R_{\mathbf{xs}}}, \end{cases} \tag{3}$$

where $\mathbf{E} = (E_i, E_j)$ is referred to as the 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 (3) is across the whole image domain. It takes similar form to [26], 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 [26] and many other vector field based methods have suffered from as mentioned earlier. Instead, we show its fundamental relationship to the MAC model [6] 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.

In the MAC model [6], 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 $(\hat{I}_x, \hat{I}_y)$. The image plane is considered as a 2D plane in a 3D space whose origin coincides with the origin of the image coordinates and 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:

$$\mathbf{O}(\mathbf{x}) = (-1)^\lambda(-\hat{I}_y(\mathbf{x}), \hat{I}_x(\mathbf{x}), 0), \tag{4}$$

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 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:

$$\mathbf{B}(\mathbf{x}) = \frac{\mu_0}{4\pi} \sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s})\mathbf{O}(\mathbf{s}) \times \frac{\hat{\mathbf{R}}_{\mathbf{xs}}}{R^2_{\mathbf{xs}}}, \tag{5}$$

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:

$$\mathbf{F}_m(\mathbf{x}) \propto \boldsymbol{\Upsilon}(\mathbf{x}) \times \mathbf{B}(\mathbf{x}). \tag{6}$$

The magnetostatic active contour (MAC) model is then formulated as [6]:

$$C_t = \alpha g(\mathbf{x})\kappa\hat{\mathbf{N}} + (1 - \alpha)(\mathbf{F}_m(\mathbf{x}) \cdot \hat{\mathbf{N}})\hat{\mathbf{N}}, \tag{7}$$

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 is thus ignored.

As shown in Appendix A, the magnetic field $\mathbf{B}$ in (5) can be derived from the proposed GCF:

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

where B is the divergence of GCF, and actually is the third component of the magnetic field $\mathbf{B}$ which is the external force in the MAC model (7). In addition, the B value can be geometrically interpreted as the change rate of the GCF along the normal direction.

Thus, the divergence of GCF, i.e. B, can be seen as the evolution velocity (external force) for attracting the contour to the boundary. The contour evolution under B can be described as:

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

where $\hat{\mathbf{N}}$ is the unit inward normal. For contour smoothing, the curvature flow is added to (9). Thus, the deformable model, based on the proposed divergence of GCF, can be formulated as follows,

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

B. GCF tensor diffusion

Due to the inherent relationship between the evolution velocity B and GCF, i.e. $B = \nabla \cdot E$, we can enhance B by refining GCF, $\mathbf{E} = (E_i, E_j)$. By considering the directional role in diffusion process, 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 [6], which then leads to the proposed diffusion scheme.

Let us incorporate (6) into (7) and re-write (7) as:

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

where $\mathbf{M} = (\hat{\mathbf{N}}, 0) \times \mathbf{\Upsilon}$ 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 act 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 (3) 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 in 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.

Since $\mathbf{B} = \frac{\mu_0}{4\pi}(0, 0, B)$ as shown in (8), the authors in [6] argued diffusing the scalar field B in the image domain was sufficient:

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

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 object boundaries, while homogeneous areas will generally have small flux density due to a lack of support from edges, resulting in substantial diffusion.

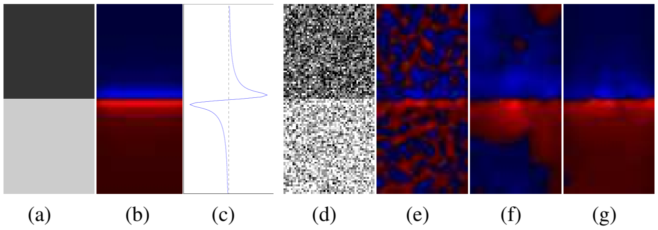

Fig. 1: (a) a test image contains a horizontal edge; (b) its divergence of GCF visualization; brightness corresponds to magnitude. Blue color indicates positive (source), and red indicates negative (sink). (c) a vertical profile of the divergence values. (d) a noisy image; (e) initial divergence; (f) diffusion result using [6]; (g) diffusion result ($\tilde{B}$) using the proposed method.

However, this diffusion has an unintended effect of canceling out positive and negative values on strong edges, which are supposed to be preserved. Let us consider a simple horizontal edge as shown in Fig. 1(a). From Figs. 1(b)&(c), we can see that the center of the edge has divergence value (or magnetic field) close to zero. Image gradient vectors on the opposite side of an edge produce opposing magnetic field. This means that the preserving term q is ineffective as the cancellation 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 the cancellation which restrains their ability to influence regions further away from edges. A smaller diffusion, however, may not meet the desired diffusion effect. Figs. 1(d) and 1(f) give an example when noise interference is strong. The diffusion in [6] is no longer effective.

For our proposed method, as described in (3), 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, e. g. [27], has been found useful in a range of applications, such as image denoising [28] and vector flow visualization [29]. Consider a class of nonlinear parabolic differential equations of the following kind:

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

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

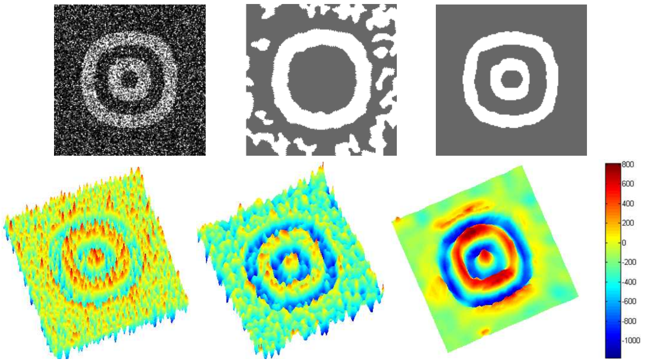

Fig. 2: First row from left: an noisy image, result obtained using the method in [6], result based on the proposed method. Second row from left: the initial B field, diffused B using Laplacian diffusion in [6], and finally the diffused result ($\tilde{B}$) using the proposed method.

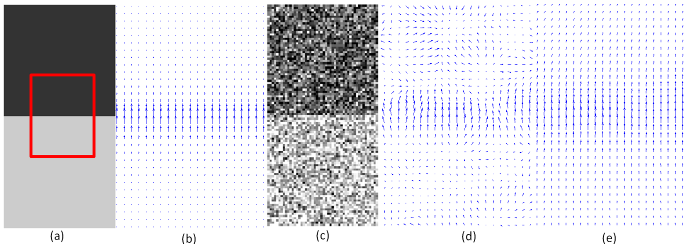

Fig. 3: Effectiveness of the proposed tensor diffusion. (a) an ideal image, (b) GCF of the clear image, (c) the noise corrupted version (d) GCF of the noisy image, (e) diffused GCF using the proposed method. The vector fields are zoomed in at the location highlighted in (a) for clear visualization.

nonlinear diffusion takes the following coupled form:

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

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 $(\cos\theta, \sin\theta)$. 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:

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

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 that the divergence and gradient operations are applied to each spatial component of $\boldsymbol{\mathcal{E}}$ separately.

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 the 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 [30]. 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 $\mathbf{E}$ provides a better attraction force, particularly in the presence of image noise and/or textures. Fig. 1 shows a comparison of the proposed diffusion against that used in [6] on a noise corrupted image. In Fig. 2, the original B field, the diffused B 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 [6] and the result using the proposed method. Also, note that the divergence of GVF is a signed scalar field (as shown in Fig. 1, red denotes sink/negative and blue denotes source/positive). A vertical profile of the divergence values clearly show that it exhibits characteristics of a region indication function. This makes it different to conventional edge based methods, although it is derived from simple image gradients. Thus, with GCF it is possible to achieve global solutions and initialization independence.

The following summarizes the uniqueness of proposed image feature. Firstly, the GCF computation takes into account both gradient direction and edge strength in a global fashion, unlike the previous methods that often only make use of edge strength such as VFC [31] or are merely based on local fittings of gradient information such as GVF [32]. Secondly, we employ tensor diffusion to refine the GCF to cope with excessive image noise. We apply linear diffusion in the direction of the vector field for enhancing the edge continuity and a Perona-Malik type diffusion orthogonal to the field for reducing the noise interference. Fig. 3 shows an example of the vector field diffusion and its effect on noisy data. Finally, unlike previous works such as GVF [32] and VCF [31], GCF is not directly used in energy minimization. Thus, the proposed method does not suffer from those various convergence issues that are well documented in the literature. The divergence of GCF provides a more reliable geometric feature to drive the contours towards object boundaries. In the following sections, we show two different approaches to energy minimization in order to show the unique properties of the proposed GCF.

C. Level set implementation with a regularized scheme

To demonstrate the uniqueness of the GCF, we provide a gradient descent optimization scheme with level set regularization that leads to global minima so that the proposed method can handle arbitrary initializations. It is worth noting that the conventional edge based techniques, such as [4], [5], [10], [26], can not achieve such results even using the proposed gradient descent method.

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 Ω is an open set representing the image domain and $\tilde{B}$ is the refined divergence of 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 [21], 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 [28] 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 [33], 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 [34] 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 [33] 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 minimizes steep surfaces and also smooths 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, i.e. initialization independent.

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 a signed distance function.

D. Global minimizer with convex relaxation

The energy functional (18) of the deformable model is homogeneous of degree one in the level set function (weighted length/area minimization flow) and the relevant minimization problem is non-convex, which cannot guarantee its gradient descent to a stationary state in a conventional scheme. Recent advances in optimization theory [19] allow us to apply convex functionals [14] in order to achieve global minima. In the following, we infer how the global minimizer of the derived deformable model can be achieved by following an inference fashion similar to [14].

Without loss of generality, we use a length minimization flow (total variation flow) instead of the previous weighted one in (18). The contour evolution can therefore be described by minimizing the following energy functional:

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

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

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

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

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

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

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

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. To address this issue, we restrict Φ in a closed set $[0, 1]$ for functional minimization. Thus, this can convert the energy functional (26) to a convex functional according to the following theorem.

Theorem 1:[1] The global minimizer for the energy functional (23) can be obtained by solving the following convex minimization in the image domain Ω:

$$min_{0\leq\Phi\leq 1}\{\mathfrak{E}_3 := \alpha\int_{\Omega}|\nabla\Phi(\mathbf{x})|\mathrm{d}\mathbf{x} + (1-\alpha)\int_{\Omega}\tilde{B}(\mathbf{x})\Phi(\mathbf{x})\mathrm{d}\mathbf{x}\} \tag{27}$$

where $\tilde{B}(\mathbf{x})$ is the diffused refined divergence of GCF ($\tilde{B} = \nabla\cdot\boldsymbol{\mathcal{E}}$) and then setting $\Sigma = \{\mathbf{x} : \Phi(\mathbf{x}) \geq \mu\}$ for a.e. $\mu \in [0, 1]$.

Following Theorem 1, the global minimizer of the deformable model (10) can be found by solving (27). In [35], 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:

$$\frac{\partial\Phi}{\partial t} = \begin{cases} max\{\alpha\nabla\cdot(\frac{\nabla\Phi}{|\nabla\Phi|}) - (1-\alpha)\tilde{B}, 0\} & if \quad \Phi = 0, \\ min\{\alpha\nabla\cdot(\frac{\nabla\Phi}{|\nabla\Phi|}) - (1-\alpha)\tilde{B}, 0\} & if \quad \Phi = 1, \\ \alpha\nabla\cdot(\frac{\nabla\Phi}{|\nabla\Phi|}) - (1-\alpha)\tilde{B} & otherwise, \end{cases} \tag{28}$$

where Φ is normalized to keep $0 \leq \Phi \leq 1$ during the curve evolution.

For numerical reasons, an isotropic Gaussian filtering of the level set function is used to constrain it close to an signed distance function (SDF) in the evolution, i.e. $\Phi_\sigma = \Phi * G_\sigma$, which is similar to the method proposed in [36]. This regularization based on two considerations. One is that the curvature term is degraded as a Laplacian function when the SDF condition ($|\nabla\Phi| = 1$) is satisfied. Thus, it is feasible that a Gaussian filtering is used instead of the Laplacian operation in (28) due to their equivalence [28]. The other is that the Gaussian filtering can help smooth the level set function while keeping it close to an SDF during the evolution. Consequently, the curvature term in (28) can be actually omitted because of the utilization of the Gaussian filtering in the algorithm implementation. Thus, (28) can be simplified as:

$$\frac{\partial\Phi}{\partial t} = \begin{cases} max\{-(1-\alpha)\tilde{B}, 0\} & if \quad \Phi = 0, \\ min\{-(1-\alpha)\tilde{B}, 0\} & if \quad \Phi = 1, \\ -(1-\alpha)\tilde{B} & otherwise. \end{cases} \tag{29}$$

It is worth noting that the regularized solution scheme (22) in Section II(C) is in fact functionally analogous to this global minimizer, where the LSF regularization is achieved by employing a Heaviside function to keep the SDF of the level set function in the context of local minimization that helps enhancing the capture range. The regularization in (22) improves the local minimization scheme presented in (19). With the regularized Heaviside function, the right side of (22) is approximately equivalent to (29) when $|\nabla\Phi| <= 1$. When $|\nabla\Phi| > 1$, the regularization in (22) becomes a Laplacian diffusion, which is similar to the isotropic Gaussian filtering of the curve front used in (29).

E. Active surfaces using GCF

In the higher dimensional cases, the GCF can be obtained by extending the definition in (3) 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:

$$\mathbf{E} = (E_{i_1}, ..., E_{i_n}) = (\nabla_{i_1} I * k(\mathbf{x}), ..., \nabla_{i_n} I * k(\mathbf{x})). \tag{30}$$

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

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

$$\begin{cases} E_i(\mathbf{x}) = \nabla_i I * k(\mathbf{x}) = \sum_{\mathbf{s}\neq\mathbf{x}} \frac{\nabla_i I(\mathbf{s})}{R^2_{\mathbf{xs}}} = \sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s})\frac{\hat{I}_x(\mathbf{s})}{R^2_{\mathbf{xs}}}, \\ E_j(\mathbf{x}) = \nabla_j I * k(\mathbf{x}) = \sum_{\mathbf{s}\neq\mathbf{x}} \frac{\nabla_j I(\mathbf{s})}{R^2_{\mathbf{xs}}} = \sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s})\frac{\hat{I}_y(\mathbf{s})}{R^2_{\mathbf{xs}}}, \\ E_k(\mathbf{x}) = \nabla_k I * k(\mathbf{x}) = \sum_{\mathbf{s}\neq\mathbf{x}} \frac{\nabla_k I(\mathbf{s})}{R^2_{\mathbf{xs}}} = \sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s})\frac{\hat{I}_z(\mathbf{s})}{R^2_{\mathbf{xs}}}. \end{cases} \tag{31}$$

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

$$B = \nabla\cdot\mathbf{E}(\mathbf{x}) = \nabla\cdot(E_i, E_j, E_k). \tag{32}$$

Thus, by combining (32) with (29), a level set solution of surface evolution is naturally formulated for the 3D extension of the deformable model (10). The GCF refinement in higher dimensional spaces can be similarly applied to that in 2D.

III. Experimental Results

The proposed method was tested on a variety of synthetic and real data in both 2D and 3D. Both qualitative and quantitative comparison were carried out against a number of state-of-the-art techniques, namely MAC, GCCV and SBCV.

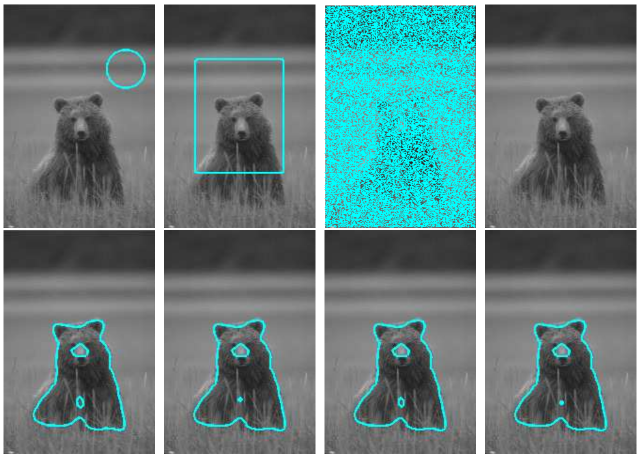

Fig. 4: Results using the proposed method for weak boundaries. Row 1 shows various initializations and Row 2 presents the converged results.

[1] The proof of Theorem 1 is similar to that of Theorem 2 in [14].

TABLE I: Quantitative results of GCCV, SBCV and the proposed method in 2D (unit: %.)

SD		Ring			GacGray			Fourcircles			Diversity		
		Bg	*Fg*	*Overall*	*Bg*	*Fg*	*Overall*	*Bg*	*Fg*	*Overall*	*Bg*	*Fg*	*Overall*
0.1	GCCV	94.6	**99.5**	95.6	94.7	86.8	92.4	38.3	**95.7**	47.2	93.9	**99.1**	94.7
	SBCV	99.5	98.5	99.2	99.8	92.9	97.8	**99.6**	71.9	95.3	99.1	93.1	98.2
	Proposed	**99.6**	98.7	**99.4**	**99.9**	**95.6**	**98.8**	95.9	93.7	**95.6**	**99.6**	95.9	**99.0**
0.2	GCCV	85.1	96.3	87.4	85.7	78.8	83.7	44.5	**86.5**	51.0	79.7	**96.0**	82.2
	SBCV	99.1	97.0	98.4	99.6	**93.3**	**97.8**	**99.1**	71.7	**94.9**	99.3	91.1	98.0
	Proposed	**99.4**	**97.4**	**99.1**	**99.9**	91.9	97.6	96.3	85.1	94.5	**99.6**	94.5	**98.8**
0.3	GCCV	78.9	92.6	81.6	79.7	72.6	77.6	35.7	**85.6**	43.4	71.1	**92.7**	74.5
	SBCV	98.5	96.3	97.9	98.9	89.3	96.1	**99.8**	42.5	90.9	99.3	79.0	96.1
	Proposed	**98.5**	**97.0**	**98.2**	**99.7**	**92.7**	**97.6**	99.3	78.7	**94.5**	**99.8**	86.7	**97.7**
0.4	GCCV	75.4	89.0	78.1	75.4	69.7	73.7	37.7	**82.3**	44.6	66.8	**89.9**	70.4
	SBCV	96.9	97.2	97.0	99.3	87.2	95.8	**99.9**	21.7	87.8	99.0	77.7	95.3
	Proposed	**97.8**	**97.6**	**97.7**	**99.5**	**92.6**	**97.5**	96.0	80.1	**93.8**	**99.5**	82.6	**96.8**
0.5	GCCV	70.6	83.2	73.5	73.0	66.7	71.1	39.3	**78.1**	45.3	64.2	86.6	67.7
	SBCV	95.1	96.2	95.2	99.1	83.0	94.4	**99.9**	11.7	86.2	**99.2**	79.5	96.1
	Proposed	**97.2**	**97.1**	**97.1**	**99.4**	**91.8**	**97.2**	99.4	75.6	**93.6**	99.1	**87.2**	**97.2**
Overall	GCCV	80.9	92.1	83.2	81.7	74.9	79.7	39.1	**85.6**	46.3	75.1	**92.9**	77.9
	SBCV	97.8	97.0	97.5	99.3	89.1	96.4	**99.7**	43.9	91.0	99.2	84.1	96.7
	Proposed	**98.5**	**97.6**	**98.7**	**99.7**	**92.9**	**97.7**	97.4	82.7	**94.4**	**99.5**	89.4	**97.9**

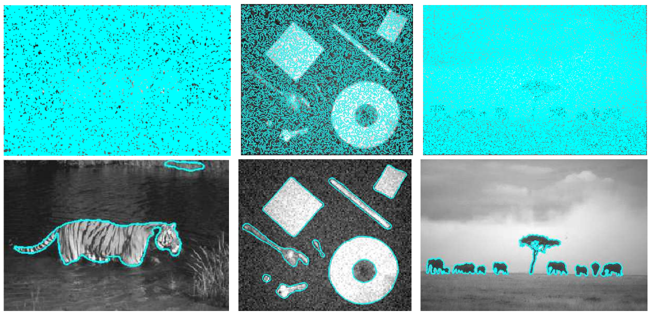

Fig. 5: Example results of the proposed method with random initializations.

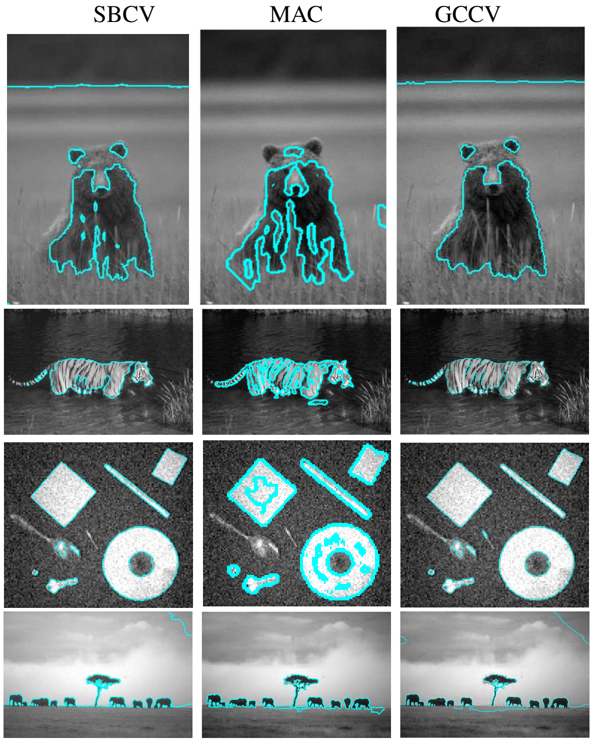

Fig. 6: Results of SBCV, MAC and GCCV for the images used in Figs. 4 and 5.

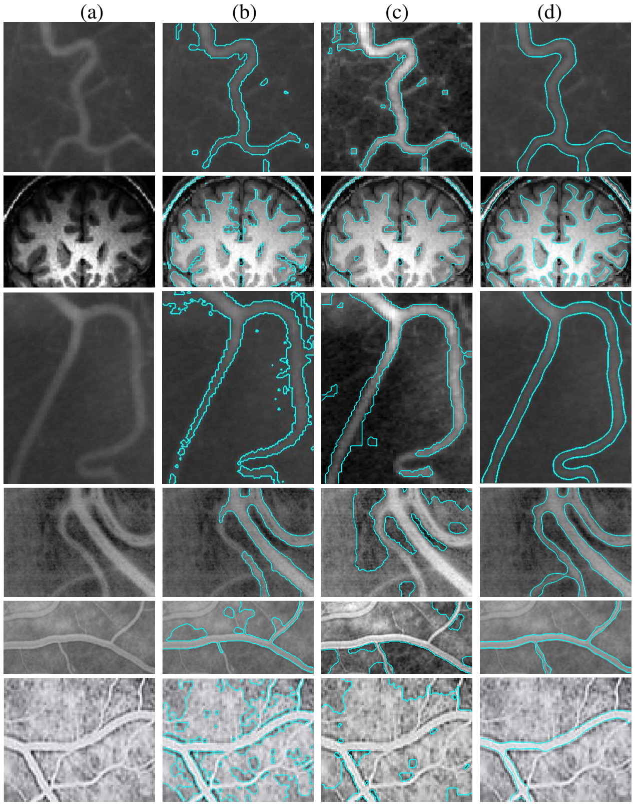

Fig. 7: Results without initialization. (a) original image, (b) SBCV, (c) GCCV and (d) the proposed method.

A. Parameter Setting

For the computation of GCF in (3) and (31), the kernel radius is set to $N/2$ (N is the smallest coordinate of the image size); the parameter K' in (15) is set to 0.5; the balancing parameter α in (22) and (29) is uniformly set to 0.1. The kernel radius does not require tuning, and its only requirement is that the kernel size needs to match or to be bigger than the image domain. The diffusion parameter takes its value generally between 0 and 1. The larger the value, the more

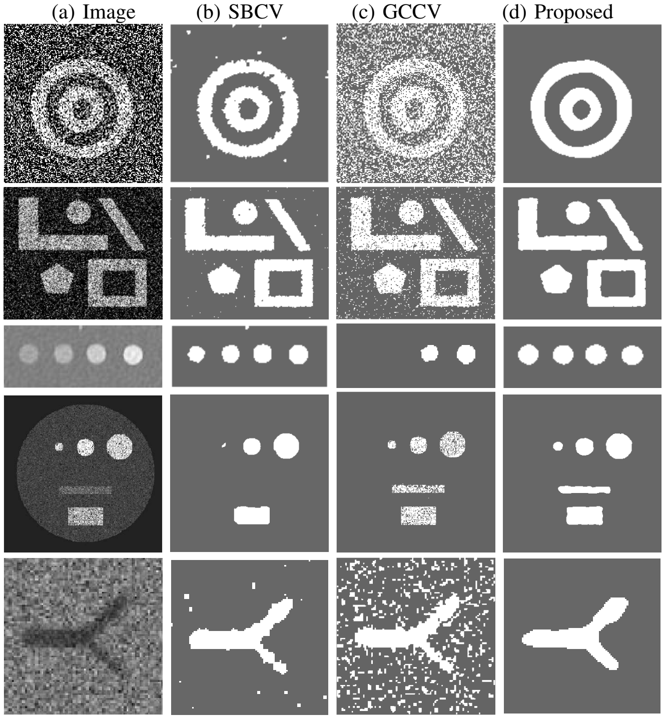

Fig. 8: Results on noisy images. (a) original image, (b) SBCV, (c) GCCV, (d) the proposed method.

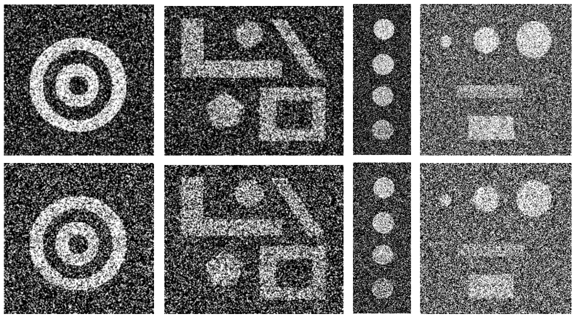

Fig. 9: Some example images used for quantitative analysis as shown in Table I. Row 1: sd = 0.3 and Row 2: sd=0.5.

the diffusion. In our experiments we fixed it in the mid range, i.e. 0.5. The curvature weighting coefficient is always kept small since it only locally smooths the contour or surface, and typically it is not considered as a tuning parameter. For obtaining the best performance for GCCV and SBCV, a series of fidelity values are used for identifying the best result as advised in [16] for SBCV and the most appropriate parameters are set as suggested in [17], [15] for GCCV. Note that, in some cases, the relevant parameters for these methods 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 SBCV.

B. Arbitrary Initialization

With its ability of create new components away from zero level set, the gradient descent scheme (22) of the proposed deformable model performed very similarly to the global minimizer in (29). The deformable model can be implemented

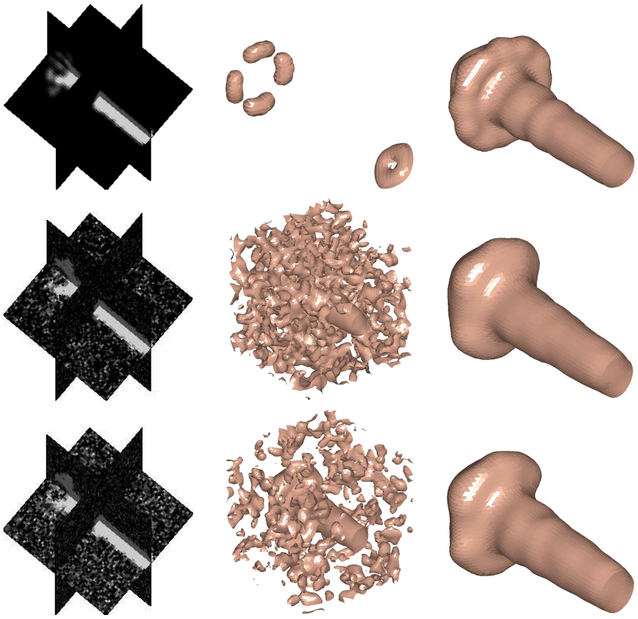

Fig. 10: Results using the proposed method on the fuel injection data ($64 \times 64 \times 64$). Row 1 shows the original dataset and the result. The datasets in Rows 2 and 3 are corrupted the Gaussian noise (SD=0.01, 0.05). Column 1 shows the 3-D view of example slices; the initial surface is shown in Column 2; the converged result is in the final column.

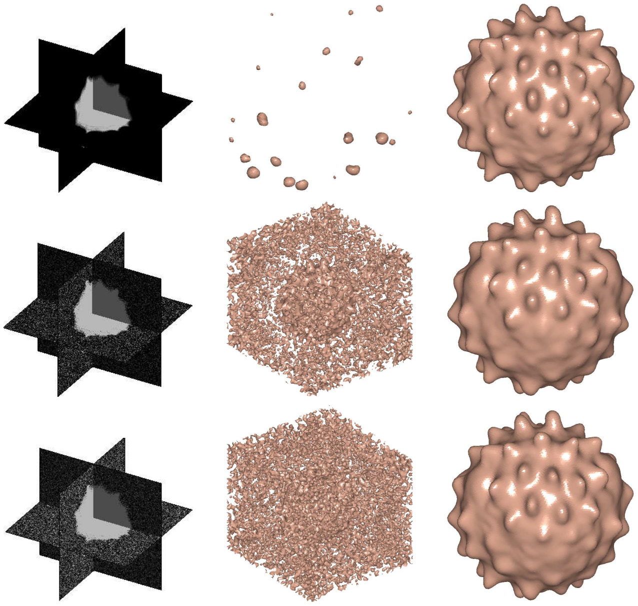

Fig. 11: Results using the proposed method in the Daisy Pollen Grain data (192 × 180 × 168). Row 1 is with the original dataset; Rows 2 and 3 are with the noisy datasets (SD=0.05, 0.1). Column 1 shows the 3-D view of example slices; Column 2 presents the initial surface; the final column is the converged result.

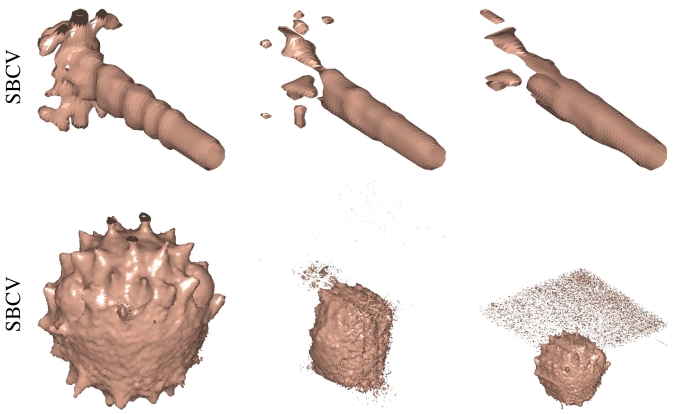

Fig. 12: Results obtained using SBCV for the Daisy and Fuel volumes. Column 1: noise free; Column 2: the data with the Gaussian noise of SD=0.01 (Fuel Injection) and SD=0.05 (Daisy); Column 3: the data with the Gaussian noise of SD=0.05 (Fuel Injection) and SD=0.1 (Daisy).

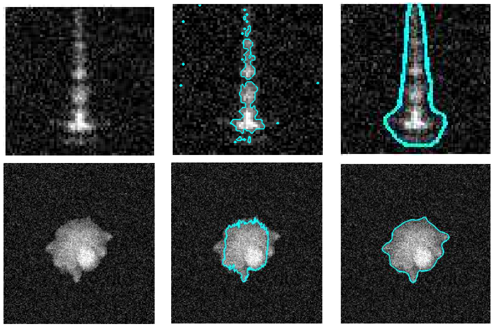

Fig. 13: Results using SBCV (Column 2), and the proposed method (Column 3) for Slice 28 in Fuel Injection and Slice 44 in Daisy.

and applied with arbitrary initializations. We apply a variety of initializations including initialization-free to both of these two implementations, see the examples in Figs. 4, 5, 7 and 8. The results show both implementations perform robustly with arbitrary initializations and consistently converge to the object boundaries.

C. Weak Boundary

The proposed method can effectively deal with incomplete edge information in the images, where the performance of the traditional global minimizers [15], [16] are affected. Fig. 4 presents the results by applying the proposed method to a weak edge/poor contrast case with various initializations including circle, rectangle, random and initialization-free. It is shown that our method can extract the features properly and handle weak boundary robustly with arbitrary initializations. In contrast, the results in the first row of Fig. 6 illustrate that the performance of SBCV, MAC and GCCV are seriously affected by incomplete boundary information and poor contrast even though the running parameters are carefully tuned. It is worth noting that GCCV is much more time-consuming. We discuss the computational issue in Section III-H.

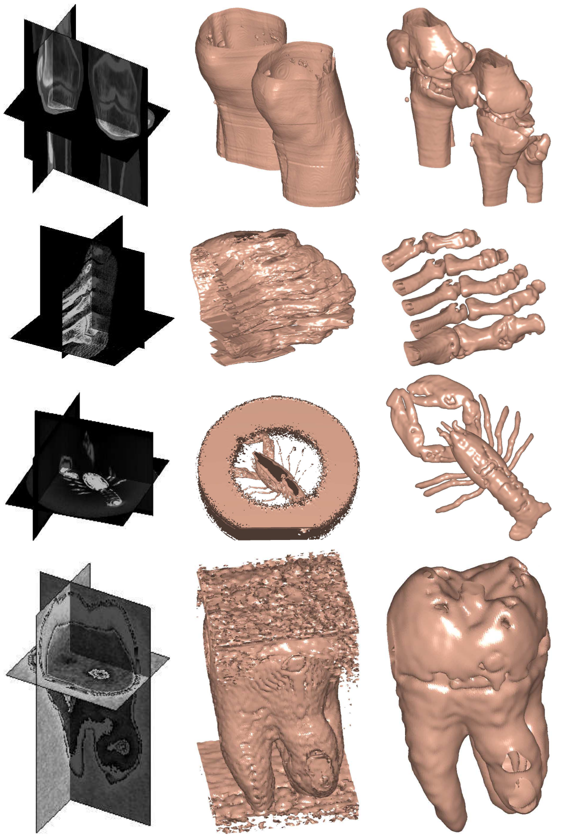

Fig. 14: Results using SBCV (Column 2), and the proposed method (Final column) in the biomedical datasets of the Knee (379 × 229 × 305), Foot (256 × 256 × 256), Lobster (301 × 324 × 56) and Tooth (92 × 78 × 161). Column 1 shows the 3-D view of example slices.

D. Complex Geometry and Deep Concavity

In the situation of complex geometries and deep concavities, thanks to global interaction for the GCF computation, our deformable model shows much better performance in comparison with the traditional methods. In Fig. 5, three real images are used to demonstrate this. The delicate structural details of the objects were well captured, such as the shapes of tail and neck in “tiger”, the shapes of small items in “miscellany” and the shapes of nose and legs in “elephants”. In comparison, Fig. 6 shows that the performances of SBCV, MAC and GCCV were all seriously affected by complex geometries and deep concavities, in that either the object geometries could not be efficiently preserved or the concavities were not effectively detected.

E. Poor Contrast

In the case of poor contrast, such as vessel imaging and magnetic resonance imaging as shown in Fig. 7, the traditional methods usually suffer from the blurred features and result in spurious edges. Our proposed method benefits from the global gradient interaction and the adaptive tensor diffusion. Its superior performance of boundary capture in poor contrast is demonstrated in Fig. 7. The results in Fig. 7(b)(c) show that

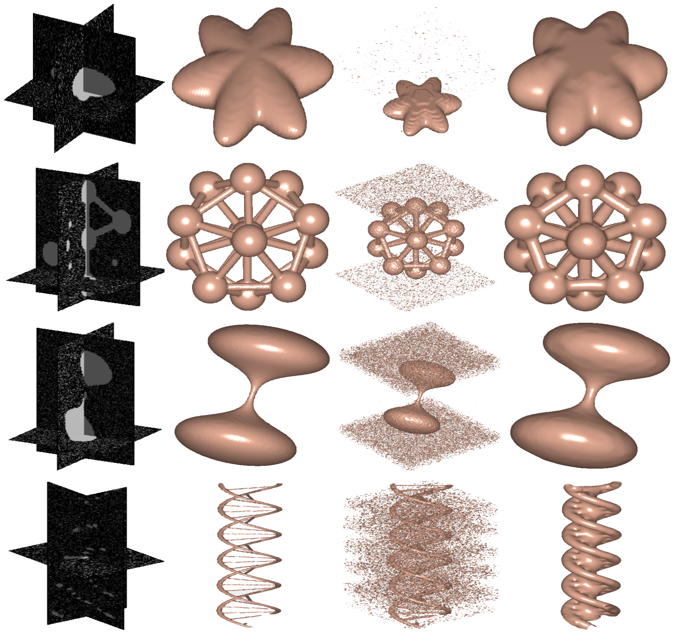

Fig. 15: Results using SBCV (Column 3) and the proposed method (Final column) in the synthetic noisy datasets ($128 \times 128 \times 64$, $sd = 0.03$) of Star, Molecule, Naval-string and DNA. Column 1 shows the 3-D view of example slices ((25,64,64),(64,64,25),(64,64,25),(32,32,64)); Column 2 is the ground truth.

the performance of both SBCV and GCCV were significantly compromised by the low image contrast.

F. Noise Sensitivity

Noise interference is a critical factor to affect the capture range of object boundaries using the edge-based approach. In the proposed method, the GCF nonlinear diffusion scheme enhances the ability of dealing with noise interference. To study the performance towards noise interference, five synthetic images with various noise conditions are used, as shown in Fig. 8. It can be seen that our method can successfully extract the object boundaries even though the excessive amount of noise and artifacts lead to very low contrast or insufficient boundary information. The performances of SBCV and GCCV were in varying degrees affected by the noise, particularly when noise level is high.

For quantitatively comparing with GCCV and SBCV, four series of synthetic noisy datasets (Gaussian noise: zero mean, standard deviation (SD) ranging from 0.1 to 0.5) were generated. Example noisy images from the dataset (SD=0.3, SD=0.5) are shown in Fig. 9. The results are presented in Table I. On noisy images SBCV performed better than GCCV. SBCV relies on known region constants as a prior, whilst GCCV uses K-means clustering to estimate the region constants. These unknown constants, in GCCV, are represented in a vector-valued function and subsequently in the derived energy function. The iterative optimization is taken place in a higher dimensional space. Any errors in estimating those constants are thus carried forward to optimization and to the segmentation results. In the case of excessive image noise, the K-means estimation may not be reliable, which in turn affects the final result. The proposed method performed better than SBCV and significantly outperformed GCCV.

G. 3D Active Surfaces

In the experiment of active surfaces, the volumetric image data used is courtesy of the Volume Library (http://www9.informatik.uni-erlangen.de) and the Volvis medical datasets (www.volvis.org). In Fig. 10, a CT Fuel Injection scan (Row 1) and its generated noisy datasets (Rows 2-3) are used for surface reconstruction using our deformable model. Column 1 illustrates the 3D view of the example slices. Column 2 presents the initial surfaces created by initialization-free surface evolution. Column 3 shows the converged surfaces that demonstrate our model can recover the 3D structures properly and deal with noise effectively. Another example of dealing with noise interference is shown in Fig. 11, where the image data is acquired for Daisy Pollen Grain using a confocal laser scanning microscopy. Similarly, the original dataset and two generated noisy datasets are presented in Column 1. The initial result and the converged result are illustrated in Columns 2 and 3, respectively. For comparison, Fig. 12 shows the surface reconstruction using SBCV. It can be seen that SBCV was severely affected by noise and weak edges, and failed to recover the object shape accurately. Two slices in the resulting volumes are shown in Fig. 13. Note, the computation of GCCV in 3D is extremely demanding, which makes it impractical to implement and compare in 3D. This will be discussed in detail in Subsection III-H.

To further demonstrate its ability in surface reconstruction, the proposed method was applied to recover the surfaces of several biomedical datasets. In Fig. 14, the dataset contains the Knee, Foot, Lobster and Tooth scans. For each volume, its 3D slice view, the converged results for SBCV and our method are presented. These recovered surfaces in Column 4 show that our method is capable of capturing the complex structures of the objects, and obtain much better results than SBCV.

In addition, for quantitative comparison, four series of noisy 3D datasets, namely ‘star’,‘molecular’, ‘navalstring’ and ‘DNA’, are generated by using the known ground truth datasets at the SD levels of 0.01,0.02,0.03,0.04,0.05. Fig. 15 shows the converged results using SBCV (Column 3) and our method (Column 4) in the dataset with SD=0.03. 3D slice view and the ground truth are presented in Columns 1 and 2 respectively. Table II provides the details of SBCV and our method on all noisy datasets, which includes the segmentation accuracies in background, foreground and overall datasets. It is shown that our method outperformed SBCV in almost all cases except the background segmentation of ‘DNA’ at SD=0.02.

H. Computational Complexity

In regard to the computational complexity of our methods, for the 2D case, the main computation comes from the calculation of the GCF and its diffusion process. For the calculation of the GCF ((3) and (8)), it is performed by convolving the image gradient ($N \times N$) with an inverse

TABLE II: Quantitative comparison between SBCV and the proposed method in 3D (unit: %).

SD		Star			Molecule			Navalstring			DNA		
		Bg	*Fg*	*Overall*	*Bg*	*Fg*	*Overall*	*Bg*	*Fg*	*Overall*	*Bg*	*Fg*	*Overall*
0.01	SBCV	98.1	85.7	97.6	98.9	80.1	97.5	99.2	91.3	98.7	**98.2**	50.5	96.6
	Proposed	**99.8**	**98.6**	**99.8**	**99.8**	**97.2**	**99.6**	**99.2**	**94.9**	**98.9**	98.1	**94.6**	**98.0**
0.02	SBCV	96.6	85.9	96.1	97.6	80.3	96.3	**99.2**	91.3	98.6	**99.7**	47.9	97.9
	Proposed	**99.8**	**99.8**	**99.8**	**99.8**	**99.3**	**99.5**	99.1	**95.6**	**98.8**	98.1	**94.6**	**97.9**
0.03	SBCV	96.1	85.9	95.7	98.8	80.3	97.4	96.8	91.4	96.4	92.9	59.4	91.8
	Proposed	**99.5**	**100.0**	**99.5**	**99.9**	**96.3**	**99.6**	**99.5**	**95.6**	**99.3**	**98.1**	**94.7**	**98.0**
0.04	SBCV	95.6	86.0	95.2	98.7	80.4	97.3	97.9	91.5	97.4	95.8	56.1	94.5
	Proposed	**99.8**	**99.9**	**99.8**	**99.8**	**96.4**	**99.6**	**99.2**	**97.5**	**99.0**	**98.1**	**94.7**	**97.9**
0.05	SBCV	94.9	86.1	94.6	96.4	80.8	95.2	96.4	91.6	96.0	93.6	59.3	92.4
	Proposed	**99.7**	**99.9**	**99.7**	**99.8**	**96.4**	**99.6**	**99.9**	**96.5**	**99.7**	**98.1**	**94.7**	**97.9**
Overall	SBCV	96.3	85.9	95.8	98.1	80.4	96.7	97.9	91.4	97.4	96.0	54.6	94.6
	Proposed	**99.7**	**99.6**	**99.7**	**99.8**	**97.1**	**99.6**	**99.4**	**96.0**	**99.1**	**98.1**	**94.7**	**97.9**

TABLE III: Computational time comparison.

Image size	GCCV		SBCV		Proposed	
	Itr	CPU(s)	Fidelity	CPU(s)	Itr	CPU(s)
307×432	500	18 032	1×10^{-3}	2.8	20	6.2
868×551	200	12 302	1×10^{-4}	4.1	35	11.3
579×481	200	6430	1×10^{-3}	3.4	30	6.7
917×582	150	29 344	1×10^{-4}	4.5	15	9.1

distance kernel $((2R+1)\times(2R+1)))$. By using fast Fourier transform (FFT) in the frequency domain, this can be efficiently conducted. The computational order are then estimated as $O((N+2R)^2log(N+2R)^2) = O((N+2R)^2log(N+2R))$ [37]. Due to the typical values of the kernel radius R being $O(N)$, the complexity of this part in the proposed method is $O(N^2logN)$. For tensor diffusion, the numerical implementation is performed using the additive operation scheme (AOS) and its computational complexity is $O(N^2)$ [27]. Therefore, the proposed method is totally computational of the order $O(N^2logN)$. Similarly, the computational complexity in 3D is $O(N^3logN)$.

The computational complexity of SBCV is $O(N^2)$ in 2D and $O(N^3)$ in 3D. As described in [17], the computation using GCCV is very expensive because GCCV has to search the solution in a higher dimensional space. In fact, the image size (N) in computation is the actual size multiplying $N_u \times N_v^2$ where N_u, N_v^2 are the numbers of vector fields used in the searching space. Therefore, it is very time-consuming although the computational complexity is also $O(N^2)$ for the two-phase case in 2D.

For quantitative comparison of computational complexity, Table III shows the consumed time by applying different methods for the images in Fig. 6, using a computer with Intel i7-3770 CPU, 32 GB RAM. In Table III, “Itr” stands for the converged iterations. “CPU” is the computational time. For SBCV, “Fidelity” is the parameter used to control the convergence. The proposed method is close to SBCV and is much more efficient than GCCV. In fact, GCCV is so inefficient that it is impractical to perform in 3D. Also, note that the proposed method converges much faster than GCCV.

IV. Conclusion

We proposed a novel gradient convolution field GCF, whose divergence, can be effectively used as an external force for formulating a deformable model. The GCF utilizes both edge strength and edge orientation and it has been shown that its divergence carries the characteristics of a region indication function. This feature allowed us to employ global minimizers in order to achieve initialization dependence that arguably has been an open challenge for edge based deformable models. GCF can be easily extended to high dimensional space with limited computational overhead. Moreover, we have shown that by diffusing the GCF, instead of its divergence, it handles image noise rather well. For solving the deformable model, a global solution scheme is obtained by using convex relaxation. We also showed that it is possible to achieve similar results using a gradient descent with level set regularization. This demonstrates the unique characteristics of the proposed method. Both qualitative and quantitative comparisons were carried against GCCV, SBCF, and MAC. The proposed method consistently achieved better performance, particularly if images contain intensity inhomogeneity, noise, weak edge, and complex geometry.

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:

$$\mathbf{B}(\mathbf{x}) = \nabla \times \mathbf{A}(\mathbf{x}), \quad \mathbf{A}(\mathbf{x}) = \frac{\mu_0}{4\pi}\sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s})\frac{\mathbf{O}(\mathbf{s})}{R_{\mathbf{xs}}}. \qquad (33)$$

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. $(A_i(\mathbf{x}), A_j(\mathbf{x}), 0)$:

$$A_i(\mathbf{x}) = \frac{\mu_0}{4\pi}\sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s})\frac{-\hat{I}_y(\mathbf{s})}{R_{\mathbf{xs}}}, \;\; A_j(\mathbf{x}) = \frac{\mu_0}{4\pi}\sum_{\mathbf{s}\neq\mathbf{x}} f(\mathbf{s})\frac{\hat{I}_x(\mathbf{s})}{R_{\mathbf{xs}}}, \qquad (34)$$

where we consider $\lambda = 1$ (see (4)). 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:

$$\begin{aligned}\mathbf{B}(\mathbf{x}) &= \nabla \times \mathbf{A}(\mathbf{x}) \\ &= (0, 0, \frac{\partial A_j}{\partial x} - \frac{\partial A_i}{\partial y}) \\ &= (0, 0, \frac{\partial E_i}{\partial x} + \frac{\partial E_j}{\partial y}) \\ &= (0, 0, \nabla \cdot \mathbf{E})\end{aligned} \tag{35}$$

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

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] 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.

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

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

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

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

[14] 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.

[15] 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.

[16] 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.

[17] E. Brown, T. Chan, and X. Bresson, “Completely convex formulation of the chan-vese image segmentation model,” *J. Comput. Vis.*, vol. 98, pp. 103–121, 2012.

[18] J. Moreno, V. Prasath, H. Proenca, and K. Palaniappan, “Fast and globally convex multiphase active contours for brain mri segmentation,” *Compu. Vis. Imag. Understand.*, vol. 125, pp. 237–250, 2014.

[19] G. Strang, “Maximal flow through a domain,” *Math. Programming*, vol. 26, pp. 123–143, 1983.

[20] W. Fleming and R. Rishel, “An integral formula for total gradient variation,” *Arch. Math.*, vol. 11, pp. 218–222, 1960.

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

[22] 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.

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

[24] E. Brown, T. Chan, and X. Bresson, “A convex relaxation method for a class of vector-valued minimization problems with applications to mumford-shah segmentation,” *UCLA CAM Report*, vol. 10, no. 43, 2010.

[25] 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.

[26] 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.

[27] 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.

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

[29] 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.

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

[31] C. Li, C. Kao, J. Gore, and Z. Ding, “Implicit active contours driven by local binary fitting energy,” in *CVPR*, 2007, pp. 1–7.

[32] C. Xu and J. Prince, “Snakes, shapes, & gradient vector flow,” *IP*, vol. 7, no. 3, pp. 359–369, 1998.

[33] 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.

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

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

[36] 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.

[37] 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.