

Adaptive Regularization for Image Segmentation using Local Image Curvature Cues

Josna Rao¹, Rafeef Abugharbieh¹ and Ghassan Hamarneh²

¹Biomedical Image & Signal Computing Lab, University of British Columbia, Canada

²Medical Image Analysis Lab, Simon Fraser University, Canada

{josnar,rafeef}@ece.ubc.ca, hamarneh@cs.sfu.ca

Abstract. Image segmentation techniques typically require proper weighting of competing data fidelity and regularization terms. Conventionally, the associated parameters are set through tedious trial and error procedures and kept constant over the image. However, spatially varying structural characteristics, such as object curvature, combined with varying noise and imaging artifacts, significantly complicate the selection process of segmentation parameters. In this work, we propose a novel approach for automating the parameter selection by employing a robust structural cue to prevent excessive regularization of trusted (i.e. low noise) high curvature image regions. Our approach autonomously adapts local regularization weights by combining local measures of image curvature and edge evidence that are gated by a signal reliability measure. We demonstrate the utility and favorable performance of our approach within two major segmentation frameworks, graph cuts and active contours, and present quantitative and qualitative results on a variety of natural and medical images.

1 Introduction

Regularization plays a crucial role in improving the robustness and applicability of image segmentation techniques. Through the use of weighted regularization terms in conjunction with data fidelity terms, images plagued by high levels of deterioration, i.e. noise or poor edge contrast, are prevented from causing excessive irregularities and inaccuracies in the resultant segmentation. The vast majority of existing segmentation methods are predominantly based on parameter-laden optimization procedures designed to produce ‘optimal’ segmentations at their minimum. These methods commonly involve a highly sensitive tradeoff between the aforementioned regularization (smoothing) terms and data fidelity terms. Depending on how differently these competing energy terms are weighted, the resulting segmentation can greatly differ. Examples of widely used optimization-based segmentation methods with this sensitive tradeoff include active contours techniques [1–4], graph cut methods [5], optimal path approaches [6] and numerous variations thereof. In fact, addressing the issue of how to best balance competing cost terms is of great importance to many related algorithmic formulations in computer vision. More generally, this tradeoff is seen in likelihood

versus prior in Bayesian methods [7] and loss versus penalty in machine learning [8].

Determining the optimum balance between regularization and adherence to image content has predominantly been done empirically and in an ad-hoc manner. However, natural and medical images commonly have objects which exhibit complicated and spatially varying boundary behavior, and often suffer from significant inhomogeneous image artifacts, e.g. the spatially varying bias field commonly observed in magnetic resonance (MR) images [9]. Compensating for such image deteriorations by uniformly increasing the level of regularization, until the most degraded region of the image is properly regularized, may result in excessive smoothing in those regions that do not require that much regularization. Subsequently, this results in a loss in segmentation accuracy, particularly for objects with highly curved boundaries. This commonly results in a painstaking and unreliable parameter-tweaking process.

Most reported approaches to segmentation keep a uniform level of regularization across the image or along an object boundary, i.e. one that does not vary spatially and is determined empirically. As addressed in McIntosh and Hamarneh [10], adapting the regularization weights *across a set of images* is necessary for addressing the variability present in real image data. Although an optimal regularization weight can be found for a single image in a set [10], the same weight may not be optimal for all regions of that image. In [11], a max-margin approach is used to learn the optimal parameter setting. In [12], Kolmogorov et al. solved the optimization problem for a range of parameters.

In recent years, spatially adaptive regularization has been acknowledged as a necessary requirement for improving the accuracy of energy-minimizing segmentations. In an earlier work [13], we proposed an adaptive regularization framework based on estimating the level of image reliability through local data cues reflecting both structure gradient and noise. Our approach in [13] demonstrated a clear advantage to spatially adaptive reliability-based regularization when compared to standard uniform regularization methods. Erdem and Tari [14] proposed a method for modulating diffusivity for Mumford-Shah segmentation approaches through the use of data-driven local cues and contextual feedback, specifically focusing on edge (gradient) consistency, edge continuity, and texture cues. Kokkinos et al. [15] proposed a spatially adaptive texture estimation measure through an amplitude/frequency modulation model of images that allows for a probabilistic discrimination between edges, textured and smooth regions. In [15], a texture cue, a loosely defined context-based classifier cue, and an intensity cue were used to distinguish between texture edges and edges between different objects. Only the latter edges were then used to dampen the curve evolution and define the segmentation boundary. Malik et al. [16] proposed Normalized Cuts to regularize segmentation in textured regions through the use of local texture and orientation cues. Gilboa et al. [17] presented a graph-cut based segmentation framework with spatially varying regularization through edge weights in the graph using a gradient magnitude-based cue.

These previous spatially adaptive methods focused on modulating regularization through local gradient direction and magnitude, texture, and noise estimation cues. In this paper, we advocate the need to integrate, for the first time, curvature cues into a spatially-adaptive regularization scheme. Object boundaries typically exhibit details of various scales, i.e. parts of the boundary can be smooth while other parts exhibit highly curved features. It is therefore inappropriate to enforce the same level of regularization in these different regions of varying degrees of curvature. In [18–20], for example, it was observed that high curvature points are anatomically and structurally important and thus form a good basis for feature matching over a set of data for image registration. Therefore, such high curvature parts of an object boundary should not be excessively regularized, otherwise important geometrical details are lost. The key idea of our approach is to decrease the regularization in reliable high curvature regions of the object to be segmented. To this end, we propose a new regularization scheme where structural curvature information calculated from the image is used to control the regularization and to better preserve the object shape in the presence of poor image quality.

It is important to distinguish our proposed curvature based spatial modulation of regularization from earlier works incorporating curvature, which fall under one of two classes. One class uses the *curvature of an evolving contour as an internal energy* to locally control the contour evolution in order to smoothen high curvature contour segments, e.g. [1, 21]. The other class treats *image curvature as an external energy* in order to attract the evolving contour to high curvature regions in the image, e.g. [1, 18]. In contrast, our proposed method uses estimates of *local image curvature to modulate the spatial regularization* by adaptively balancing the relative contributions of internal vs. external energies in the optimization process.

In summary, we propose a local image curvature-based structural cue that is robust to noise, is computed automatically, and does not require any prior knowledge or preprocessing steps. In order to showcase the utility of our approach, we incorporate this structural cue into two popular segmentation frameworks, graph cuts [22, 23] and active contours [24]. We validate our method on real natural and medical images, and compare its performance against two alternative approaches for regularization: using the best possible spatially uniform (fixed) weight, and using a curvature-oblivious spatially adaptive regularization cue based on a signal reliability approach [13]).

2 Methods

Our regularization technique focuses on energy-minimizing segmentation, where the objective is to find a segmentation $C(x, y)$ that labels every pixel p in an image $I(x, y) : \Omega \subset \mathbf{R}^2 \rightarrow \mathbf{R}$, e.g., object vs. background. We use an adaptive regularization weight $w(x, y) \in [0, 1]$ that varies across the image, and incorporate this weight into a general-form energy functional as follows:

$$E(C(x, y), w(x, y)) = w(x, y)E_{int}(C(x, y)) + (1 - w(x, y))E_{ext}(C(x, y)|I) \quad (1)$$

where E_{int} is the internal cost term contributing to the regularization of the segmentation in order to counteract the effects of image artifacts. E_{ext} is the external cost term contributing to the contour's attraction to desired image features, e.g. edges.

Our novel approach to balancing internal and external energy terms employs data cues that autonomously gauge and adapt the required level of regularization in local image regions. This required level of regularization should be different in two distinct scenarios: (a) high curvature boundary, and (b) poor image quality. The first situation requires low regularization to prevent loss of structural details, and the second situation requires high regularization to prevent erratic segmentation behavior due to noise. We set $w(x, y)$ in (1) such that reliable high curvature regions will have less regularization.

2.1 Local Image Curvature Cue

Let $I(x, y; \sigma) = G_\sigma(x, y) * I_o(x, y)$ be a smoothened image where $I_o(x, y)$ is the original image and σ is the Gaussian scale parameter. The unit vector tangent to the iso-intensity contour $C_I(x, y; \sigma)$ and passing through a point (x, y) is given as:

$$\mathbf{t}(x, y; \sigma) = \frac{1}{\sqrt{I_{x,\sigma}^2(x, y) + I_{y,\sigma}^2(x, y)}} \begin{bmatrix} I_{y,\sigma}(x, y) \\ -I_{x,\sigma}(x, y) \end{bmatrix} \quad (2)$$

where $I_{x,\sigma}$ and $I_{y,\sigma}$ are the image derivatives along x and y , respectively, at scale σ . Denoting the Hessian matrix of $I(x, y; \sigma)$ by $H_\sigma(x, y)$, the local image curvature $K(x, y; \sigma)$ can be calculated as [25, 26]:

$$K(x, y; \sigma) = |\mathbf{t}^T H_\sigma \mathbf{t}|. \quad (3)$$

Note that we used the absolute value on the right hand side of (3) since we are not concerned with differentiating between convex and concave curvature. We follow the method in [27] where equation (3) is enhanced to have a stronger response near edges by multiplication with the gradient magnitude raised to some power, which we chose as 2. The enhanced curvature estimate becomes

$$\tilde{K}(x, y; \sigma) = \left| \frac{I_{y,\sigma}^2 I_{xx,\sigma} - 2I_{x,\sigma} I_{y,\sigma} I_{xy,\sigma} + I_{x,\sigma}^2 I_{yy,\sigma}}{\sqrt{I_{x,\sigma}^2 + I_{y,\sigma}^2}} \right|. \quad (4)$$

To determine the curvature values of differently sized structures in the image, we automate the scale selection process by using the normalized scale coordinates of [27]. As the amplitude of the image spatial derivatives decreases with increasing scale, to compare the curvature values across different scales, the curvature must be scale-normalized. \tilde{K}_{norm} is determined through scale-normalized coordinates $\xi = x/\sigma$. The normalized derivative operator with respect to ξ becomes $\partial_\xi = \sigma \partial_x$. Substituting the scale normalized coordinates into (4) results in the following normalized rescaled curvature:

$$\tilde{K}_{norm}(x, y; \sigma) = \sigma^3 \tilde{K}(x, y; \sigma). \quad (5)$$

After the curvature values at each scale have been normalized, the final curvature cue at every pixel is determined by selecting the scale at which \tilde{K}_{norm} assumes a maximum value [27]:

$$K_m(x, y) = \max_{\sigma} \tilde{K}_{norm}(x, y, \sigma). \quad (6)$$

The curvature measure (6) is sensitive to noise and might inaccurately give rise to a strong response at non-structure, high-noise regions of the image. Following the concept of cue gating, as proposed in [16], where gradient information is suppressed in high texture regions, we thus define a noise-gated curvature cue, $K_G(x, y)$, that suppresses our curvature cue in high noise regions as follows:

$$K_G(x, y) = (1 - N(x, y)) K_m(x, y). \quad (7)$$

$N(x, y)$ is a noise measure calculated using local image spectral flatness as follows [13]:

$$N(x, y) = \frac{\exp\left(\frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \ln S(\omega_x, \omega_y) d\omega_x d\omega_y\right)}{\frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} S(\omega_x, \omega_y) d\omega_x d\omega_y} \quad (8)$$

where $S(\omega_x, \omega_y) = |F(\omega_x, \omega_y)|^2$ is the power spectrum of the image, $F(\omega_x, \omega_y)$ is the Fourier transform of the image and (ω_x, ω_y) are the two spatial radian frequencies. $N(x, y)$ and $K_m(x, y)$ are normalized to the range $[0, 1]$. This noise measure responds best to white noise-like patterns (i.e. occupying a very wide and flat spectrum).

2.2 Curvature-Based Regularization

Our noise-gated curvature cue in (7) is used to augment our noise-gated edge evidence $E_G(x, y)$ (proposed in [13]), which we also normalize to $[0, 1]$:

$$E_G(x, y) = (1 - N(x, y)) |\nabla I(x, y)|. \quad (9)$$

Both noise-gated, local image cues, curvature K_G and edge E_G , are now used to control $w(x, y)$ in (1). A meaningful way for setting $w(x, y)$ should satisfy the following requirements: (i) in highly trusted (noise-gated) edge evidence, little regularization is needed, regardless of the curvature strength; and (ii) in regions with low edge evidence, we set the regularization to be inversely proportional to the trusted (noise-gated) curvature, such that high curvature regions are not overly regularized. Note that ‘high’ curvature or edge evidence means a value close to 1 as all our cues are normalized. Thus, we form the adaptive weight as follows:

$$w(x, y) = 1 - E_G(x, y)^{(1 - K_G(x, y))}. \quad (10)$$

If $E_G(x, y)$ is large (approaching 1), the exponent has little effect on the resulting weight, and requirement (i) is satisfied. If $E_G(x, y)$ is low and $K_G(x, y)$ is non-zero, the noise-gated edge evidence will be raised to a power $(1 - K_G(x, y)) \approx 0$, resulting in a lower $w(x, y)$, satisfying requirement (ii). Note that the detrimental effects from noise are handled by this model through the noise-gating of the cues. We refer to $E_G(x, y)^{(1 - K_G(x, y))}$ as the curvature-modulated image reliability measure.

2.3 Incorporation of Texture Cue

In many natural images, large gradients and large curvature values can arise from texture edges rather than from edges representing object boundaries. To prevent texture edges from being included in the final edge set of the image, we must ensure greater regularization occurs in textured regions. We employ a texture measure $T(x, y)$ from Erdem and Tari [14] that estimates the probability of a pixel being near a texture edge:

$$T(x, y) = 1 - \exp(-\gamma(\min(\rho^1(x, y), \rho^2(x, y)))) \quad (11)$$

where γ is a decay rate parameter and $\rho^1(x, y)$ and $\rho^2(x, y)$ represent the p-values returned from the Wilcoxon Mann-Whitney test for sampled distributions lying between regions to the left and right of (x, y) , and above and below of (x, y) . If texture exists around (x, y) , the differences between the distributions will be large and the resulting minimum p-values will be low, producing a low $T(x, y)$.

We incorporate the texture cue into our framework by modifying (9) to form the noise- and texture-gated edge evidence term as follows:

$$E_{G,T}(x, y) = T(x, y) (1 - N(x, y)) |\nabla I(x, y)|. \quad (12)$$

Incorporating (12) into our spatially adaptive weight produces:

$$w(x, y) = 1 - E_{G,T}(x, y)^{(1-K_G(x, y))}. \quad (13)$$

2.4 Structural Cue Modulated Graph Cuts Segmentation

We first incorporated our adaptive weights¹ $w(p)$ into a graph cuts (GC) based segmentation [22, 23]. The segmentation energy in this case becomes:

$$E(f) = \sum_{p,q \in \mathcal{N}} w(p) E_{int}(f_p, f_q) + \sum_{p \in P} (1 - w(p)) E_{ext}(f_p) \quad (14)$$

where $f \in \mathcal{L}$ is the labeling for all pixels $p \in P$, \mathcal{L} is the space of all possible labellings, and P is the set of pixels in image I . In GC, E_{int} is the interaction penalty between pixel pairs (i.e. the penalty of assigning labels f_p and f_q to neighboring pixels p and q), E_{ext} measures how well label f_p fits pixel p given the observed data, and \mathcal{N} is the set of interacting pairs of pixels. $E_{ext}(f_p)$ is proportional to the difference between the intensity of p and the mean intensity of seeds labelled with f_p . $E_{int}(f_p, f_q) = 0$ if $f_p = f_q$ and 1 otherwise.

2.5 Structural Cue Modulated Active Contours Segmentation

We also implemented our proposed adaptive regularization within the popular active contours without edges (AC) segmentation framework by Chan and Vese [24]. The segmentation $C(x, y)$ in (1) is represented here via a Lipschitz

¹ We use p to reflect graph vertices representing an image pixel at (x, y) .

function, $\phi(x, y) : \Omega \rightarrow \mathbf{R}$, where pixels interior to the zero-level set of ϕ are labeled as object and exterior pixels as background. The segmentation energy, $E(\phi)$ is given by:

$$\begin{aligned} E(\phi(x, y)) = & \mu \int_{\Omega} \delta(\phi(x, y)) |\nabla \phi(x, y)| dx dy + \\ & + \lambda_1 \int_{\Omega} |I(x, y) - c_1|^2 H(\phi(x, y)) dx dy + \\ & + \lambda_2 \int_{\Omega} |I(x, y) - c_2|^2 (1 - H(\phi(x, y))) dx dy \end{aligned} \quad (15)$$

where the first term is the internal (regularization) energy equal to the contour length, $\delta(z)$ is the dirac function and $H(z)$ is the Heaviside function. The latter two terms in (15) are external (data) terms, and c_1 and c_2 are the averages of $I(x, y)$ inside and respectively outside the zero-level set of ϕ [24]. λ_1 , λ_2 and μ are constants that control the balance between smoothing and data adherence of the contour. We modified (15) to incorporate spatially adaptive regularization by replacing λ_1 , λ_2 and μ with an adaptive convex weighting as follows:

$$\begin{aligned} E(\phi(x, y)) = & \int_{\Omega} w(x, y) \delta(\phi(x, y)) |\nabla \phi(x, y)| dx dy + \\ & + \int_{\Omega} (1 - w(x, y) + \epsilon) |I(x, y) - c_1|^2 H(\phi(x, y)) dx dy + \\ & + \int_{\Omega} (1 - w(x, y) + \epsilon) |I(x, y) - c_2|^2 (1 - H(\phi(x, y))) dx dy \end{aligned} \quad (16)$$

where ϵ prevents a zero data force term (i.e. to prevent impeding curve evolution). We selected $\epsilon = 0.1$.

We minimize (16) with respect to $\phi(x, y)$ to determine the corresponding Euler-Lagrange equation for $\phi(x, y)$, which is derived in the supplementary material (see also [28]). We then solve for $\phi(x, y)$ iteratively by parameterizing the gradient descent with an artificial time $t \geq 0$ to produce the PDE for $\phi(t, x, y)$ as

$$\begin{aligned} \frac{\partial \phi}{\partial t} = & \delta(\phi(x, y)) \nabla w(x, y) \cdot \frac{\nabla \phi(x, y)}{|\nabla \phi(x, y)|} + w(x, y) \delta(\phi(x, y)) \operatorname{div} \left(\frac{\nabla \phi(x, y)}{|\nabla \phi(x, y)|} \right) \\ & - (1 - w(x, y) + \epsilon) \delta(\phi(x, y)) \left[|I(x, y) - c_1|^2 - |I(x, y) - c_2|^2 \right] = 0 \end{aligned} \quad (17)$$

where $\phi(0, x, y)$ represents the initial contour provided to the method.

3 Results and Discussion

Using MATLAB code on a PC with 3.6 GHz Intel Core Duo processor and 2GB of RAM, we ran a series of tests using a GC wrapper [22], and an implementation of AC [29], both of which were modified as proposed in Sections 2.4 and 2.5. We tested various natural images where structural features play an important role and which are available at the McGill Calibrated Color Database [30]. We

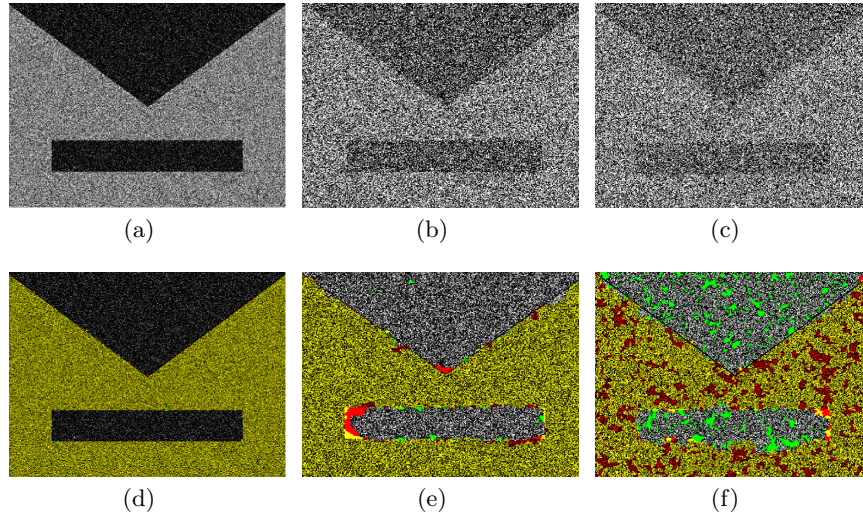


Fig. 1. (color figure) Segmentation of grey object in synthetic image corrupted by AWGN with increasing standard deviation. (a), (b), (c) Original images with std. dev. of 0.05, 1.05, and 1.90, respectively. (d), (e), (f) Corresponding segmentations from the proposed adaptive weight (*green*) and the least-error fixed weight (*red*) where *yellow* regions are where the segmentations overlap. At the high noise level of (c), the segmentation (f) begins to form holes and inaccuracies.

also tested on magnetic resonance imaging (MRI) data from BrainWeb [31]. To demonstrate the advantage of our method, we compared against segmentation results from using the least-error fixed regularization weight, and against segmentation results from using a spatially adaptive regularization weight solely based in image reliability without any curvature-modulation [13]. For quantitative analysis, we used a set of 18 coronal brain MRI slices with ground truth segmentations and performed ANOVA to ascertain the improvements in segmentation accuracy afforded by our method. Computationally, the proposed method required less than a minute to calculate the regularization weights for a 768×576 image. For GC, a low number of random seeds (0.3% of image pixels for each label) were selected automatically by using the ground truth. For AC, we used an initial contour of a 50×50 square placed in the center of images, and used the same initial contour for comparison tests against the alternate methods.

We first analyze the robustness of the spatially adaptive regularization weight (Section 2.2). Figs. 1(a), 1(b), and 1(c) show synthetic images corrupted by increasing levels of average white Gaussian noise (AWGN). The GC adaptive weight segmentation for the images corrupted by noise levels of 0.05 and 1.05 std. dev. (Figs. 1(d) and 1(e), respectively) adheres to the corners of the object and does not leak outside of the object, unlike the fixed weight segmentation in red. At an extremely high noise level of 1.90 std. dev. shown in Fig. 1(c), the resulting adaptive weight segmentation (Fig. 1(f)) begins to show holes and

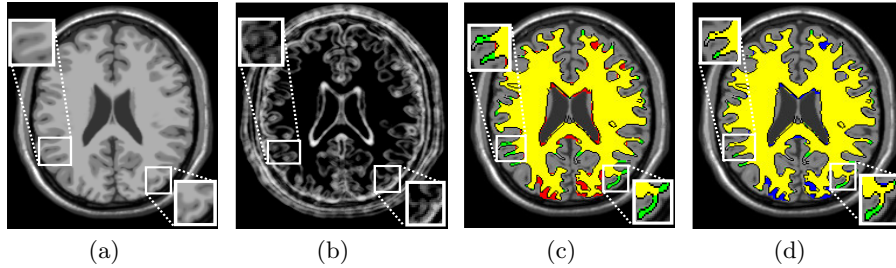


Fig. 2. (color figure) Segmentation of MR data from BrainWeb using GC with curvature-modulated regularization. (a) T1 slice with 20% intensity non-uniformity. (b) Curvature-modulated reliability calculated by our proposed method. Black intensities corresponds to 0 (low reliability/high regularization) and white to 1. Note higher reliability in cortical folds. (c) Comparison of segmentations from the proposed adaptive weight (*green*) to the least-error fixed weight (*red*), and (d) to the non-structural image reliability weight (*blue*). *Yellow* regions are where the segmentations overlap, and ground truth contour is shown in *black*. Proposed weights result in better segmentation of high curvature cortical folds (see *green*) with minimal leakage into the background, unlike other methods (see *red* and *blue* leakage regions).

degradation. Analysis of the Dice similarity coefficient between the adaptive weight GC segmentations and the the ground truth, for the synthetic image of Fig. 1 over various noise levels, showed that segmentation accuracy begins to drop at noise levels greater than 1.75 std. dev.

We next present results of GC segmentation with our proposed regularization framework on MR images from BrainWeb [31]. Fig. 2(a) shows a T1 image with an intensity inhomogeneity of 20%. High curvature-modulated reliability in the cortical folds (Fig. 2(b)) results in lower regularization in these regions. The overlayed GC segmentations (Fig. 2(c)) using the adaptive regularization weight versus the least-error fixed weight shows greater segmentation accuracy in high curvature regions. Additionally, the proposed method shows improvements over the existing non-structural image reliability framework (Fig. 2(d)).

Fig. 3(a) shows the same T1 image of Fig. 2(a) but with a noise level of 7%. The resulting curvature-modulated reliability map (Fig. 3(b)) is not corrupted by the noise and still enforces greater regularization in high curvature cortical folds, as seen in the resultant segmentation comparisons of Fig. 3(c) and Fig. 3(d). At higher noise levels, our proposed curvature modulation results in a more accurate segmentation than the standard least-error uniform weight, and even more accurate than the noise-cue image reliability approach.

We also tested GC with our proposed regularization framework on a series of natural images, such as the flower shown in (Fig. 4(a)), where this image has been corrupted by AWGN with a standard deviation of 0.3. From this image, we produced the curvature-modulated reliability mapping in (Fig. 4(b)). The higher curvature-modulated reliability in the petal tip regions allows for a more accurate segmentation when compared to the least-error fixed weight segmentation

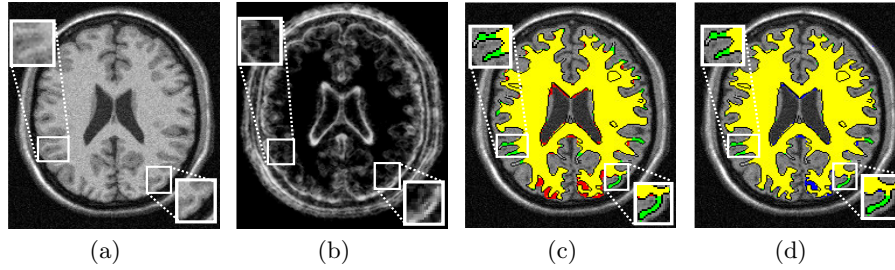


Fig. 3. (color figure) Segmentation of noisy MR data from BrainWeb using GC with curvature-modulated regularization. (a) T1 slice with 7% noise level. (b) Curvature-modulated reliability. (c) Comparison of segmentations from the proposed adaptive weight (*green*) to the least-error fixed weight (*red*), and (d) to the non-structural image reliability weight (*blue*). Even in a high noise case, cortical folds are correctly segmented with proposed weights.

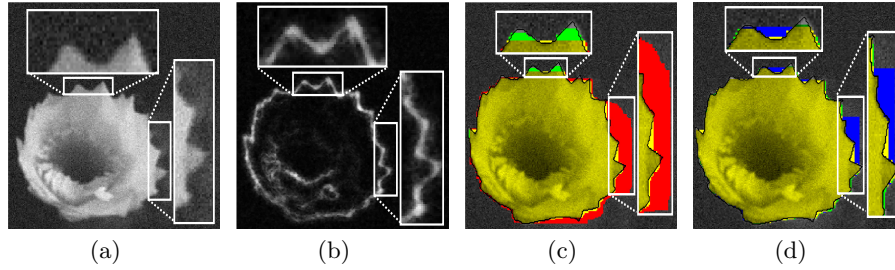


Fig. 4. (color figure) GC segmentation of flower image. (a) Original image with AWGN of standard deviation 0.3. (b) Curvature-modulated reliability (higher in petal tip and crevice regions). (c) Comparison of segmentations from the proposed adaptive weight (*green*) to the least-error fixed weight (*red*), and (d) to the non-structural image reliability weight (*blue*) with overlapping regions in *yellow*. The proposed weights provided the best segmentation of the petal tips and had the least amount of leakage.

(Fig. 4(c)) and the non-structural image reliability weight segmentation (Fig. 4(d)) which, as expected, required higher regularization in the detailed petal tip regions, resulting in leakage into the background.

We demonstrate the AC segmentation with our regularization framework on the dandelion image of Fig. 5(a). Iterations were run until the contour evolution converged (at most 700 iterations). The low curvature-modulated reliability (Fig. 5(b)) in regions outside the flower prevents the resulting segmentation from including objects in the background, unlike the fixed weight segmentation (Fig. 5(c)) which leaked into the background of the image (see the red region). Additionally, the proposed method segments the petal tips more correctly than the non-structural image reliability segmentation as shown in Fig. 5(d) (where our segmentation in green captures all petal tips).

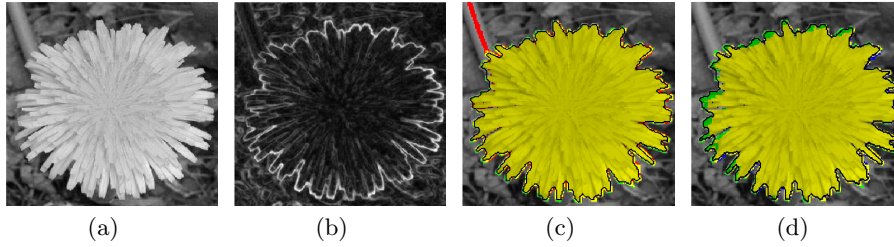


Fig. 5. (Color figure, refer to e-copy). Active Contours segmentation of a natural image. (a) Original image. (b) Curvature-modulated reliability calculated by our proposed method. (c) Comparison of segmentations from the proposed adaptive weight (*green*) to the least-error fixed weight (*red*), and (d) to the non-structural image reliability weight (*blue*). *Yellow* regions are where segmentations overlap. In (c), high regularization in the background prevents the segmentation from the proposed weights from leaking, unlike the fixed-weight method in *red*. In (d), only our segmentation in *green* captures all petal tip details.

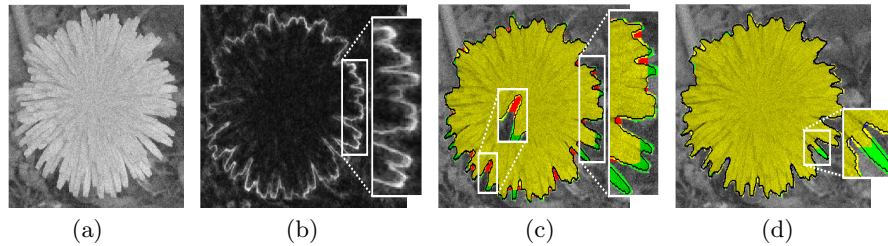


Fig. 6. (color figure) GC segmentation of natural image (a) Original image corrupted by AWGN with standard deviation of 0.3. (b) Curvature-modulated reliability calculated by our proposed method. (c) Comparison of segmentations from the proposed adaptive weight (*green*) to the least-error fixed weight (*red*), and (d) to the non-structural image reliability weight (*blue*) with overlapping regions in *yellow*. Proposed method provides best segmentation of high curvature petal tips and crevices with minimal leakage into the background.

We segmented the same dandelion again but with corruption by AWGN of standard deviation 0.3 (image values normalized to range between 0 and 1), as shown in Fig. 6(a). The curvature-modulated reliability (Fig. 6(b)) produces lower regularization weights in the petal tips and petal crevices. In Fig. 6(c), the fixed-weight segmentation excessively regularizes in the petal region, resulting in leakage (shown in red). Our method does not leak into the background and is able to capture the petal tips (shown in green). Similarly, in Fig. 6(d), the non-structural image reliability segmentation misses a few petal tips, which our method captured.

We demonstrate the ability of the texture-modulated weight defined in (13) (Section 2.3) to segment the textured image of Fig. 7(a) where we set the parameter γ in (11) to 0.1. The curvature modulated reliability (Fig. 7(b)) is erroneously

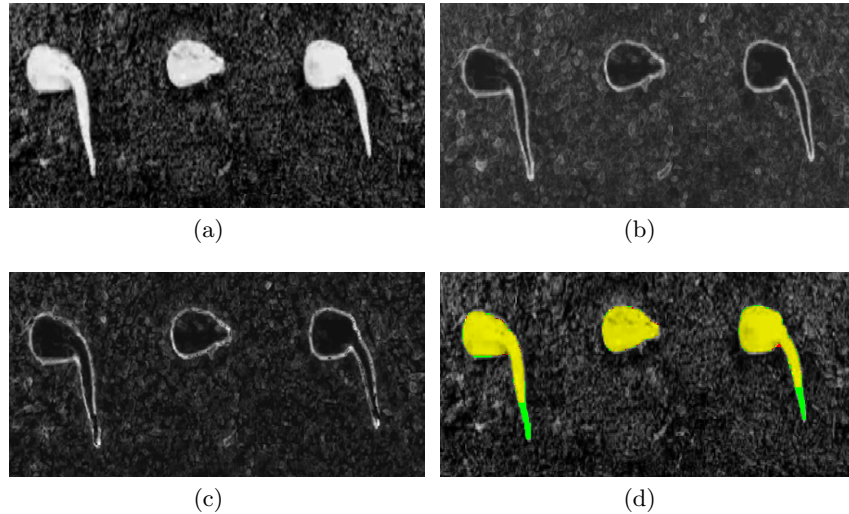


Fig. 7. (color figure) GC segmentation of textured natural image using the curvature-and-texture modulated weight. (a) Original image. (b) Curvature-modulated reliability calculated by our proposed method with no texture gating. (c) Curvature-and-texture modulated reliability. (d) Comparison of segmentations from the proposed adaptive weight (*green*) to the least-error fixed weight (*red*) with overlapping regions in *yellow*. Incorporation of a texture cue reduces leakage into the background, and proposed curvature cue reduces regularization in the protrusion region of the plant.

large for regions with texture. The curvature-and-texture modulated reliability shown in Fig. 7(c) is lower for the texture edges. The resulting GC segmentation is shown in Fig. 7(d). The higher curvature regions of the plant seedlings are accurately segmented by the adaptive weight due to lower regularization in these regions.

Quantitatively, we found significant improvements with the proposed method on the set of 18 brain MRI slices. Using the AC segmentation method with our proposed regularization framework to segment for cortical white matter, and validating with ground truth data, we found our method to produce an average Dice similarity of 78.4% (standard deviation of 0.0402) compared to 72.52% for the least-error fixed weight segmentation (std of 0.0728) and 60.68% for the non-structural image reliability segmentation (std of 0.1481). Our proposed method was significantly more accurate than the alternate methods with p -values $\ll 0.05$. We performed GC segmentations of the cortical white matter on the same dataset and found an average Dice similarity of 89.91% (std of 0.0317) from our proposed method, compared to 86.20% (std of 0.0486) for the least-error fixed weight segmentation and 88.90% (std of 0.0331) for the non-structural image reliability segmentation. Again, our proposed method was significantly more accurate with all p -values $\ll 0.05$. For each slice, we averaged 25 GC segmentations with random seed selections to determine the Dice similarity for

that slice. In addition, we investigated the effect of removing the noise-gating of the curvature measure. We found the resulting segmentation to be 25% less accurate when tested on the image of Fig. 6(a).

4 Conclusion

The key goal of our proposed method was to prevent excessive regularization of structurally important regions in energy minimization based segmentation. We presented a novel local curvature-based structural cue for modulating regularization. This cue was made robust to noise through gating by local signal reliability. Unlike current methods that employ curvature either as an internal or an external energy term, we use curvature to balance the competing internal and external energy terms. Accordingly, highly curved yet reliable boundary regions are spared from regularization.

We incorporated our proposed regularization framework into graph cuts and active contours for image segmentation. We demonstrated superior performance when compared to non-contextual regularization weights, as well as to adaptive, but curvature-oblivious, regularization cues. Quantitative and qualitative tests demonstrated that curvature-controlled regularization improves accuracy since it is common for image data to contain object boundaries of varying degrees of curvature. Future work will focus on extending our proposed cues and regularization framework to 3D image data, as well as adopting measures for other types of noise, e.g. spatially correlated non-Gaussian noise.

References

1. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. *Int. J. of Comput. Vision* **1** (1988) 321–331
2. Caselles, V., Kimmel, R., Sapiro, G.: Geodesic active contours. *Int. J. of Comput. Vision* **22** (1997) 61–79
3. Osher, S.J., Paragios, N.: *Geometric Level Set Methods in Imaging, Vision, and Graphics*. Springer-Verlag (2003)
4. Pluempitiwiriyaew, C., Moura, J.M.F., Wu, Y.J.L., Ho, C.: STACS: New active contour scheme for cardiac MR image segmentation. *IEEE TMI* **24** (2005) 593–603
5. Boykov, Y., Funka-Lea, G.: Graph cuts and efficient N-D image segmentation. *Int. J. Comput. Vision* **70** (2006) 109–131
6. Barrett, W.A., Mortensen, E.N.: Interactive live-wire boundary extraction. *Medical Image Analysis* **1** (1997) 331–341
7. Akselrod-Ballin, A., Galun, M., Gomori, M.J., Brandt, A., Basri, R.: Prior knowledge driven multiscale segmentation of brain MRI. In: *MICCAI (2)*. Volume 4792 of *Lecture Notes in Computer Science*, Springer (2007) 118–126
8. Zhao, P., Yu, B.: Stagewise lasso. *Journal of Machine Learning Research* **8** (2007) 2701–2726
9. Samsonov, A.A., Johnson, C.R.: Noise-adaptive nonlinear diffusion filtering of MR images with spatially varying noise levels. *Magnetic Resonance in Medicine* **52** (2004) 798–806

10. McIntosh, C., Hamarneh, G.: Is a single energy functional sufficient? Adaptive energy functionals and automatic initialization. In: MICCAI (2). Volume 4792 of Lecture Notes in Computer Science., Springer (2007) 503–510
11. Szummer, M., Kohli, P., Hoiem, D.: Learning CRFs using Graph Cuts. In: ECCV, Springer (2008) 582–595
12. Kolmogorov, V., Boykov, Y., Roth, C.: Applications of parametric maxflow in computer vision. ICCV **8** (2007)
13. Rao, J., Hamarneh, G., Abugharbieh, R.: Adaptive contextual energy parameterization for automated image segmentation. In: ISVC. Volume 5875-I. (2009) 1089–1100
14. Erdem, E., Tari, S.: Mumford-Shah regularizer with contextual feedback. Journal of Mathematical Imaging and Vision **33** (2009) 67–84
15. Kokkinos, I., Evangelopoulos, G., Maragos, P.: Texture analysis and segmentation using modulation features, generative models, and weighted curve evolution. IEEE PAMI **31** (2009) 142–157
16. Malik, J., Belongie, S., Leung, T.K., Shi, J.: Contour and texture analysis for image segmentation. Int. J. of Comput. Vision **43** (2001) 7–27
17. Gilboa, G., Darbon, J., Osher, S., Chan, T.: Nonlocal convex functionals for image regularization. UCLA CAM-report (2006) 06–57
18. Cohen, I., Ayache, N., Sulger, P.: Tracking points on deformable objects using curvature information. In: ECCV. (1992) 458–466
19. Schmid, C., Mohr, R., Bauckhage, C.: Evaluation of interest point detectors. Int. J. of Comput. Vision **37** (2000) 151–172
20. Glocker, B., Komodakis, N., Paragios, N., Navab, N.: Approximated curvature penalty in non-rigid registration using pairwise MRFs. In: ISVC. Volume 5875-I. (2009)
21. Evans, L.C., Spruck, J.: Motion of level sets by mean curvature. I. Journal of Differential Geometry **33** (1991) 635–681
22. Bagon, S.: MATLAB wrapper for graph cuts. <http://www.wisdom.weizmann.ac.il/~bagon> (2006)
23. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE PAMI **20** (2001) 1222–1239
24. Chan, T.F., Vese, L.A.: Active contours without edges. IEEE Trans. Image Processing **10** (2001) 266–277
25. Kitchen, L., Rosenfeld, A.: Gray-level corner detection. Pattern Recognition Letters **1** (1982) 95 – 102
26. Donias, M., Baylou, P., Keskes, N.: Curvature of oriented patterns: 2-D and 3-D estimation from differential geometry. In: ICIP. (1998) I: 236–240
27. Lindeberg, T.: On scale selection for differential operators. In: Scan. Conf. on Image Analysis. (1993) 857–866
28. Rao, J., Abugharbieh, R., Hamarneh, G.: Adaptive regularization for image segmentation using local image curvature cues. Technical Report TR 2010-08, School of Computing Science, Simon Fraser University, Burnaby, BC, Canada (2010)
29. Wu, Y.: Chan Vese active contours without edges. <http://www.mathworks.com/matlabcentral/fileexchange/23445>. (2009)
30. Olmos, A., Kingdom, F.A.A.: McGill calibrated colour image database. <http://tabby.vision.mcgill.ca>. (2004)
31. Cocosco, C.A., Kollokian, V., Kwan, R.K.S., Evans, A.C.: BrainWeb: Online interface to a 3D MRI simulated brain database. In: NeuroImage. Volume 5., Academic Press (1997)

Supplementary Material:

Adaptive Regularization for Image Segmentation using Local Image Curvature Cues

Josna Rao¹, Rafeef Abugharbieh¹ and Ghassan Hamarneh²

¹Biomedical Image & Signal Computing Lab, University of British Columbia, Canada

²Medical Image Analysis Lab, Simon Fraser University, Canada
{josnar,rafeef}@ece.ubc.ca, hamarneh@cs.sfu.ca

A Appendix: Derivation of Evolution Equations for Active Contours without Edges

We implemented our proposed adaptive regularization within the popular Active Contours Without Edges(AC) segmentation framework by Chan and Vese [1]. The segmentation is represented here via a Lipschitz function, $\phi(x, y) : \Omega \rightarrow \mathbf{R}$, where pixels p interior to the zero-level set of $\phi(x, y)$ are labeled as objects and exterior pixels as background. We use (x, y) to reflect the location of a pixel p . In the original formulation [1], the segmentation energy, $E(\phi)$ is given by:

$$\begin{aligned} E(\phi(x, y)) = & \mu \int_{\Omega} \delta(\phi(x, y)) |\nabla \phi(x, y)| dx dy + \\ & + \lambda_1 \int_{\Omega} |I(x, y) - c_1|^2 H(\phi(x, y)) dx dy + \\ & + \lambda_2 \int_{\Omega} |I(x, y) - c_2|^2 (1 - H(\phi(x, y))) dx dy \end{aligned} \quad (\text{A.1})$$

where the first term represents the contour length and is the internal energy (regularizing) term, $H(x)$ is the Heaviside function, and $I(x, y)$ is the original image. The latter two terms in (A.1) are external (data) terms, and c_1 and c_2 are the averages of $I(x, y)$ inside and respectively outside the contour, as defined in [1]. λ_1 , λ_2 and μ are constants that control the balance between smoothing and data adherence of the contour. Additionally, we use the notation

$$|\nabla \phi(x, y)| = \sqrt{\left(\frac{\partial \phi}{\partial x}\right)^2 + \left(\frac{\partial \phi}{\partial y}\right)^2} \quad (\text{A.2})$$

We modified (A.1) to incorporate spatially adaptive regularization by replacing λ_1 , λ_2 and μ with an adaptive convex weighting as follows:

$$\begin{aligned} E(\phi(x, y)) &= \int_{\Omega} w(x, y) \delta(\phi(x, y)) |\nabla \phi(x, y)| dx dy \\ &+ \int_{\Omega} (1 - w(x, y) + \epsilon) |I(x, y) - c_1|^2 H(\phi(x, y)) dx dy + \\ &+ \int_{\Omega} (1 - w(x, y) + \epsilon) |I(x, y) - c_2|^2 (1 - H(\phi(x, y))) dx dy \end{aligned} \quad (\text{A.3})$$

where ϵ prevents a zero data force term (i.e. to prevent impeding curve evolution). We selected $\epsilon = 0.001$.

We determine $\phi(x, y)$ that minimizes (A.3) by using the Euler-Lagrange equation to solve the gradient descent PDE :

$$\frac{\partial \phi}{\partial t} = -\frac{\partial E}{\partial \phi} = -\left[\frac{\partial L}{\partial \phi} - \frac{d}{dx} \frac{\partial L}{\partial \phi_x} - \frac{d}{dy} \frac{\partial L}{\partial \phi_y} \right] = 0 \quad (\text{A.4})$$

where

$$\begin{aligned} L &= w(x, y) \delta(\phi(x, y)) |\nabla \phi(x, y)| + \\ &+ (1 - w(x, y) + \epsilon) |I(x, y) - c_1|^2 H(\phi(x, y)) + \\ &+ (1 - w(x, y) + \epsilon) |I(x, y) - c_2|^2 (1 - H(\phi(x, y))) \end{aligned} \quad (\text{A.5})$$

and where we use the notation $\phi_x = \frac{\partial \phi}{\partial x}$.

We first determine the partial derivative $\frac{\partial L}{\partial \phi}$ as follows:

$$\begin{aligned} \frac{\partial L}{\partial \phi} &= \frac{\partial}{\partial \phi} w(x, y) \delta(\phi(x, y)) |\nabla \phi(x, y)| + \\ &+ \frac{\partial}{\partial \phi} (1 - w(x, y) + \epsilon) |I(x, y) - c_1|^2 H(\phi(x, y)) + \\ &+ \frac{\partial}{\partial \phi} (1 - w(x, y) + \epsilon) |I(x, y) - c_2|^2 (1 - H(\phi(x, y))) \end{aligned} \quad (\text{A.6})$$

which in expanded format is:

$$\begin{aligned} \frac{\partial L}{\partial \phi} &= \frac{\partial}{\partial \phi} w(x, y) \delta(\phi(x, y)) |\nabla \phi(x, y)| + \\ &+ \frac{\partial}{\partial \phi} (1 - w(x, y) + \epsilon) |I(x, y) - c_1|^2 H(\phi(x, y)) + \\ &+ \frac{\partial}{\partial \phi} (1 - w(x, y) + \epsilon) |I(x, y) - c_2|^2 - \frac{\partial}{\partial \phi} (1 - w(x, y) + \epsilon) |I(x, y) - c_2|^2 H(\phi(x, y)) \end{aligned} \quad (\text{A.7})$$

We use the property

$$\frac{\partial}{\partial \phi} H(\phi(x, y)) = \delta(\phi(x, y)) \quad (\text{A.8})$$

and the fact that

$$\frac{\partial}{\partial \phi}(1 - w(x, y) + \epsilon) |I(x, y) - c_2|^2 = 0 \quad (\text{A.9})$$

to simplify (A.7) as follows:

$$\begin{aligned} \frac{\partial L}{\partial \phi} &= w(x, y) \delta_\phi(\phi(x, y)) |\nabla \phi(x, y)| + \\ &+ (1 - w(x, y) + \epsilon) \delta(\phi(x, y)) \left[|I(x, y) - c_1|^2 - |I(x, y) - c_2|^2 \right] \end{aligned} \quad (\text{A.10})$$

We then determine $\frac{\partial L}{\partial \phi_x}$ by:

$$\begin{aligned} \frac{\partial L}{\partial \phi_x} &= \frac{\partial}{\partial \phi_x} w(x, y) \delta(\phi(x, y)) |\nabla \phi(x, y)| + \\ &+ \frac{\partial}{\partial \phi_x} (1 - w(x, y) + \epsilon) \delta(\phi(x, y)) |I(x, y) - c_1|^2 H(\phi(x, y)) + \\ &+ \frac{\partial}{\partial \phi_x} (1 - w(x, y) + \epsilon) \delta(\phi(x, y)) |I(x, y) - c_2|^2 (1 - H(\phi(x, y))) \end{aligned} \quad (\text{A.11})$$

Note that

$$\frac{\partial}{\partial \phi_x} (1 - w(x, y) + \epsilon) \delta(\phi(x, y)) |I(x, y) - c_1|^2 H(\phi(x, y)) = 0 \quad (\text{A.12})$$

and

$$\frac{\partial}{\partial \phi_x} (1 - w(x, y) + \epsilon) \delta(\phi(x, y)) |I(x, y) - c_2|^2 (1 - H(\phi(x, y))) = 0, \quad (\text{A.13})$$

Additionally, we note that

$$\begin{aligned} &\frac{\partial}{\partial \phi_x} w(x, y) \delta(\phi(x, y)) |\nabla \phi(x, y)| \\ &= w(x, y) \delta(\phi(x, y)) \frac{\partial}{\partial \phi_x} |\nabla \phi(x, y)| \\ &= w(x, y) \delta(\phi(x, y)) \frac{2\phi_x}{2\sqrt{\left(\frac{\partial \phi}{\partial x}\right)^2 + \left(\frac{\partial \phi}{\partial y}\right)^2}} \\ &= w(x, y) \delta(\phi(x, y)) \frac{\phi_x}{|\nabla \phi(x, y)|} \end{aligned} \quad (\text{A.14})$$

Thus we can simplify (A.11) to

$$\frac{\partial L}{\partial \phi_x} = w(x, y) \delta(\phi(x, y)) \frac{\phi_x}{|\nabla \phi(x, y)|} \quad (\text{A.15})$$

Similarly, we obtain the following for $\frac{\partial L}{\partial \phi_y}$:

$$\frac{\partial L}{\partial \phi_y} = w(x, y) \delta(\phi(x, y)) \frac{\phi_y}{|\nabla \phi(x, y)|} \quad (\text{A.16})$$

We then take the derivative of (A.15) with respect to x as follows:

$$\frac{d}{dx} \left(\frac{\partial L}{\partial \phi_x} \right) = \frac{d}{dx} \left(w(x, y) \delta(\phi(x, y)) \frac{\phi_x}{|\nabla \phi(x, y)|} \right) \quad (\text{A.17})$$

Using the product rule

$$\frac{d(AB)}{dx} = A \frac{dB}{dx} + B \frac{dA}{dx}, \quad (\text{A.18})$$

and noting that due to the chain rule,

$$\frac{d}{dx} \delta(\phi(x, y)) = \delta_\phi(\phi(x, y)) \phi_x, \quad (\text{A.19})$$

we expand (A.17) as follows:

$$\begin{aligned} \frac{d}{dx} \left(\frac{\partial L}{\partial \phi_x} \right) &= \frac{d}{dx} (w(x, y)) \delta(\phi(x, y)) \frac{\phi_x}{|\nabla \phi(x, y)|} + \\ &+ w(x, y) \frac{d}{dx} (\delta(\phi(x, y))) \frac{\phi_x}{|\nabla \phi(x, y)|} + w(x, y) \delta(\phi(x, y)) \frac{d}{dx} \left(\frac{\phi_x}{|\nabla \phi(x, y)|} \right) \\ &= w_x(x, y) \delta(\phi(x, y)) \frac{\phi_x}{|\nabla \phi(x, y)|} + w(x, y) \delta_\phi(\phi(x, y)) \phi_x \frac{\phi_x}{|\nabla \phi(x, y)|} + \\ &\quad + w(x, y) \delta(\phi(x, y)) \frac{d}{dx} \frac{\phi_x}{|\nabla \phi(x, y)|} \\ &= w_x(x, y) \delta(\phi(x, y)) \frac{\phi_x}{|\nabla \phi(x, y)|} + w(x, y) \delta_\phi(\phi(x, y)) \frac{\phi_x^2}{|\nabla \phi(x, y)|} + \\ &\quad + w(x, y) \delta(\phi(x, y)) \frac{d}{dx} \frac{\phi_x}{|\nabla \phi(x, y)|} \end{aligned} \quad (\text{A.20})$$

Similarly, we take the derivative of (A.16) with respect to y as follows:

$$\begin{aligned} \frac{d}{dy} \left(\frac{\partial L}{\partial \phi_y} \right) &= w(x, y) \delta_\phi(\phi(x, y)) \frac{\phi_y^2}{|\nabla \phi(x, y)|} + \\ &+ w_y(x, y) \delta(\phi(x, y)) \frac{\phi_y}{|\nabla \phi(x, y)|} + w(x, y) \delta(\phi(x, y)) \frac{d}{dy} \frac{\phi_y}{|\nabla \phi(x, y)|} \end{aligned} \quad (\text{A.21})$$

We then combine (A.20) and (A.21). We first note that

$$\frac{\phi_x^2 + \phi_y^2}{|\nabla \phi(x, y)|} = |\nabla \phi(x, y)| \quad (\text{A.22})$$

and thus

$$w(x, y) \delta_\phi(\phi(x, y)) \left(\frac{\phi_x^2}{|\nabla \phi(x, y)|} + \frac{\phi_y^2}{|\nabla \phi(x, y)|} \right) = w(x, y) \delta_\phi(\phi(x, y)) |\nabla \phi(x, y)| \quad (\text{A.23})$$

We also note that

$$w_x(x, y) \frac{\phi_x}{|\nabla \phi(x, y)|} + w_y(x, y) \frac{\phi_y}{|\nabla \phi(x, y)|} = \nabla w(x, y) \cdot \frac{\nabla \phi(x, y)}{|\nabla \phi(x, y)|}. \quad (\text{A.24})$$

Additionally, using (A.18) we note that

$$\begin{aligned} & \frac{d}{dx} \frac{\phi_x}{|\nabla \phi(x, y)|} + \frac{d}{dy} \frac{\phi_y}{|\nabla \phi(x, y)|} \\ = & \phi_x \frac{d}{dx} \left(\frac{1}{|\nabla \phi(x, y)|} \right) + \frac{1}{|\nabla \phi(x, y)|} \frac{d}{dx} \phi_x + \phi_y \frac{d}{dy} \left(\frac{1}{|\nabla \phi(x, y)|} \right) + \frac{1}{|\nabla \phi(x, y)|} \frac{d}{dy} \phi_y \\ = & \left[\phi_x \frac{d}{dx} \left(\frac{1}{|\nabla \phi(x, y)|} \right) + \phi_y \frac{d}{dy} \left(\frac{1}{|\nabla \phi(x, y)|} \right) \right] + \frac{1}{|\nabla \phi(x, y)|} \left[\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} \right] \\ = & \nabla \phi \cdot \nabla \frac{1}{|\nabla \phi(x, y)|} + \frac{1}{|\nabla \phi(x, y)|} \text{div}(\nabla \phi) \end{aligned} \quad (\text{A.25})$$

Since the divergence of a scalar function γ and a vector \mathbf{F} is as follows:

$$\text{div}(\gamma \mathbf{F}) = (\nabla \gamma) \cdot \mathbf{F} + \gamma \text{div}(\mathbf{F}), \quad (\text{A.26})$$

we can simplify (A.25) by using the substitution $\gamma = \frac{1}{|\nabla \phi(x, y)|}$ and $\mathbf{F} = \nabla \phi$ as follows:

$$\frac{d}{dx} \frac{\phi_x}{|\nabla \phi(x, y)|} + \frac{d}{dy} \frac{\phi_y}{|\nabla \phi(x, y)|} = \text{div} \left(\frac{\nabla \phi(x, y)}{|\nabla \phi(x, y)|} \right) \quad (\text{A.27})$$

Thus, we note

$$\begin{aligned} & w(x, y) \delta(\phi(x, y)) \left(\frac{d}{dx} \frac{\phi_x}{|\nabla \phi(x, y)|} + \frac{d}{dy} \frac{\phi_y}{|\nabla \phi(x, y)|} \right) \\ = & w(x, y) \delta(\phi(x, y)) \text{div} \left(\frac{\nabla \phi(x, y)}{|\nabla \phi(x, y)|} \right) \end{aligned} \quad (\text{A.28})$$

From (A.23), (A.24), and (A.28), we combine (A.20) and (A.21) as follows:

$$\begin{aligned} & \frac{d}{dx} \left(\frac{\partial L}{\partial \phi_x} \right) + \frac{d}{dy} \left(\frac{\partial L}{\partial \phi_y} \right) = w(x, y) \delta(\phi(x, y)) |\nabla \phi(x, y)| + \\ & + \delta(\phi(x, y)) \nabla w(x, y) \cdot \frac{\nabla \phi(x, y)}{|\nabla \phi(x, y)|} + w(x, y) \delta(\phi(x, y)) \text{div} \left(\frac{\nabla \phi(x, y)}{|\nabla \phi(x, y)|} \right) \end{aligned} \quad (\text{A.29})$$

We substitute (A.10) and (A.29) into the Euler-Lagrange (A.4) to obtain the final evolution equation that corresponds to the functional (A.3).

$$\begin{aligned} \frac{\partial \phi}{\partial t} = & \delta(\phi(x, y)) \nabla w(x, y) \cdot \frac{\nabla \phi(x, y)}{|\nabla \phi(x, y)|} + w(x, y) \delta(\phi(x, y)) \text{div} \left(\frac{\nabla \phi(x, y)}{|\nabla \phi(x, y)|} \right) + \\ & -(1 - w(x, y) + \epsilon) \delta(\phi(x, y)) \left[|I(x, y) - c_1|^2 - |I(x, y) - c_2|^2 \right] \end{aligned} \quad (\text{A.30})$$

References

1. Chan, T.F., Vese, L.A.: Active contours without edges. IEEE Transactions on Image Processing **10** (2001) 266–277 [1](#)