Decoupled Active Contour (DAC) for Boundary Detection

A. K. Mishra, *Student Member, IEEE,* P. W. Fieguth, *Member, IEEE,* and D. A. Clausi, *Senior Member, IEEE*

***Abstract*— The accurate detection of object boundaries via active contours is an ongoing research topic in computer vision. Most active contours converge towards some desired contour by minimizing a sum of internal (prior) and external (image measurement) energy terms. Such an approach is elegant, but suffers from a slow convergence rate and frequently mis-converges in the presence of noise or complex contours.**

To address these limitations, a decoupled active contour (DAC) is developed which applies the two energy terms separately. Essentially, the DAC consists of a measurement update step, employing a Hidden Markov Model (HMM) and Viterbi search, and then a separate prior step, which modifies the updated curve based on the relative strengths of the measurement uncertainty and the non-stationary prior. By separating the measurement and prior steps, the algorithm is less likely to mis-converge; furthermore, the use of a Viterbi optimizer allows the method to converge far more rapidly than energy-based iterative solvers.

The results clearly demonstrate that the proposed approach is robust to noise, can capture regions of very high curvature, and exhibits limited dependence on contour initialization or parameter settings. Compared to five other published methods and across many image sets, the DAC is found to be faster with better or comparable segmentation accuracy.

***Index Terms*— Snake, deformable model, active contour, importance sampling, Viterbi algorithm, statistical data fusion**

I. Introduction

The task of locating exact boundaries of objects in cluttered and noisy environments has many applications in object tracking [1], content based image and video retrieval [2], [3], robotics [4], image composition [5], [6] and biomedical engineering [7], [8]. Energy minimizing splines, also known as deformable snakes or active contours [9], [10], [11], [12], [13], are the key approaches in the computer vision literature for such boundary extraction problems.

The principal idea in active-contour based boundary extraction is to minimize the sum of internal (prior) and external (image-based) energies to obtain an optimum boundary. The internal energy typically asserts a first- or second-order smoothness constraint on the boundary, whereas the external energy applies a "force" on the boundary, creating an attractive force, typically towards areas of high gradient. Since the original development of snake methods [9], a great many variations have been developed, falling broadly into three classes: parametric [9], [14], [15], [16], [17], [13], [1], [18], [19], non-parametric or geometric active contour [11], [10], [20], [21], [22], [23] and physics inspired particle based [24], [25].

Despite this large number of approaches, none of the parametric active contours are able to handle the problems associated with image noise contamination, complex high curvature boundaries, algorithm parameter sensitivity, initialization sensitivity, ineffective stopping criteria and slow convergence rate. Non-parametric approaches are initialization independent and are able to handle high curvature regions and topology naturally. However, non-parametric active contours are not able to extract the boundaries of objects having open or broken edges. Furthermore, non-parametric active contours are comparatively slower and more sensitive to noise compared to parametric methods.

In tackling these issues, our research led to a novel parametric decoupled active contour (DAC) method, sharing the same formulation and origins of conventional deformable models. However, instead of discretizing and iteratively minimizing the total energy, as in most approaches, our method alternately minimizes the external energy within a specified region, then separately asserts the prior constraints to force the boundary to satisfy required smoothness. The adaptivity of the method to sharp corners is satisfied by importance sampling [26] on the basis of curvature. Although the DAC is parametric, there is limited parameter sensitivity as the parameter values are derived implicitly from the image curvature and gradient.

To validate the DAC's claims, we have evaluated the performance of DAC compared to five state-of-the-art segmentation algorithms [9], [16], [17], [23], [21], [27], using three test approaches: a comprehensive quantitative segmentation accuracy evaluation on the Weizmann database [27], a quantitative evaluation of boundary identification accuracy and convergence speed using natural and synthetic images, and an evaluation of DAC capabilities through illustrative examples. The experimental results show a dramatic improvement of the DAC compared to existing parametric active contours, across all five criteria of high curvature, noise, parameter sensitivity, initialization, and speed. Compared to nonparametric approaches, DAC performed better in finding single-object boundaries in the presence of noise and background clutter at a lower computational cost.

The remainder of the paper is organized as follows. Section II reviews existing active contours. Section III explains the proposed methods, with results and comparisons shown in Section IV.

II. Background

This section reviews the different type of active contour models, their solution, and corresponding limitations.

A. Active Contour Models

Parametric or implicit active contour models were first purposed by Kass et al. [9], implicitly expressed as an energy minimizing curve $v(s) = (x(s), y(s))$, $s \in [0, 1]$, with normalized arclength s. The goal is to deform the spline to minimize the

A. K. Mishra, P. W. Fieguth, and D. A. Clausi are with the Vision and Image Processing (VIP) Research Group, Systems Design Engineering, University of Waterloo, Waterloo, ON, Canada.

energy functional:

$$E = \int_0^1 \left(\underbrace{\alpha(s)v_s^2(s)}_{Elastic} + \underbrace{\beta(s)v_{ss}^2(s)}_{Thin-plate} - \underbrace{\gamma(s)E_{ext}(v(s))}_{External} \right) ds \quad (1)$$

where $v_s(s)$ and $v_{ss}(s)$ are the first and second derivatives of $v(s)$ with respect to arclength s, and the parameters $\alpha(s), \beta(s)$ and $\gamma(s)$ are the weight factors for penalties on slope, curvature and the external force of the active contours, respectively. Further, $\alpha(s), \beta(s)$ and $\gamma(s)$ may be a function of spatial location, but are constants in practice.

The internal energy, representing the prior, is some weighted sum of elastic and thin-plate energies, whereas the external potential is computed from the image I in a manner dependant upon the application. Typically, the contour should converge to image edges, so the energy is made a function of the magnitude of the image gradient (g)

$$E_{ext} = (g(I))^2 = (|\delta G_\sigma * I|)^2 \quad (2)$$

for some first derivative of Gaussian (δG_σ) with bandwidth σ, where $*$ is the convolution operator.

The curve deforms under the (typically opposing) influences of internal and external energies. The internal energy seeks to make the curve smooth, while the external energy forces the curve towards object boundaries. We identify three inherent problems with the classical snake:

(i) There is a delicate balance required between the energy parameters $\alpha(s), \beta(s)$ and $\gamma(s)$, and it is typical for parameter settings to lead to mis-convergence.

(ii) The standard external and internal forces are not able to pull the snake into concave regions.

(iii) In many cases (1) is numerically stiff, meaning that the iterative solution of (1) tends to be slow, and this worsens if the active contour is initialized far from the true solution, implying a limited capture range.

To increase the convergence speed and capture range, most variations on the parametric active contour have concentrated on altering the external energy, such as the pressure based balloon force [28], distance transformed image gradient [14], [15], gradient vector flow [16] and vector field convolution [17] of image gradient. Leymarie et al. [1] suggested a better termination criterion with detailed analysis of various parametric snake models for tracking applications. For handling contours of high curvature, Wong et al. [18] proposed a segmented snake model, which follows a dual optimization approach, with a rough estimate obtained using a classical snake, followed by a recursive split and merge approach.

A fundamentally separate approach is formulated in geometric terms, such as the geometric active contour (GAC) of Malladi et al. [11] and Caselles et al. [10]. The contour is described in the level-set framework of Osher et al. [29], which allows curve splitting and merging to be handled more naturally than with a spline. Caselles et al. [30] and Siddiqi et al. [31] reformulate the level-set framework by introducing gradient-weighted length and area terms to the total energy. To avoid local minima, Cohen et al. [32] proposed a minimal path approach to find the global minima associated with the active contour. Mumford-Shah [33] and Chane-Vese [20], [21] developed another region based active contour model which can be solved in a level set [29] framework. The Chane-Vese model does not use the image gradient as its stopping criterion, and so works efficiently even if there are very weak edges. Recently, Sundaramoorthi et al. [34] have identified the problems associated with geometric active contour energy minimization techniques using Riemannian space. They have reformulated a generic geometric active contour by redefining the notion of gradient using a Sobolev type inner product, while using the level set methods as the evolution framework. To improve computational complexity, Bresson et al. [23] proposed a new global optimization method, the fast active contour (FAC), similar to the GAC model, but with a dual optimization method. Eric et al. [35] tried a hybrid approach to improve the convergence speed and initialization robustness of level set based active contour by combining k-means clustering with the level set framework.

Recently, a new category of active contour models have been proposed, motivated by laws of physics. Examples include the gravitational force active contour model [15], the charged particle model active contour [24] (CPM), and the magneto-static active contour model (MAC) [25]. The CPM considers each pixel to be a charged particle attracted by electric fields generated from the image gradients, claimed to reduce snake initialization sensitivity. Xie et al. [25] claim that the MAC more effectively captures complex boundaries than the CPM, however the computational and algorithmic complexities of the MAC are high.

B. Solution Techniques

The solution to parametric, geometric and physically motivated active contour models are generally obtained via a discretized optimization of some energy, such as in (1). Many algorithms follow from an Euler-Lagrange approach, essentially leading to a gradient descent optimization of the energy function. Some geometric and physical models may be solved in higher dimensions using level-set approaches [11], [25].

A wide variety of algorithms have been proposed to address convergence stability and complexity, such as the aforementioned FAC [23]. To improve stability, Amini et al. [12] first proposed dynamic programming to perform the active contour energy optimization, later used more broadly [7], [36], [37], [38]. Williams and Shah [39] introduced a fast greedy approach to find global minima of energy functionals.

Because completely unsupervised segmentations based on dynamic programming can converge to unwanted boundaries, Mortensen et al. [6] first introduced intelligent scissors – a user-interactive dynamic programming based graph search method to locate exact boundaries. Further, GrabCut [40], Lazy snapping [5] and Bayesian matting [41] were introduced for user-interactive image segmentation as well.

C. Limitations of Existing Active Contours

The motivation for DAC stem from the following five basic limitations associated with existing active contour solutions:

1) **Robustness to Noise and Background Clutter:** Most active contour solutions are susceptible to local minima, many of which are created by the presence of noise and background clutter. In order to avoid the convergence of the snake towards noise and background clutter Nascimento et al. [42] proposed a robust snake based upon stroke grouping and expectation maximization (EM), but such methods are very time consuming, parameter dependent

and insensitive to high curvature boundaries. Alternatively, denoising models have been incorporated into the active contour formulation [23].

2) **High Curvature Object Boundaries:** A key challenge in prior-model based parametric active contours is the capture of regions of high curvature, since most prior models penalize curvature, but decreased curvature penalties lead to increased noise sensitivity. Wong et al. [18] proposed a segmented snake, splitting the snake into a number of separated regions based on curvature, followed by later merging. Analogous to a weakened prior, the split/merge snake is similarly sensitive to noise.
3) **Parameter Dependency:** Parameter tuning is a common problem with active contour algorithms [9], [16], [17]. The selection of appropriate parameters is generally tedious and image dependent, and the parameter sensitivity precludes the use of such algorithms in certain applications.
4) **Sensitivity to Initialization:** The classical parametric snake is sensitive to initial position, normally manually supervised, and often fails to converge or takes inordinately long to converge. To avoid these issues Cohen et al. [28], Xu et al. [16] and Li and Acton [17] proposed the balloon force, gradient vector flow snakes and vector field convolution snakes to extend the capture range, however these two methods suffer from lower convergence rates and increased noise sensitivity.
5) **Stopping Criteria and Rate of Convergence:** The active contour evolves iteratively to minimize some energy functional, ideally stopping once the global minimum is reached. Because of noise and image clutter the energy functional is not smooth, causing algorithms to be either trapped or delayed for many iterations at local minima. To have a criterion by which the iterative solver is halted is necessary, or global optimizers need to be developed such as those based on dynamic programming [12], [6], simulated annealing [38], or graph search methods [40]. Both the iterative solvers and global optimizers are slow to converge. Although active contours might otherwise be a promising approach for image segmentation and tracking, the slow rate of convergence precludes active contours from being considered in real-time applications.

III. Decoupled Active Contour (DAC)

To overcome the five limitations mentioned in Section II-C, a novel unsupervised parametric decoupled active contour (DAC) has been designed and built for identifying the boundary of a single object. Because of the slow rate of convergence of standard active contour methods, it would be desirable to find an optimal, non-iterative solution. A non-iterative solution can be identified, in principle, via a Bayesian perspective, with the internal, external and total energy of the active contour analogous to the prior, measurement and posterior of a random field. Minimizing the total energy of the active contour is equivalent to maximizing the posterior (MAP problem), a complex solution space with a significant chance of being trapped in a local minimum. The complex, nonlinear nature of the MAP problem makes an optimal, non-iterative method essentially impossible, however the key insight is that the two parts of the snake criterion (1) can, in fact, be individually locally optimized in a non-iterative fashion. Although the overall algorithm remains iterative, alternating between the external and internal criteria, each iteration converges far more rapidly than the direct iterative solutions to (1). Therefore, DAC simplifies the optimization step by first ignoring the prior and employing a Viterbi algorithm to obtain a sub-optimal solution that is consistent with the measurements, and secondly by asserting the prior via a Bayesian linear least squares estimator.

The DAC method consists of three steps, illustrated in Fig. 1:

A. **Hidden Markov Model (HMM):** The visual boundary finding problem is modeled as an HMM and a Viterbi search is used to find the solution by dynamic programming. In the absence of image noise and shape prior, the Viterbi [43] search will identify all of the strongest local boundaries. Details are developed in Section III-A.

B. **Importance Sampling on Curvature:** If snake points are uniformly spaced on the curve, there will be an excess of points in areas of gentle curvature and too few points in areas of high curvature. To make a single algorithm work in both smooth and high-curvature portions of a curve, a non-stationary prior is essential, which is accomplished by placing more snake points in high-curvature areas, weakening the prior, and fewer in smooth areas, strengthening the prior. Therefore the curvature of the Viterbi boundary is computed and importance sampled to generate non-uniform samples. Details are developed in Section III-B.

C. **Statistical Data Fusion:** The non-stationary prior constraints need to be traded-off against the strength and significance of the image gradients. The fused curve is estimated statistically, as described in Section III-C.

A. Visual Boundary Extraction Using HMM

We seek a non-iterative solution to finding a curve $v(s)$ to minimize E_{ext} of (1). To avoid undoing the work of previous iterations, we wish to constrain the search space to seek an optimum in the vicinity of $v^{\mathrm{p}}(s)$, the solution from the previous iteration. The visual boundary extraction can be formulated as an HMM on a discrete space, which allows the solution to be found using a standard Viterbi search. We begin by discretizing the snake, a collection of $q-1$ discrete straight segments with q discrete locations (the head and tail of the snake do not have to be connected):

$$v\left(s_j\right) = v_{z_j}\big|_{z_j=0} = \left(x_{j,0}, y_{j,0}\right),\ s_j \in [0,1],\ j \in [1,q] \qquad (3)$$

As illustrated in Fig. 2, at each of the q discretized snake locations a set of $u+1$ points are defined, lying normal to the curve. Constant-length normals lead to the possibility of a self-intersecting curve. To deal with this problem, we initially tried an approach similar to dual front propagation technique [44], where a medial axis growing and shrinking technique was used to generate a search space. However, such an approach is computationally expensive and fails to grow normals outwards from the curve in areas of high curvature. Instead, we iteratively test and prune the length of the normals until we obtain nonintersecting normals.

Given the set of non-intersecting normals, the exhaustive set of $(u+1)^q$ possible snakes over which to optimize may be represented as the possible transitions through an ordered graph, as shown in Figs. 3 and 4. We define

$$z_j \in \left[-\frac{u}{2}, \ldots, \frac{u}{2}\right],\ \ j \in [1,q],\ \ u \text{ is even} \qquad (4)$$

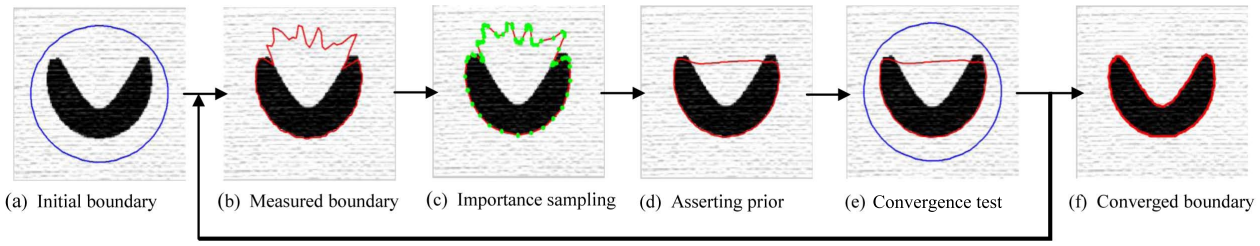


Fig. 1. Steps involved in DAC on a synthetic V-shaped object. (a) The blue circle shows the initial snake. (b) The red line shows the sub-optimal snake after a Viterbi search. (c) The small green dots are the particles generated using curvature-based importance sampling on the curvature of the sub-optimal Viterbi snake. (d) The red line shows the estimated snake following the Bayesian estimator. (e) A test of convergence between the current and previous snakes. If not converged, return to step (b). (f) The converged snake.

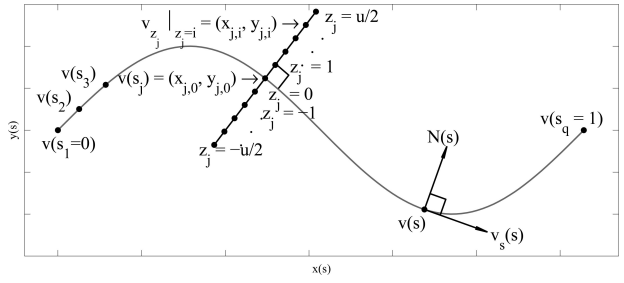


Fig. 2. Description of DAC terminology. $s_j \in [0,1]$ where $j \in [1,q]$ is the discretized arclength along the active contour; v is the position in (x,y) space; $v_s(s)$ and $N(s)$ are the tangent and normal vectors at $v(s)$. The jth normal is constructed at s_j, with $u+1$ possible values (small circles) for the snake at that normal.

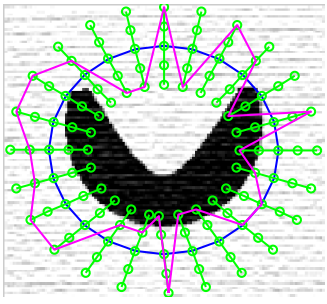


Fig. 3. Circular trellis with an initial snake (blue circle), one possible, albeit unlikely snake (magenta), normals (green lines) and nodes (small circles along the normals).

to represent the jth state, such that z_j selects which of the $u+1$ evenly distributed points a given snake passes through at the jth normal. Given a sequence of states

$$\underline{z} = [z_1, z_2, \ldots, z_q], \tag{5}$$

if we can determine the energy of this state sequence, then we have a basis to find the optimal $\underline{z}$. Efficient solvers of such graph problems, such as the Viterbi method [43], allow a cost to be associated with each state value z_j, and with each state transition $z_j \rightarrow z_{j+1}$. It is possible to assert prior models with penalties involving more than two successive state values (such as a penalty on curvature (1), which requires *three* successive points):

- We can let each state represent a *pair* of points, with the consequence that the size of the graph explodes (see right panel of Fig. 4), or
- We can separate the optimizing of E_{ext} from the assertion of the prior, as we have proposed for the DAC, and just use a simpler graph (see left panel of Fig. 4).

The HMM will therefore be limited to simple inter-point constraints. Since the snake prior is to be asserted later, at this point only the length of the snake is penalized, such that the state transition probability is represented by

$$\begin{aligned} p(z_j, z_{j+1}) &= \frac{1}{\sqrt{2\pi}\sigma_s} \exp\left(-\frac{\nabla v(z_j, z_{j+1})}{2\sigma_s^2}\right) \\ &= \frac{1}{\sqrt{2\pi}\sigma_s} \exp\left(-\frac{\left\| v_{z_j} - v_{z_{j+1}} \right\|}{2\sigma_s^2}\right) \end{aligned} \tag{6}$$

where

$$v_{z_j}\big|_{z_j=i} = (x_{j,i}, y_{j,i}) \tag{7}$$

represents the points along the snake, and therefore $\|\nabla v\|$ is the corresponding arclength between two successive points on the snake. The term σ_s represents the spatial standard deviation and is computed from the distribution of Euclidian distances between all pairs of points between the jth and $(j+1)$th normals. An alternate transition probability might be the integral of edge map weighted arclength ($\int_{v_{z_j}}^{v_{z_{j+1}}} \phi ds$) between v_{z_j} and $v_{z_{j+1}}$, where $\phi = 1/(1+g)$ is the edge map of the image. This has the benefit of evaluating the edge map along each arc, rather than only at discrete points, however computing the edge map information is a computationally intensive process, and in practice was found to offer little to no benefit over the proposed approach based on Euclidian distance.

Next, the state probability for z_j must be related to the external energy (normally related to the image gradient) at the state location v_{z_j}, thus the state probability is defined as

$$p(\psi(z_j)|z_j) = \frac{1}{\sigma_{\psi(z_j)}} \exp\left(-\frac{\psi(z_j)}{\sigma_{\psi(z_j)}}\right) \tag{8}$$

where the measurement is defined as

$$\psi(z_j) = \frac{1}{1 + E_{ext}(z_j)} \tag{9}$$

and $\sigma_{\psi(z_j)}$ is the standard deviation of the measurements along the normal j and computed locally from $\psi(z_j)$. $E_{ext}(z_j)$ is calculated as per (2).

At this point we have a first-order lattice (Fig. 4, left) with state and state-transition probabilities defined. In principle, the optimum contour can be found by solving the joint maximization

$$\max_{z_1, z_2, \cdots z_q} p\big(z_1, z_2, \cdots, z_q\big). \tag{10}$$

However in practice there is no need to find the optimum, particularly because the proposed algorithm remains iterative, and

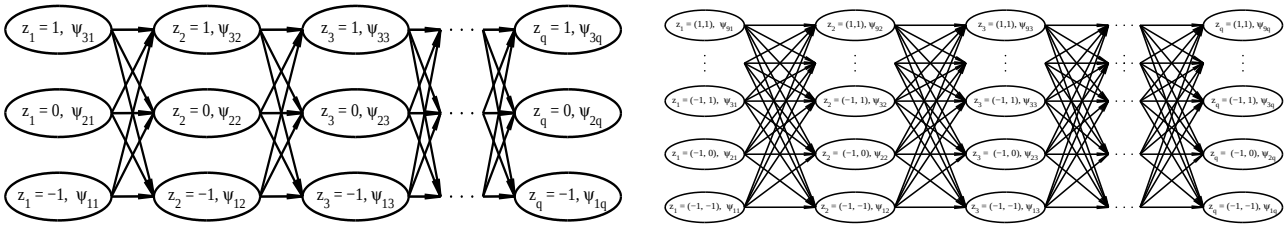


Fig. 4. First and second order trellis, where $z_1, z_2, \ldots, z_q$ are the hidden states, and $\psi(z_j) = \psi_{z_j=i,j}$ are the observations. The second-order trellis, right, has a far higher representation, storage, and computational complexity than the first-order trellis, left. In both cases, $u = 2$.

Algorithm 1 $[\underline{v}^{\mathrm{v}}]$ = Function Viterbi($\underline{z}, q, u$) [43]

Initialization: j=1, $\Theta(z_1) = p(\psi(z_1)|z_1)$, $\Gamma(z_1) = \{0\}$ and $z_1 = [-\frac{u}{2} \cdots \frac{u}{2}]$
while $j \leq q$ **do**
  $\Theta\left(z_{j+1}\right) = \max_{z_j} \left(\Theta(z_j) \times p(z_j, z_{j+1})\right) \times p(\psi(z_j)|z_j)$
  $\Gamma(z_{j+1}) = \arg\max_{z_j} \left(\Theta(z_j) p(z_j, z_{j+1})\right)$
  $j = j + 1$
end while
Termination: $z_q^{\mathrm{v}} = \arg\max_{z_q} \left(\Theta(z_q)\right), v_q^{\mathrm{v}} = (x_{q,z_q^{\mathrm{v}}}, y_{q,z_q^{\mathrm{v}}})$
$j = q - 1$
while $j > 2$ **do**
  $z_j^{\mathrm{v}} = \Gamma(z_{j+1}^{\mathrm{v}}), v_j^{\mathrm{v}} = (x_{j,z_j^{\mathrm{v}}}, y_{j,z_j^{\mathrm{v}}})$
  $j = j - 1$
end while

because (10) is only one part of the complete criterion (1). Therefore finding a sub-optimal path using the Viterbi algorithm [43], as described in Algorithm. 1 is preferable.

Since the Viterbi search is a sub-optimal algorithm, it is dependent on both the initial state z_1 and the ordering of the vertices. In principle this dependence could be removed by minimizing over initial state and ordering, however experimentally we found that this adds little in terms of accuracy, but greatly in terms of computational burden. By increasing the computational cost of the Viterbi method from $O(u^2 q)$ to $O(u^2 q^2)$, where typically $q = 300$, and where the Viterbi method accounts for approximately half of the total complexity of the DAC, this minimization slows the method by a factor of approximately 150. Given the limited benefit and considerable cost, and given the robust convergence seen in the results, we propose to use a regular Viterbi method.

The state sequence $z_j^{\mathrm{v}}, j \in [1, q]$ resulting from the Viterbi optimization gives rise to the sub-optimal curve

$$v_j^{\mathrm{v}} = v^{\mathrm{v}}(s_j) = (x_{j,z_j^{\mathrm{v}}}, y_{j,z_j^{\mathrm{v}}}),\ j \in [1, q]. \tag{11}$$

B. Curvature Guided Importance Sampling

Active contours generally consider stationary elastic and thin-plate constraints to constitute the prior model of the object. Such a prior represents a poor hypothesis for complex, high curvature objects because the degree to which prior smoothness should be asserted varies with the boundary curvature. Therefore, the DAC approximates the smoothness of complex boundaries using stationary elastic and rigidity constraints, but with non-stationary sampling intervals of the active contour points along the snake, which will ensure more samples (with correspondingly shorter active contour segments) near high curvature regions.

The curvature κ of a parametric curve $v(s) = (x(s), y(s))$ is expressed in terms of the tangent vector $v_s(s)$, such that the curvature is related to the rate of change of the tangent vector

$$\kappa(s) \times N(s) = v_{ss}(s) \tag{12}$$

where $N(s)$ is the normal vector to the parametric curve $v(s)$, and where v_s, v_{ss} represent the first and second derivatives of v. From [45], (12) can be rewritten as

$$\kappa(s) = \frac{x_s(s) y_{ss}(s) - y_s(s) x_{ss}(s)}{(x_s^2 + y_s^2)^{3/2}} \tag{13}$$

where the derivatives are computed numerically using forward differences. Given the Viterbi boundary $\underline{v}^{\mathrm{v}}$, we can use discrete derivatives in (13) to compute the sampled curvature $\underline{\kappa}$. We propose to resample, with a sampling density proportional to the negative exponential of curvature:

$$\Delta v(s) = \frac{c_1}{\sigma_\kappa} \exp\left(-\frac{|\kappa(s)|}{\sigma_\kappa}\right) \tag{14}$$

where $\Delta v(s)$ is the discrete arclength of an individual segment of curve $v(s)$ and σ_κ is the standard deviation of sampled curvature κ. If we fix q, the total number of points, then the proportionality parameter c_1 is set to preserve the arc-length of the curve.

Since arbitrarily high curvature is possible, leading to arbitrarily fine discretization, for practical reasons we seek to limit the discretization interval as

$$\Delta v^m(s_j) = \begin{cases} \Delta v_{\max} & \text{if } \Delta v(s_j) > \Delta v_{\max} \\ \Delta v_{\min} & \text{if } \Delta v(s_j) < \Delta v_{\min} \\ \Delta v(s_j) & \text{otherwise,} \end{cases} \tag{15}$$

where

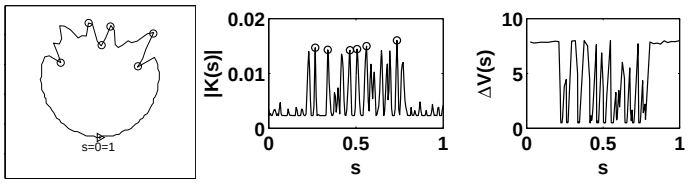
$$\frac{1}{\Delta v_{\max}} \ll q \ll \frac{1}{\Delta v_{\min}} \tag{16}$$

to generate q samples $\underline{v}^m$ having a sampling interval of $\Delta v^m(s_j)$.

Fig. 5 illustrates the resampling procedure. The inverse relationship between curvature and sampling interval are displayed in panels (b) and (c). The maximal and minimal extents of the sampling interval are determined by the user specified values of $\Delta v_{\max} = 8$ and $\Delta v_{\min} = 0.5$.

C. Statistical Estimation

Because of the complexity of introducing all but the most trivial shape priors into the Viterbi boundary extraction, the calculation of $\underline{v}^v$ and $\underline{v}^m$ have not permitted the assertion of a meaningful prior. Therefore, to limit the computational complexity of the Viterbi step, but to retain the assertion of a prior, a separate step of fusing a shape prior into $\underline{v}^m$ is required. This fusion step is the subject of this section.



(a) Example snake (b) κ vs. s (c) $\Delta v(s)$ vs. s

Fig. 5. (a) A Viterbi boundary. (b) Absolute value of the curvature along the length of the snake. (c) Arclength of each segment. The red dots identify six prominent high curvature boundary points.

Because each element of $\underline{v}$ is, itself, a two-vector coordinate $v_j^m = (x_j^m, y_j^m)$, to talk about the estimation of $\underline{v}$ is mathematically ambiguous. Therefore, we will explicitly discuss the estimation of the components $\underline{x}, \underline{y}$. Let $\underline{v}^t = (\underline{x}^t, \underline{y}^t)$ be the *true* and unknown discretized snake points that need to be estimated, considering $\underline{v}^t$ to be a random vector with the components of $\underline{v}^t$ satisfying a prior model

$$\underline{x}^t \sim N(\underline{\mu}_x, P), \quad \underline{y}^t \sim N(\underline{\mu}_y, P). \tag{17}$$

The boundary constraint P follows from the prior term implied in (1). The extracted and resampled boundary $\underline{v}^m$ represents a measurement of $\underline{v}^t$, thus

$$\underline{x}^m = C_x \underline{x}^t + \underline{\nu}_x, \quad \underline{y}^m = C_y \underline{y}^t + \underline{\nu}_y, \tag{18}$$

where $C_x = C_y = I$, since each snake point is measured, and where $\underline{\nu} = [\underline{\nu}_x, \underline{\nu}_y]$ is the measurement noise, itself having statistics

$$\underline{\nu}_x, \underline{\nu}_y \sim N(0, R). \tag{19}$$

A Bayesian estimate [46], [47] of $\underline{v}$ for each component of $\underline{v}$ can be obtained by minimizing the expected error norm, such that

$$\begin{aligned} \underline{\hat{x}} &= \underline{\mu}_x + \left(R^{-1} + P^{-1}\right)^{-1} R^{-1} \left(\underline{x}^m - \underline{\mu}_x\right) \\ &= \underline{\mu}_x + K_g \left(\underline{x}^m - \underline{\mu}_x\right). \end{aligned} \tag{20}$$

The term $K_g = \left(R^{-1} + P^{-1}\right)^{-1} R^{-1}$ is the Kalman gain, weighting the measurement residual. The remaining task is to select $\underline{\mu}$, P and R, discussed in the next section.

D. Bayesian Model Determination

The active contour literature [9], [16], [28], [30] considers the constraints on boundary curvature as the prior model of the object boundary. In practice, such prior models are obtained using an extensive training approach [48]. However an unsupervised, broadly applicable, and non-stationary prior is preferred.

In practice, the continuous integral of some function of $v(s)$ is replaced with a discrete norm over the sampled snake $\underline{v}$:

$$\|A\underline{x}\|_2 + \|A\underline{y}\|_2 \approx \int_0^1 \left(\alpha(s) |v_s|^2 + \beta(s) |v_{ss}|^2\right) ds \tag{21}$$

Each row of A asserts some discretized constraint on $\underline{x}$ or $\underline{y}$, where A is banded (here penta-diagonal, for a second-order constraint). The inverse relationship between prior constraints and prior covariance lead to the constraints A implicitly specifying the prior as

$$P = \left(A^T A\right)^{-1}. \tag{22}$$

Although we treat the weighting factors $\alpha(s)$, $\beta(s)$ as constant, a non-stationary prior is acquired by placing the measured samples non-uniformly on the object boundary (as will be explained in Section III-E).

The deterministic portion of the prior model is the mean, $\underline{\mu}$. We would normally consider $\underline{\mu} = 0$, in the absence of any specific deterministic knowledge of the contour shape, however we can use $\underline{\mu}$ to create biases in snake evolution, leading to expansion or contraction forces. The mean $\underline{\mu} = (\underline{\mu}_x, \underline{\mu}_y)$ is defined as

$$\underline{\mu}_x = \underline{\mu}_x^c + (1 + \tau)\, \underline{\delta}_x, \tag{23}$$

and similarly for $\underline{\mu}_y$, where $\underline{\mu}^c$ is the center of mass of $\underline{v}^m$, and $\underline{\delta}$ is a circle of points with a radius equal to the average separation of $\underline{v}^m$ from $\underline{\mu}^c$. To avoid bias due to varying snake point densities, the shape center is normalized with respect to arc length:

$$\underline{\mu}_x^c = \left(\frac{\sum_{j=1}^{q} x_j^m |x_{j+1}^m - x_j^m|}{\sum_{j=1}^{q} |x_{j+1}^m - x_j^m|} \right). \tag{24}$$

The constant τ in (23) is a growth factor on the average radius, creating an expanding force if $\tau > 0$, and a contracting force if $\tau < 0$, motivated from the balloon force snake [28].

With a prior model $(\underline{\mu}, P)$ defined, the key question is the relative weighting of the measurements versus the prior in (20), a weighting which is controlled by diagonal covariance R, where

$$R_{jj} = r\left(v^m(s_j)\right). \tag{25}$$

We wish the variances to be a function of their respective image gradients, such that the larger the gradient magnitude g (2), the more certain the measurement, and the closer R is to zero. The limiting cases of the measurement variance should be approximated by

$$r(g) = \begin{cases} r_{\max} & \text{if } g = 0 \\ \frac{r_{\max}}{2} & \text{if } g = \mu_g \\ 0 & \text{if } g = \infty \end{cases}, \tag{26}$$

a relationship easily satisfied using

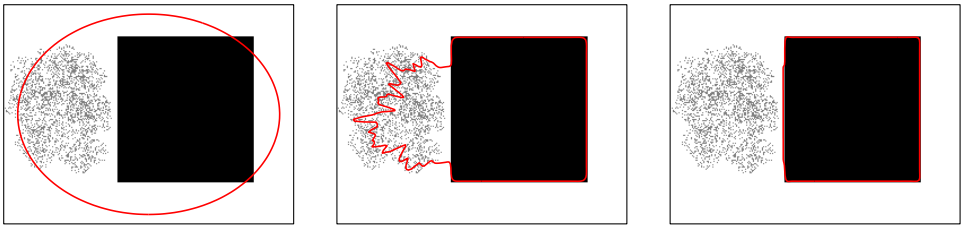
$$r(g) = r_{\max} \frac{f(g)}{1 + f(g)} \tag{27}$$

where

$$f(g) = \frac{1}{\sigma_g} \exp\left(-\frac{g - \mu_g}{\sigma_g}\right). \tag{28}$$

The rationale behind f is to scale r on the basis of the average gradient μ_g and standard deviation of gradients σ_g along the observed boundary $\underline{v}^m$ in the image. Because noise pixels do not form a continuous boundary, therefore near a noisy region the value of g is smaller, and that of σ_g larger, leading to a larger value of r, implying lower measurement certainty.

The effectiveness of this gradient-sensitive measurement variance is demonstrated in Fig. 6. In particular, Fig. 6(b) shows the convergence of the active contour by the use of fixed $R = I$, meaning that this fixed R fails to discriminate between the gradients generated by noise and by an object. In contrast, using the image-dependent R from (27), we weaken the measurements in noisy regions, giving more weight to the prior, pulling the snake away from such regions, leading to the excellent convergence shown in Fig. 6(c).



(a) Initialization (b) Fixed $R = I$ (c) Variable R from (27)

Fig. 6. Demonstration of the role of measurement noise co-variance (R) in the convergence of active contours in the presence of noise. Panel (a) shows the initial snake overlapping the square object of interest beside a noise cloud. Panel (b) shows the erroneous solution generated with fixed $R = I$. Panel (c) shows the correct solution using variable R from (27). Because the calculated gradient is spatially averaged, gradients near continuous (true) boundaries are stronger than those at discontinuous (noisy) ones.

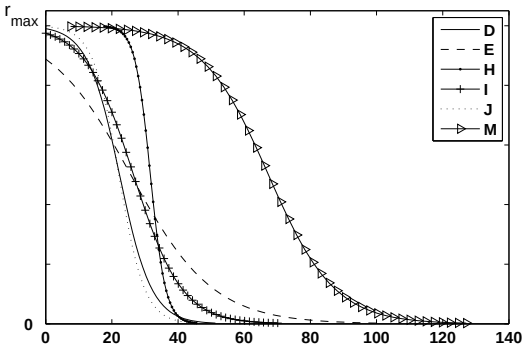


Fig. 7. Examples of measurement noise variance $r(g)$ as a function of image gradient for images D, E, H, I, J and M in Figs. 9 and 10.

To better understand the role of measurement variance, Fig. 7 plots the measurement noise variance (R) for images D, E, H, I, J and M of Figs. 9 and 10; these images were chosen for their varying contrast, non-homogeneity, noise, and background clutter. For uniform contrast foreground and background images (H and J), $\underline{r}$ is similar to a step function, because the distribution of image gradient magnitude follows a narrow band (lower σ_g). For nonuniform contrast and noisy images (D, E and I), $\underline{r}$ changes towards a inverted sigmoid function, due to a larger variance in the distribution of image gradient magnitude. For background cluttered images (M), $\underline{r}$ is significantly different due to larger values of the mean and variance of image gradient.

E. Non-Stationary Prior

The prior actually developed in Section III-D has constant $\alpha(s)$ and $\beta(s)$, and is therefore stationary. An argument for a non-stationary prior was motivated in Section III-B, leading to the rationale for importance sampling. We are now ready to evaluate this claim.

Recall from (17) the prior

$$\underline{x} \sim N(\mu_x, P) \approx N\left(\mu_x, (A^T A)^{-1}\right) \tag{29}$$

where A is a linear penalty term,

$$\|A\underline{x}\|_2 \approx \int_0^1 \left(\alpha(s)\,|x_s|^2 + \beta(s)\,|x_{ss}|^2\right) ds \tag{30}$$

such that A synthetically approximates the first and higher derivatives on the basis of local differences. Correctly computing a discrete derivative requires taking into account the sampling interval. However because the constraint, represented by A, is *stationary*, and does *not* take the interval into account, therefore the effective result is to induce a non-stationary discretization of the derivative, essentially meaning that $\alpha(s)$ is space-varying, an increasing function of discretization interval.

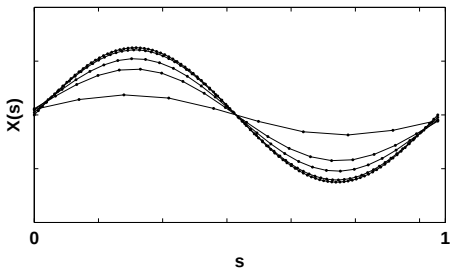


Fig. 8. Illustration of regularization as function of the number of samples with constant penalty factor $(\alpha(s), \beta(s))$ and R. As the number of samples decreases the curve inherently becomes more smooth as the prior is asserted more strongly.

Algorithm 2 $\underline{v}^c$ = Function DAC($\underline{z}, q, u$)

1: $k = 1$, $ASD = \infty$, $\epsilon = 10^{-4}$
2: Initialize object boundary as $v(s_j), s_j \in [0,1], j \in [1,q]$
3: **while** $ASD \geq \epsilon$ **do**
4: Create the trellis and compute the trellis parameters $p(z_j, z_{j+1})$ and $p(\psi(z_j)|z_j)$
5: $[\underline{v}^{\mathrm{v}}]$ = Viterbi($\underline{z}, q, u$)
6: Generate $\underline{v}^m$ applying importance sampling on $\underline{v}^{\mathrm{v}}$ following Section III-B
7: Compute $\hat{\underline{v}}_k$ using linear Bayesian estimator (20)
8: Compute the ASD between $\hat{\underline{v}}_k$ and $\hat{\underline{v}}_{k-1}$ using (31)
9: $v(s_j) = \hat{v}_{k_j}, s_j \in [0,1], j \in [1,q]$.
10: $k = k + 1$.
11: **end while**
12: Assign $v_j^c = v(s_j), \quad j \in [1,q]$ as the converged boundary.

The resulting phenomenon is illustrated in Fig. 8. Suppose we have sinusoidal measurements, but a zero (flat) prior. For fixed values of α and β, as the number of measurement points increases the influence of the prior on the sinusoidal measurements drops, implying a weaker prior. As the samples become sparser, the regularized curve flattens. The net result is a space-varying prior model, penalizing more strongly in areas of low curvature, and less so in areas of higher curvature.

The pseudocode for the complete DAC algorithm is provided in Algorithm 2.

IV. Testing and Results

This section describes the experimental data sets, methods compared, experimental setup and demonstrates quantitative and qualitative evaluation of DAC compared to five other methods.

A. Evaluation Data Sets and Criterion

Three types of tests are established for evaluating the performance of DAC. Test 'A' (Weizmann database [27]) evaluates the average segmentation accuracy of DAC compared to other methods. Test 'B' (eighteen natural and synthetic images) evaluates the quantitative and visual boundary identification accuracy and computational cost. The eighteen images selected for Test 'B' depict various characteristics: 1) complex background and object of interest with weak edges (Images A and D of Fig. 9, and P and R of Fig. 16), 2) natural and synthetic high curvature images (Images C-H of Fig. 9 and N-O of Fig.12), and 3) synthetic and natural noisy images (Fig. 10). Test 'C' evaluates DAC capabilities mentioned in Section II-C through illustrative examples.

B. Methods Compared

The performance of DAC is compared to the traditional snake (TS) [9], gradient vector flow snake (GVFS) [16], vector field convolution snake [17], active contour without edges (ACWE) [21] using Chan-Vese model and fast active contour (FAC) [23]. The current DAC implementation is designed for unsupervised segmentation of a single complex object in noisy and cluttered environments. The theory behind DAC is derived from traditional active contour concepts, so TS is considered for a base comparison. GVFS and VFC increase the capture range making these approaches less sensitive to initialization.

Region based segmentation approaches [11], [21], [23], [30] usually employ more global information in defining object boundaries and can outperform parametric active contours. Therefore, ACWE [21], a region based approach that is able to capture high curvature regions, is tested. Further, to validate DAC's speed performance, FAC [23] is tested since it is designed as a very fast active contour method. Further, FAC and ACWE are designed to work on noisy images by employing a smoothing model to extract the active contour, so ACWE and FAC provide a basis on which to evaluate DAC's noise robustness.

For completeness, we have used a sample image to compare DAC to level set [11] (LS), intelligent scissors (IS) [6], and greedy snake (GS) [39] methods. These methods have characteristics that do not support extensive test comparisons to DAC. LS is designed for multiple-object segmentation and oversegments the single-object images of this paper. IS requires user interaction, whereas DAC and other methods are unsupervised. GS was tested across all test images, but was unable to segment any image properly.

C. Experimental Setup

For comparison purposes, published MATLAB code for the gradient vector flow (GVFS) snake [16], VFC [17], fast active contour for global minimization (FAC) [23] and active contour without edge (ACWE) [21] were downloaded from [51], [52], [53] and [54]. The codes obtained from these websites were written for experimental research purposes without optimization. We modified these codes using MATLAB vector optimization to allow fair speed comparison with DAC.

TABLE II

Single Segment Average F-measure score (mean ± standard deviation) of DAC, GVFS [16], TS [9] and VFC [17] over 100 images of Weizmann data base [27]

Algorithm	Average F-measure Score
DAC	0.88±0.060
GVFS	0.75±0.120
TS	0.72±0.153
VFC	0.77±0.140

The initial location of the snake was specified manually for all images in Test 'B' as shown in Figs. 9, 10, and 12, while for Test 'A' the initial snake was always chosen as an ellipse. For DAC, TS, and GVFS, the snakes were initialized with $q = 250$ discrete points. The suitable range of parameters for TS, GVFS and VFC were chosen from [16] and [17]. Since ACWE and FAC are region based approaches, ACWE was initialized with a closed area as shown in Figs. 9, 10, and 12 and the optimal parameters were chosen from [21] and [23]. However, FAC was initialized with the original image, because FAC is completely independent of initial solution. For all methods but DAC, σ_s was set using the value provided in the respective papers. For all experiments, DAC uses $r_{max} = 2000$, $\Delta v_{\min} = 0.5$ and $\Delta v_{\max} = 8$, $\alpha = 1, \beta = 0.5$, $\tau = 0$, $\epsilon = 10^{-4}$ and $\sigma_s = 0.5$. For each normal, the number of points u was automatically selected so that the normal touched no other nearby normal. The $u + 1$ points on each normal were separated by one pixel. Experiments were performed on a 2.4 GHz, 1G RAM, Intel P4 computer.

D. Test 'A': Segmentation Using Weizmann [27] Database

We have evaluated the segmentation performance of DAC on the Weizmann database [27] consisting of 100 images, each of which contains one foreground object, differing from the background on the basis of intensity, texture, or other low level cues. The single object F-measure score [27], a popular segmentation accuracy measure index, is shown in Table II along with its standard deviation. From the results, we found the Weizmann database images are not suitable for ACWE [21] and FAC [23], because of the variations in intensities of the foreground object. Therefore, we have not reported the F-measure score for these two methods. The methods tested here (DAC, GVFS, TS, VFC) only consider the grey level yet texture features are required to properly segment six of the images. If these six images were removed, the F-measure would have been higher for all four methods. Over the 100 images in the database, DAC provides higher accuracy relative to the other tested methods.

E. Test 'B': Boundary Accuracy and Convergence Speed

To measure the quantitative dissimilarity between converged boundary $\underline{v}^c$ and true boundary $\underline{v}^t$ we have defined the average shortest distance (ASD), defined as the average shortest distance of the converged boundary from the ground truth and ground truth from the converged boundary:

$$ASD = \frac{1}{2q}\sqrt{\sum_{i=1}^{q} \min_{j=1,2\cdots n} \left(\left\|v^t(s_j) - v^c(s_i)\right\|_2\right)} + \frac{1}{2n}\sqrt{\sum_{j=1}^{n} \min_{i=1,2\cdots q} \left(\left\|v^t(s_j) - v^c(s_i)\right\|_2\right)}. \quad (31)$$

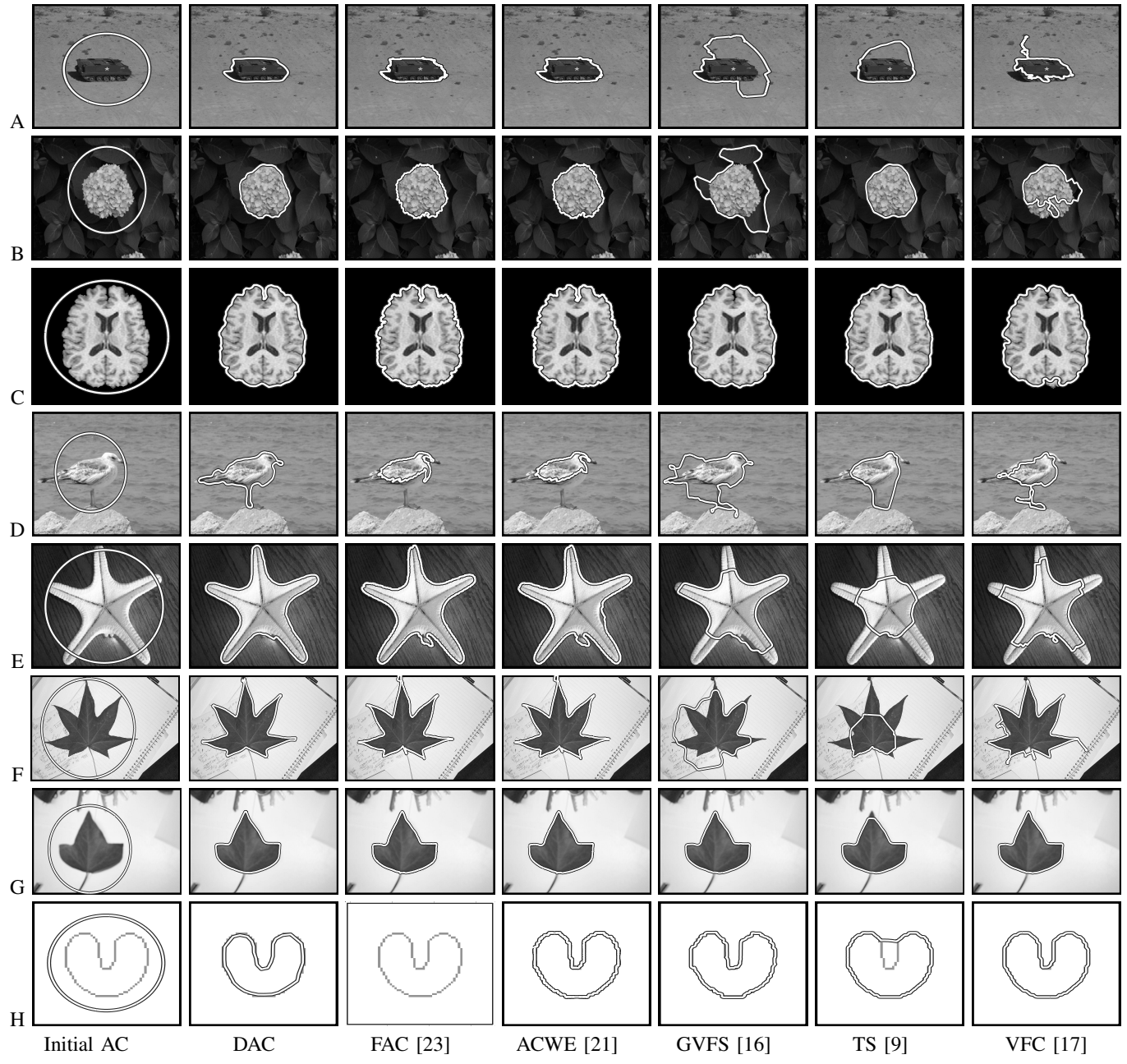


Fig. 9. Six snake methods applied to eight different images obtained from [15], [16], [49], [50], [27]. Column 1 shows the test images along with the common initial snake. Column 2 plots the boundary found by the proposed DAC. Columns 3 to 7 show results using five other snake methods [23], [21], [16], [9], [17]. In each panel, the white line with black border shows the final contour. DAC is the only method that properly identifies the object boundary in each image; no other method works for all eight images.

TABLE I

AVERAGE SHORTEST DISTANCE (ASD) IN PIXELS AND EXECUTION TIME (ET) IN SECONDS OF DAC COMPARED TO FIVE OTHER METHODS USING EIGHT TEST IMAGES IN FIG. 9. BOLD TEXT INDICATES BEST (OR CLOSE TO BEST) PERFORMANCE ACROSS ALL METHODS FOR A PARTICULAR IMAGE.

	Average shortest distance (ASD) [pixels]						Execution time [seconds]					
	DAC	FAC	ACWE	GVFS	TS	VFC	DAC	FAC	ACWE	GVFS	TS	VFC
(A)	**0.89**	**0.56**	**0.68**	29.17	15.68	13.52	**13**	67	377	208	391	167
(B)	**0.58**	**0.87**	**0.80**	13.86	**0.63**	7.62	**3**	15	59	109	397	86
(C)	**1.60**	**1.60**	**1.70**	**2.40**	**2.80**	3.52	**7**	25	70	90	255	67
(D)	**2.81**	12.92	14.26	6.48	9.61	7.57	**11**	35	190	146	351	103
(E)	**1.40**	**1.90**	**2.70**	9.60	36.9	8.65	**5**	33	159	148	543	109
(F)	**2.20**	**2.00**	**1.12**	26.62	60.3	**4.81**	**14**	120	636	357	581	312
(G)	**0.60**	**0.65**	**0.62**	**0.97**	**2.32**	**0.5**	**6**	119	612	323	596	283
(H)	**3.40**	104	**3.00**	**3.60**	10.6	**3.13**	**5**	86	134	73	536	53

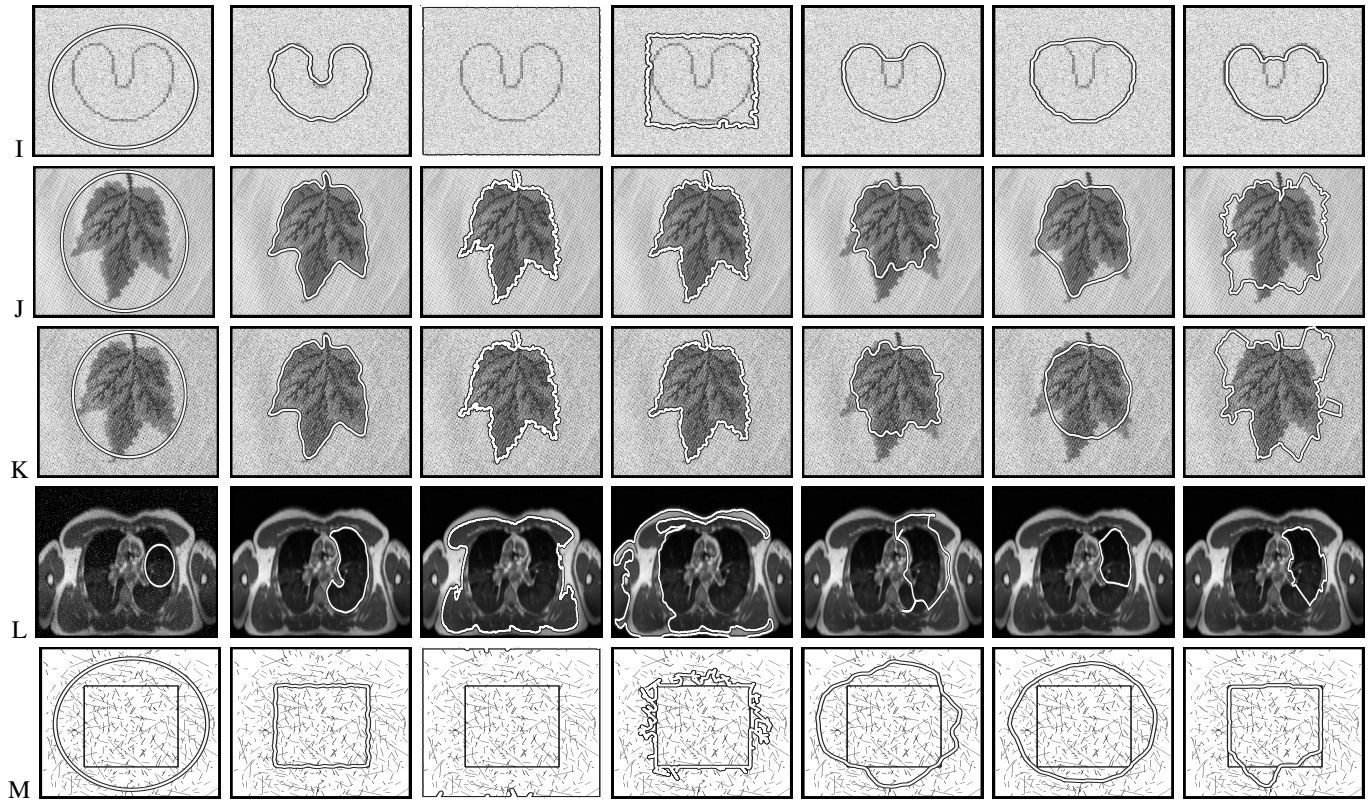


Fig. 10. Six snake methods are applied to five test images containing noise and clutter with the same organization as Fig. 9. Image (I) shows a U-shape contaminated with Gaussian noise of $\sigma = 30.2$dB. (J) and (K) show a natural textured image with natural noise like feature and additive Gaussian noise of $\sigma = 7.2$dB. (L) shows the image of a thorax contaminated with additive salt and pepper noise of $\sigma = 5$dB. The last row (M) shows a synthetic image on a cluttered background. The other four methods are generally ineffective in the presence of noise and background clutter.

The number of points on the converged boundary (q) and ground truth (n) are not necessarily equal.

The qualitative and quantitative results of DAC compared to FAC [23], ACWE [21], GVFS [16], TS [9] and VFC [17] for both natural and synthetic images are presented in Figs. 9 and 10 and Tables I and III. Figs. 9 and 10 shows separate images for the initial contour and the final solution for all five methods. Tables I and III show the quantitative values of ASD (31) and execution time (ET) for all six methods. We chose suitable test parameters from the respective papers of all compared methods for as fair a comparison as possible.

In Figs. 9 and 10, DAC accurately detects the desired object boundary for all images. Because of the non-stationary curvature-dependent prior (P) and image statistic dependent measurement uncertainty R, DAC is able to differentiate between true high curvature and the false high curvature (generated due to noise or background clutter) with no parameter tuning. In Tables I and III, DAC is consistently significantly faster and with an accuracy comparable to or significantly better than all compared methods.

FAC [23] integrates the concept of geometric active contour [30] with denoising model of Rudin et al. [55]. FAC uses fast and efficient numerical schemes for contour evolution, therefore showing a relatively fast speed in Tables I and III. However, FAC fails for images D, H, I, L and M. The object of interest of image D consists of multiple homogenous regions, images H, I, and L have small intensity differences between background and foreground, and FAC's denoising model smoothed the object of interest along with the background clutter and treated them as a single object for image M.

ACWE [21] employs a level set based curve evolution technique to solve the Mumford-Shah model [33] with a piecewise constant approximation, such that the background and foreground can be represented using two constants. Therefore, ACWE model does not use the gradient as the stopping criterion and instead uses intensity homogeneity constraints. ACWE successfully identified the region of interest for most of the images. ACWE failed for images D, I, L and M because the region of interest could not be described by a single constant (D), ACWE converges to a local minimum generated by noise or clutter (I, M), or because the object of interest is not properly closed (L).

TS [9] failed to locate the true boundary for most images. TS uses an iterative gradient descent optimization technique (same as for GVFS [16] and VFC [17]); as a result, TS is sensitive to local minima and has a slower convergence rate. Furthermore, TS uses local image gradient as the external energy and, as a result, an initial solution far from the true solution may be trapped or take a long time to converge.

To attract an initial active contour from a greater distance, GVFS and VFC use a partial differential and a kernel based diffusion technique to spread out the external energy throughout the image. As a result, both GVFS and VFC are less sensitive to the initial solution and converge faster compared to TS. The main cause for the failure of GVFS is twofold. First, the diffusion function used for spreading the image potential does not work properly if the nature of the gradient is not simple. Second, high curvature boundaries hinder the GVFS to converge towards

TABLE III

AVERAGE SHORTEST DISTANCE (ASD) IN PIXEL AND EXECUTION TIME (ET) IN SECOND OF DAC COMPARED TO FIVE OTHER METHODS FOR FIVE DIFFERENT NOISY IMAGES IN FIG. 10. BOLD TEXT INDICATES BEST PERFORMANCE ACROSS ALL METHODS FOR A PARTICULAR IMAGE.

	Average shortest distance (ASD) [pixels]						Execution time (ET) [second]					
	DAC	FAC	ACWE	GVFS	TS	VFC	DAC	FAC	ACWE	GVFS	TS	VFC
(I)	**3.70**	50.7	39.9	7.30	13.9	8.48	**15**	37	223	148	683	137
(J)	**1.60**	**2.05**	**1.69**	8.84	5.81	11.29	**10**	44	281	306	596	257
(K)	**2.80**	**1.77**	**1.44**	7.96	19.45	10.64	**18**	26	162	189	601	129
(L)	**3.00**	55.3	41.8	9.90	13.55	8.24	**16**	26	171	94	539	78
(M)	**3.50**	71.8	13.44	38.6	43.7	9.14	**9**	74	468	192	603	209

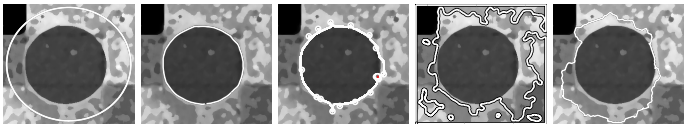


Fig. 11. Performance of DAC, intelligent scissors [6] (LS), level set [11] (LS) and greedy snake [39] (GS) in a cluttered environment.

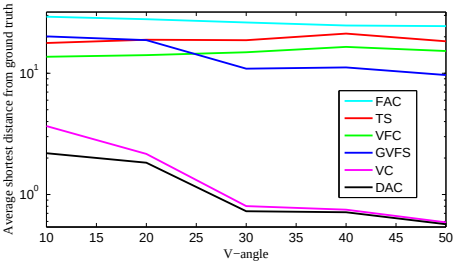


Fig. 13. The ASD (31) as a function of V-angle for DAC, FAC, ACWE, GVFS, TS and VFC from Fig. 12. The DAC and ACWE methods are similarly successful, although the DAC outperforms at tighter angles.

the correct solution. VFC uses a noise robust computationally efficient kernel based diffusion technique to handle complex boundary edges. Therefore, VFC performed better than GVFS, both in terms of accuracy and speed. The primary reasons for VFC failure are due to incorrect diffusion of gradients and convergence to local minima in the presence of background clutter and noise.

Finally, as a qualitative comparison to other methods, in Fig. 11 the segmentation result of DAC is compared to level set [11] (LS), intelligent scissors (IS) [6], and greedy snake (GS) [39].

F. Test 'C': Evaluation of DAC Capabilities

This section experimentally assesses the DAC in terms of the five criteria identified in Section II-C: 1) noise and background clutter, 2) high curvature regions, 3) parameter sensitivity, 4) initialization sensitivity, and 5) stopping criteria and rate of convergence.

1) Robustness to Noise and Background Clutter: The performance of DAC in the presence of noise and background clutter is demonstrated in Figs. 9 and 10 and Tables I and III using natural and synthetic images. The DAC accurately identified the object boundary for all test images. Our method distinguishes between noise and true high-curvature segments in three ways:

1) Because the gradient is computed via convolution, as in (2), there is some degree of smoothing, which reduces noise more than structure.
2) Because there is a minimum step size $\Delta v_{min} > 0$, a given noise point will be felt by zero, or at most one, measured points, whereas a curvature segment will be felt, and constrained, by multiple measurements.
3) Finally, the computation of the measurement variance r in (28) leads to a preference for curves along uniform gradients, rather than curves which encounter varying gradient levels.

The limited number of measurements and greater value of σ_g allow DAC to pass through points of noise and background clutter.

2) High curvature object boundaries: The ability of DAC to capture high curvature regions compared to FAC [23], ACWE [21], GVFS [16] and TS [9] is tested on a set of five synthetic images in which a concavity is made tighter and tighter, a problem of considerable difficulty for most parametric contours. Fig. 13 plots the ASD as a function of angle, and results for two of the images are shown in Fig. 12. Only the ACWE and DAC perform well on this test set; the GVFS, TS and VFC fail to capture the high curvature region as these are both parametric active contours that do not incorporate a non-stationary prior to reduce the penalty near high curvature corners. The DAC's abilities to capture high curvature boundaries was demonstrated via natural images (Test 'B') and the tests on the Weizmann data set (Test 'A').

3) Parameter independency: DAC uses a fixed set of parameters ($\alpha = 1$ and $\beta = 0.5$ in (21)) for all of the images in all of the experiments in this paper. Although most existing methods [9], [16], [23], [21] require the user to manually set a fixed prior and other parameters, a fixed prior is inappropriate for a non-stationary boundary, and image-dependent parameters are inconvenient when working with varied images. The non-stationary prior inferred by the DAC from importance sampling simplifies parameter issues significantly. Essentially, DAC translates the problem of parameter tuning into the problem of seeking a good edge probability by carefully choosing the measurement uncertainty matrix R and non-stationary prior P.

As discussed under Test 'A', DAC accurately identified the object boundary for all of these diverse images without a single change to any parameter settings in all cases.

4) Initialization insensitivity: Fig. 14 shows the convergence of four different initializations. Whether the snake expands or contracts is a function of the external mean term $\underline{\mu}$ (24), (23) (the external mean need to be set manually). In contrast, the traditional snake (TS) requires an initial snake close to its solution to ensure speed of convergence and accuracy [16]. Although GVFS increases the capture range, the method is not able to find the correct boundary if initialized far from the true boundary. These phenomena can be observed from Figs. 9, 10, and 12. FAC and ACWE are region based approaches, so they are not sensitive

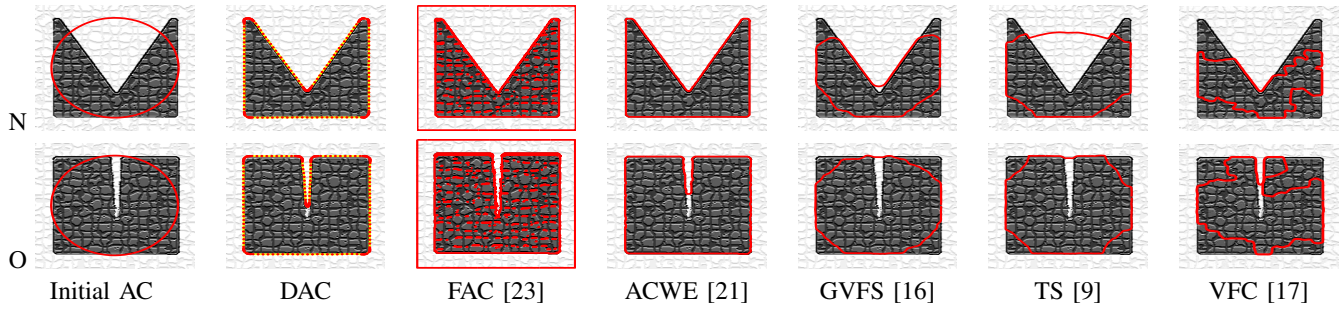


Fig. 12. A test of convergence into a concave region, a challenging task for a snake algorithm. For the proposed DAC, in Column 2, the dots plot the placement of sample points illustrating the curvature-dependent sampling. Of the other five methods, only the ACWE [21] performs similarly, although at much greater computational cost. The degree of convergence of the panels in this figure is plotted in Fig. 13.

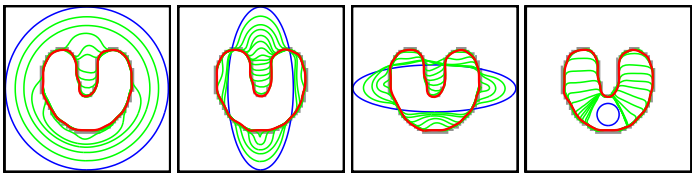


Fig. 14. Convergence pattern of DAC for a variety of initial positions given a U-shaped object. For each case, DAC demonstrates insensitivity to initial snake position by converging to the correct solution in each case.

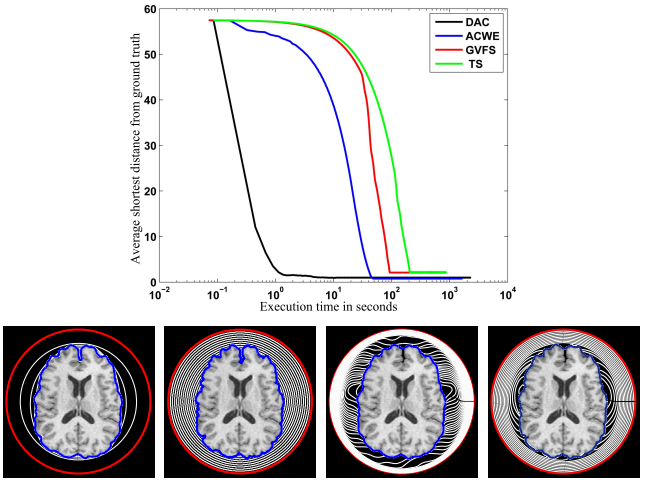


Fig. 15. Convergence rate for different methods for brain image. Each method is able to capture the true object boundary but with a different convergence rate. Red, white, and blue lines (second row of Fig. 15) are the initial, intermediate and final snake positions respectively. The ASD from ground truth is shown in first row of Fig. 15.

to initial positions.

5) Stopping criteria and convergence rate: The comparison of computational complexity is ambiguated by the difficulty of assessing convergence in algorithms which converge very slowly. For all examples in this paper, DAC reached the ASD limit ($\epsilon = 10^{-4}$) quite quickly and so always terminated on the basis of ASD convergence. For the other compared methods, the slower convergence rate frequently led a simple threshold to lead to early termination. Therefore we manually tested the convergence for all images displayed in this paper.

A much more convincing demonstration, independent of stopping criterion, is shown in Fig. 15, plotting ASD as a function of computation time. The convergence rate of DAC is fast, roughly two orders of magnitude faster, compared to other methods as noted in Tables I and III. Such an improvement in convergence rate offers potential for more advanced segmentation tasks, such as real-time tracking or three-dimensional problems.

G. Parametric vs. Non-Parametric Active Contours

Parametric and nonparametric methods of segmentation are fundamentally very different, making any comparison difficult. Aspects of the trade-offs between the two approaches are illustrated in images (P,Q,R and S) of Fig. 16. Non-parametric active contours inherently perform multiple object segmentation (P and Q) and can capture high-curvature boundaries, since there is no explicitly modeled boundary. Similarly, non-parametric methods are robust to non-uniform and low-contrast boundaries (R). In contrast, non-parametric approaches can not handle discontinuous boundaries (S) and are usually computationally slower than parametric methods.

We make no claims regarding the superiority of DAC, or parametric methods in general, over non-parametric ones. Our claim is that for single-object segmentation in the presence of clutter, DAC is able to outperform other parametric and non-parametric methods, both in terms of convergence accuracy and computational complexity. Clearly for multi-object images (P and Q [21]) and low-contrast settings (R [21]), nonparametric methods are the more natural choice.

V. Conclusions and Future Efforts

A novel active contour method, the DAC, is designed for accurate boundary extraction, despite image noise and complex object geometries. Each iteration of the DAC is carried out in three steps: a Viterbi search to find the image gradients, importance resampling to generate a non-stationary prior, and a Bayesian estimator to update the boundary by incorporating prior shape constraints.

Validation of the DAC is demonstrated experimentally on noisy, cluttered natural and synthetic images. DAC is demonstrated to be robust to noise, requires no parameter tuning, is able to capture high curvature regions, and is insensitive to initialization.

Of all of the parametric active contours tested for finding the boundary of a single object, DAC is the only one which detected all boundaries accurately for a fixed set of parameters. The limitation of DAC and other parametric methods is that they are designed to find single objects, as opposed to non-parametric methods, which inherently can identify multiple objects.

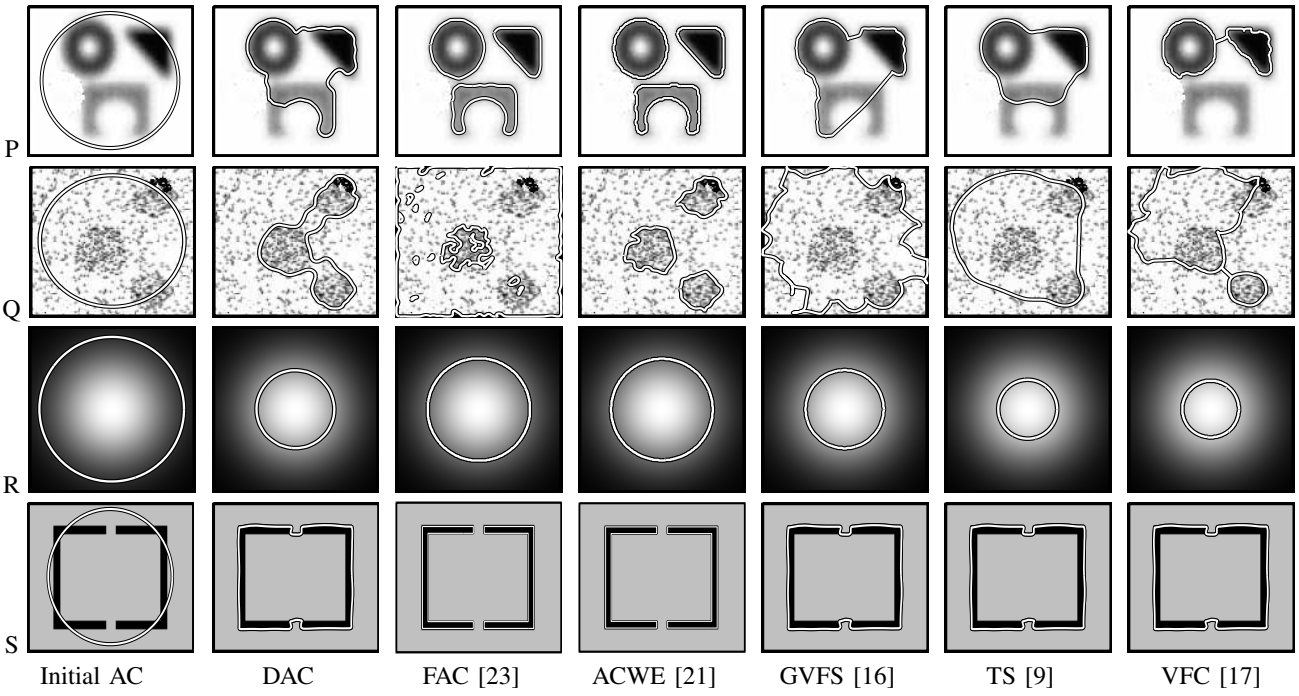


Fig. 16. Four images, illustrating certain tradeoffs between parametric and non-parametric active contours. Parametric methods explicitly model a single curve, thus nonparametric methods are more naturally suited to the segmentation of multiple objects (P and Q [21]). Parametric and Nonparametric methods also differ in how they treat low-contrast (R [21]) and discontinuous (S) boundaries.

The computational time of DAC is dramatically lower relative to other parametric and non-parametric approaches by two orders of magnitude, a dramatic improvement. Future work is aimed at further computational improvements, in addition to the possibility of extensions to three-dimensional deformable surfaces.

ACKNOWLEDGMENT

This research has been sponsored by the Natural Sciences and Engineering Research Council (NSERC) of Canada through individual Discovery Grants as well as GEOIDE (GEOmatics for Informed Decisions) which is a Network of Centres of Excellence under NSERC. We would like to thank the anonymous reviewers for their constructive feedback.

REFERENCES

[1] F. Leymarie and M. Levine, "Tracking deformable objects in the plane using an active contour model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 6, pp. 617–634, 1993.

[2] S. Sclaroff and L. Liu, "Deformable shape detection and description via model-based region grouping," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 5, pp. 475–489, 2001.

[3] N. Alajlan, M. Kamel, and G. Freeman, "Geometry-based image retrieval in binary image databases," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 6, pp. 1003–1013, 2008.

[4] M. Mata, J. Armingol, J. Fernández, and A. De, "Object learning and detection using evolutionary deformable models for mobile robot navigation," *Robotica*, vol. 26, no. 1, pp. 99–107, 2008.

[5] Y. Li, J. Sun, C. Tang, and H. Shum, "Lazy snapping," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 303–308, 2004.

[6] N. M. Eric and A. B. William, "Intelligent scissors for image composition," in *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM, 1995, pp. 191–198.

[7] A. Mishra, A. Wong, W. Zhang, D. Clausi, and P. Fieguth, "Improved interactive medical image segmentation using enhanced intelligent scissors (EIS)," in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Vancouver, Canada, August 2008, pp. 3083–6.

[8] C. Davatzikos and J. Prince, "An active contour model for mapping the cortex," *IEEE Transactions on Medical Imaging*, vol. 14, no. 1, pp. 65–80, March 1995.

[9] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988.

[10] V. Caselles, F. Catte, T. Coll, and F. Dibos, "A geometric model for active contours in image processing," *Numerische Mathematik*, vol. 66, pp. 1–31, 1993.

[11] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Shape modeling with front propagation: A level set approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 2, pp. 158–175, 1995.

[12] A. Amini, T. E. Weymouth, and R. Jain, "Using dynamic programming for solving variational problems in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 9, pp. 855–867, 1990.

[13] P. Brigger, J. Hoeg, and M. Unser, "B-spline snakes: a flexible tool for parametric contour detection," *IEEE Transactions on Image Processing*, vol. 9, no. 9, pp. 1484–1496, 2000.

[14] L. Cohen and I. Cohen, "Finite-element methods for active contour models and balloons for 2-d and 3-d images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1131–1147, 1993.

[15] F. Shih and K. Zhang, "Locating object contours in complex background using improved snakes," *Computer Vision Graphics and Image Understanding*, vol. 105, no. 2, pp. 93–98, 2007.

[16] C. Xu and J. Prince, "Snakes, shapes, and gradient vector flow," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 359–369, 1998.

[17] B. Li and T. Acton, "Active contour external force using vector field convolution for image segmentation," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2096–2106, 2007.

[18] Y. Wong, P. Yuen, and C. Tong, "Segmented snake for contour detection," *Pattern Recognition*, vol. 31, no. 11, pp. 1669–1679, November 1998.

[19] G. Sundaramoorthi, A. Yezzi, Jr., and A. Mennucci, "Coarse-to-fine segmentation and tracking using sobolev active contours," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 5, pp. 851–864, 2008.

[20] T. Chan, B. Sandberg, and L. Vese, "Active contours without edges for vector-valued images," *Journal of Visual Communication and Image Representation*, vol. 11, no. 2, pp. 130–141, 2000.

[21] T. Chan and L. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.
[22] S. Zhu and A. Yuille, "Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 9, pp. 884–900, 1996.
[23] X. Bresson, S. Esedoglu, P. Vandergheynst, J. Thiran, and S. Osher, "Fast global minimization of the active contour/snake model," *Journal of Mathematical Imaging and Vision*, vol. 28, no. 2, pp. 151–167, 2007.
[24] A. Jalba, M. Wilkinson, and J. Roerdink, "CPM: A deformable model for shape recovery and segmentation based on charged particles," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 10, pp. 1320–1335, 2004.
[25] X. Xianghua and M. Majid, "MAC: Magnetostatic active contour model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 4, pp. 632–646, 2008.
[26] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.
[27] S. Alpert, M. Galun, T. Basri, and A. Brandt, "Image segmentation by probabilistic bottom-up aggregation and cue integration." in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA, June 2007.
[28] L. Cohen, "On active contour models and balloons," *Computer Vision Graphics and Image Understanding*, vol. 53, no. 2, pp. 211–218, March 1991.
[29] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations," *Journal of Computational Physics*, vol. 79, no. 1, pp. 12–49, 1988.
[30] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *International Journal of Computer Vision*, vol. 22, no. 1, pp. 61–97, 1997.
[31] K. Siddiqi, S. Zucker, Y. BérubéLauzière, and A. Tannenbaum, "Area and length minimizing flows for shape segmentation," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 1997, pp. 621–627.
[32] L. Cohen and R. Kimmel, "Global minimum for active contour models: A minimal path approach," *International Journal of Computer Vision*, vol. 24, no. 1, pp. 57–78, 1997.
[33] D. Mumford and J. Shah, "Boundary detection by minimizing functionals," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, June 1985, pp. 22–26.
[34] G. Sundaramoorthi, A. Yezzi, Jr., and A. Mennucci, "Sobolev active contours," *International Journal of Computer Vision*, vol. 73, no. 3, pp. 345–366, 2007.
[35] F. Eric, F. Gibou, and R. Fedkiw, "A fast hybrid k-means level set algorithm for segmentation," 4th Annual Hawaii International Conference on Statistics and Mathematics, Honolulu Hawaii, USA, Tech. Rep., 2002.
[36] A. Amini, T. Weymouth, and R. Jain, "Using dynamic programming for solving variational problems in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 9, pp. 855–867, 1990.
[37] S. Chandran and A. Potty, "Energy minimization of contours using boundary conditions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 5, pp. 546–549, 1998.
[38] G. Storvik, "A Bayesian approach to dynamic contours through stochastic sampling and simulated annealing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 10, pp. 976–986, 1994.
[39] D. Williams and M. Shah, "A fast algorithm for active contours and curvature estimation," *Computer Vision Graphics and Image Understanding*, vol. 55, no. 1, pp. 14–26, 1992.
[40] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 309–314, 2004.
[41] H. Chang, Q. Yang, and C. Pan, "An iterative Bayesian approach for digital matting," in *IEEE International Conference on Pattern Recognition*, 2006, pp. II: 122–125.
[42] J. Nascimento and J. Marques, "Adaptive snakes using the EM algorithm," *IEEE Transactions on Image Processing*, vol. 14, no. 11, pp. 1678–1686, November 2005.
[43] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. IT-13, no. 2-3, pp. 260–269, 1967.
[44] H. Li and J. Yezzi, "Local or global minima: Flexible dual-front active contours," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 1–14, 2007.
[45] D. A. Forsyth and J. Ponce, *Computer Vision A Modern Approach*. Prentice Hall, Upper Saddle River, New Jersey 07458, 2003.
[46] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill Companies, 3rd edition, 1991.
[47] E. L. Lehmann and G. Casella, *Theory of Point Estimation*. Springer, 2nd ed, 1998.
[48] N. Vaswani and R. Chellappa, "Principal components null space analysis for image and video classification," *IEEE Transactions on Image Processing*, vol. 15, no. 7, pp. 1816–1830, 2006.
[49] R. Kwan, A. Evans, and G. Pike, "MRI simulation-based evaluation of image-processing and classification methods," *IEEE Transactions on Medical Imaging*, vol. 18, no. 11, pp. 1085–1097, 1999.
[50] K. Bay, "Starfish- they are huge," 2007, http://kahunabay.wordpress.com/2007/09/07/starfish-they-are-huge/.
[51] C. Xu and J. Prince, "Johns Hopkins University's Image Analysis and Communications lab's GVF software," 1999, http://iacl.ece.jhu.edu/projects/gvf/.
[52] B. Li and T. Acton, "Vector field convolution," 2007, http://viva.ee.virginia.edu/research-vfc/.
[53] X. Bresson, S. Esedoglu, P. Vandergheynst, J. Thiran, and S. Osher, "Fast global minimization of the active contour model," 2007, http://www.math.ucla.edu/ xbresson/code.html.
[54] S. Lankton, "Georgia Institute of Technology," 2007, http://www.shawnlankton.com/2007/05/active-contours/.
[55] I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, no. 1-4, pp. 259–268, 1992.

Akshaya K. Mishra received the M.Tech degree from the Indian Institute of Technology (IIT), Kharagpur, India, in 2004. He worked as a research engineer at Read-Ink Technology Pvt Ltd., Bangalore, India from 2004 to 2006, where he was working on developing architecture for on-line and off-line handwritten character recognition. Currently, he is a full time Ph.D student at University of Waterloo. His research interest includes handwritten character recognition, active contour based image and video segmentation, tracking, denoising and data fusion.

Paul W. Fieguth (S87M96) received the B.A.Sc. degree from the University of Waterloo, Waterloo, ON, Canada, in 1991 and the Ph.D. degree from the Massachusetts Institute of Technology, Cambridge, in 1995, both in electrical engineering. He joined the faculty at the University of Waterloo in 1996, where he is currently Associate Professor in Systems Design Engineering. He has held visiting appointments at the Cambridge Research Laboratory, at Oxford University, Oxford, U.K., and the Rutherford Appleton Laboratory, U.K., and at INRIA/Sophia, France, with postdoctoral positions in Computer Science at the University of Toronto and in Information and Decision Systems at MIT. His research interests include statistical signal and image processing, hierarchical algorithms, data fusion, and the interdisciplinary applications of such methods, particularly to remote sensing.

David. A. Clausi (S93M96SM03) received the B.A.Sc., M.A.Sc., and Ph.D. degrees in systems design engineering from the University of Waterloo, Waterloo, ON, Canada, in 1990, 1992, and 1996, respectively. He started his academic career in 1997 as an Assistant Professor in geomatics engineering with the University of Calgary, Calgary, AB , Canada. In 1999, he returned to the University of Waterloo and was awarded tenure and promotion to Associate Professor in 2003, where he is currently with the Department of Systems Design Engineering. He has an extensive publication record in diverse fields of remote sensing and computer vision. His primary research interest is automated interpretation of synthetic aperture radar sea-ice imagery in support of the operational activities of the Canadian Ice Service. The research results have successfully led to commercial implementations. Dr. Clausi has received numerous scholarships, conference paper awards, and two Teaching Excellence Awards.