Level-set MCMC Curve Sampling and Geometric Conditional Simulation

Ayres Fan
John W. Fisher III
Alan S. Willsky
Stochastic Systems Group

February 16, 2007

Outline

1. Overview
2. Curve evolution
3. Markov chain Monte Carlo
4. Curve sampling
5. Conditional simulation
6. 2.5D Segmentation

Overview

- Curve evolution attempts to find a curve C (or curves C_i) that best segment an image (according to some model)
- Goal is to minimize an energy functional E(C) (view as a negative log likelihood)
- Find a local minimum using gradient descent

Sampling instead of optimization

- Draw multiple samples from a probability distribution **p** (e.g., uniform, Gaussian)
- Advantages:
  - Naturally handles multi-modal distributions
  - Can get out of local minima
  - Higher-order statistics (e.g., variances)
  - Conditional simulation

Outline

1. Overview
2. Curve evolution
3. Markov chain Monte Carlo
4. Curve sampling
5. Conditional simulation
6. 2.5D Segmentation

Planar curves

- A curve is a function $\vec{C} : [0, 1] \rightarrow \mathbb{R}^2$
- We wish to minimize an energy functional with a data fidelity term and regularization term:

$$\mathsf{E}(\vec{C}) = \mathsf{D}(y|\vec{C}) + \mathcal{R}(\vec{C})$$

- This results in a gradient flow:

$$\frac{\mathrm{d}\vec{C}}{\mathrm{d}t}(p) = \vec{\mathsf{F}}(p)$$

- We can write any flow in terms of the normal:

$$\frac{\mathrm{d}\vec{C}}{\mathrm{d}t}(p) = \mathsf{f}(p)\vec{\mathcal{N}}(p)$$

Euclidean curve shortening flow

- Let $\mathsf{E}(\vec{C}) = \oint_{\vec{C}} \mathsf{d}s$
- This energy functional is smaller when C is shorter
- Gradient flow is direction that minimizes the curve length the fastest
- Use Euler-Lagrange and we see

$$\frac{\mathsf{d}\vec{C}}{\mathsf{d}t}(p) = -\kappa(p)\vec{\mathcal{N}}(p)$$

where κ is curvature, N is the outward normal

Level-Set Methods

- A curve is a function (infinite dimensional)
- A natural implementation approach is to use marker points on the boundary (snakes)
  - Reinitialization issues
  - Difficulty handling topological change
- Level set methods instead evolve a surface (one dimension higher than our curve) whose zeroth level set is the curve (Sethian and Osher)

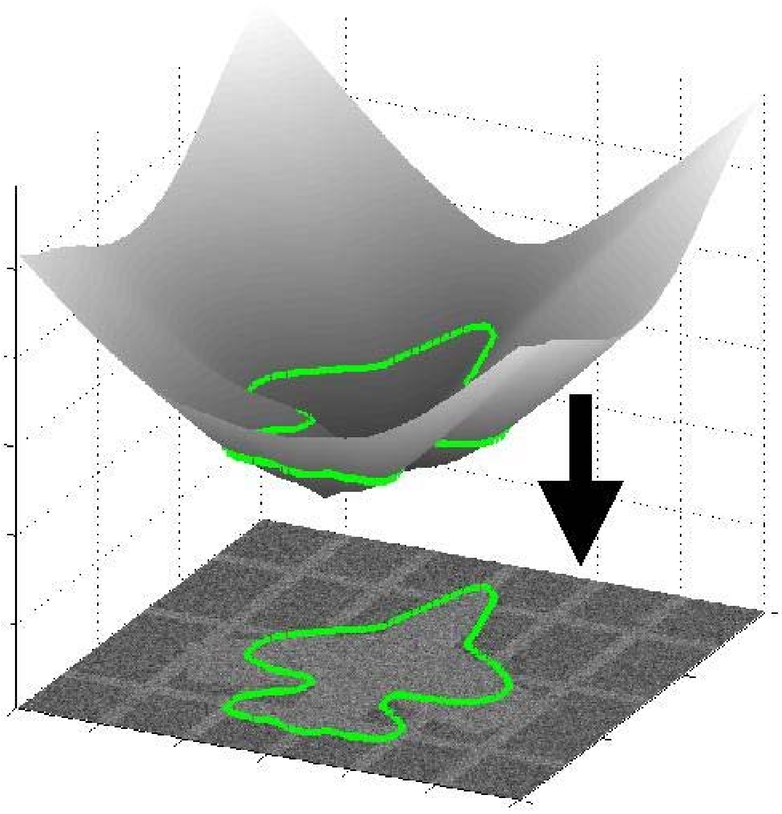
Embedding the curve

- Force level set Ψ to be zero on the curve

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

- Chain rule gives us

$$\begin{aligned} \frac{\mathrm{d}\Psi}{\mathrm{d}t} &= -\frac{\mathrm{d}\vec{C}}{\mathrm{d}t} \cdot \nabla\Psi \\ &= -\mathrm{f}||\nabla\Psi|| \end{aligned}$$



Popular energy functionals

- Geodesic active contours (Caselles et al.):

$$E(\vec{C}) = \oint_{\vec{C}} \frac{ds}{1 + |\nabla I|^2}$$

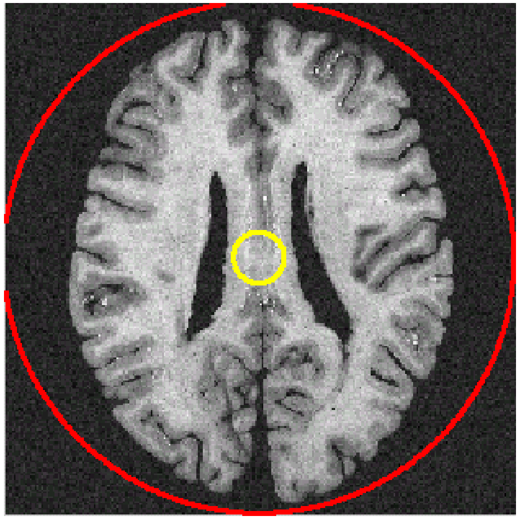
- Separating the means (Yezzi et al.):

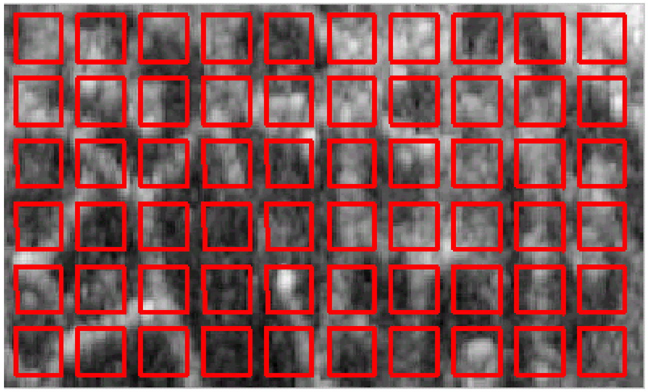
$$E(\vec{C}) = (\mu_{\text{in}} - \mu_{\text{out}})^2$$

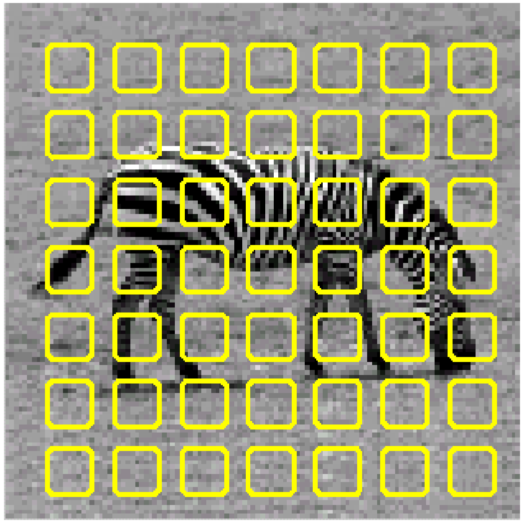
- Piecewise constant intensities (Chan and Vese):

$$E(\vec{C}) = \iint_{R_0} (y - \mu_0)^2 dx + \iint_{R_1} (y - \mu_1)^2 dx + \alpha \oint_{\vec{C}} ds$$

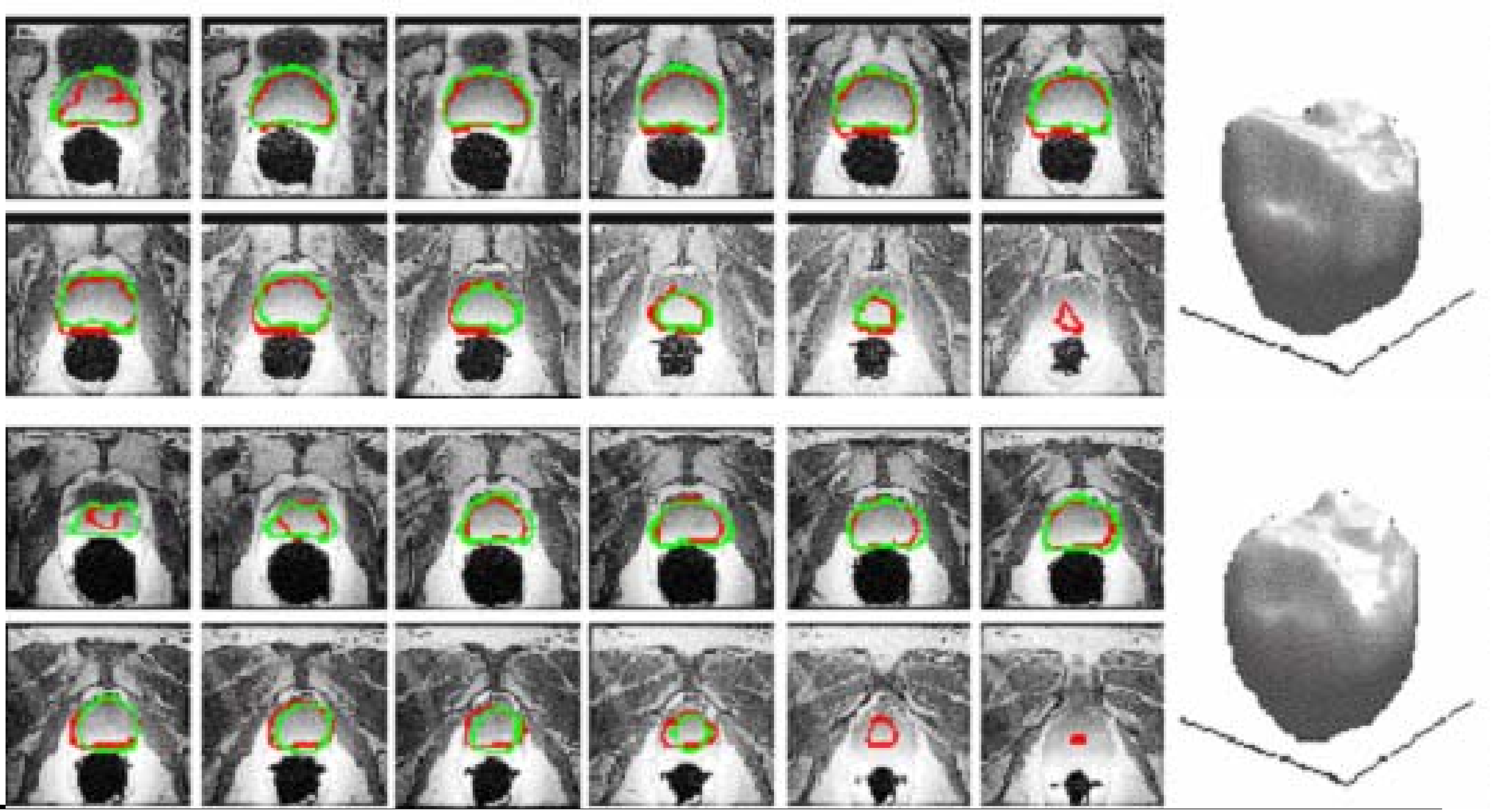
Examples-I







Examples-II



Outline

1. Overview
2. Curve evolution
3. Markov chain Monte Carlo
4. Curve sampling
5. Conditional simulation
6. 2.5D Segmentation

General MAP Model

$$p(x|y;S) \propto p(y|x;S)p(x;S)$$

- For segmentation:
  - x is a curve
  - y is the observed image (can be vector)
  - S is a shape model
  - Data model usually IID given the curve
- We wish to sample from p(x|y;S), but cannot do so directly

Markov Chain Monte Carlo

- Class of sampling methods that iteratively generate candidates based on a previous iterate (forming a Markov chain)
- Instead of sampling from p(x|y;S), sample from a proposal distribution **q** and keep samples according to an acceptance rule **a**
- Examples include Gibbs sampling, Metropolis-Hastings

Metropolis-Hastings

- Metropolis-Hastings algorithm:
  1. Start with x^0
  2. At time t, generate candidate φ^t (given x^{t-1})
  3. Calculate Hastings ratio:

$$r^t = \frac{\mathsf{p}(\phi^t)}{\mathsf{p}(x^{t-1})} \cdot \frac{\mathsf{q}(x^{t-1}|\phi^t)}{\mathsf{q}(\phi^t|x^{t-1})}$$

  4. Set $x^t = \varphi^t$ with probability $\min(1, r^t)$, otherwise $x^t = x^{t-1}$
  5. Go back to 2

Asymptotic Convergence

- We want to form a Markov chain such that its stationary distribution is p(x):

$$\mathsf{p}(x) = \int \mathsf{p}(\phi)\mathsf{T}(x|\phi)\mathsf{d}\phi$$

- To guarantee asymptotic convergence, sufficient conditions are:

1) Ergodicity
2) Detailed balance

$$\mathsf{p}(x^{t-1})\mathsf{q}(\phi^t|x^{t-1})\mathsf{a}(\phi^t|x^{t-1}) = \mathsf{p}(\phi^t)\mathsf{q}(x^{t-1}|\phi^t)\mathsf{a}(x^{t-1}|\phi^t)$$

Outline

1. Overview
2. Curve evolution
3. Markov chain Monte Carlo
4. Curve sampling
5. Conditional simulation
6. 2.5D Segmentation

MCMC Curve Sampling

- Generate perturbation on the curve:

$$\vec{C}'(s) = \vec{C}(s) + f(s)\vec{\mathcal{N}}(s)\mathrm{d}t$$

- Sample by adding smooth random fields:

$$f \sim \mathsf{N}(0, \Sigma)$$

- Σ controls the degree of smoothness in field
- Note for portions where *f* is negative, shocks may develop (so called prairie fire model)
- Implement using white noise and circular convolution

Smoothness issues

- While detailed balance assures asymptotic convergence, may need to wait a very long time
- In this case, smooth curves have non-zero probability under **q**, but are very unlikely to occur
- Can view accumulation of perturbations as
  $$F(s) = \sum_i f_i(s) = h \circledast \sum_i n_i$$
  (h is the smoothing kernel, n_i is white noise)
- Solution: make **q** more likely to move towards high-probability regions of **p**

Adding mean force

- We can add deterministic elements to f (i.e., a mean to **q**):

$$f \sim \mathrm{N}(-\kappa + \gamma, \Sigma)$$

$$f(s) = \beta r(s) - \alpha\kappa(s) + \gamma$$

- The average behavior should then be to move towards higher-probability areas of **p**
- In the limit, setting f to be the gradient flow of the energy functional results in always accepting the perturbation

Coverage/Detailed balance

- It is easy to show we can go from any curve C_1 to any other curve C_2 (shrink to a point)
- For detailed balance, we need to compute probability of generating *C´* from *C* (and vice versa)

$$\begin{aligned}\vec{C}'(s) &= \vec{C}(s) + f(s)\vec{\mathcal{N}}(s)\mathrm{d}t \\ \vec{C}(s) &= \vec{C}'(s) + f'(s)\vec{\mathcal{N}}'(s)\mathrm{d}t\end{aligned}$$

- Probability of going from *C* to *C´* is the probability of generating *f* (which is Gaussian) and the reverse is the probability of *f ´* (also Gaussian)

Approximations to q

- Relationship between *f* and *f´* complicated due to the fact that the normal function changes
- *f´* does not always exists (given an *f*). Unknown what conditions on f are necessary to guarantee existence.
- Various levels of exactness
  - Assume $\vec{\mathcal{N}} = \vec{\mathcal{N}}'$ (then *f´* = -*f*)
  - Locally-linear approximation
    $$f'(s) = -f(s)/ < \vec{\mathcal{N}}(s), \vec{\mathcal{N}}'(s) >$$
  - Trace along $\mathcal{N}'$ (technical issues)
- Unknown how approximations affect convergence

Synthetic noisy image

- Piecewise-constant observation model:

$$y(x) = \mu(x) + n(x)$$

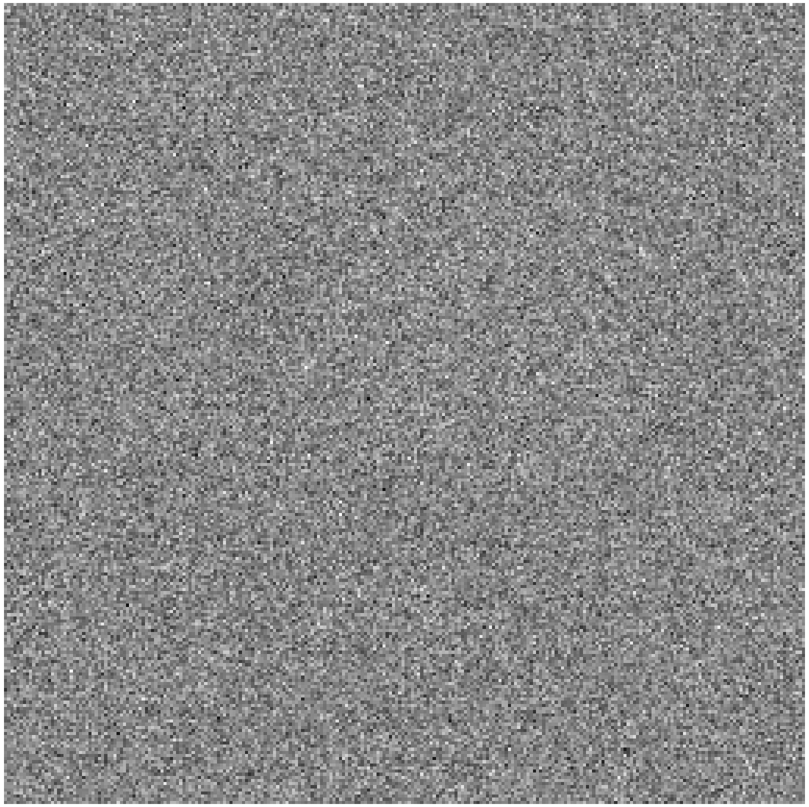
- Chan-Vese energy functional:

$$E(\vec{C}) = \iint_{R_0} (y-\mu_0)^2 dx + \iint_{R_1} (y-\mu_1)^2 dx + \alpha \oint_{\vec{C}} ds$$

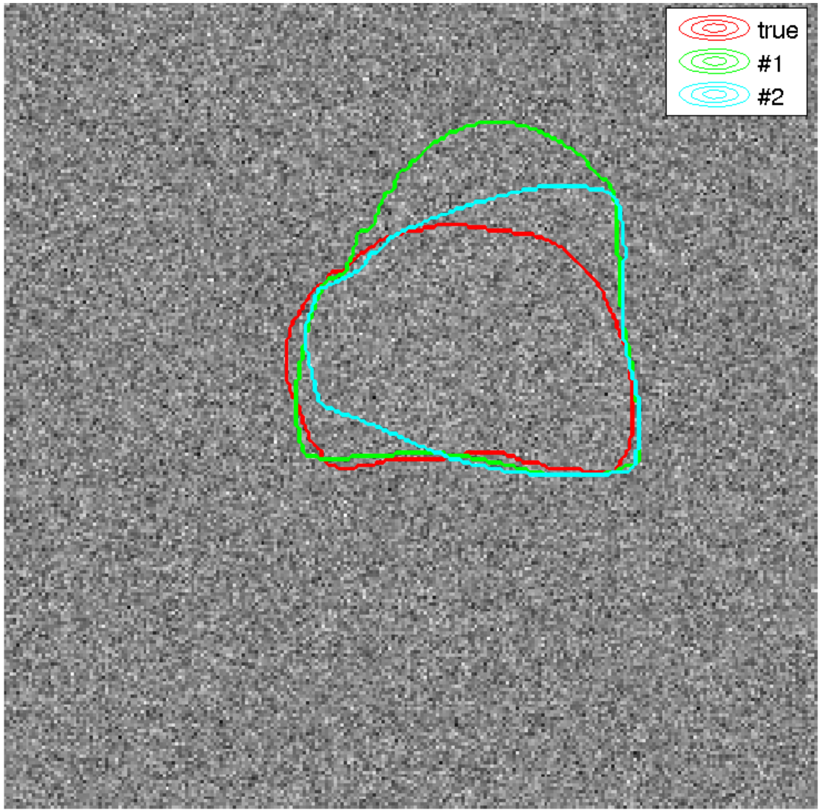
- Probability distribution ($T=2\sigma^2$):

$$\mathrm{p}(\vec{C}) = \frac{1}{Z}\exp(-\mathrm{E}(\vec{C})/T)$$

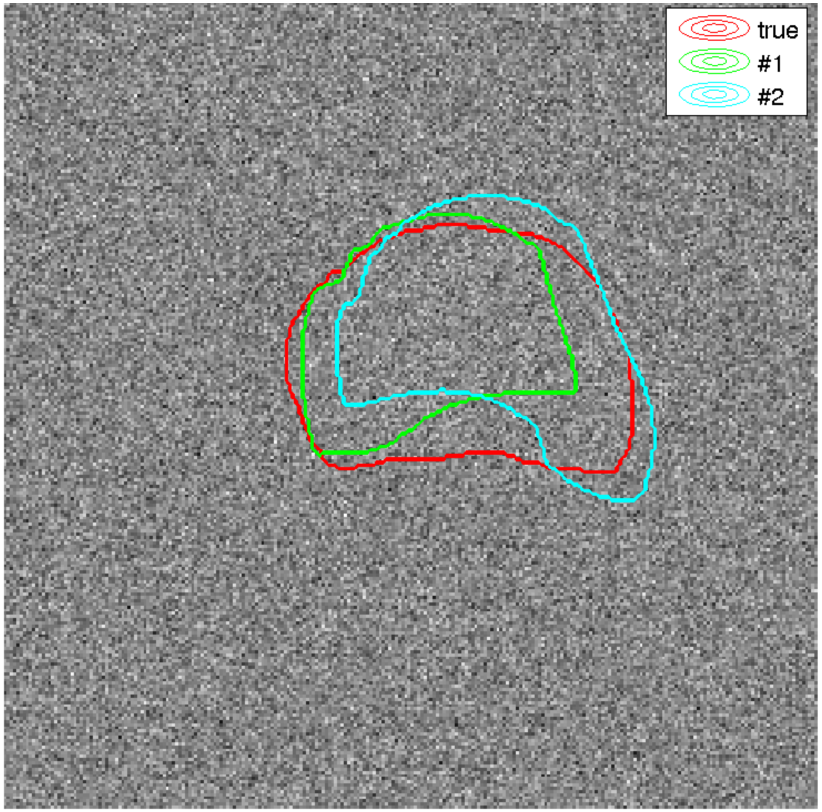
Prostate in a Haystack



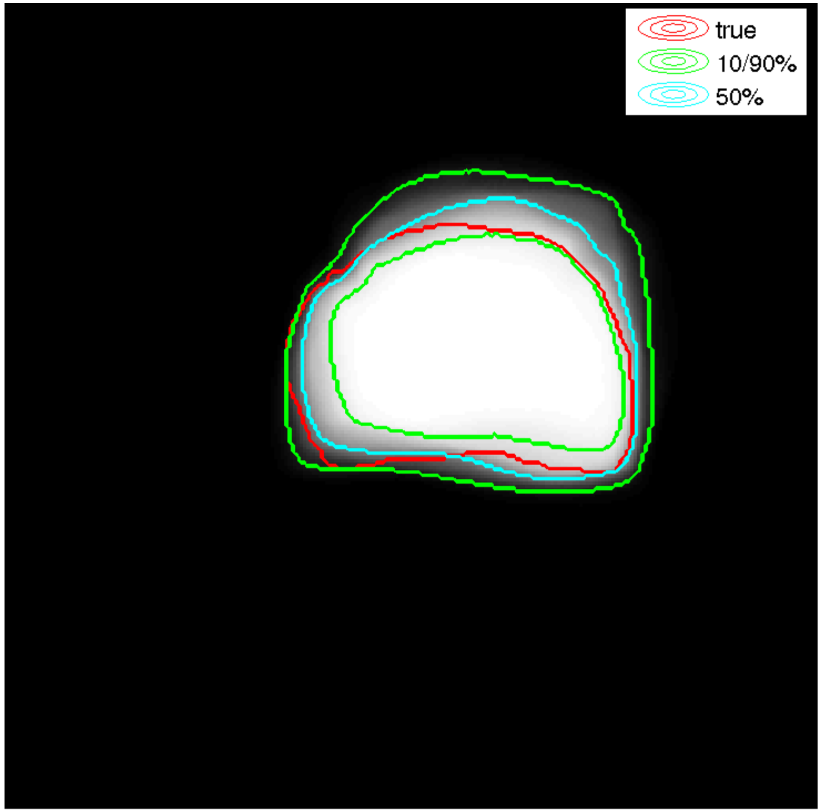
Most likely samples



Least likely samples



Confidence intervals

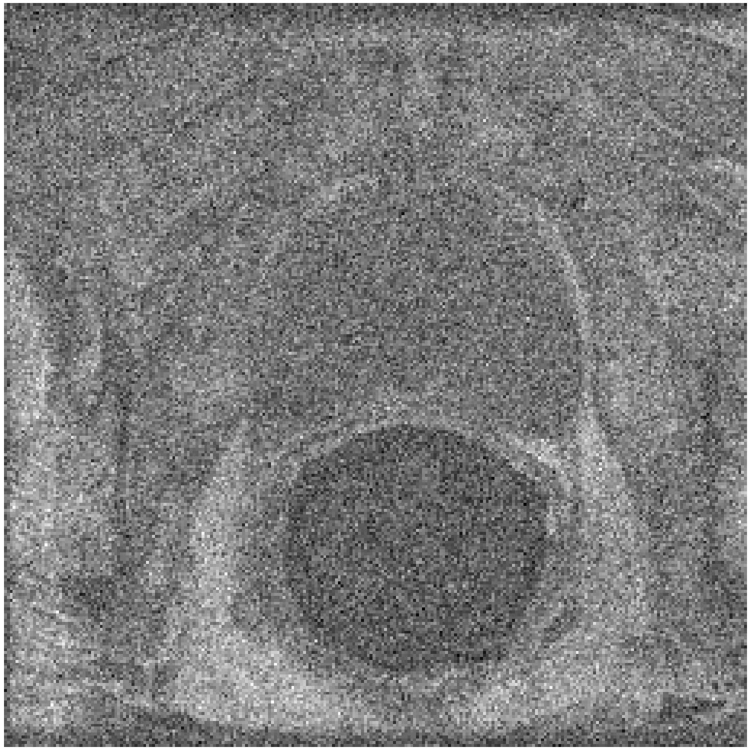


When "best" is not best

- In this example, the most likely samples under the model are not the most accurate according to the underlying truth
- 10%/90% "confidence" bands do a good job of enclosing the true answer
- Histogram image tells us more uncertainty in upper-right corner
- "Median" curve is quite close to the true curve
- Optimization would result in subpar results

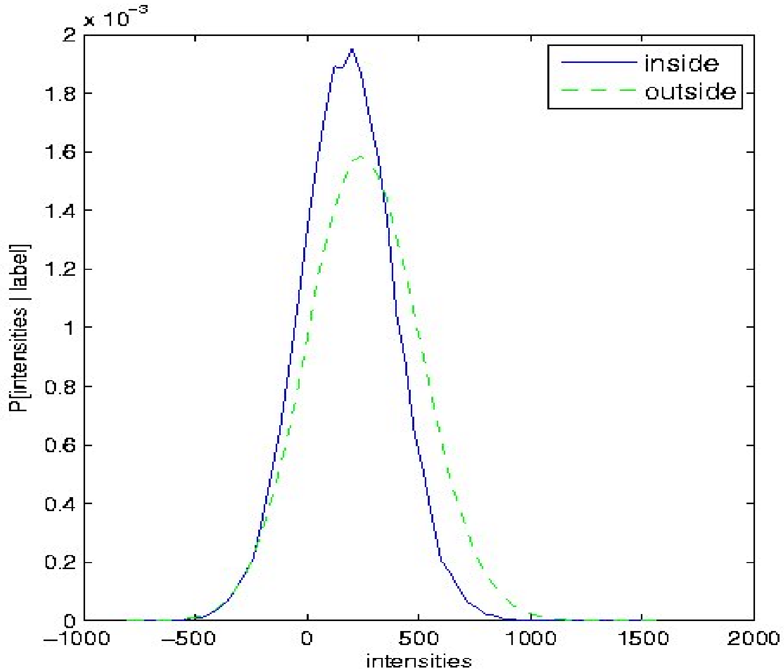
Bias-corrected prostate

- “Expert” segmentation, add noise (simulate body coil image)



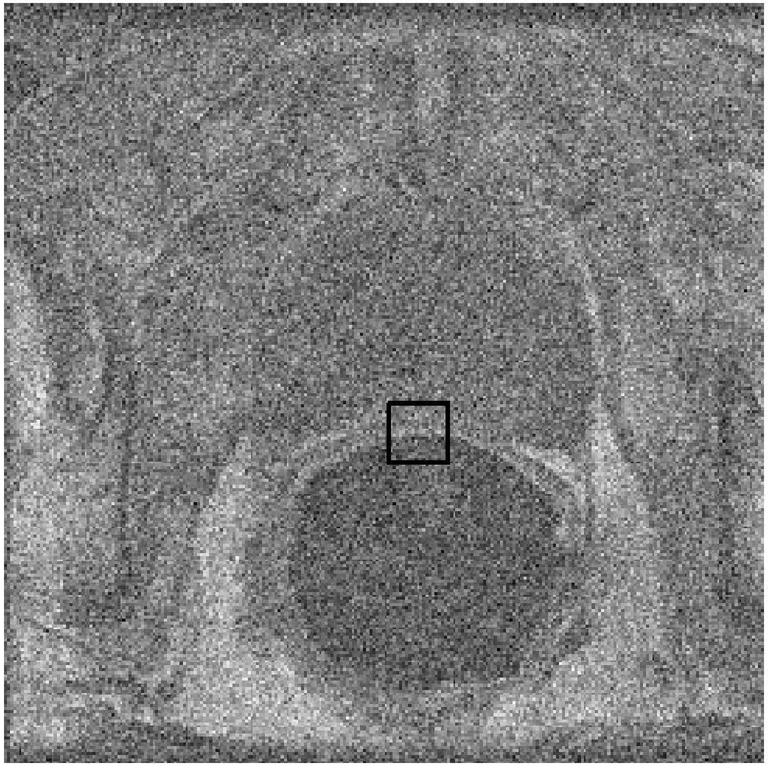
Learn probability densities

- Use histograms
- Learn pdf inside p(y|1) and pdf outside p(y|0) and assume iid given curve:

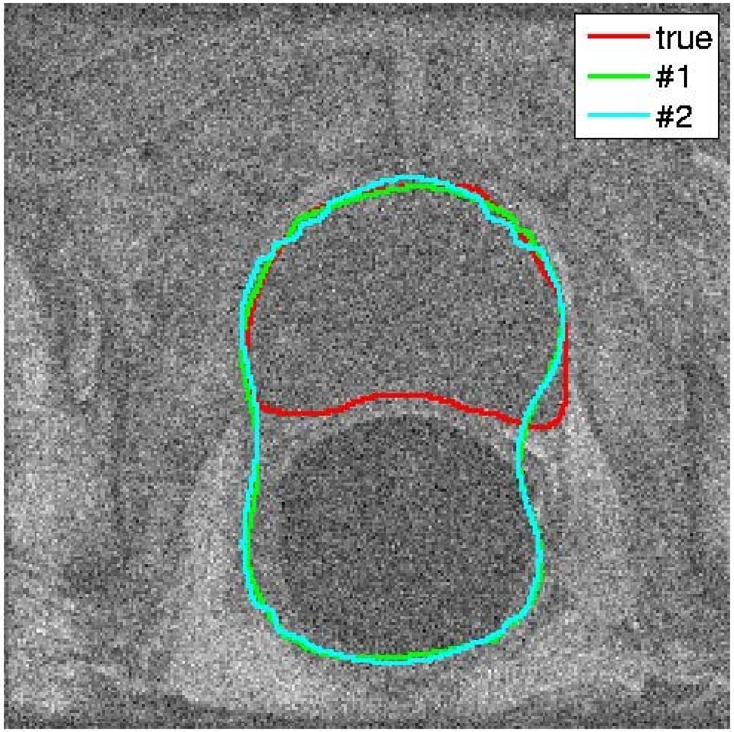


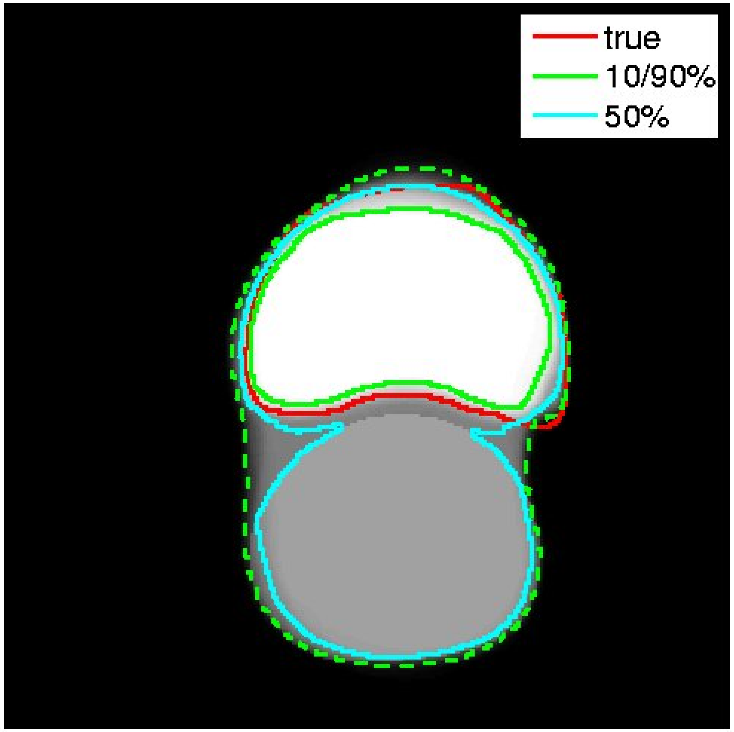
$$E(\vec{C}) = -\int_{\Omega} \log \mathrm{p}(y(x)|\mathcal{H}(-\Psi(x)))\mathrm{d}x + \alpha \oint_{\vec{C}} ds$$

Initialization

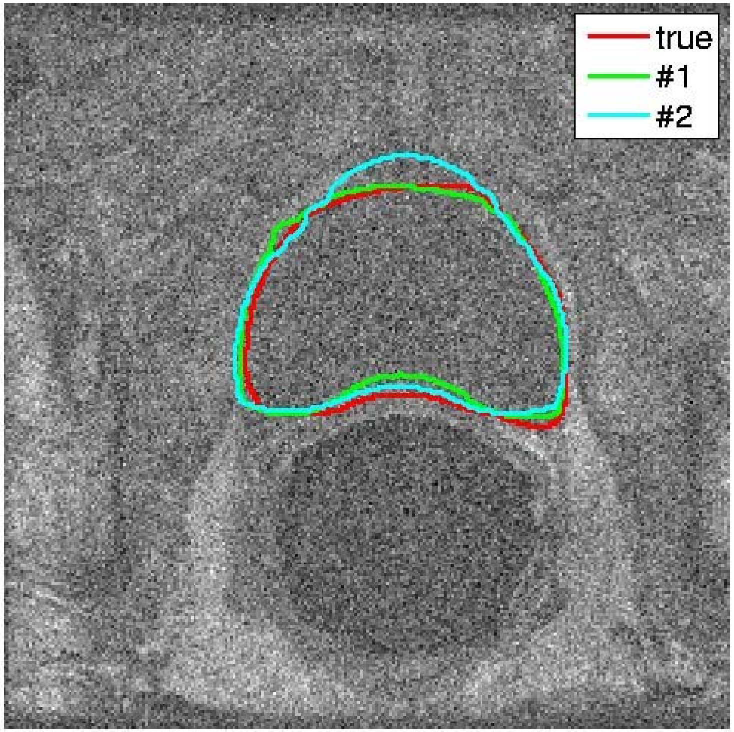


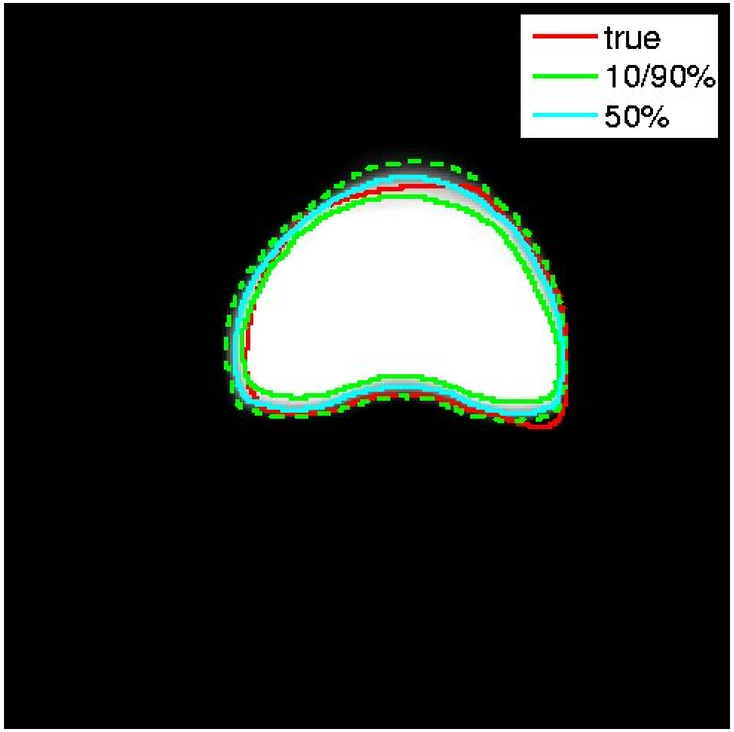
Results





Prostate-only cluster





Multimodality and convergence

- Natural multimodal distribution
- When starting near one mode, need a lot of time to traverse valley between modes
- Clustering helps with presenting results
- Interesting work to be done in learning dimensionality of manifold and local approximations

Outline

1. Overview
2. Curve evolution
3. Markov chain Monte Carlo
4. Curve sampling
5. Conditional simulation
6. 2.5D Segmentation

User Information

- In many problems, the model admits many reasonable solutions
- Currently user input largely limited to initialization
- We can use user information to reduce the number of reasonable solutions
  - Regions of inclusion or exclusion
  - Partial segmentations
- Can help with both convergence speed and accuracy
- Interactive segmentation

Conditional simulation

- With conditional simulation, we are given the values on a subset of the variables
- We then wish to generate sample paths that fill in the remainder of the variables (e.g., simulating Brownian motion)

Simulating curves

- Say we are given C_s, a subset of C (with some uncertainty associated with it)
- We wish to sample the unknown part of the curve C_u
- One way to view is as sampling from:

$$\mathsf{p}(y|\vec{C})\mathsf{p}(\vec{C}) = \mathsf{p}(y|\vec{C})\mathsf{p}(\vec{C}_u|\vec{C}_s)\mathsf{p}(\vec{C}_s)$$

- Difficulty is being able to evaluate the middle term as theoretically need to integrate p(C)

Simplifying Cases

For special cases, evaluation of $\mathsf{p}(\vec{C}_u|\vec{C}_s)$ is tractable:

1. When *C* is low-dimensional (can do analytical integration or Monte-Carlo integration)
2. When C_s is assumed to be exact
3. When p(*C*) has special form (e.g., independent, Markov)
4. When willing to approximate

- When implementing conditional simulation, modify q to be compatible with new conditional probability

Gravity inversion

- Supplement to standard seismic data to segment bottom salt using an array of surface gravimeters (~10^{-15} N accuracy)
- Subtract base effects (geoid, centrifugal force, etc.) to leave salt effects:

$$\vec{g}(i) = G \int_{\Omega} \frac{\rho(x;\vec{C})\widehat{r}_i(x)}{||\vec{r}_i(x)||^2} \mathrm{d}x$$

- Assume constant density inside and outside:

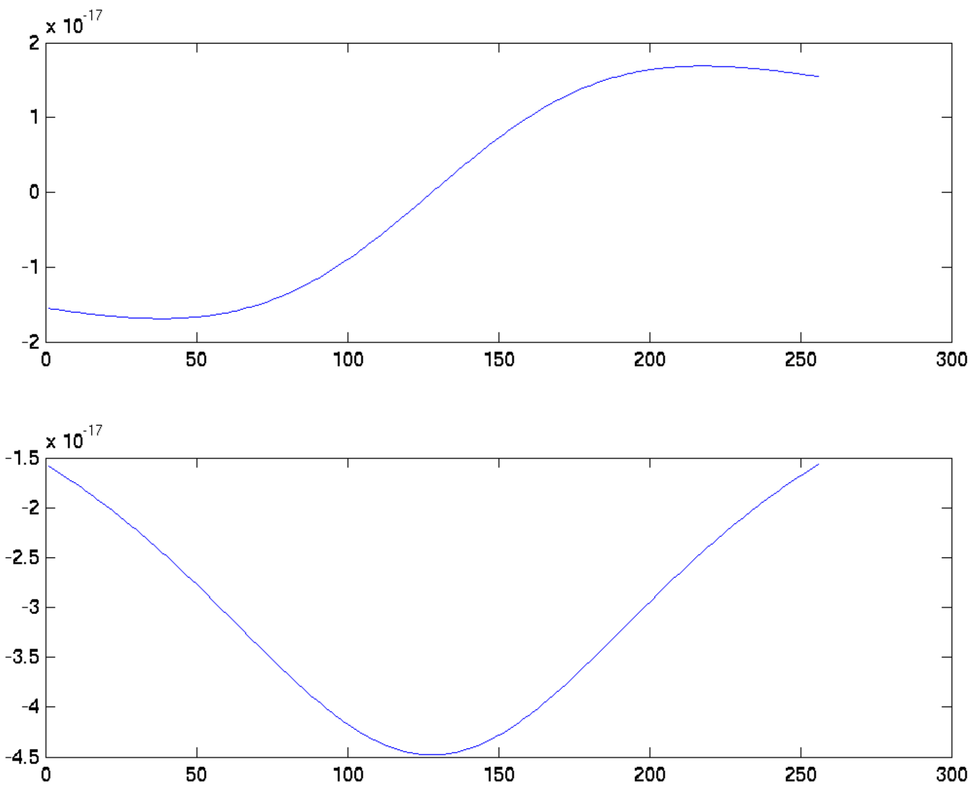
$$\rho(x;\vec{C}) = \triangle\rho\mathcal{H}(-\Psi(x))$$

- Model energy as L2 estimation error (probability as Boltzmann distribution):

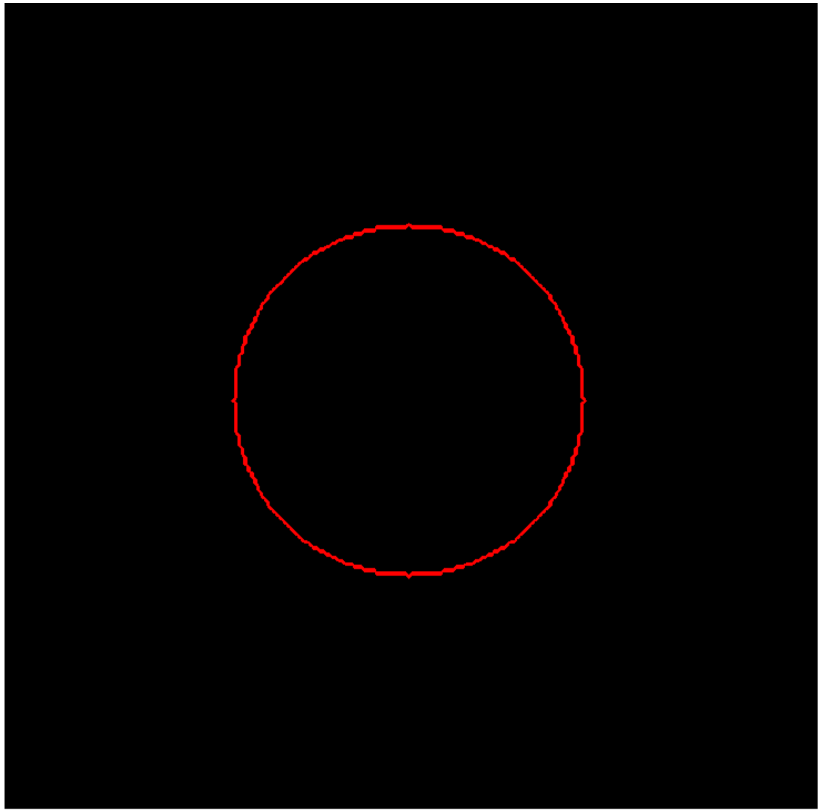
$$\mathrm{E}(\vec{C}) = \sum_{i=1}^{N_{\mathrm{array}}} ||\vec{g}_{\mathrm{obs}}(i) - \vec{g}(i;\vec{C})||^2 + \alpha \oint_{\vec{C}} \mathrm{d}s$$

A strange segmentation problem

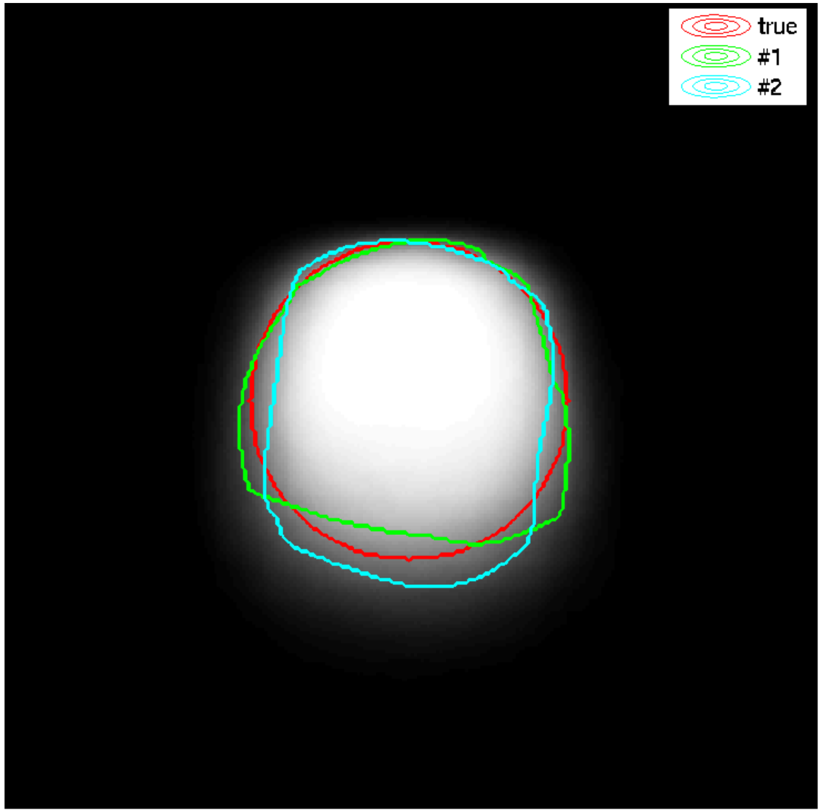
X- and z-components of gravity profile for synthetic salt body



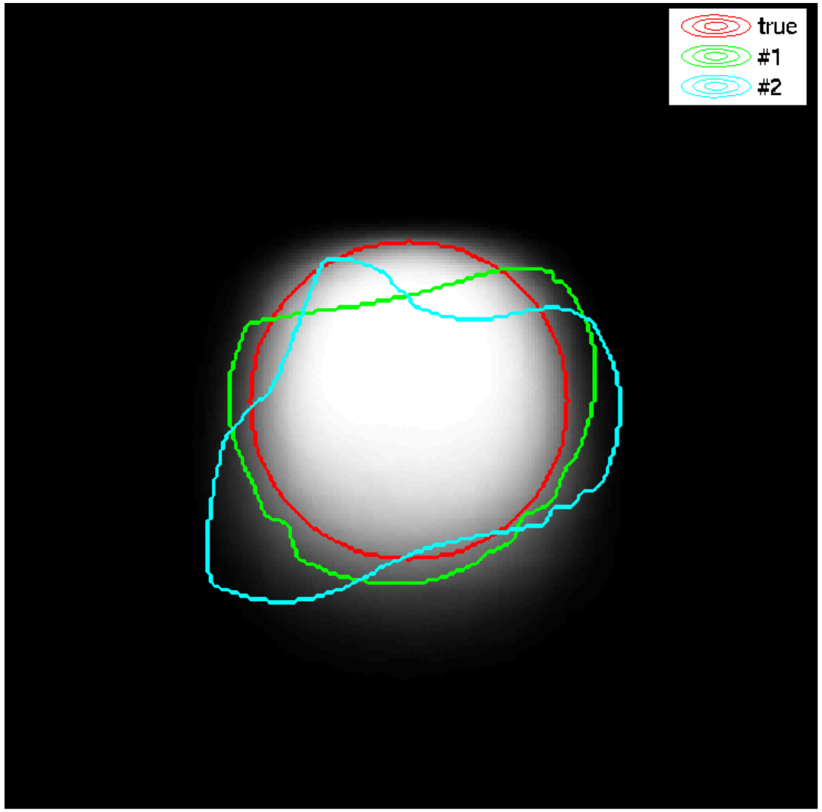
Circle salt



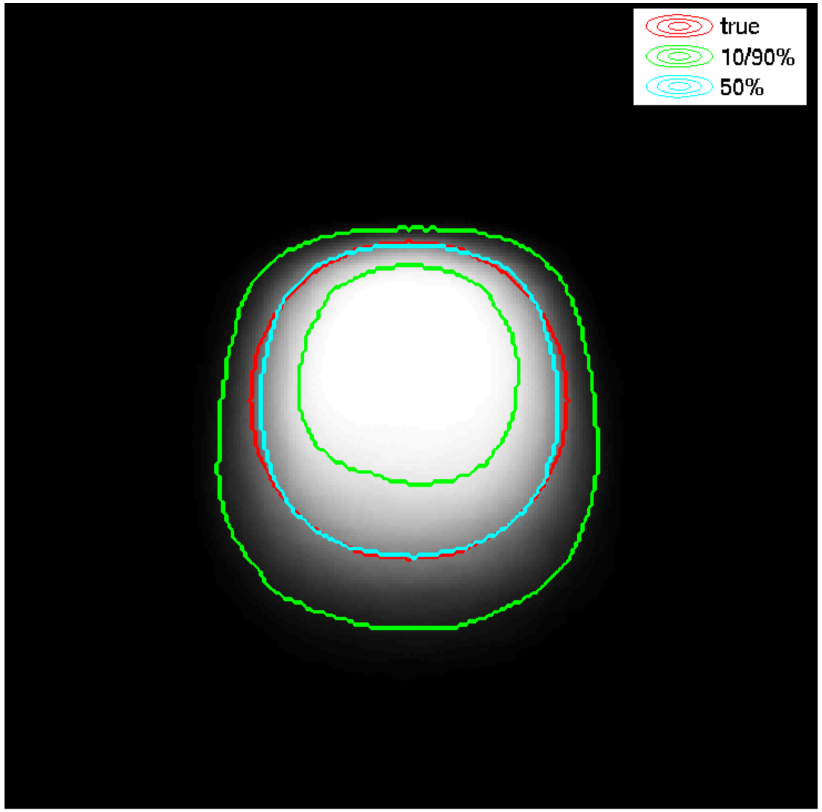
Most likely samples



Least likely samples



Confidence intervals

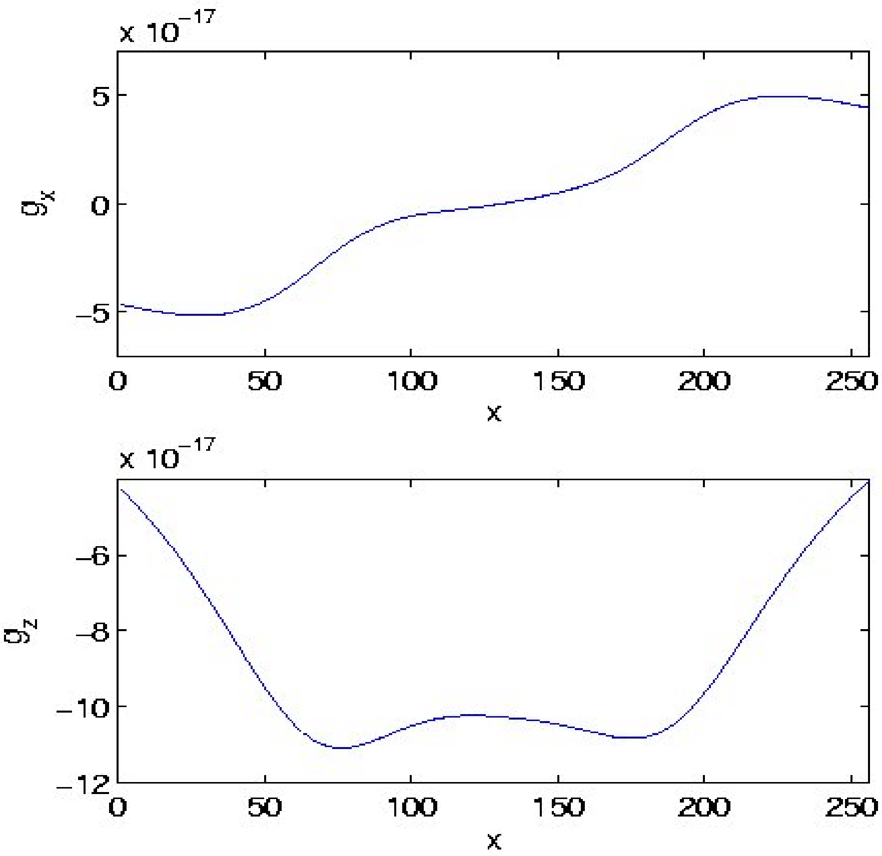


Notable features

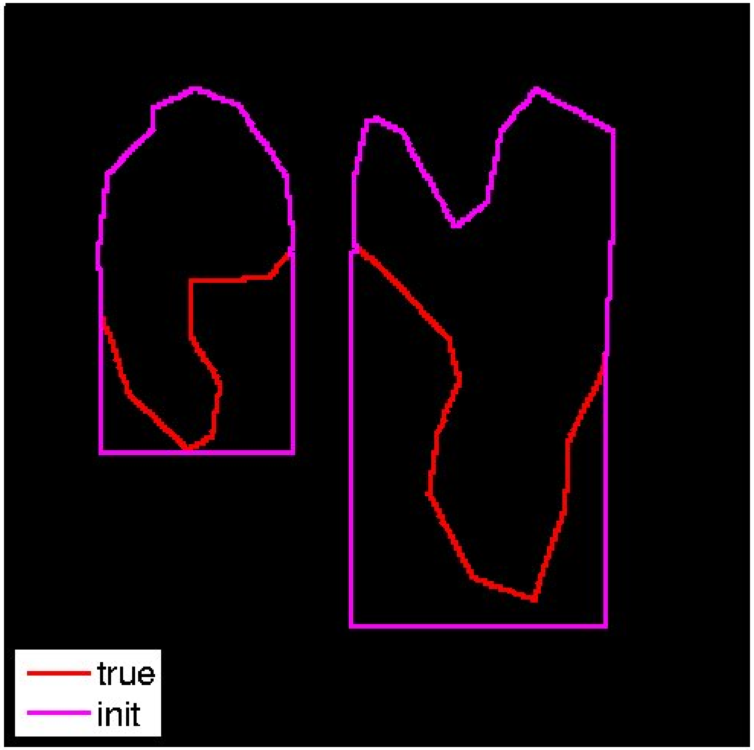
- Measurement points << image pixels, but we can do a reasonable job
- Much higher uncertainty at the bottom than the top (weaker measurements)
- Less uncertainty in middle than on sides
- Median of histogram not necessarily related to median of distribution

More complex example

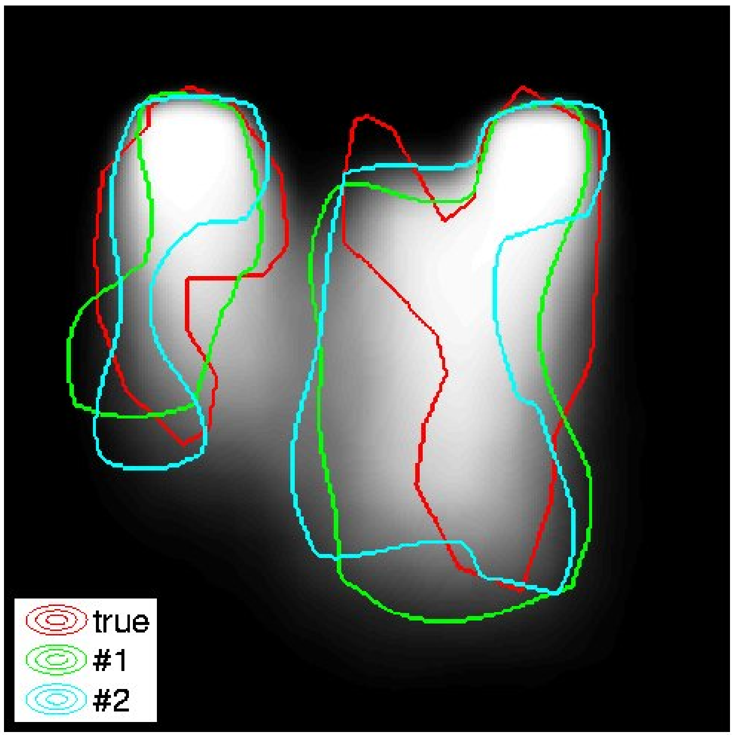
- Same x- and z-components of gravity
- Synthetic image with more complex geometry



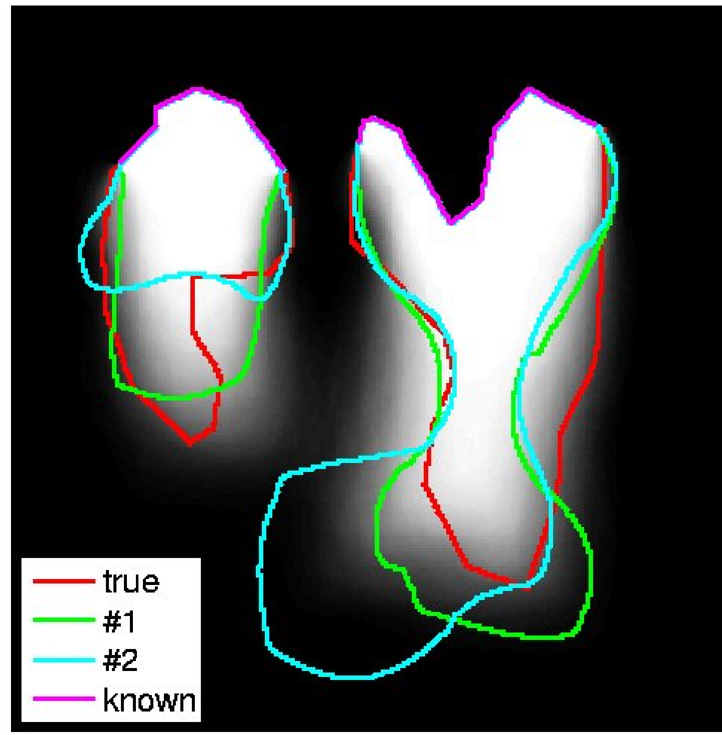
Initialization



Most likely samples

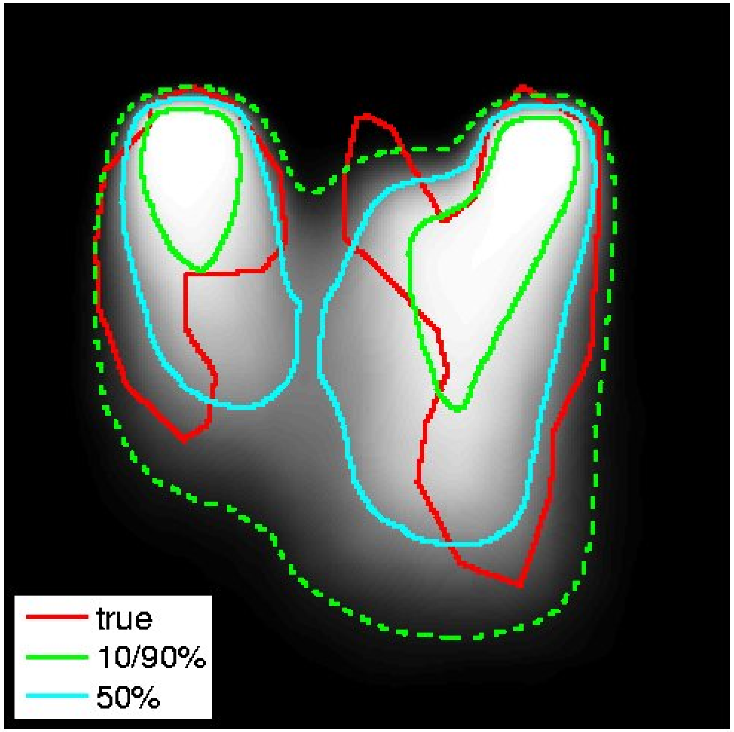


Regular

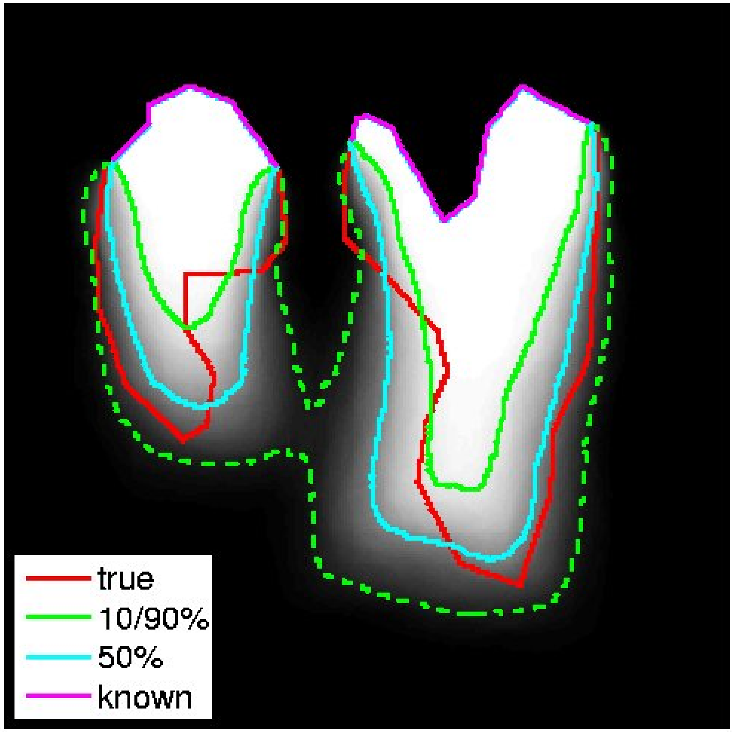


Conditionally simulated

Marginal confidence bounds

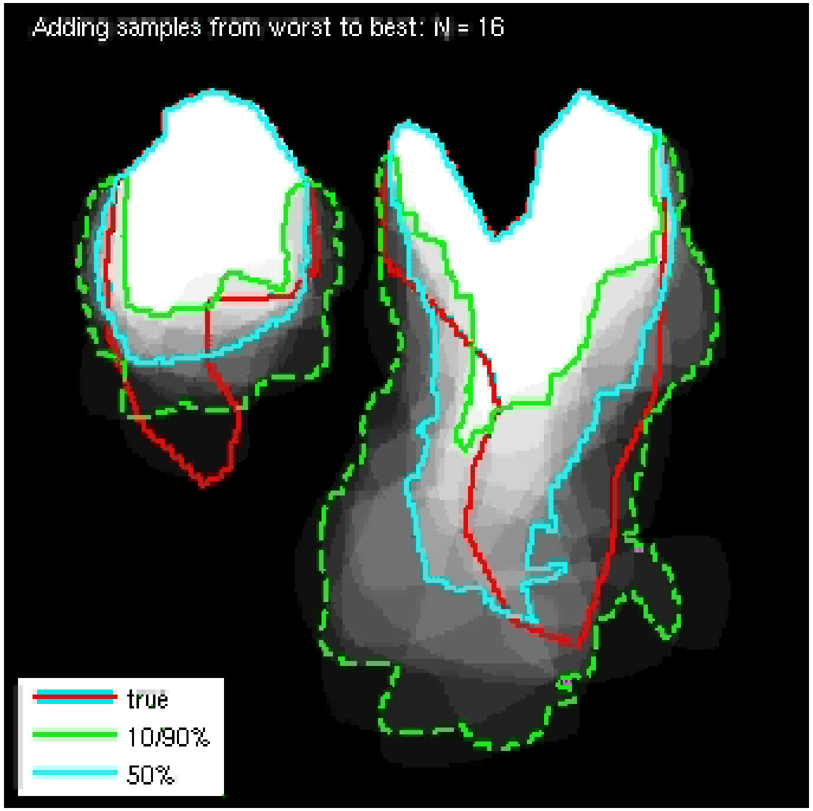


Regular



Conditionally simulated

Aggregating samples



Outline

1. Overview
2. Curve evolution
3. Markov chain Monte Carlo
4. Curve sampling
5. Conditional simulation
6. 2.5D Segmentation

Chan-Vese in 3D

- Energy functional with surface area regularization:

$$E(\vec{C}) = \frac{1}{2\sigma_1^2}\iiint_{R_1}(y-\mu_1)^2 d\mathbf{x} + \frac{1}{2\sigma_2^2}\iiint_{R_2}(y-\mu_2)^2 d\mathbf{x}$$
$$+\alpha \oiint_{\vec{C}} dA$$

- With a slice-based model, we can write the regularization term as:

$$\oiint_{\vec{C}} dA = \sum_{i=1}^{n-1} \iint_{\vec{c}_i \oplus \vec{c}_{i+1}} dA$$

where $\vec{c}_i \oplus \vec{c}_{i+1}$ is the surface between c_i and c_{i+1}

Zero-order hold approximation

- Approximate volume as piecewise-constant "cylinders":

$$\vec{C}(s,z) = \vec{c}_i(s), \;\; \forall \; |z - i\triangle z| < \frac{\triangle z}{2}$$

- Then we see that the surface areas are:

$$\iint_{\vec{c}_i \oplus \vec{c}_{i+1}} dA = \frac{\triangle z}{2} \oint_{\vec{c}_i} ds + \frac{\triangle z}{2} \oint_{\vec{c}_{i+1}} ds + \iint_{R^{\mathrm{diff}}_{i,i+1}} d\mathbf{x}$$

- We see terms related to the curve length and the difference between neighboring slices
- Upper bound to correct surface area

Overall regularization term

- Adding everything together results in:

$$\underbrace{\oiint_{\vec{C}} dA = \triangle z \sum_{i=1}^{n} \oint_{\vec{c}_i} ds}_{\text{self potentials}} + \underbrace{\sum_{i=1}^{n-1} \int\int_{R_{i,i+1}^{\text{diff}}} dx}_{\text{edge potentials}}$$

2.5D Approach

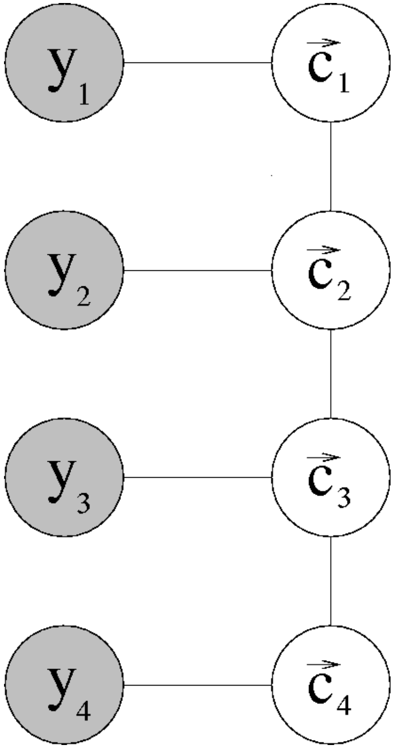
- In 3D world, natural (or built-in) partition of volumes into slices
- Assume Markov relationship among slices
- Then have local potentials (e.g., PCA) and edge potentials (coupling between slices)
- Naturally lends itself to local Metropolis-Hastings approach (iterating over the slices)

2.5D Model

- We can model this as a simple chain structure with pairwise interactions
- This admits the following factorization:

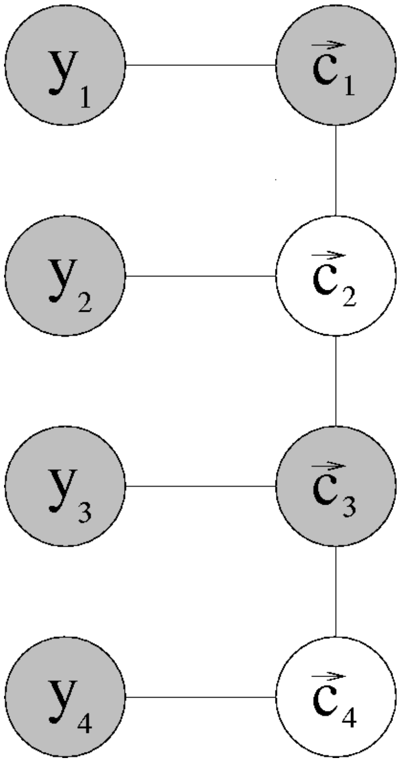
$$\mathrm{p}(Y|\vec{C}) = \prod_{i=1}^{n} \mathrm{p}(y_i|\vec{c}_i)$$

$$\mathrm{p}(\vec{C}) = \prod_{i=1}^{n} \psi_i(\vec{c}_i) \prod_{i=1}^{n-1} \psi_{i,i+1}(\vec{c}_i, \vec{c}_{i+1})$$



Partial segmentations

- Assume that we are given segmentations of every other slice
- We now want to sample surfaces conditioned on the fact that certain slices are fixed
- Markovianity tells us that c_2 and c_4 are independent conditioned on c_3

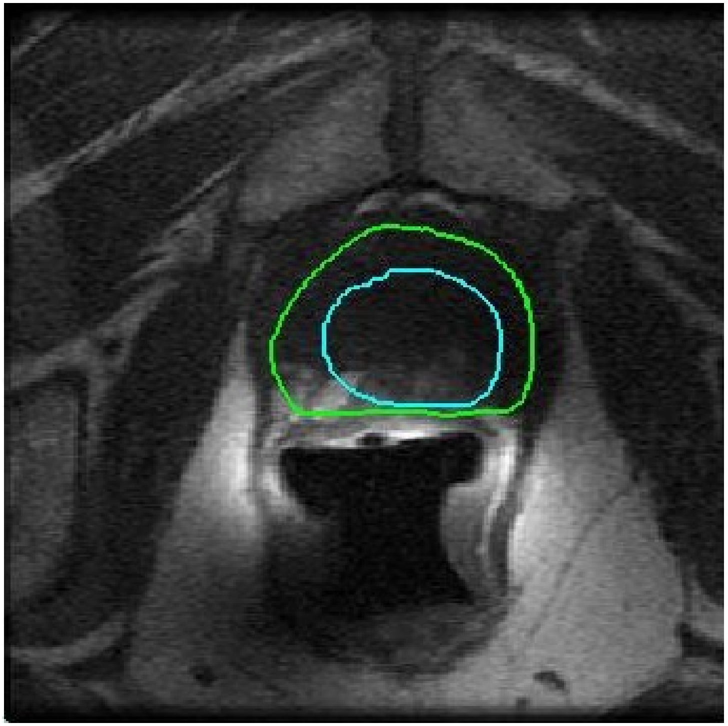


Log probability for c_2

- We can then construct the probability for c_2 conditioned on its neighbors using the potential functions defined previously:

$$\begin{aligned}\ell(\vec{c}_2|y_2,\vec{c}_1,\vec{c}_3) &= \ell(y_2|\vec{c}_2) + \\ &\alpha[\triangle z \oint_{\vec{c}_2} ds + \mathrm{d}_{\mathrm{SAD}}(\vec{c}_2,\vec{c}_1) + \mathrm{d}_{\mathrm{SAD}}(\vec{c}_2,\vec{c}_3)]\end{aligned}$$

Results



Neighbor segmentations

Our result (cyan) with expert (green)

Larger spacing

- Apply local Metropolis-Hastings algorithm where we sample on a slice-by-slice basis
- Theory shows that asymptotic convergence is unchanged
- Unfortunately larger spacing similar to less regularization
- Currently have issues with poor data models that need to be resolved

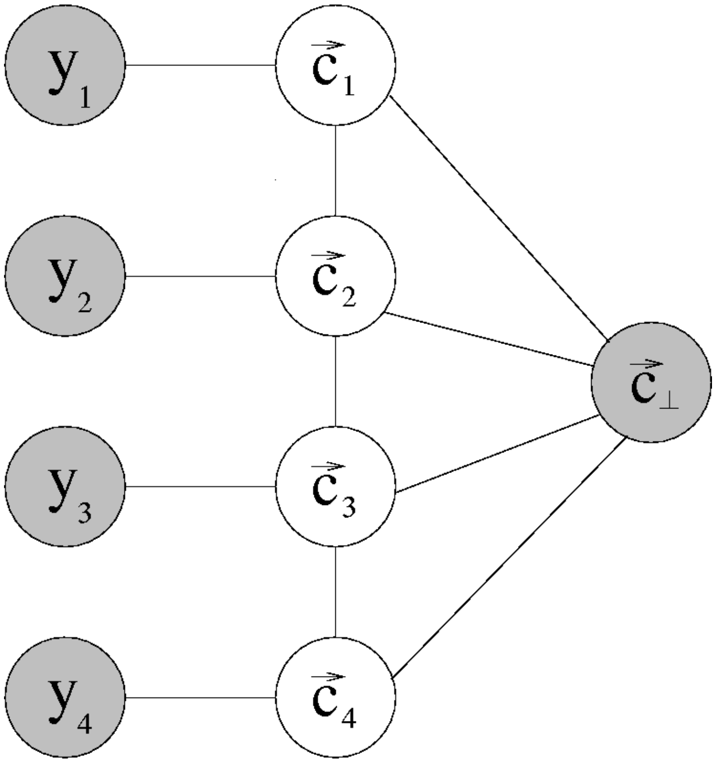
Full 3D

- In medical imaging, have multiple slice orientations (axial, sagittal, coronal)
- In seismic, vertical and horizontal shape structure expected
- With a 2.5D approach, this introduces complexity to the graph structure

Incorporating perpendicular slices

- $c_\perp$ is now coupled to all of the horizontal slices
- $c_\perp$ only gives information on a subset of each slice
- Specify edge potentials as, e.g.:

$$\Psi_{i,\perp}(\vec{c}_i, \vec{c}_\perp) = \exp\left(- \int_0^{L_y} (\Gamma(x_0, y, i\triangle z) - \Gamma_\perp(x_0, y, i\triangle z))^2 dy \right)$$



Other extensions

- Additional features can be added to the base model:
  - Uncertainty on the expert segmentations
  - Shape models (semi-local)
  - Exclusion/inclusion regions
  - Topological change (through level sets)

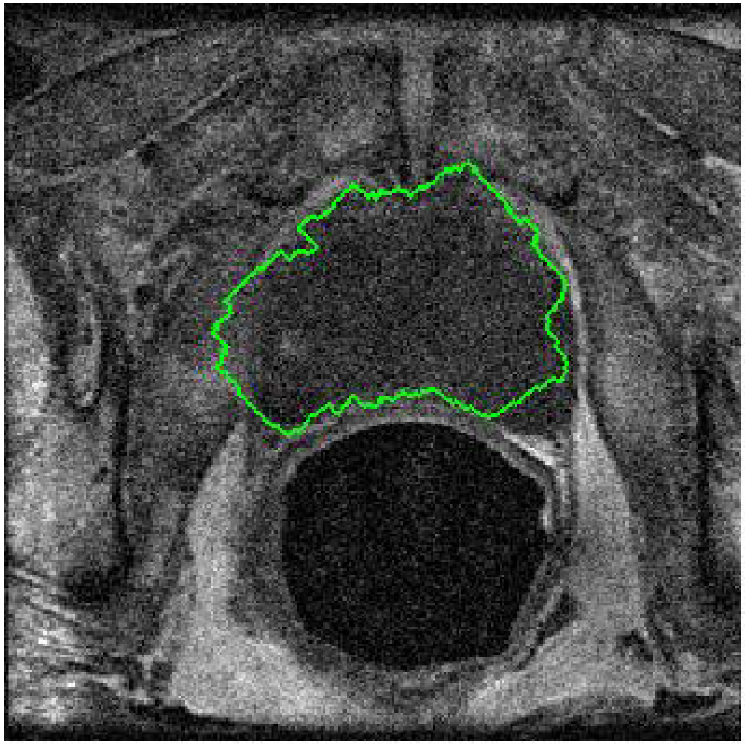
Conclusion

- Computationally feasible algorithm to sample from space of curves
- Approximate detailed balance
- Demonstrated utility for robustness to noise, multimodal distributions, displaying uncertainty
- Can generate arbitrary shapes with relatively complex geometry
- Conditional simulation provides a natural framework to incorporate partial user segmentations on a slice-by-slice level

Further research

- Multiple curves, known/unknown topology
- We would like q to naturally sample from space of smooth curves (different perturbation structure)
- Speed always an issue for MCMC approaches
  - Multiresolution perturbations
  - Parameterized perturbations (efficient basis)
  - Hierarchical models
- Using samples to explore the geometry of shape manifolds

Smooth curve + smooth perturbations ≠ smooth curve



SAD Target

- We define symmetric area difference (SAD) as:

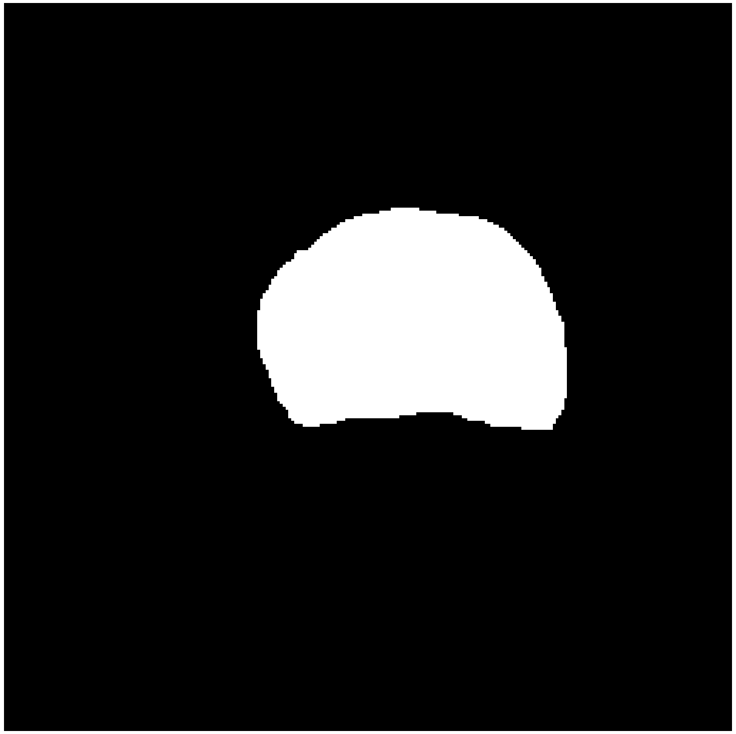
$$\mathrm{d}_{\mathrm{SAD}}(\Psi_1, \Psi_2) = \int_\Omega (\mathcal{H}(-\Psi_1(x)) - \mathcal{H}(-\Psi_2(x)))^2 \mathrm{d}x$$

- Use a Boltzmann distribution:

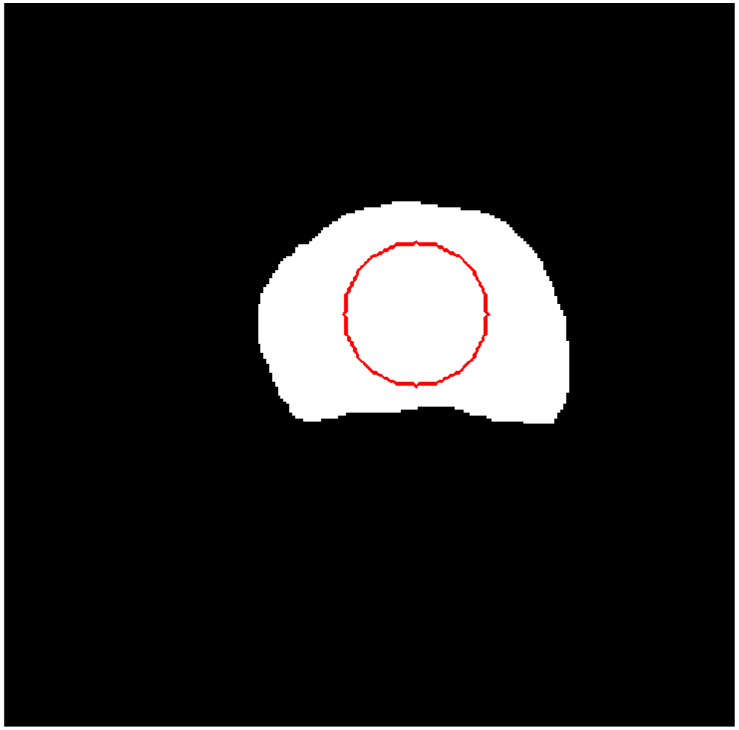
$$\mathrm{p}(\vec{C}|\vec{C}_0) = \frac{1}{Z}\exp(-\mathrm{d}_{\mathrm{SAD}}(\vec{C}, \vec{C}_0)/T)\mathrm{p}(\vec{C})$$

- T is a parameter we can use to control how likely we are to keep less likely samples
- We will keep a sample with T log(2) additional errors with probability ½
- Single mode distribution

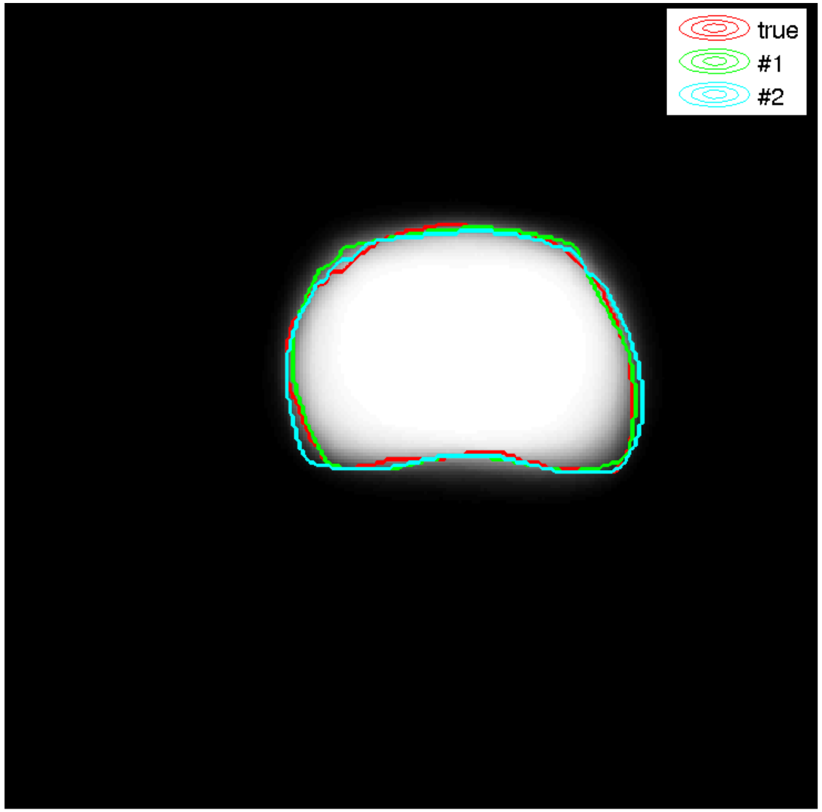
Target Shape



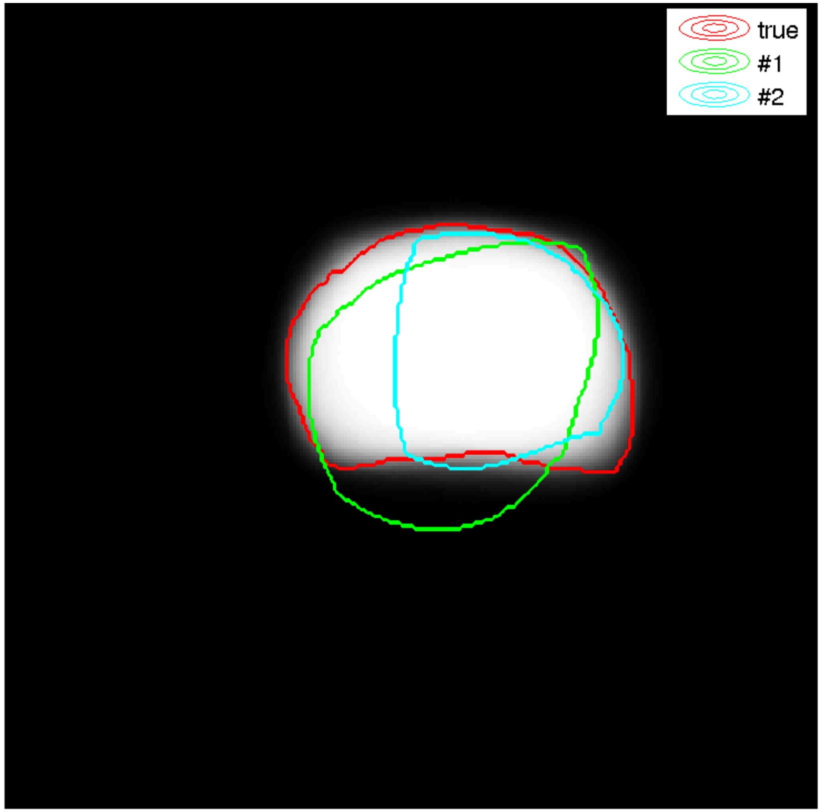
Initialization



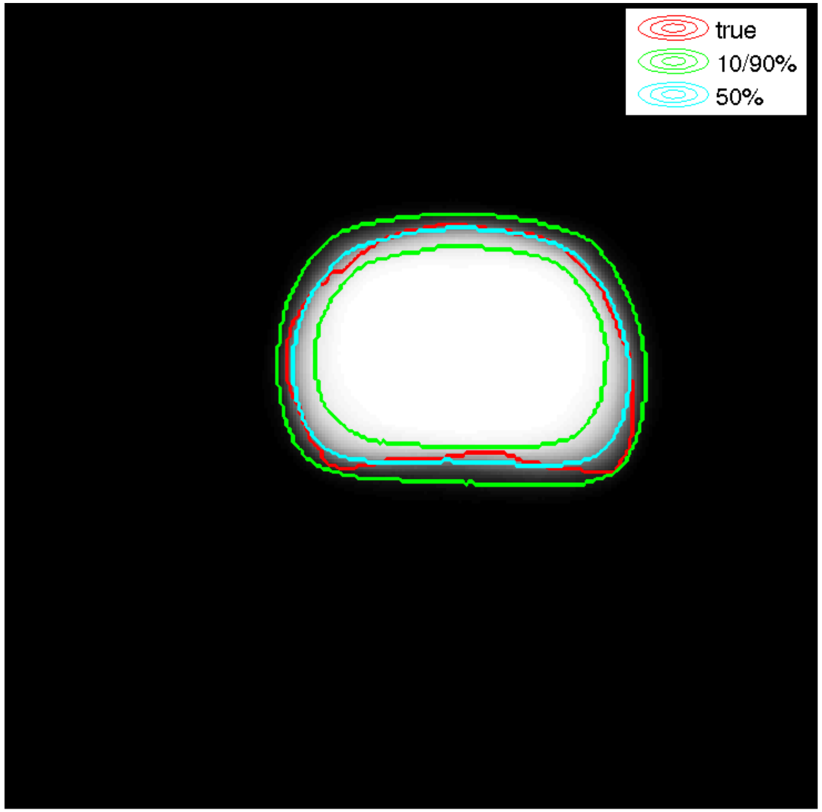
Most likely samples



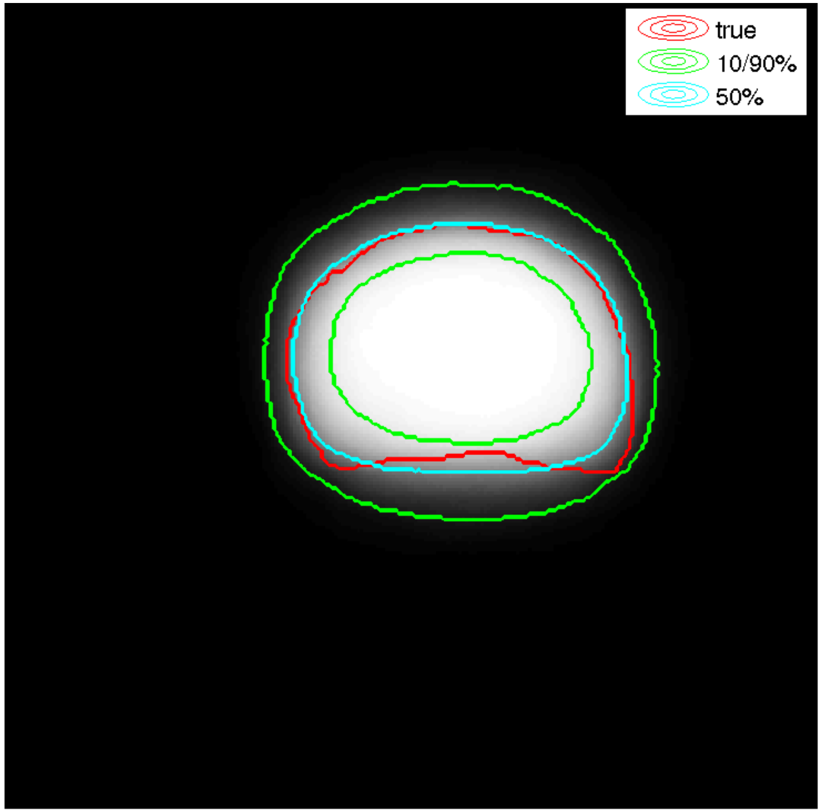
Least likely samples



“Confidence intervals”



Doubling the temperature



Results