

# MCMC Curve Sampling for Image Segmentation

AYRES FAN<sup>1</sup>, JOHN W. FISHER III<sup>1,2</sup>, WILLIAM M. WELLS III<sup>2,3</sup>, JAMES J. LEVITT<sup>3,4</sup>, AND ALAN S. WILLISKY<sup>1</sup>

<sup>1</sup>Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, USA  
<sup>2</sup>Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA  
<sup>3</sup>Dept. of Radiology, Brigham and Women's Hospital, Harvard Medical School, Boston, MA, USA  
<sup>4</sup>Dept. of Psychiatry, VA Boston Healthcare System, Harvard Medical School, Boston, MA, USA  
{fan, willsky}@mit.edu, {fisher, sw}@csail.mit.edu



## Introduction

- Hybrid Markov Chain Monte Carlo (MCMC) and curve evolution sampling approach.
- We generate samples using correlated Gaussian curve perturbations and show how to apply the Metropolis-Hastings acceptance rule to provide detailed balance.
- This leads to an iterative algorithm to sample from probability distributions on curves.
- Extend to a *conditional simulation* approach for interactive segmentation where part of the curve is specified and the remainder is sampled conditioned on the known part.
- Major advantages:
  - Naturally handle problems with multi-modal distributions.
  - Most probable samples similar to running a global optimizer.
  - Multiple samples can be combined together to provide higher-order curve statistics (e.g., error estimates).
  - Conditional simulation approaches can substantially lessen work required for manual segmentation and dramatically reduce uncertainty for low-contrast problems.

## Background

### Markov Chain Monte Carlo

Markov Chain Monte Carlo (MCMC) methods are a class of algorithms designed to generate samples from a target distribution  $\pi(x)$ , which is difficult to sample from directly. Instead a Markov chain with transition probability  $T(y|x)$  and stationary distribution  $\pi(x)$  is constructed:

$$\pi(z) = \int \pi(x)T(z|x)dx \quad (1)$$

**Detailed balance** is a sufficient condition for this to hold:

$$\pi(z)T(x|z) = \pi(x)T(z|x) \quad (2)$$

If a chain is ergodic and detailed balance holds, sequential samples from  $T(z|x)$  asymptotically become samples from  $\pi(x)$ .

**Metropolis-Hastings** is a general MCMC algorithm developed by Metropolis *et al.* (1953) and extended by Hastings (1970). The transition probability  $T(z|x)$  is defined as the product of a proposal distribution  $q(y|x)$  and an acceptance probability  $a(y|x)$ . A candidate sample is generated from  $q$ , and the **Hastings ratio** is computed:

$$\eta(y|x) = \frac{\pi(y)q(x|y)}{\pi(x)q(y|x)} \quad (3)$$

Then the acceptance rule states  $z = y$  with probability  $\min(1, \eta(y|x))$ . Otherwise  $z = x$ . Detailed balance automatically follows.

### Curve Evolution

Given an image  $I$  defined on an image domain  $\Omega \subset \mathbb{R}^2$ , curve evolution methods find a curve  $\vec{C} : [0, 1] \rightarrow \Omega$  that minimizes an **energy functional**  $E(\vec{C})$  using gradient descent. This results in a geometric PDE flow:

$$\frac{d\vec{C}}{dt}(p) = f(p)\vec{N}_{\vec{C}}(p) \quad (4)$$

where  $f$  is a force function and  $\vec{N}_{\vec{C}}$  is the normal to the curve.

A classical example of this is the Euclidean curve shortening flow:

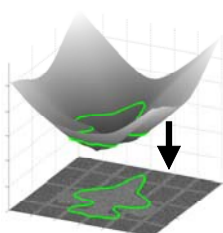
$$\frac{d\vec{C}}{dt}(p) = -\kappa_{\vec{C}}(p)\vec{N}_{\vec{C}}(p) \quad (5)$$

This flow minimizes the energy  $E(\vec{C}) = \int_{\vec{C}} ds$  where  $ds = \|\vec{C}'(p)\|dp$  is differential arc length. This generally has a smoothing effect, and it is common to use as a regularizing prior term.

**Level set methods** are a standard numerical implementation which evolve a surface  $\Psi$  (one dimension higher than our curve) whose zeroth level set is  $\vec{C}$ . (Osher and Sethian 1988) Flow on a curve is related to a flow on the level set:

$$\frac{d\Psi}{dt} = \frac{d\vec{C}}{dt} \cdot \nabla \Psi = f(\vec{N}_{\vec{C}} \cdot \nabla \Psi) = f\|\nabla \Psi\| \quad (6)$$

Need to do **velocity extension** to make  $f$  defined on all of  $\Omega$ .



## Sampling Framework

We define an algorithm to sample curves using a Metropolis-Hastings MCMC algorithm which transforms the problem of sampling from  $\pi$  into one of generating many samples from  $q$ . We view the curve evolution energy functional  $E(\vec{C}; I)$  as a negative log probability distribution. This means we write our target distribution as:

$$\pi(\vec{C} | I) \propto \exp(-E(\vec{C}; I)) \quad (7)$$

In this framework, we implicitly define  $q(\vec{\Gamma} | \vec{C})$  by explicitly defining how to sample from it. Candidate samples are generated through random Gaussian perturbations, and then we approximately compute the probability of those perturbations. Similar approach to Tu and Zhu (2002). Largest difference is we address the issue of detailed balance, so we are generating samples, not doing optimization.

### Overall Algorithm

- Set  $\vec{C}^{(0)}$  to some initial value (deterministic or random) and  $t = 1$ .
- Generate candidate sample  $\vec{\Gamma}^{(t)} \sim q(\vec{\Gamma} | \vec{C}^{(t-1)})$  by creating Gaussian perturbation  $f^{(t)}$ . This results in  $\vec{\Gamma}^{(t)}(p) = \vec{C}^{(t-1)}(p) + f^{(t)}(p)\vec{N}_{\vec{C}^{(t-1)}}(p)\delta t$  for some positive constant  $\delta t$ . Implement with level sets, but no topology change allowed.
- Compute Hastings ratio  $\eta(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)})$  which requires evaluation of the forward and reverse perturbation probabilities  $q(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)})$  and  $q(\vec{C}^{(t-1)} | \vec{\Gamma}^{(t)})$  as well as the target distribution probabilities  $\pi(\vec{C}^{(t-1)})$  and  $\pi(\vec{\Gamma}^{(t)})$ .
- Accept or reject  $\vec{\Gamma}^{(t)}$  with probability  $\eta(\vec{\Gamma}^{(t)} | \vec{C}^{(t-1)})$  to obtain the current iterate value  $\vec{C}^{(t)}$ .
- Increment  $t$  and return to Step 2.

### Forward Perturbation

To make the sampling process geometric, we define all perturbations relative to a canonical arc length curve parameterization. We denote the arc length-parameterized curve as  $\vec{C}_a$  and drop the time superscripts for simplicity. We then generate random, correlated Gaussian noise:

$$f(p) = \mu_{\vec{C}_a}(p) + h \otimes n(p) \quad (8)$$

These perturbation are added to the normal of the curve:

$$\vec{\Gamma}(p) = \vec{C}_a(p) + f(p)\vec{N}_{\vec{C}_a}(p)\delta t \quad (9)$$

Our standard choice for  $\mu$  is a smoothing term:

$$\mu_{\vec{C}_a}(p) = -\alpha\kappa_{\vec{C}_a}(p) + \gamma\vec{C}_a \quad (10)$$

The probability of the forward perturbation is (approximating as discretized vectors):

$$q(\vec{\Gamma} | \vec{C}) = p(f) = p(\mu_{\vec{C}_a} + Hn) \propto \exp\left(-\frac{n^T n}{2\sigma^2}\right) \quad (11)$$

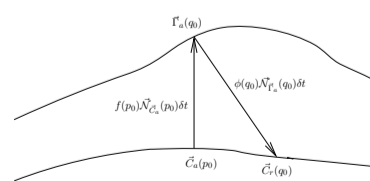
### Reverse Perturbation

The reverse perturbation (going from  $\vec{\Gamma}$  back to  $\vec{C}$ ) can be written as:

$$\vec{C}_r(q) = \vec{\Gamma}_a(q) + \phi(q)\vec{N}_{\vec{\Gamma}_a}(q)\delta t \quad (12)$$

$$\phi(q) = \mu_{\vec{\Gamma}_a}(q) + h \otimes \nu(q) \quad (13)$$

Here  $\vec{C}_r$  is geometrically identical to  $\vec{C}$  but with a different parameterization. We can find  $\phi$  by tracing back along the normal of  $\vec{\Gamma}_a$ .



This computation can be approximated by forming locally-linear approximations to  $\vec{C}_a$  to obtain:

$$\hat{\phi}(q_0) = -\frac{f(p_0)}{\langle \vec{N}_{\vec{C}_a}(p_0), \vec{N}_{\vec{\Gamma}_a}(q_0) \rangle} \quad (14)$$

The curve parameters  $p_0$  and  $q_0$  are defined so that  $\vec{\Gamma}(p_0) = \vec{\Gamma}_a(q_0)$ . This results in a reverse perturbation probability:

$$q(\vec{C} | \vec{\Gamma}) = p(\phi) = p(\mu_{\vec{\Gamma}_a} + H\nu) \propto \exp\left(-\frac{\nu^T \nu}{2\sigma^2}\right) \quad (15)$$

## Experimental Results

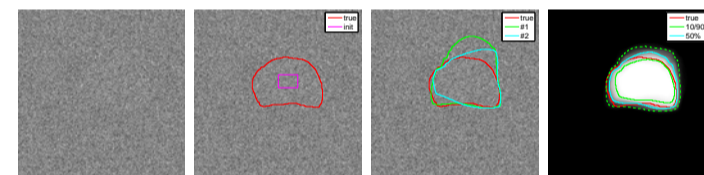
We demonstrate the algorithm on three examples: a synthetic image with very low SNR, a noisy prostate magnetic resonance (MR) image, and a thalamus MR segmentation problem. For each example, 1000 samples are generated. We use three main techniques to visualize the output:

- Plotting the **most likely samples**. These can be viewed as a proxy for what a global optimization approach would capture.
- Histogram images**. For each pixel  $x$ , we count the samples for which  $x$  is inside the curve. This value  $\Phi(x)$  is the marginal distribution over segmentation labels at each  $x \in \Omega$ .
- Marginal confidence bounds**. Given a histogram image, we plot the level contours (the  $k$ th level contour is the set of points  $x$  such that  $\Phi(x) = k$ ), which 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).

### Synthetic Example

Generate image as piecewise constant plus Gaussian noise (forward model implied by Chan-Vese energy). Overall SNR is -20 dB. Energy functional is:

$$E(\vec{C}) = \iint_{\mathcal{R}_c} (I - m_1)^2 dx + \iint_{\mathcal{R}_c^c} (I - m_0)^2 dx + \alpha \int_{\vec{C}} ds \quad (16)$$



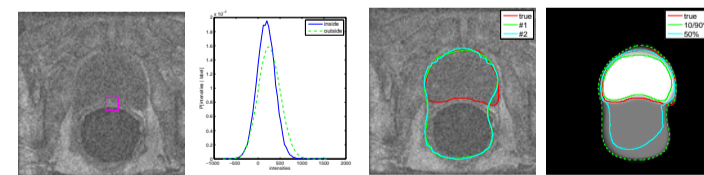
In this case, due to the specific noise realization, the most probable samples are actually not very good in the upper-right corner. Histogram image and confidence bounds tell us likely range of locations for the true location whereas optimization would only give us an incorrect answer.

### Prostate Segmentation

The images were obtained from a prostate MR image captured with a surface coil, and the bias field was removed using the technique in Fan *et al.* (2003). Gaussian noise was added to then simulate a T1-weighted body coil image. We assume that pixels are iid given the curve and learn non-parametric histogram distributions  $p(I(x)|0)$  and  $p(I(x)|1)$  (shown below) to specify the pixel intensity distribution outside and inside the curve respectively. The overall data likelihood term is

$$p(I | \vec{C}) = \prod_x p(I(x) | \mathcal{H}(\Psi_{\vec{C}}(x))) \quad (17)$$

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



We can see that this distribution has three primary modes: one around the correct prostate segmentation; another which segments just the rectum; and the third encompasses both the prostate and the rectum. The curves that segment the two regions together are actually from the most likely mode due to the learned intensity models and simplistic iid assumption.

While the aggregate marginal statistics do not appear to be providing very useful information, we can cluster the samples into prostate-only, rectum-only, and prostate and rectum segmentations.

We show the most-likely samples and the marginal confidence boundaries for the prostate-only cluster. Note that an optimization-based approach would have only found the prostate and rectum cluster due to the multi-modality.

## Conditional Simulation

In many problems, multiple reasonable solutions may exist due to low SNR or ill-posedness. Conditional simulation involves sampling part of the solution conditioned on the rest being known (e.g., pinned Brownian motion). For curve sampling, this means that part of the curve is specified. This approach is much more feasible with sampling than it would be doing constrained optimization in high-dimensional spaces.

Let  $\vec{C}_k : [0, b] \rightarrow \Omega$  be the known portion of the curve, and  $\vec{C}_u : [b, 1] \rightarrow \Omega$  be the unknown portion. When there is no uncertainty associated with  $\vec{C}_k$ , the conditional target distribution  $\tilde{\pi}(\vec{C}_u | \vec{C}_k)$  can be written as:

$$\tilde{\pi}(\vec{C}_u | \vec{C}_k) \propto p(I | \vec{C})p(\vec{C}_u, \vec{C}_k) / p(\vec{C}_k) \propto \pi(\vec{C} | I) \quad (18)$$

We see that evaluation of the target distribution is unchanged. The difference is that some of the curve does not change with time, so the proposal distribution must be modified so that perturbed curves remain on the manifold of curves which contain  $\vec{C}_k$ . To do so, we simply multiply our perturbation  $f(p)$  by a smooth scalar field  $d(p)$  which is 0 on  $[0, b]$ .

### Thalamus Segmentation

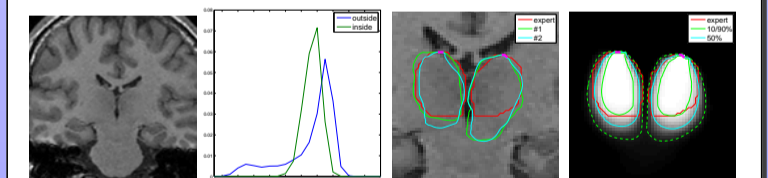
Low contrast makes it difficult to distinguish the thalamus from surrounding cerebral tissue. One approach to this problem is to use shape models (Pohl *et al.* 2004). Here we apply our conditional simulation approach which requires much less training and allows more user control over the segmentation process.

Note that the thalamus has two halves which we track with independent curves  $\mathcal{C} = \{\vec{C}_i\}_{i=1}^2$ . We perturb each curve individually, and the curves are combined into a joint label map  $\lambda_{\mathcal{C}}(x)$  which is 1 if  $x$  is inside any  $\vec{C}_i$ . The evaluation of  $\pi$  is now done relative to  $\lambda_{\mathcal{C}}(x)$ .

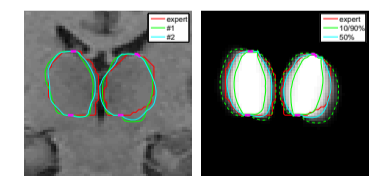
We apply a similar non-parametric data intensity model as for the prostate, except the probability distribution is only defined in a band 10-pixels thick around the curve location.

$$p(I | \mathcal{C}) = \prod_{\substack{\{x\} \text{ i.s.t.} \\ |\Psi_{\vec{C}_i}(x)| \leq d_0}} p(I(x) | \lambda_{\mathcal{C}}(x)) \quad (19)$$

We begin by fixing a point on the top of each half of the thalamus (without any constraints, the samples generated are extremely poor):



We can see there is greater uncertainty (in terms of the gap in the confidence bounds) at the bottom. If we iteratively introduce additional expert information at the bottom, we can refine the results:



## Conclusion

- Introduced a framework combining curve evolution and MCMC methods to sample from probability distributions on curves.
- Showed how to evaluate probability of perturbations to ensure detailed balance.
- Demonstrated benefits over optimization-based methods:
  - Noisy synthetic example: global maximum is skewed by the noise, but marginal confidence bounds bracket the true curve location.
  - Prostate: multi-modal distribution. Clustering to characterize different modes.
  - Thalamus: conditional simulation to perform semi-automatic segmentation.