

# MCMC Curve Sampling and Geometric Conditional Simulation

Ayres Fan<sup>a</sup>, John W. Fisher III<sup>a</sup>, Jonathan Kane<sup>b</sup>, and Alan S. Willsky<sup>a</sup>

<sup>a</sup>Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology,  
77 Massachusetts Ave, Cambridge, MA USA;

<sup>b</sup>Shell E&P, 3737 Bellaire Blvd., Houston, TX USA

## ABSTRACT

We present an algorithm to generate samples from probability distributions on the space of curves. Traditional curve evolution methods use gradient descent to find a local minimum of a specified energy functional. Here, we view the energy functional as a negative log probability distribution and sample from it using a Markov chain Monte Carlo (MCMC) algorithm. We define a proposal distribution by generating smooth perturbations to the normal of the curve, update the curve using level-set methods, and show how to compute the transition probabilities to ensure that we compute samples from the posterior. We demonstrate the benefits of sampling methods (such as robustness to local minima, better characterization of multi-modal distributions, and access to some measures of estimation error) on medical and geophysical applications. We then use our sampling framework to construct a novel semi-automatic segmentation approach which takes in partial user segmentations and conditionally simulates the unknown portion of the curve. This allows us to dramatically lower the estimation variance in low-SNR and ill-posed problems.

**Keywords:** Image segmentation, sampling, MCMC, conditional simulation, gravity inversion

## 1. INTRODUCTION

Active contours or curve evolution are image segmentation methods which track the location of a contour that divides an image into coherent regions. The earliest implementations by Kass *et al.* tracked discrete marker points on the contour<sup>1</sup> while level set methods<sup>2,3</sup> were later introduced to more naturally handle topological changes and avoid the reinitialization problems that often troubled marker-point approaches.

Curve evolution methods originated from research in the differential geometry community studying the behavior of curves evolving in time according to a geometric partial differential equation (PDE). Let  $\Omega \subset \mathbb{R}^2$  be the image domain,  $I : \Omega \rightarrow \mathbb{R}$  be a scalar-valued image, and  $\vec{C} : [0, 1] \rightarrow \Omega$  be a closed curve (*i.e.*,  $\vec{C}(0) = \vec{C}(1)$ ). We introduce an artificial time variable  $t$  which parameterizes a family of curves  $\vec{C}(\cdot, t)$ . Then a geometric PDE has the form:

$$\frac{\partial \vec{C}}{\partial t}(p) = f(p)\vec{N}_{\vec{C}}(p), \quad p \in [0, 1] \quad (1)$$

where  $f(p)$  is an arbitrary so-called force function and  $\vec{N}_{\vec{C}}(p)$  is the outward unit normal to the curve.

Active contour methods are formulated by specifying an energy functional of the curve  $E(\vec{C} | I)$  and evolving the curve according to the gradient descent of that functional. A classical energy functional is Euclidean curve length:

$$E(\vec{C} | I) = \oint_{\vec{C}} ds \quad (2)$$

with  $ds$  being the differential arc length along the curve. The resulting force function is  $f(p) = -\kappa_{\vec{C}}(p)$  where  $\kappa_{\vec{C}}$  is curvature. This flow will shrink an arbitrary curve to a point without self intersection and generally has a smoothing effect on the curve.<sup>3</sup> Due to this property, a curve length penalty is often used as a regularization term.

Region-based energy functionals<sup>4,5</sup> (which separate regions using the image statistics) are widely used now due to their robustness to noise. A commonly-used region-based energy functional is one introduced by Chan and Vese:<sup>4</sup>

$$E(\vec{C}) = \int_{\mathcal{R}(\vec{C})} (I(\mathbf{x}) - m_0)^2 d\mathbf{x} + \int_{\Omega \setminus \mathcal{R}(\vec{C})} (I(\mathbf{x}) - m_1)^2 d\mathbf{x} + \alpha \oint_{\vec{C}} ds. \quad (3)$$

---

Corresponding author: Ayres Fan. E-mail: fan@mit.edu, Telephone: +1 617 253 6172

Computational Imaging VI, edited by Charles A. Bouman, Eric L. Miller, Ilya Pollak,  
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 6814, 681407, © 2008 SPIE-IS&T · 0277-786X/08/\$18

Here  $\mathcal{R}(\vec{C})$  is the region enclosed in  $\vec{C}$ , and  $m_0$  and  $m_1$  are mean intensities inside and outside the curve (which can be specified *a priori* or learned during the curve evolution). From a statistical perspective, this functional can be viewed as corresponding to an observation model where the true intensities inside and outside the curve are constant, and the observed intensities are corrupted by white Gaussian noise.

Level set methods are a standard way to implement curve evolution algorithms<sup>3</sup> by creating a surface  $\Psi(\mathbf{x})$  whose zeroth level set is the curve:

$$\Psi(\vec{C}(p)) = 0 \quad \forall p \in [0, 1] . \quad (4)$$

By convention, we force  $\Psi$  to be negative inside the curve and positive outside the curve. We then wish to evolve  $\Psi$  in time so that the zeroth level set tracks  $\vec{C}$ . To do this we differentiate both sides of equation (4) with respect to  $t$ . Applying the chain rule results in:

$$\frac{\partial \Psi}{\partial t} = -\frac{\partial \vec{C}}{\partial t} \cdot \nabla \Psi . \quad (5)$$

Note that  $\frac{\partial \vec{C}}{\partial t}$  is only defined on the zeroth level set of  $\Psi$ . The PDE is extended to the rest of  $\Omega$  by propagating the force along the normal direction to the curve in a technique known as *velocity extension*.<sup>3</sup>

Optimization-based methods typically return a single local optimum. Finding the global maximum of a non-concave posterior distribution in general requires exhaustive search. Furthermore, a single local optimum provides little insight as to how close the result is to the global optimum or how confident one should be in the answer. For low signal-to-noise ratio (SNR) or ill-posed problems, this is problematic because there are many local optima, and there can even be multiple answers which plausibly explain the data.

A common alternative is to view  $E(\vec{C})$  as the negative log of a probability density

$$\pi(\vec{C} | I) \propto \exp(-E(\vec{C}; I)) . \quad (6)$$

Rather than restricting ourselves to finding modes of the posterior, this probabilistic formulation allows us to consider methods to draw samples from  $\pi$ . The randomness built into sampling methods can help avoid local minima, but perhaps of more importance, multiple samples can help characterize multi-modal distributions and estimation uncertainty by more fully exploring the configuration space.

Several recent approaches have applied standard probabilistic approaches to curve-based methods. Sun *et al.*<sup>6</sup> use particle filtering to track the left ventricle in cardiac magnetic resonance (MR). De Bruijne and Nielsen<sup>7</sup> use particle filtering and Juan *et al.*<sup>8</sup> present a stochastic optimization method to find global minima of energy functionals. Tu and Zhu<sup>9</sup> use a Markov chain Monte Carlo (MCMC) method that bears similarities to the approach presented here. Their primary focus is on finding global optima, and in fact cannot be used to draw samples from the posterior due to the lack of detailed balance. We will show examples here of noisy images where the global maximum *a posteriori* estimate does not provide a satisfactory segmentation due to the large amount of noise, whereas a constellation of samples can help provide greater information as to likely locations for the true segmentation. Additionally, multi-modal distributions are better represented by sampling methods, and we also demonstrate how sampling methods can be used to do partially-constrained segmentation approaches.

In general,  $\pi(\vec{C} | I)$  is a complex distribution and non-trivial to sample. Here, we present an approach to iteratively generate samples from arbitrary distributions over curves using MCMC methods. One of our key results is to understand how to ensure that detailed balance holds when sampling from the space of closed curves (a necessity to ensure that we asymptotically generate true samples from the posterior). We show how to adapt these sampling methods to use user input to perform conditional simulation in order to reduce the estimation variance, and demonstrate our approach on applications in medical imaging and geophysics.

## 2. FORMULATION

MCMC methods are useful for situations when one wishes to draw samples from a distribution  $\pi(\vec{C})$ , but it is not possible to do so directly. Instead a so-called proposal distribution  $q$  is defined, and samples from  $q$  are accepted or rejected in such a way as to guarantee that samples from  $\pi$  are generated asymptotically. MCMC methods were originally developed by Metropolis *et al.*,<sup>10</sup> and Geman and Geman<sup>11</sup> introduced the idea to use MCMC to segment images with a Markov random

field (MRF) statistical prior. Here our goal is to extend these methods to draw samples from a posterior distribution on the space of curves using non-parametric representations. The advantage of sampling curves instead of from MRFs is that curve sampling is an inherently geometric process that enables one to work explicitly in the space of shapes. This allows one to, *e.g.*, encode statistical properties of shape directly into the model and enables global object characteristics (whereas MRFs are inherently localized statistical descriptions).

With MCMC methods, an ergodic Markov chain with  $\pi(\vec{C} | I)$  as its stationary distribution<sup>12</sup> is constructed\*. We can then start the chain from any state  $\vec{C}_0$ , simulate it, and have the probability distribution of the state asymptotically approach  $\pi(\vec{C} | I)$ . The transition probability of the chain  $\mathbb{T}(\vec{C}^{(t-1)} \rightarrow \vec{C}^{(t)})$  is written as the product of two terms: a proposal distribution  $q(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)})$  and an acceptance probability function  $a(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)})$ . Then a sample  $\vec{C}^{(t)}$  from  $\mathbb{T}(\vec{C}^{(t-1)} \rightarrow \vec{C})$  can be drawn by accepting a sample  $\vec{\Gamma}^{(t)}$  from  $q(\vec{\Gamma} | \vec{C}^{(t-1)})$  with probability  $a(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)})$ ; otherwise  $\vec{C}^{(t)} = \vec{C}^{(t-1)}$ .

The proposal distribution is chosen so as to be easy to sample from, as MCMC methods change the problem of sampling from  $\pi$  to one of drawing many samples from  $q$ . It is also advantageous for speed reasons if samples from  $q$  tend to be in high-probability regions of  $\pi$ . A sufficient condition for discrete state spaces for  $\pi(\vec{C} | I)$  to be the stationary distribution of the chain is *detailed balance*:

$$\pi(\vec{C} | I) \mathbb{T}(\vec{C} \rightarrow \vec{\Gamma} | I) = \pi(\vec{\Gamma} | I) \mathbb{T}(\vec{\Gamma} \rightarrow \vec{C} | I) . \quad (7)$$

For continuous state spaces, a bit more care needs to be taken, but a similar statement can generally be made. See Neal<sup>12</sup> for details.

## 2.1 Metropolis-Hastings

Hastings<sup>13</sup> created the MCMC sampling approach commonly referred to as the Metropolis-Hastings algorithm by extending Metropolis' work to provide detailed balance when  $q$  is asymmetric ( $q(\vec{C} | \vec{\Gamma}) \neq q(\vec{\Gamma} | \vec{C})$ ). For an iterate  $\vec{C}^{(t-1)}$  and a candidate sample  $\vec{\Gamma}^{(t)}$ , the Metropolis-Hastings acceptance probability is defined as

$$a(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)}) = \min \left( 1, \eta(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)}) \right) \quad (8)$$

where the Hastings ratio  $\eta$  is

$$\eta(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)}) = \frac{\pi(\vec{\Gamma}^{(t)}) q(\vec{C}^{(t-1)} | \vec{\Gamma}^{(t)})}{\pi(\vec{C}^{(t-1)}) q(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)})} . \quad (9)$$

Using the ratio of the proposal distributions in the Hastings ratio ensures that the samples do not get overly attracted to high probability regions of  $q$ .

## 2.2 Bayesian segmentation model

To construct a Metropolis-Hastings sampling algorithm, we must specify  $\pi(\vec{C} | I)$  and  $q(\vec{\Gamma} | \vec{C})$ . Note that while the formulation we present is continuous, any implementation must necessarily be a discrete approximation. The choice of  $\pi$  is problem dependent (though one must be able to evaluate it). For the proposal distribution, arbitrary choices of  $q$  can result in a slow mixing rate. The more likely  $q$  is to generate high-probability samples from  $\pi$ , the faster convergence will be. For the proposal distribution, we choose to implicitly define  $q$  by explicitly defining the method of sampling from it. To generate a candidate sample  $\vec{\Gamma}$  from a previous iterate  $\vec{C}$ , we add a random perturbation to the normal of  $\vec{C}$  (the arc length-parameterized version of  $\vec{C}$ ):

$$\vec{\Gamma}(p) = \vec{C}_a(p) + f(p) \vec{\mathcal{N}}_{\vec{C}}(p) \delta t \quad (10)$$

where  $f(p)$  is a random field. The problem of generating  $\vec{\Gamma}$  is now the problem of generating  $f(p)$ . Note that we choose to define the perturbation relative to a canonical parameterization so that the perturbations are geometric and independent of the particular parameterization of  $\vec{C}$ . We implement equation (10) using a narrowband<sup>3</sup> level set approach for numerical stability reasons.

---

\*The ergodic condition implies that it is possible to get to any state from any other state with non-zero probability and that the chain is aperiodic. The stationary distribution of a Markov chain is the distribution  $\pi(\vec{C})$  that satisfies  $\pi(\vec{\Gamma}) = \int \pi(\vec{C}) \mathbb{T}(\vec{C} \rightarrow \vec{\Gamma}) d\vec{C}$  for all  $\vec{\Gamma}$ .

1. Start with an initial curve  $\vec{C}^{(0)}$  and set  $t = 1$ .
2. Create  $f^{(t)}(p) = h \circledast n^{(t)}(p) + \mu_{\vec{C}^{(t-1)}}(p)$ .
3. Compute  $\vec{\Gamma}^{(t)}(p) = \vec{C}_a^{(t-1)}(p) + f^{(t)}(p) \vec{N}_{\vec{C}^{(t-1)}}(p)$ .
4. Evaluate Hastings ratio  $\eta(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)})$  (Sec. 2.3).
5.  $\vec{C}^{(t)} = \vec{\Gamma}^{(t)}$  with probability  $\eta(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)})$ . Otherwise  $\vec{C}^{(t)} = \vec{C}^{(t-1)}$ .
6. Increment  $t$  and return to Step 2.

Table 1. Overall algorithm

For practical sampling implementations, it is necessary for both computing sample paths of  $f(p)$  and the probability of these sample paths to be fast. In this work, we focus on generating  $f(p)$  which are composed of a correlated zero-mean random process  $r(p)$  and a mean function  $\mu_{\vec{C}}(p)$ :

$$f(p) = \mu_{\vec{C}}(p) + r(p) . \quad (11)$$

We construct  $r(p)$  by circularly convolving white Gaussian noise  $n(p)$  with a smoothing kernel  $h(p)$  (e.g., a Gaussian kernel). Many other choices for generating  $r(p)$  are possible such as using Fourier or wavelet bases to provide perturbations that are more localized in space and frequency.

The mean process should be chosen to increase the convergence rate of the sampling algorithm. For instance, if  $\mu_{\vec{C}}(p)$  is the gradient flow of  $\pi(\vec{C} | I)$  (or  $\log \pi(\vec{C} | I)$ ), then samples from  $q$  are likely to move to higher probability regions of  $\pi$ . It is important, though, that the mean component does not overly dominate the random component of the perturbation to ensure the ergodicity of the chain.

Many reasonable choices are possible for the mean perturbation. Here, we define it as

$$\mu_{\vec{C}}(p) = -\kappa_{\vec{C}}(p) + \gamma_{\vec{C}} \quad (12)$$

where  $\gamma_{\vec{C}}$  is a positive value (which can be dependent on  $\vec{C}$  or a fixed constant) counteracting the curve shortening term  $\kappa_{\vec{C}}$ . As discussed earlier, a flow of  $f(p) = -\kappa_{\vec{C}}(p)$  is a regularizing flow which creates smooth curves. Empirical results show that without this deterministic smoothing force, generating smooth curves from a chain of  $q$  evaluations is so unlikely that the sampling procedure will not generate smooth curves in a reasonable amount of time. While each  $r(p)$  is smooth, the sum of  $N$  of these random fields is somewhat similar to convolving  $h(p)$  with a random white Gaussian process with variance proportional to  $N$ .

### 2.3 Detailed balance

Metropolis-Hastings sampling requires that we be able to evaluate both the forward and reverse proposal distributions. This computation needs to be understood in order to ensure detailed balance and guarantee that our samples come from the posterior. In the case of our curve perturbations, this is non-trivial because  $q$  is asymmetric due to the presence of the mean component. The mean component biases the proposal distribution to generate perturbations in a certain direction. Both the forward and reverse directions of the proposal distribution have the same bias but their perturbations are opposite of each other (to go from  $\vec{\Gamma}$  to  $\vec{C}$  requires a perturbation in the opposite direction of moving from  $\vec{C}$  to  $\vec{\Gamma}$ ). So if a perturbation around the mean was generated to go from  $\vec{C}$  to  $\vec{\Gamma}$ , a perturbation in the opposite direction of the mean is required to go from  $\vec{C}$  to  $\vec{\Gamma}$ . The former is a high-probability event while the latter is a low-probability event, thus  $q$  cannot be symmetric.

The perturbation defined in equation (10) is a differential in the direction of the normal, so each random perturbation maps one curve uniquely to another. This remains approximately true for small finite  $\delta t$ . Thus evaluating  $q(\vec{\Gamma} | \vec{C})$  is equivalent to evaluating the probability of generating  $f(p)$ .

For a numerical implementation of equation (11), we generate a Gaussian noise vector  $\mathbf{n}$  of fixed length, multiply it by a circulant matrix  $\mathbf{H}$  (which implements the circular convolution), and add a mean vector  $\mu_{\vec{C}}$ . This results in a Gaussian random vector  $\mathbf{f} \sim N(\mu_{\vec{C}}, \mathbf{H}\mathbf{H}^T)$ . Note that  $\mathbf{f}$  is deterministically generated from  $\mathbf{n}$  so

$$p_{\mathbf{f}}(\mathbf{H}\mathbf{n} + \mu_{\vec{C}}) \propto p_{\mathbf{n}}(\mathbf{n}) \propto \exp(-\frac{1}{2}\mathbf{n}^T \mathbf{n}) . \quad (13)$$

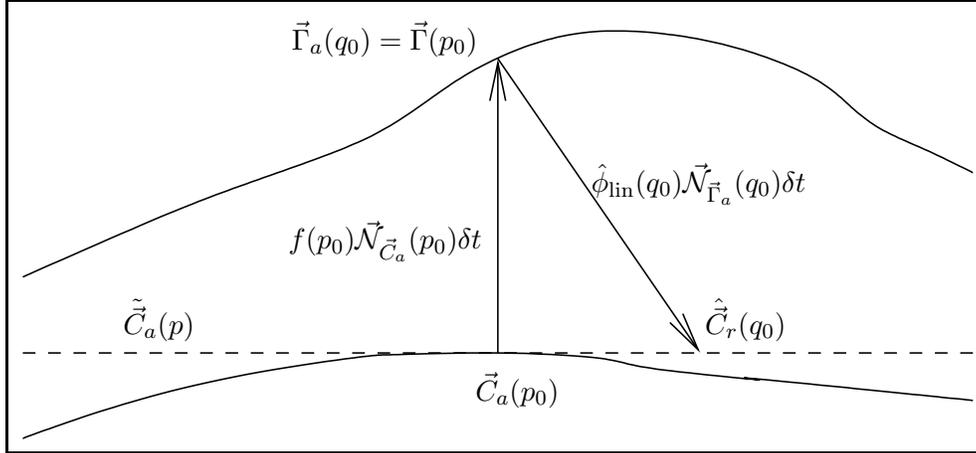


Figure 1. An example of the backtracing operation which must be done to compute the reverse perturbation. The linear approximation to  $\vec{C}$  is shown as the dashed line.

Unfortunately, computing the probability of the reverse perturbation  $q(\vec{C} | \vec{\Gamma})$  (which is necessary for detailed balance) is less straightforward. The following analysis demonstrates how to overcome this issue. The analogs to equations (10) and (11) are:

$$\vec{C}_r(q) = \vec{\Gamma}_a(q) + \phi(q)\vec{N}_{\vec{\Gamma}_a}(q)\delta t \quad (14)$$

$$\phi(q) = \mu_{\vec{\Gamma}}(q) + h(q) \otimes \nu(q) \quad (15)$$

with  $\mu_{\vec{\Gamma}}(q) = -\kappa_{\vec{\Gamma}}(q) + \gamma_{\vec{\Gamma}}$ . Note that we use a different parameter value  $q$  to indicate that the parameterizations between  $\vec{C}_a$  and  $\vec{\Gamma}_a$  are not directly compatible, and the matching curve  $\vec{C}_r$  in (15) is geometrically equivalent to  $\vec{C}$ , but does not have the same parameterization.

The main difficulty arises because  $\vec{N}_{\vec{C}} \neq \vec{N}_{\vec{\Gamma}}$  otherwise  $\phi(q)$  would just be  $-f(q)$ . In fact, for finite  $\delta t$ , it is possible to construct an  $f$  such that no  $\phi$  can satisfy equation (14). For small  $\delta t$  and smooth curves  $\vec{C}$  (say  $\kappa_{\vec{C}}(p)$  bounded), it is generally possible to construct  $\phi(q)$  by taking each point  $q_0$  on  $\vec{\Gamma}(q)$  and tracing along the normal  $\vec{N}_{\vec{\Gamma}}(q_0)$  until the line intersects with  $\vec{C}$  at some corresponding point  $p_0$  (essentially solving equation (14) for each  $q_0 \in [0, 1]$ ).

This is cumbersome, so in practice we approximate that computation by constructing linear approximations at each point  $q_0$  to  $\vec{C}$  and  $\vec{\Gamma}$  (using the tangent to the curve at  $p_0$  and tracing back along the normal). See Fig. 1 for an illustration of this process. This results in the following closed-form approximation:

$$\phi(q_0) \approx -\frac{f(p_0)}{\vec{N}_{\vec{C}}(p_0) \cdot \vec{N}_{\vec{\Gamma}}(q_0)}. \quad (16)$$

Note that when  $\vec{N}_{\vec{C}}(p_0) = \vec{N}_{\vec{\Gamma}}(q_0)$ , we get the expected  $\phi(q_0) = -f(p_0)$ . More accurate approximations can be found by using higher-order terms such as curvature. Note that this explicit correspondence we construct here means that even though we implement the perturbations using level sets, topological change is not valid for our Markov chain. To allow splitting or merging of regions, a jump-diffusion process must be used.<sup>9</sup>

Once we have computed  $\phi(q)$ , computing its probability is straightforward:

$$p_\phi(\mathbf{H}\nu + \mu_{\vec{\Gamma}}) = p_\nu(\nu) \propto \exp(-\frac{1}{2}\nu^T \nu). \quad (17)$$

We can then use the analysis we just performed (which is a detail that most implementations ignore) to ensure detailed balance by computing the ratio of the forward and reverse proposal distributions as

$$\frac{q(\vec{C}^{(t-1)} | \vec{\Gamma}^{(t)})}{q(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)})} = \exp\left(-\frac{1}{2}\left[\|\nu^{(t)}\|^2 - \|\mathbf{n}^{(t)}\|^2\right]\right) \quad (18)$$

with  $\nu^{(t)} = \mathbf{H}^{-1}(\phi^{(t)} - \mu_{\bar{\Gamma}^{(t)}})$ .

We summarize the algorithm in Table 1.

### 3. CONDITIONAL SIMULATION

One advantage of sampling-based approaches is that it enables the natural handling of situations when we are given exact information about part of the solution. With an optimization-based approach, this would require one to do constrained optimization which can be quite difficult in high-dimensional problems. Standard semi-automatic segmentation algorithms such as Livewire<sup>15</sup> can generally only fill in small gaps between user-specified points. With a sampling approach, we can use a technique often used in geophysics referred to as *conditional simulation*.<sup>16</sup> The basic idea is to generate samples from a multi-dimensional probability distribution conditioned on knowing some of the dimensions (in our case portions of the curve).

Let  $\vec{C}_k : [0, \beta] \rightarrow \Omega$  be the known part of the curve ( $\beta \in [0, 1]$ ) and  $\vec{C}_u : [\beta, 1] \rightarrow \Omega$  be the unknown part of the curve. Then

$$\vec{C}(s) = \begin{cases} \vec{C}_k(s) & : s \in [0, \beta] \\ \vec{C}_u(s) & : s \in [\beta, 1] \end{cases} \quad (19)$$

$\vec{C}_k(\beta) = \vec{C}_u(\beta)$  and  $\vec{C}_k(0) = \vec{C}_u(1)$  for  $\vec{C}$  to be closed. While we will only discuss the case where  $\vec{C}_k$  is on a single contiguous interval  $[0, \beta]$ , it is straightforward to generalize this approach for multiple intervals.

Now we wish to sample from

$$\pi(\vec{C}_u | I, \vec{C}_k) \propto p(I | \vec{C}_u, \vec{C}_k) p(\vec{C}_u | \vec{C}_k) . \quad (20)$$

Note that

$$p(\vec{C}_u | I, \vec{C}_k) = \frac{p(\vec{C}_u, \vec{C}_k | I)}{p(\vec{C}_k)} = \frac{p(\vec{C} | I)}{p(\vec{C}_k)} . \quad (21)$$

Thus, we can see that

$$\pi(\vec{C}_u | I, \vec{C}_k) \propto p(\vec{C} | I) \propto p(I | \vec{C}) p(\vec{C}) , \quad (22)$$

and computing the conditional target distribution is the same computation as in our regular sampling algorithm (and part of  $\vec{C}$  does not change with time).

To ensure that our samples from our proposal distribution stay on the manifold of curves that contain  $\vec{C}_k$ , we simply need to modify our proposal distribution to impose zero variance on part of the curve. A simple way to implement this is to multiply the random perturbations we defined in Sec. 2.2 by a scalar field:

$$r(s) = d(s) (\mu(s) + h(s) \otimes n(s)) \quad (23)$$

with  $d(s) = 0$  for  $s \in [0, \beta]$ ,  $d(s) = 1$  for  $s \in [\beta + \epsilon, 1 - \epsilon]$  and  $\epsilon > 0$ . From  $(\beta, \beta + \epsilon]$  and  $[1 - \epsilon, 1)$ , we should have  $d(s)$  smoothly transition from 0 to 1 so that there is not a strong variance mismatch at the end points of  $\vec{C}_k$ .

Computationally, this is equivalent to multiplying our random vector  $\mathbf{r}$  by a diagonal matrix  $\mathbf{D}$  resulting in  $\mathbf{r} \sim \mathbf{N}(\mathbf{D}\boldsymbol{\mu}, \mathbf{D}\mathbf{H}\mathbf{H}^T\mathbf{D})$ . Note that this is a degenerate probability distribution as some entries of  $\mathbf{r}$  have zero variance. Thus we should only evaluate  $q(\cdot | \cdot)$  using the the perturbation  $\mathbf{r}_u$  on the unknown part of the curve. Otherwise the computation is identical to that described in Sec. 2.3.

### 4. RESULTS

In this section we present results on two main applications: prostate magnetic resonance (MR) segmentation and a geological inversion problem. All images used were  $256 \times 256$ . For each application, we generated 1000 samples  $\{\vec{C}_i\}_{i=1}^{1000}$  from  $\pi(\vec{C} | I)$ . Computation time per sample ranged from 10 to 120 seconds on a Pentium 4 workstation depending on the complexity of the specific model and the nature of the configuration space. Each sample is generated independently of the others, so parallel computing systems can easily be used to increase sample throughput.

To display the results of the sampling methods, we will use three main visualization approaches:

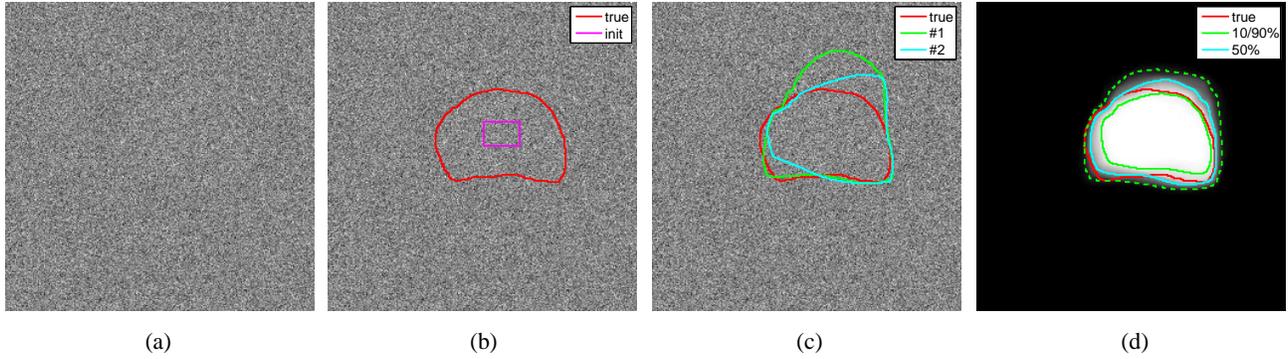


Figure 2. Results for the synthetic Gaussian example. (a) Observed image. (b) True boundary and initial curve. (c) Two most likely samples. (d) Marginal confidence bounds and histogram image.

1. Plotting the most likely samples (*e.g.*, Fig. 2(c)).
2. Histogram images (*e.g.*, Fig. 2(d)). For each  $\mathbf{x}$  we count the number of  $\vec{C}_i$  for which  $\mathbf{x}$  is inside the curve (*i.e.*,  $\Psi_{\vec{C}_i}(\mathbf{x}) < 0$ ). This is thus the marginal distribution over segmentation labels at each  $\mathbf{x}$ .
3. Marginal confidence bounds (*e.g.*, Fig. 2(d)). Given a histogram image, we plot the level contours. All points with probability 0.5 of being inside the curve can be viewed as analogous to a median contour. Other level contours can be viewed as confidence bounds (*e.g.*, the 10% confidence bound is the contour outside of which all pixels were inside fewer than 10% of the samples).

These visualizations are only possible because we are able to draw a large number of statistically-correct samples from the posterior distribution. Confidence bounds have been used in some previous image segmentation or reconstruction approaches,<sup>17,18</sup> but those dealt with parametric shape representations (so the uncertainty was over a finite set of parameters). It is important to note that our confidence representations are marginal statistics from infinite-dimensional non-parametric shape distributions.

While we do not present any direct comparisons to gradient-based curve evolution methods, the most likely samples we show can be viewed as a proxy for what an optimization-approach that can find global optima would capture. In the examples we show below, the global optimum does not provide a satisfactory segmentation (even for the synthetic example where the image is generated exactly according to the data model used in segmentation) due to the high levels of noise and clutter in the images. This is similar to trying to estimate the mean of a Gaussian random variable with variance  $\sigma^2$  from  $N$  samples. As the variance of the maximum likelihood (ML) estimate is  $\sigma^2/N$ , we would not be surprised for the ML estimate to be incorrect by  $\pm\sigma/\sqrt{N}$  (which can be very different from the true answer if  $\sigma$  is large). Note though that having access to higher-order statistics such as error variances can help constrain the range of likely answers, much as we do below.

#### 4.1 Synthetic Gaussian

In this section, we generate a synthetic image  $I$  exactly according to the implicit Chan-Vese model:

$$I(\mathbf{x}) = m(\mathbf{x}) + n(\mathbf{x}) \quad (24)$$

where  $m(\mathbf{x})$  is piecewise constant (0 outside the curve, 1 inside the curve) and  $n(\mathbf{x})$  is white Gaussian noise with  $\sigma = 10$  (for a SNR of  $-20$  dB), and the shape of the true curve that of a 2D prostate slice. Our probability density then is (3) exponentiated with the mean values inside and outside the curve known *a priori*. The observed image is shown in Fig. 2(a) with no readily apparent structure due to the low SNR. The true location of the contour is shown in Fig. 2(b).

In this example, the most-likely samples shown in Fig. 2(c) are actually not close to the true boundary in the upper-right corner of the object. While the model is correct, the noise levels in this image are high enough to create regions where the incorrect segmentation is more likely under the model. Note, though, that the marginal confidence bounds in

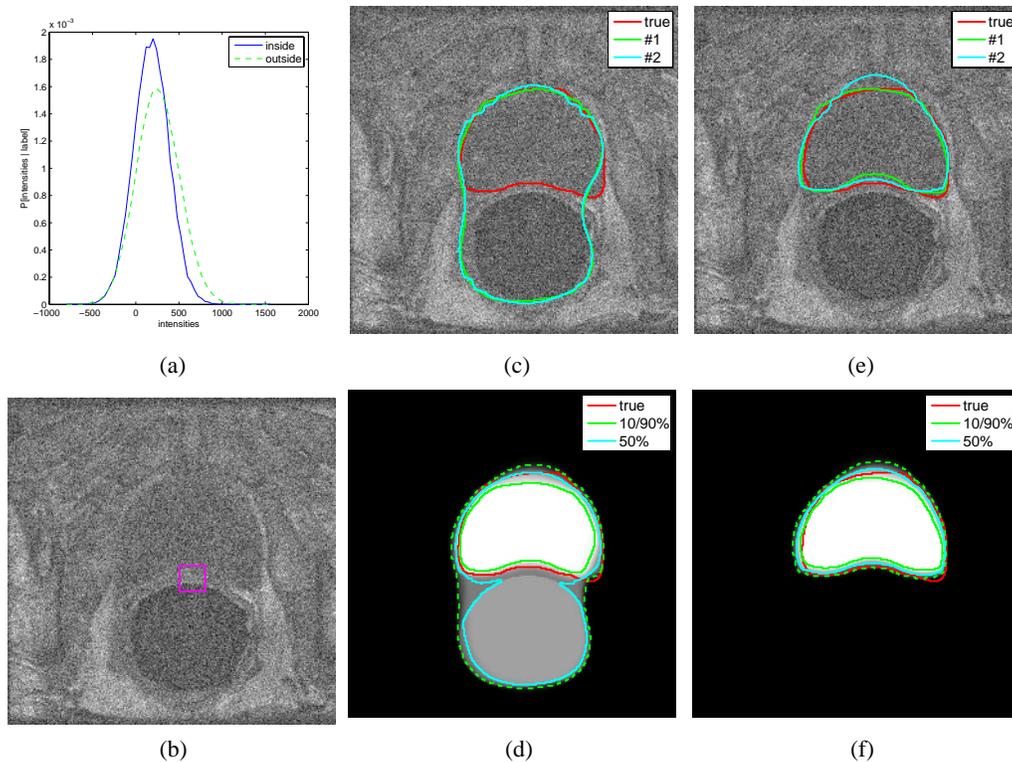


Figure 3. Prostate segmentation using non-parametric intensity distributions. (a) Pixel intensities for each class. (b) Initial curve. (c) Two most likely samples. (d) Marginal confidence bounds and histogram image. (e)-(f) Most likely samples and marginal for prostate-only cluster.

Fig. 2(d) show that the true contour location is mostly bracketed between the 90% and 10% confidence contours. The median contour is also quite close to the true boundary location indicating that the typical sample is perhaps providing more useful information than the most-likely sample. There is also a more diffuse histogram image (and a larger gap between the confidence bounds) in the upper right corner of the prostate which indicates greater uncertainty in that portion of the curve.

## 4.2 Prostate segmentation

Here we present results on a noisy T1-weighted prostate MR image. We assume that pixels are independent and identically distributed (iid) given the curve and learn (from segmented training data) non-parametric histogram distributions  $p(I(\mathbf{x})|0)$  and  $p(I(\mathbf{x})|1)$  (shown in Fig. 3(a)) to specify the pixel intensity distribution outside and inside the curve respectively. Then the data likelihood term is

$$\pi(I|\vec{C}) = \prod_{\mathbf{x}} p(I(\mathbf{x})|\mathcal{H}(\Psi_{\vec{C}}(\mathbf{x}))) \quad (25)$$

with  $\mathcal{H}$  the Heaviside function. We use a standard curve length penalty as the prior.

The histogram image and the marginal confidence bounds in Fig. 3(d) show that this distribution has three primary modes: one is around the correct prostate segmentation (the red contour); another segments just the rectum (the dark region located below the prostate); and the third encompasses both the prostate and the rectum. As can be seen in Fig. 3(c), the curves that segment the two regions together are actually from the most likely mode. The reason for this can be seen in the image intensity distributions. Due to the noise and our simple iid model, even though the rectum appears to be darker than the prostate, all pixel intensities below some threshold (approximately a value of 350) result in a higher likelihood for that pixel to be inside the curve. Hence the most likely curves are actually the ones that segment the prostate and rectum together.

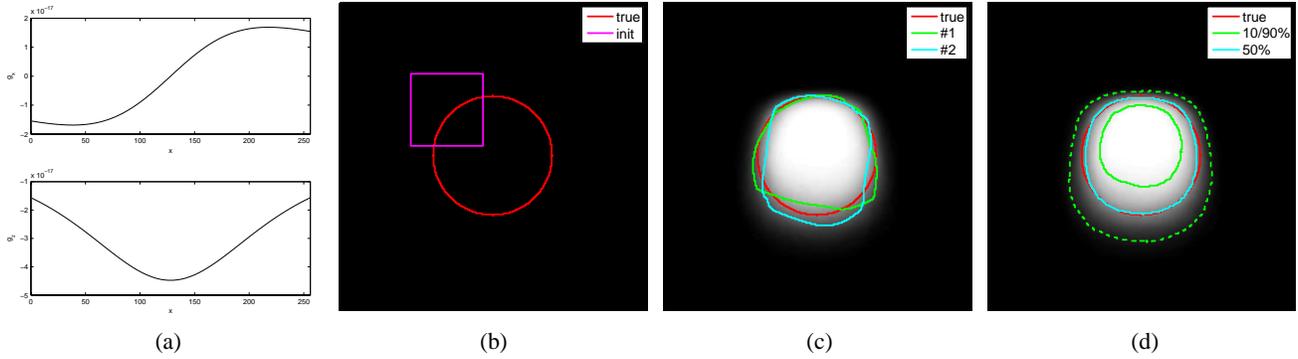


Figure 4. Synthetic circular salt body example. (a) Observed gravity profile. (b) Initial curve and true boundary. (c) Two most likely samples. (d) Marginal confidence bounds and histogram image.

Without having any additional *a priori* information, it would be difficult to say which of these three scenarios is the correct one. In fact, it is possible in some applications where multiple modes all provide reasonable explanations of the data. One approach we can take here is to utilize the information our sampling procedure provides to us. While the aggregate marginal statistics do not appear to be providing very useful information (though the 90% confidence boundary is located within the true prostate boundary), it is straightforward to create three clusters of samples. An expert user or a shape-driven classifier could then pick the best cluster to use. We show the most-likely samples and the marginal confidence boundaries for the prostate-only cluster in Fig. 3(e) and (f).

### 4.3 Gravity inversion

In petroleum geophysical applications, it is often important to be able to accurately locate underground salt bodies. Fluid does not flow through salt bodies, so water, oil, and gas tend to become trapped next to salt bodies. The tops of salt bodies are usually easy to locate using standard seismic tomographic imaging techniques, but locating the bottoms is more difficult due to the strong attenuation of seismic waves in salt bodies. Without accurate knowledge of the salt boundaries, high-quality seismic imaging below the salt body is impossible.

Gravity inversion is one technique that can be used to localize salt bodies.<sup>19</sup> Salt is less dense than the surrounding rock, so the presence of salt will cause a deviation from the expected gravitational field at the earth's surface which can be measured by very precise gravimeters. We will discuss a 2D version of the formulation that naturally generalizes to 3D. The measurements are preprocessed by subtracting various effects (such as the geoid and centrifugal force from the rotation of the earth) so the resulting observed values only depend on the mass distribution  $\rho(\mathbf{x})$  inside the image domain  $\Omega$ . Note that this is a very ill-posed problem. In the example we show, there were 512 surface measurements from which we try to reconstruct an image containing 65,536 pixels.

We assume there is an array of evenly-spaced gravimeters located at  $\mathbf{x}_i = (i\Delta x, 0)$  having associated vector gravity measurements  $\vec{g}_i = (g_{x,i}, g_{z,i})$  for  $i \in \{1, 2, \dots, N_g\}$ . The measurements can be modeled as:

$$\vec{g}_i(\rho) = G \int_{\Omega} \frac{\rho(\mathbf{x})(\mathbf{x} - \mathbf{x}_i)}{\|\mathbf{x} - \mathbf{x}_i\|^3} d\mathbf{x} \quad (26)$$

with  $G$  the universal gravitational constant.

The densities are relatively constant within salt and non-salt regions, so we assume  $\rho$  takes on a simple form:

$$\rho(\mathbf{x}) = \rho_0 + \Delta\rho\mathcal{H}(-\Psi(\mathbf{x}; \vec{C})) \quad (27)$$

This results in the following forward gravity model:

$$\vec{g}_i(\vec{C}) = \vec{g}_0 + \Delta\rho G \int_{\mathcal{R}(\vec{C})} \frac{(\mathbf{x} - \mathbf{x}_i)}{\|\mathbf{x} - \mathbf{x}_i\|^3} d\mathbf{x} \quad (28)$$

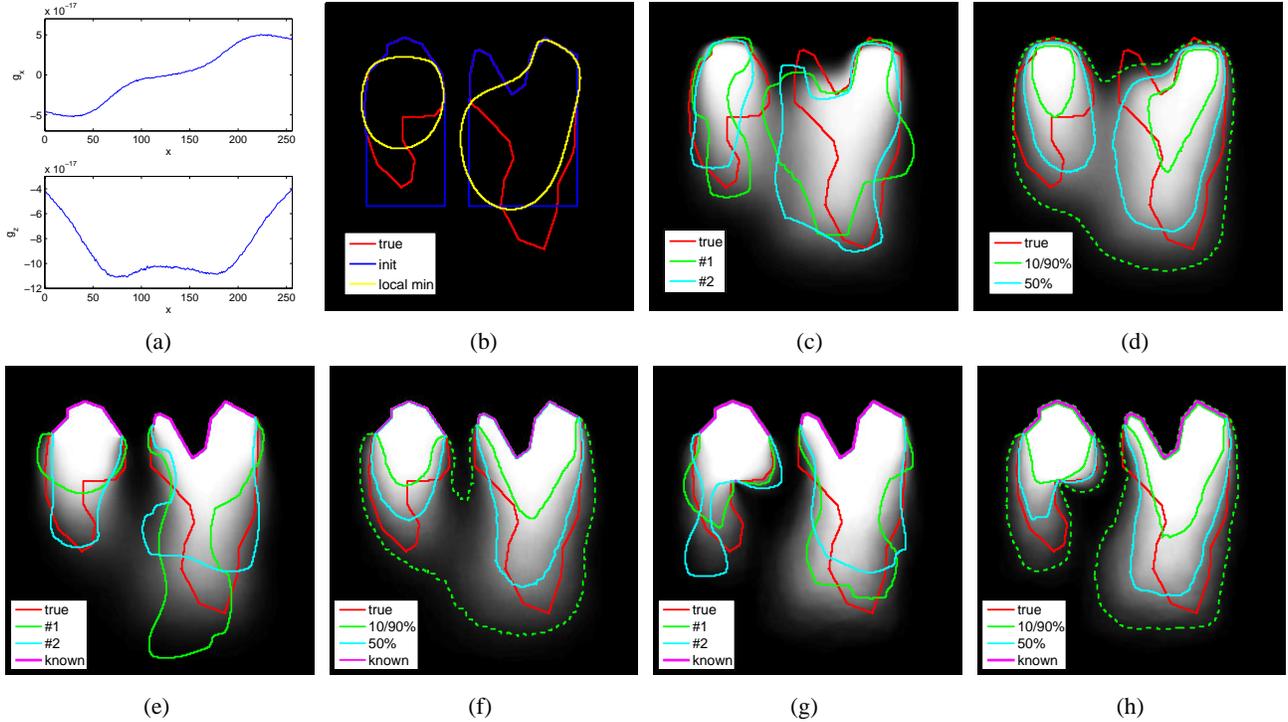


Figure 5. Synthetic salt body example with two disjoint regions and complex geometry. (a) Observed gravity profile. (b) True boundary, initial curve, and curve found using gradient-based curve evolution. Most probable samples and marginal confidence bounds for (c)-(d) unconstrained problem, (e)-(f) top-salt fixed, and (g)-(h) top-salt and an additional point fixed. The known portion of the curve is shown in magenta.

We then define our energy functional as the L2 difference between observed gravity measurements  $\vec{\xi}_i$  and forward model estimates  $\vec{g}_i(\vec{C})$  with a regularization term:

$$E(\vec{C}) = \sum_{i=1}^{N_g} \|\vec{g}_i(\vec{C}) - \vec{\xi}_i\|^2 + \alpha \oint_{\vec{C}} ds \quad (29)$$

In the results we present here, synthetic gravity measurements are generated exactly according to equation (26) using some true salt boundary  $\vec{C}^*$ .

We begin with a salt body with a simple circular geometric structure. In Fig. 4(a) we show the observed vector gravity profile, and in Fig. 4(b) the true salt boundary in red and the initial curve in magenta. The two most likely samples in Fig. 4(c) come reasonably close to the true curve, and the marginal median contour in Fig. 4(d) is quite accurate as well. Note that the error behaves as we would expect with greater uncertainty at the bottom of the salt body than at the top (due to the  $1/r^2$  term in the measurement model).

In Fig. 5, we show an example with more complex geometry. We are given *a priori* the information that there are two separate regions. If we did not know the topology beforehand, one approach would be to run the sampler multiple times, varying the number of curves. We now track two curves  $\vec{C}_1$  and  $\vec{C}_2$  which are treated independently in the proposal distribution:

$$q(\vec{\Gamma}_1, \vec{\Gamma}_2 | \vec{C}_1, \vec{C}_2) = q(\vec{\Gamma}_1 | \vec{C}_1) q(\vec{\Gamma}_2 | \vec{C}_2) . \quad (30)$$

The curves are coupled together only through  $\pi(\vec{C}_1, \vec{C}_2 | I)$ . Computation of equation (28) is largely unchanged except a pixel is considered to be salt if it is inside either curve. The two curves are allowed to overlap, but both the data and the curve length penalty make that unlikely in this case.

In Fig. 5(a) and (b), we show the observed gravity measurements with the true salt boundaries and the initial curves and the result of running an optimization-based curve evolution method (the derivation of the gradient flow can be found

in Fan<sup>14</sup>). Because the top salt boundary is usually available through other means, we allow the initialization to have this information. We show the results of running the normal sampling procedure in Fig. 5(c). In Fig. 5(d), we fix the location of the top-salt and conditionally simulate the rest of the curve. In Fig. 5(e) we further fix an additional location at the high curvature point on the left salt body (this can possibly be obtained using an exploratory well). The conditional simulation results exhibit much less sample variance, more accurate estimates, and allow much faster computation (due to reduced mixing time and reduced number of samples needed to accurately represent the distribution). As an indication of the ill-posedness of this problem, the top two samples in Fig. 5(d) have virtually identical gravity errors but are fairly different geometrically.

Conditional simulation is particularly useful for globally-coupled estimation problems. For instance, in the gravity inversion problem, the locations of the top of the salt and the bottom are correlated. If the estimate of the top is too low, the observed gravity at the surface can still be accurately explained if our estimate of the bottom is also too low (by a scale factor reflecting the  $1/r^2$  of gravity measurements). Thus fixing the location of the top of the salt to some extent fixes a range for the location of the bottom of the salt. This is not the case with energy functionals such as Chan-Vese which are more local in nature.

One thing to notice is that we have trouble accurately segmenting portions of the curve with high positive or negative curvature because our proposal distribution makes it unlikely that those types of samples will be generated. A different proposal distribution that still smooths the curve but is not based on curvature would alleviate this issue.

## 5. CONCLUSION

In this paper, we presented an approach to generate samples from probability distributions defined on spaces of curves by constructing a MCMC algorithm and showing how to properly compute the proposal distribution probabilities to ensure detailed balance and asymptotic convergence to a desired posterior distribution. The sampling approach provided robustness to local minima in low-SNR and ill-posed problems, and we showed how a large number of curve samples can be used to provide useful aggregate statistics (such as non-parametric marginal confidence bounds) about the likely location of the true curve locations. Experimental results demonstrated the usefulness of this aggregate information even when the most likely curves do not provide satisfactory segmentations.

We developed a semi-automatic segmentation algorithm which extended our sampling approach to enable conditional simulation. In many application domains of interest, segmentations are currently performed by an expert such as a radiologist or a geological interpreter. Rather than trying to remove them completely from the equation, we can create a feedback system that allows them to focus their expertise and knowledge on the most difficult portions of the problem. In fact, sampling methods can actually provide information about where the greatest uncertainty is and, thus, where expert assistance is most needed.

Future work in this space involves developing faster sampling algorithms by utilizing better proposal distributions or multiresolution methods; extending the framework to non-closed curves and unknown topology; creating uncertainty measures that provide information about the local characteristics of the shape manifold; and extending the conditional simulation algorithm to allow for uncertainty in the user-provided curve location.

## ACKNOWLEDGMENTS

We wish to thank Clare Tempany and Sandy Wells for their assistance in acquiring data. Our research was primarily supported by a grant from Shell International Exploration and Production, Inc.

## REFERENCES

1. M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Intl. J. Comp. Vis.* **1**(4), pp. 321–331, 1988.
2. S. Osher and J. Sethian, "Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulation," *J. Comp. Phys.* **79**, pp. 12–49, 1988.
3. J. Sethian, *Level Set Methods and Fast Marching Methods*, Cambridge University Press, 1999.
4. T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Imag. Proc.* **10**, pp. 266–277, 2001.
5. A. Tsai, A. Yezzi, and A. Willsky, "Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification," *IEEE Trans. Imag. Proc.* **10**(8), pp. 1169–1186, 2001.

6. W. Sun, M. Cetin, R. Chan, *et al.*, "Segmenting and tracking the left ventricle by learning the dynamics in cardiac images," in *IPMI 2005*, pp. 553–565, 2005.
7. M. de Bruijne and M. Nielsen, "Shape particle filtering for image segmentation," in *MICCAI 2004, LNCS 3216, I*, pp. 168–175, 2004.
8. O. Juan, R. Keriven, and G. Postelnicu, "Stochastic motion and the level set method in computer vision: Stochastic active contours," *Intl. J. Comp. Vis.* **69**(1), 2006.
9. Z. Tu and S. C. Zhu, "Image segmentation by data-driven Markov chain Monte Carlo," *IEEE Trans. Patt. Anal. Mach. Intell.* **24**(5), pp. 657–673, 2002.
10. N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, "Equations of state calculations by fast computing machines," *J. Chem. Phys.* **21**(6), 1953.
11. S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE PAMI* **6**, pp. 721–741, 1984.
12. R. M. Neal, "Probabilistic inference using Markov chain Monte Carlo methods," Tech. Rep. CRG-TR-93-1, Univ. of Toronto, 1993.
13. W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika* **57**(1), pp. 97–109, 1970.
14. A. Fan, *Curve Sampling and Geometric Conditional Simulation*. PhD thesis, MIT, 2007.
15. A. Falcao, J. Udupa, S. Samarasekera, S. Sharma, B. Hirsch, and R. Lotufo, "User-steered image segmentation paradigms - Live Wire and Live Lane," *Graph. Mod. and Image Proc.* **60**(4), pp. 233–260, 1998.
16. L. Wang, "Modeling complex reservoir geometries with multiple-point statistics," *Math. Geol.* **28**(7), p. 895, 1996.
17. Y. Zhang, A. O. Hero, and W. L. Rogers, "Simultaneous confidence intervals for image reconstruction problems," in *Proceedings of ICASSP*, pp. 317–320, 1994.
18. J. C. Ye, Y. Bresler, and P. Moulin, "Asymptotic global confidence regions in parametric shape estimation problems," *IEEE Trans. Inf. Theory* **46**(5), pp. 1881–1895, 2000.
19. R. Blakely, *Potential Theory in Gravity and Magnetic Applications*, Cambridge University Press, 1995.