Curve Sampling and Geometric Conditional Simulation

Ayres Fan

Ph. D. Defense

Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology

November 13, 2007

Image Segmentation

- Given an image $I : \Omega \to \mathbb{R}$ on a domain $\Omega \subset \mathbb{R}^2$, we wish to partition the image into meaningful regions.
- Traditional curve-based segmentation methods try to optimize an energy functional $\mathrm{E}(\vec{C}; I)$.
- Sampling-based methods can offer a number of benefits over optimization-based techniques:
  - Robustness to local optima.
  - Characterization of multi-modal distributions.
  - Uncertainty measures using confidence bounds and principal modes of variation.
  - Conditional simulation to create semi-automatic segmentation algorithms.

Sampling Problem

- We view the energy functional of optimization-based curve evolution approaches as the negative log likelihood of a posterior probability distribution π:

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

- The results that we present here:
  - Draw samples from $\pi(\vec{C} \,|\, I)$.
  - Show how to visualize many samples from a high-dimensional space.
  - Extend the approach to do conditional simulation.
  - Create a hybrid 2D/3D Markov model for volume segmentation.

Outline of the talk

1. Motivation and problem statement.
2. **Curve evolution and MCMC methods.**
3. MCMC curve sampling.
4. 2D curve sampling results.
5. Conditional simulation.
6. Hybrid 2D/3D models.
7. Conclusions and future work.

Curve Evolution

- Given an image I defined on an image domain $\Omega \subset \mathbb{R}^2$, curve evolution methods attempt to find a curve $\vec{C} : [0,1] \rightarrow \Omega$ that minimizes an energy functional $\mathrm{E}(\vec{C})$ using gradient descent.
- If the energy functional is geometric (*i.e.*, only depends on the geometry of $\vec{C}$, not its parameterization), this results in a geometric partial differential equation (PDE):

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

This flow is expressed in terms of a force function f times the normal function of the curve $\vec{\mathcal{N}}_{\vec{C}}$.

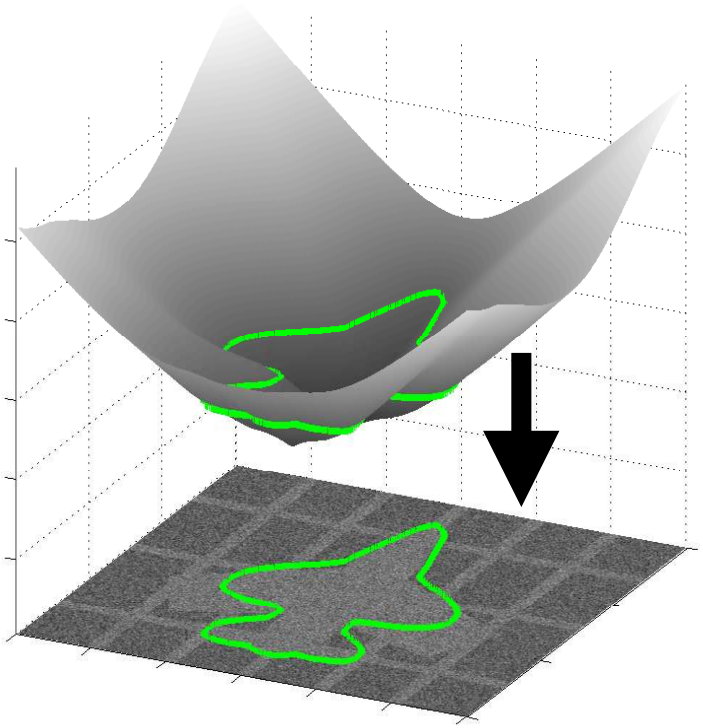
Euclidean Curve Shortening Flow

- Let $\mathrm{E}(\vec{C}) = \oint_{\vec{C}} \mathrm{d}s$ where $\mathrm{d}s = \|\vec{C}'(p)\|\mathrm{d}p$ is differential arc length.
- This energy functional is smaller when $\vec{C}$ is shorter, so the gradient flow is in the direction which minimizes the curve length the fastest. The gradient flow (which can be found using the Euler-Lagrange equation) is:

$$\frac{\mathrm{d}\vec{C}}{\mathrm{d}t}(p) = -\kappa_{\vec{C}}(p)\vec{\mathcal{N}}_{\vec{C}}(p) \ .$$

- This flow has a smoothing effect and nice geometrical properties (*e.g.*, evolution using this flow shrinks any embedded plane curve to a point without any self intersections).
- Common to use curve length as a regularizing prior term.

Level Set Methods



- A natural numerical implementation to track $\vec{C}$ is to use marker points on the boundary [Kass *et al.* 1988].
- This approach has problems with reinitialization and topological change.
- Level sets are an alternative approach which evolve a surface Ψ (one dimension higher than our curve) whose zeroth level set is $\vec{C}$. [Osher and Sethian 1988]

Level Set Methods (continued)

- Setting $\Psi(\vec{C}(p)) = 0$ for all $p \in [0,1]$ and differentiating with respect to t, we obtain:

$$\frac{\mathrm{d}\Psi}{\mathrm{d}t} = \frac{\mathrm{d}\vec{C}}{\mathrm{d}t} \cdot \nabla\Psi = (f\vec{\mathcal{N}}_{\vec{C}}) \cdot \nabla\Psi \ .$$

- Standard curve derivatives can be written in terms of Ψ:

$$\vec{\mathcal{N}}_{\vec{C}} = \frac{\nabla\Psi}{\|\nabla\Psi\|} \text{ and } \kappa_{\vec{C}} = \nabla \cdot \left(\frac{\nabla\Psi}{\|\nabla\Psi\|}\right) \ .$$

- This results in an evolution equation for Ψ of:

$$\frac{\mathrm{d}\Psi}{\mathrm{d}t} = f\|\nabla\Psi\| \ .$$

- The force function f is only defined on the curve. Velocity extension methods are a standard method to extend it to Ω.

Markov Chain Monte Carlo

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

$$\pi(z) = \int \pi(x)\mathrm{T}(z \,|\, x)\mathrm{d}x \ .$$

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

$$\pi(z)\mathrm{T}(x \,|\, z) = \pi(x)\mathrm{T}(z \,|\, x) \ .$$

- If a chain is ergodic and detailed balance holds, successive samples from $\mathrm{T}(z \,|\, x)$ *asymptotically* become samples from $\pi(x)$.

Metropolis-Hastings

- General method developed by Metropolis *et al.* (1953) and extended by Hastings (1970).
- Define transition probability as the product of a proposal distribution $\mathtt{q}(y \,|\, x)$ and an acceptance probability $\mathtt{a}(y \,|\, x)$.
- A candidate sample is generated from $\mathtt{q}$, and the Hastings ratio is computed:

$$\eta(y \,|\, x) = \frac{\pi(y)\mathtt{q}(x \,|\, y)}{\pi(x)\mathtt{q}(y \,|\, x)} \quad .$$

- Then the next iterate value $z = y$ with probability $\min(1, \eta(y \,|\, x))$. Otherwise $z = x$.
- Problem of sampling from π is now the problem of generating many samples from $\mathtt{q}$ and evaluating π.

Outline of the talk

1. Motivation and problem statement.
2. Curve evolution and MCMC methods.
3. **MCMC curve sampling.**
4. 2D curve sampling results.
5. Conditional simulation.
6. Hybrid 2D/3D models.
7. Conclusions and future work.

Curve Sampling

- We construct a curve sampling framework based on the Metropolis-Hastings algorithm.
- The target distributions $\pi(\vec{C})$ are usually based on standard curve evolution energy functionals (*e.g.*, Chan-Vese or non-parametric densities similar to Kim *et al.*).
- We define our proposal distribution $\mathtt{q}(\vec{\Gamma}^{(t)} \,|\, \vec{C}^{(t-1)})$ by specifying a method of generating samples from it using random curve perturbations.

Curve Perturbations

- For consistency, all perturbations for a curve $\vec{C}$ are defined relative to a canonical arc length parameterization $\vec{C}_{\mathrm{a}}$.
- We generate random, correlated Gaussian noise

$$f^{(t)} = \mu_{\vec{C}_{\mathrm{a}}^{(t-1)}}(p) + h \circledast n^{(t)}(p) \quad .$$

- These smooth perturbation are added to the normal of the curve:

$$\vec{\Gamma}^{(t)}(p) = \vec{C}_{\mathrm{a}}^{(t-1)}(p) + f^{(t)}(p)\vec{\mathcal{N}}_{\vec{C}_{\mathrm{a}}^{(t-1)}}(p)\delta t$$

- Our standard choice for the mean perturbation is:

$$\mu_{\vec{C}_{\mathrm{a}}^{(t)}}(p) = -\alpha\kappa_{\vec{C}_{\mathrm{a}}^{(t)}}(p) + \gamma_{\vec{C}_{\mathrm{a}}^{(t)}} \quad .$$

Convergence

- For ergodicity, we need to show *irreducibility* and aperiodicity. The latter is difficult to show, but even without it, sample averages will converge asymptotically.
- For a chain to be irreducible, any two curves $\vec{C}_0$ and $\vec{C}_1$ with non-zero probability under π must have non-zero probability of transitioning from $\vec{C}_0$ to $\vec{C}_1$ by simulating the chain.
- For "nice" curves, we can construct a discrete evolution:

$$\vec{C}(p, \tau + \delta t) = \vec{C}(p, \tau) + \frac{\delta t}{T} \left\langle \vec{C}_T(p) - \vec{C}_0(p), \vec{\mathcal{N}}_{\vec{C}_\tau}(p) \right\rangle \vec{\mathcal{N}}_{\vec{C}_\tau}(p)$$

  with non-zero probability under our Gaussian perturbation model.
- We can make arbitrary curves convex using a $-\kappa\vec{\mathcal{N}}$ flow [Gage 1986] which also has non-zero probability for our perturbation model.

Numerical Implementation

- While we view our method as having a continuous curve, any implementation must be discrete.
- At time t, we generate the random portion of the perturbation $r^{(t)}(p)$ as a discrete Gaussian vector $\boldsymbol{r}^{(t)} = \boldsymbol{H}\boldsymbol{n}^{(t)}$ (where $\boldsymbol{n}^{(t)}$ is white noise and $\boldsymbol{H}$ implements circular convolution).
- The curve is discretized on a set of points $\{p_i\}_{i=1}^{N_c} \in [0,1]$, extracting the points from the level set $\Psi_{\vec{C}_{\mathrm{a}}^{(t-1)}}$.
- The $\boldsymbol{r}^{(t)}$ values are interpolated to $\{p_i\}_{i=1}^{N_c}$ (using, *e.g.*, linear interpolation or cubic splines) and added to the mean perturbation $\mu_{\vec{C}_{\mathrm{a}}^{(t)}}(p)$ (computed on the discretization points) to form a perturbation $\boldsymbol{f}^{(t)}$.
- We apply $\boldsymbol{f}^{(t)}$ to $\vec{C}_{\mathrm{a}}^{(t-1)}$ using a narrowband level set implementation.

Detailed Balance

- To implement Metropolis-Hastings, we need to be able to calculate the Hastings ratio:

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

- The target distribution computation is application dependent.
- The probability of the forward transition is approximately the probability of generating the perturbation $\boldsymbol{f}^{(t)}$:

$$\mathtt{q}(\vec{\Gamma}^{(t)} \,|\, \vec{C}^{(t-1)}) \approx \mathrm{p}(\boldsymbol{f}) \propto \exp\left(-\frac{\boldsymbol{n}^{\mathrm{T}}\boldsymbol{n}}{2\sigma^2}\right)$$

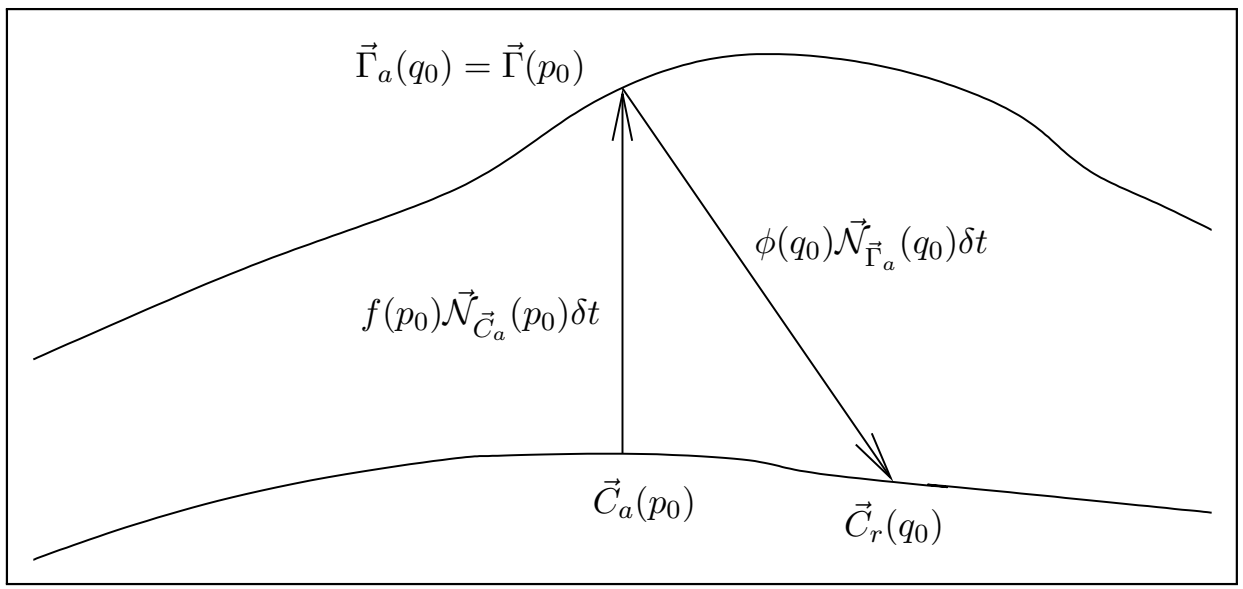
This is exact for infinitesimal δt.

Reverse Perturbation

- The reverse perturbation is the one that takes us from $\vec{\Gamma}^{(t)}$ back to $\vec{C}_{\mathrm{r}}^{(t-1)}$, a curve which is geometrically identical to $\vec{C}^{(t-1)}$:

$$\vec{C}_{\mathrm{r}}^{(t-1)}(q) = \vec{\Gamma}_{\mathrm{a}}^{(t)}(q) + \phi^{(t)}(q)\vec{\mathcal{N}}_{\vec{\Gamma}_{\mathrm{a}}^{(t)}}(q)\delta t$$

$$\phi^{(t)}(q) = \mu_{\vec{\Gamma}_{\mathrm{a}}^{(t)}}(q) + h \circledast \nu^{(t)}(q)$$



Reverse Perturbation (continued)

- Given the reverse perturbation $\phi^{(t)}$, we can employ a similar discretization as for the forward computation and approximate the reverse proposal distribution probability as:

$$\mathsf{q}(\vec{C}^{(t-1)} \,|\, \vec{\Gamma}^{(t)}) \approx \mathrm{p}(\boldsymbol{\phi}) \propto \mathrm{p}(\boldsymbol{\nu}) \propto \exp\left(-\frac{\boldsymbol{\nu}^{\mathrm{T}}\boldsymbol{\nu}}{2\sigma^2}\right)$$

- By building a linear approximation to $\vec{C}_{\mathrm{a}}^{(t-1)}$ around p_0, we can estimate the reverse perturbation at q_0 (where p_0 and q_0 are defined so that $\vec{\Gamma}^{(t)}(p_0) = \vec{\Gamma}_{\mathrm{a}}^{(t)}(q_0)$) as:

$$\hat{\phi}_{\mathrm{lin}}^{(t)}(q_0) = -\frac{f^{(t)}(p_0)}{\left\langle \vec{\mathcal{N}}_{\vec{C}_{\mathrm{a}}^{(t-1)}}(p_0),\, \vec{\mathcal{N}}_{\vec{\Gamma}_{\mathrm{a}}^{(t)}}(q_0) \right\rangle} \quad .$$

Summary of Algorithm

1. Initialize $\vec{C}^{(0)}$ to some initial value (deterministic or random). Set $t = 1$.
2. Generate candidate sample $\vec{\Gamma}^{(t)} \sim \mathtt{q}(\vec{\Gamma} \,|\, \vec{C}^{(t-1)})$ by creating a Gaussian perturbation $f^{(t)}$ and applying it to the normal:

$$\vec{\Gamma}^{(t)}(p) = \vec{C}^{(t-1)}(p) + f^{(t)}(p)\vec{\mathcal{N}}_{\vec{C}^{(t-1)}}(p)\delta t$$

   for some positive constant δt.
3. Compute Hastings ratio $\eta(\vec{\Gamma}^{(t)} \,|\, \vec{C}^{(t-1)})$. This requires evaluation of the forward and reverse perturbation probabilities $\mathtt{q}(\vec{\Gamma}^{(t)} \,|\, \vec{C}^{(t-1)})$ and $\mathtt{q}(\vec{C}^{(t-1)} \,|\, \vec{\Gamma}^{(t)})$ as well as the target distribution probabilities $\pi(\vec{C}^{(t-1)})$ and $\pi(\vec{\Gamma}^{(t)})$.
4. Accept or reject $\vec{\Gamma}^{(t)}$ with probability $\eta(\vec{\Gamma}^{(t)} \,|\, \vec{C}^{(t-1)})$ to obtain the current iterate value $\vec{C}^{(t)}$.
5. Increment t and return to Step 2.

Outline of the talk

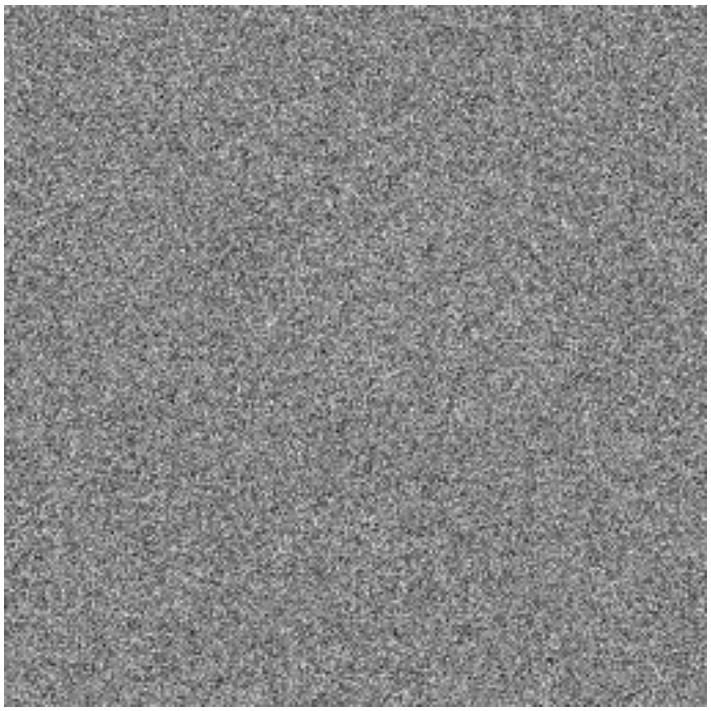
1. Motivation and problem statement.
2. Curve evolution and MCMC methods.
3. MCMC curve sampling.
4. **2D curve sampling results.**
5. Conditional simulation.
6. Hybrid 2D/3D models.
7. Conclusions and future work.

Visualizing Samples

We use four main methods for visualizing the output of our curve sampling algorithm:

1. Most likely samples
   - Close to the global maximum.
2. Histogram images
   - Given samples $\{\vec{C}_i\}_{i=1}^N$, $\Phi(\boldsymbol{x}) = \frac{1}{N}\sum_{i=1}^N \mathcal{H}(-\Psi_{\vec{C}_N}(\boldsymbol{x}))$ (*i.e.*, the percentage of samples for which $\boldsymbol{x}$ is inside).
3. Marginal confidence bounds
   - Level curves of Φ. These can give an idea of the range of likely locations of the true curve.
4. Principal modes of variation
   - Principal components analysis (PCA) on signed distance functions. [Leventon *et al.* 2000]

Synthetic Noisy Image Example



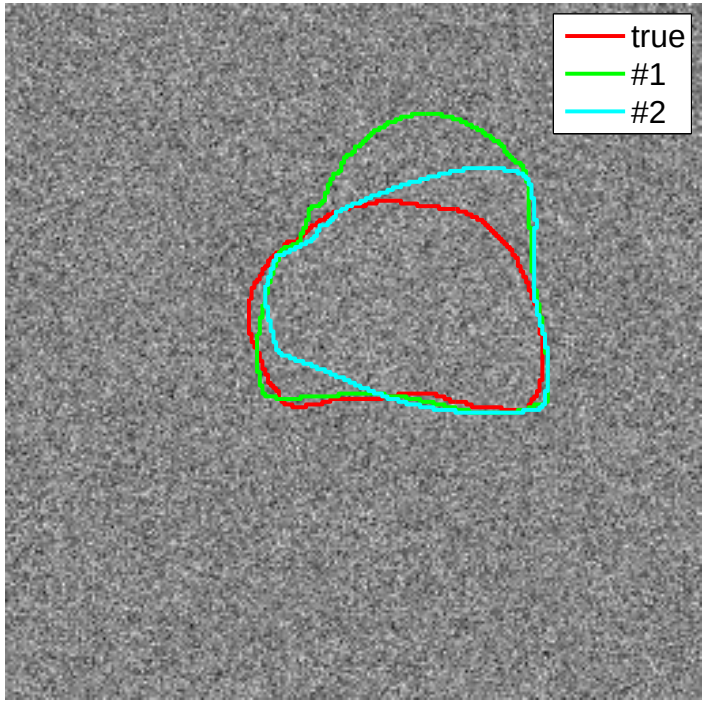
- Assume a piecewise-constant image $m(\boldsymbol{x})$ with white Gaussian noise $w(\boldsymbol{x})$:

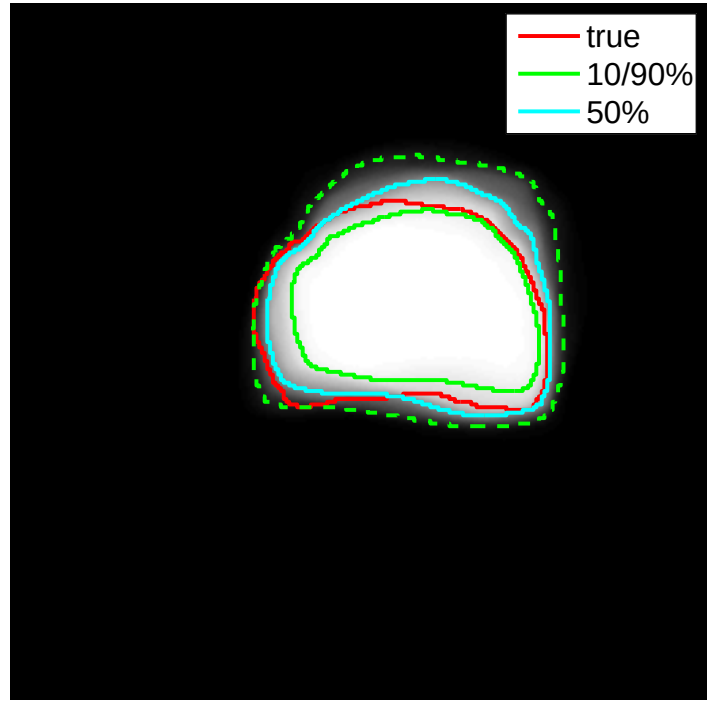
$$I(\boldsymbol{x}) = m(\boldsymbol{x}) + w(\boldsymbol{x})$$

- This corresponds to the Chan-Vese energy functional (which also adds a regularizing term):

$$\mathrm{E}(\vec{C}) = \iint\limits_{\mathcal{R}_{\vec{C}}} (I - m_1)^2 \mathrm{d}\boldsymbol{x} + \iint\limits_{\mathcal{R}^c_{\vec{C}}} (I - m_0)^2 \mathrm{d}\boldsymbol{x} + \beta \oint\limits_{\vec{C}} \mathrm{d}s$$

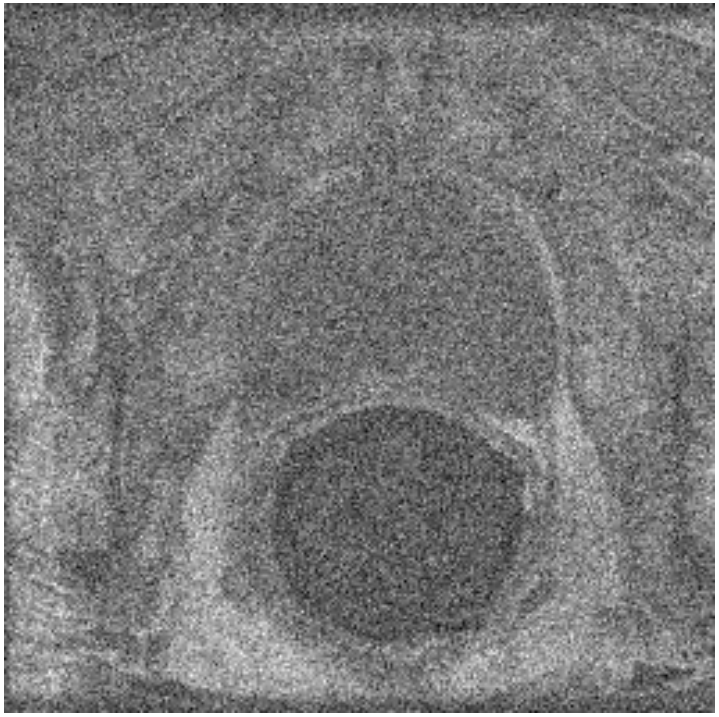
Synthetic Gaussian Results





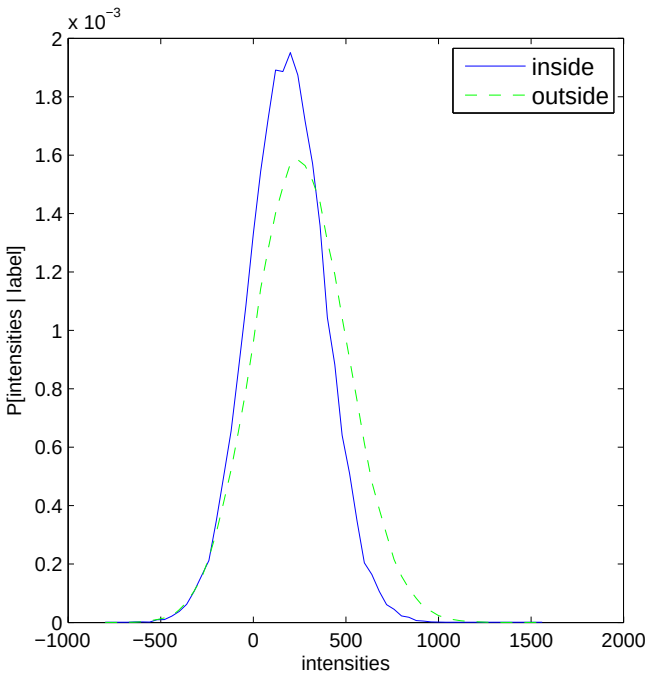
- Most likely samples not very accurate (due to specific noise configuration).
- 10/90% confidence bounds bracket the true answer.

Prostate Magnetic Resonance Example



- Segmentation of the prostate from magnetic resonance (MR) images is important for cancer staging and treatment planning.
- Here we have a bias-corrected noisy T1-weighted image that simulates a body coil image.

Non-parametric Intensity Distribution



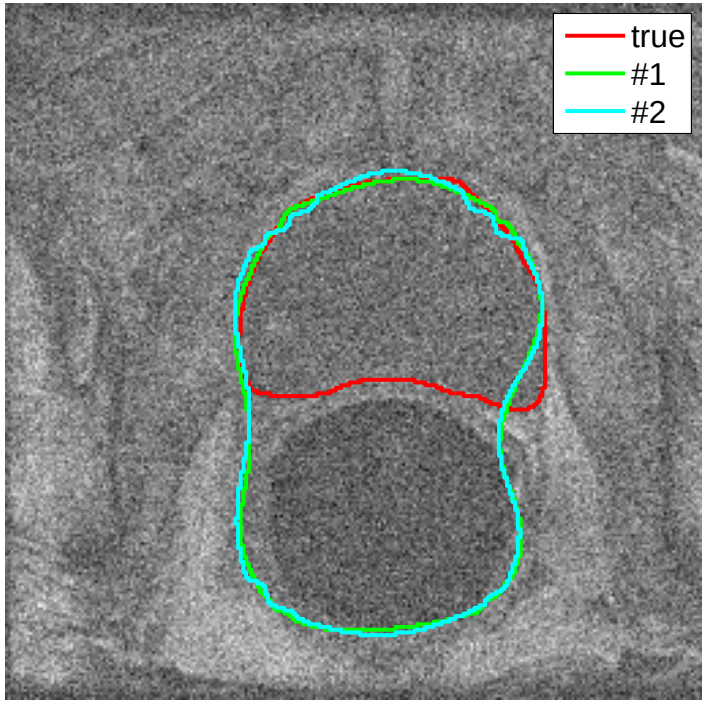
- We learn (from segmented training data) non-parametric histogram distributions $\mathrm{p}(I(\boldsymbol{x})\,|\,0)$ and $\mathrm{p}(I(\boldsymbol{x})\,|\,1)$.
- This leads to a data likelihood of:

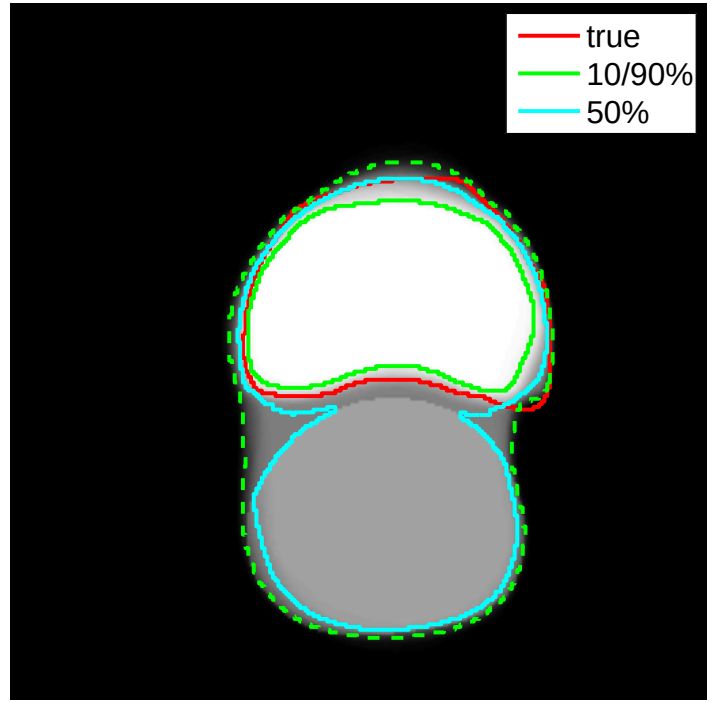
$$\mathrm{p}(I\,|\,\vec{C}) = \prod_{\boldsymbol{x}} \mathrm{p}\left(I(\boldsymbol{x})\,|\,\mathcal{H}(\Psi_{\vec{C}}(\boldsymbol{x}))\right)$$

  with $\mathcal{H}$ the Heaviside function.
- Adding in a curve length prior results in an overall target distribution of

$$\pi(\vec{C}\,|\,I) \propto \mathrm{p}(I\,|\,\vec{C})\exp(-\beta\,\mathrm{d}_{\mathrm{SAD}}(\vec{C},\vec{C}_i)) \quad .$$

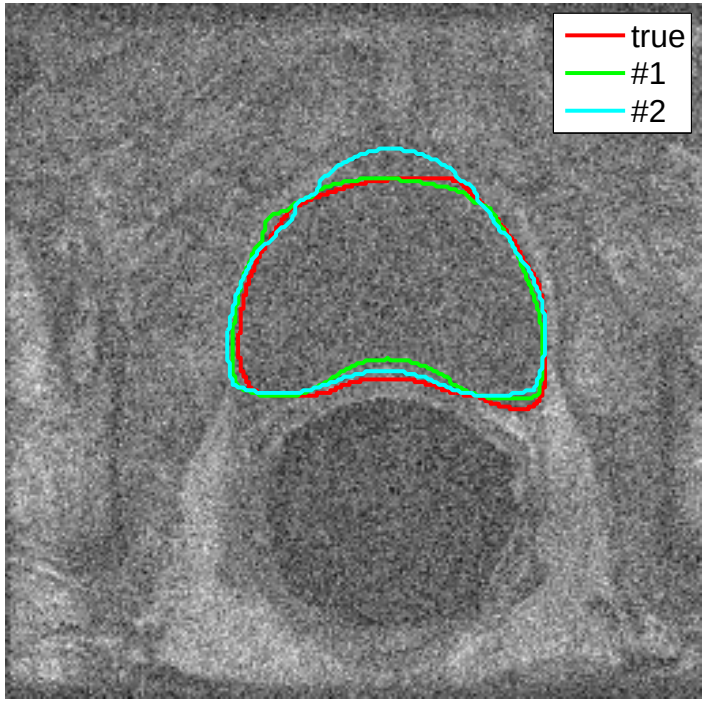
Prostate MR Results

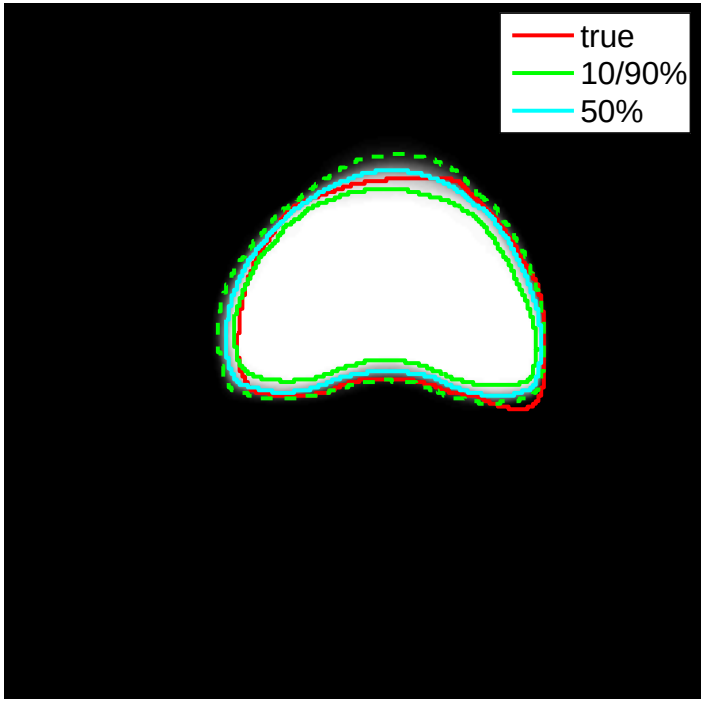




- Most likely samples actually capture the prostate and rectum.
- Multi-modality evident in the histogram image.

Clustered Prostate Samples





- We can cluster the results into prostate-only, rectum-only, and prostate-and-rectum groups.
- Here we display the results for the prostate cluster.

Outline of the talk

1. Motivation and problem statement.
2. Curve evolution and MCMC methods.
3. MCMC curve sampling.
4. 2D curve sampling results.
5. **Conditional simulation.**
6. Hybrid 2D/3D models.
7. Conclusions and future work.

Conditional Simulation

- In many problems, a model may admit many reasonable solutions. This can be due to a low signal-to-noise ratio (SNR) or an ill-posed estimation problem.
- For most segmentation algorithms, user input is limited to initialization and parameter selection.
- *Conditional simulation* involves sampling part of the solution conditioned on the rest being known (*e.g.*, pinned Brownian motion).
  - For curve sampling, part of the curve is specified. Much more feasible for sampling than constrained optimization in high-dimensional spaces.
  - Can help with both accuracy and convergence speed.
  - Leads to interactive semi-automatic segmentation approaches.

Conditional Curve Sampling

- Let $\vec{C}_{\mathrm{k}} : [0, b] \to \Omega$ be the known portion of the curve, and $\vec{C}_{\mathrm{u}} : [b, 1] \to \Omega$ be the unknown portion.
- We now wish to sample from $\tilde{\pi}(\vec{C}_{\mathrm{u}} \,|\, \vec{C}_{\mathrm{k}})$:

$$\tilde{\pi}(\vec{C}_{\mathrm{u}} \,|\, \vec{C}_{\mathrm{k}}) \propto \mathrm{p}(I \,|\, \vec{C}_{\mathrm{u}}, \vec{C}_{\mathrm{k}})\mathrm{p}(\vec{C}_{\mathrm{u}} \,|\, \vec{C}_{\mathrm{k}}) = \mathrm{p}(I \,|\, \vec{C})\mathrm{p}(\vec{C}_{\mathrm{u}} \,|\, \vec{C}_{\mathrm{k}})$$

- We note that $\mathrm{p}(\vec{C}_{\mathrm{u}} \,|\, \vec{C}_{\mathrm{k}}) = \mathrm{p}(\vec{C}_{\mathrm{u}}, \vec{C}_{\mathrm{k}})/\mathrm{p}(\vec{C}_{\mathrm{k}})$, and the denominator can generally only be obtained from $\mathrm{p}(\vec{C})$ by integrating out $\vec{C}_{\mathrm{u}}$.

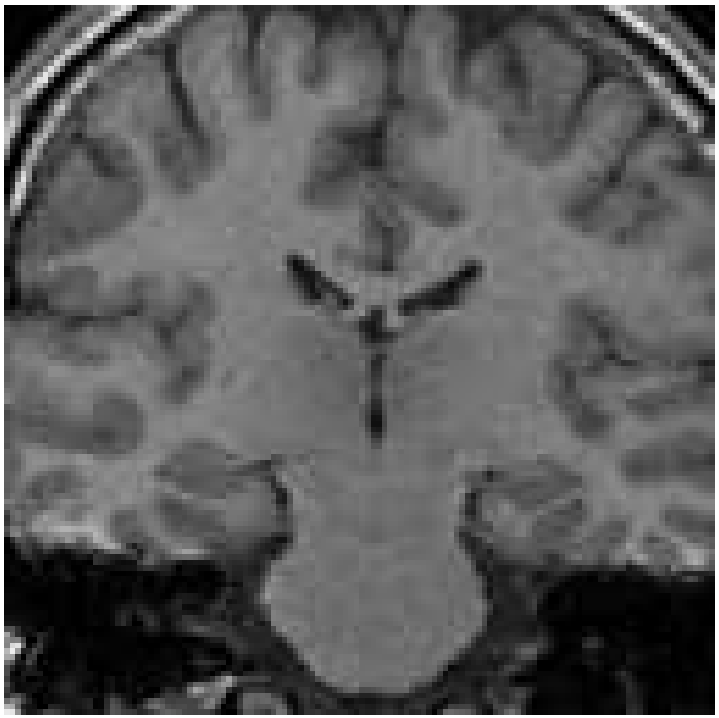
Exact Curve Information

- For special cases, evaluation of $\mathrm{p}(\vec{C}_{\mathrm{u}} \,|\, \vec{C}_{\mathrm{k}})$ is tractable:
  - $\vec{C}$ is low-dimensional.
  - $\vec{C}_{\mathrm{k}}$ is assumed to be exact.
  - $\mathrm{p}(\vec{C})$ has special form (*e.g.*, Markov structure).
- When the curve is specified exactly, we observe that

$$\tilde{\pi}(\vec{C}_{\mathrm{u}} \,|\, I, \vec{C}_{\mathrm{k}}) \propto \mathrm{p}(I \,|\, \vec{C})\mathrm{p}(\vec{C}_{\mathrm{u}}, \vec{C}_{\mathrm{k}})/\mathrm{p}(\vec{C}_{\mathrm{k}}) \propto \pi(\vec{C} \,|\, I)$$

- Thus we see that evaluation of the target distribution is unchanged. The proposal distribution must be modified so that candidate samples remain on the manifold of curves which contain $\vec{C}_{\mathrm{k}}$.
- To do so, we can multiply our earlier perturbation $f(p)$ by a scalar field $d(p)$ which is 0 on $[0, b]$.

Thalamus Segmentation



- The thalamus is a subcortical brain structure.
- Low-contrast makes it difficult to distinguish it from surrounding cerebral tissue.
- One approach to make the problem better-posed is using shape models (Pohl *et al.* 2004).
- We apply our conditional simulation approach which requires much less training and allows more user control over the segmentation process.

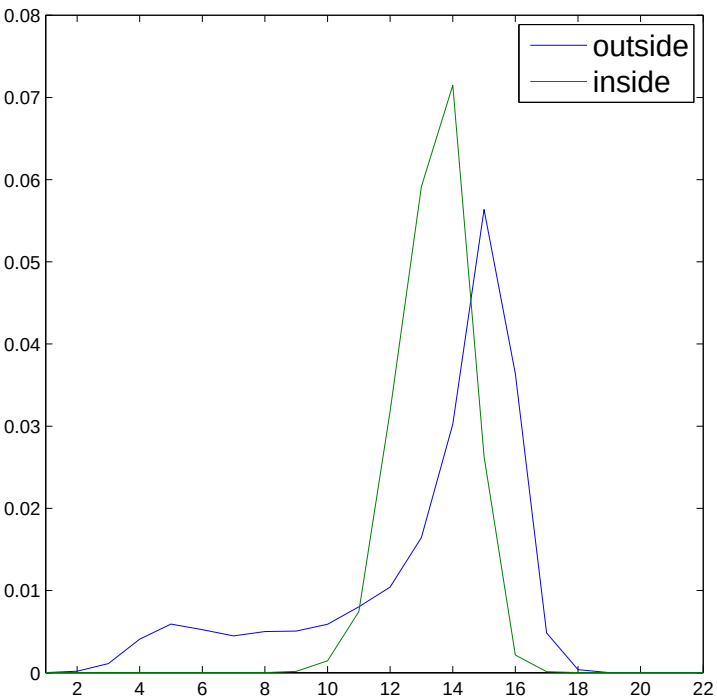
Multiple Disjoint Regions

- Disjoint regions leads to a set of multiple curves $\boldsymbol{C} = \{\vec{C}_i\}_{i=1}^{N_c}$.
- Perturb each curve individually with

$$\mathtt{q}(\boldsymbol{\Gamma} \,|\, \boldsymbol{C}) = \prod_i \mathtt{q}_i(\vec{\Gamma}_i \,|\, \vec{C}_i) \ .$$

- Curves are coupled together through the evaluation of π. Because pixel intensities in both halves of the thalamus are drawn from the same distribution, we combine the curves into a joint label map $\lambda_{\boldsymbol{C}}(\boldsymbol{x})$ which is 1 if $\boldsymbol{x}$ is inside any $\vec{C}_i$.
- If curves represent objects with different statistics, we would need to resolve ambiguities caused by overlap.

Thalamus Model



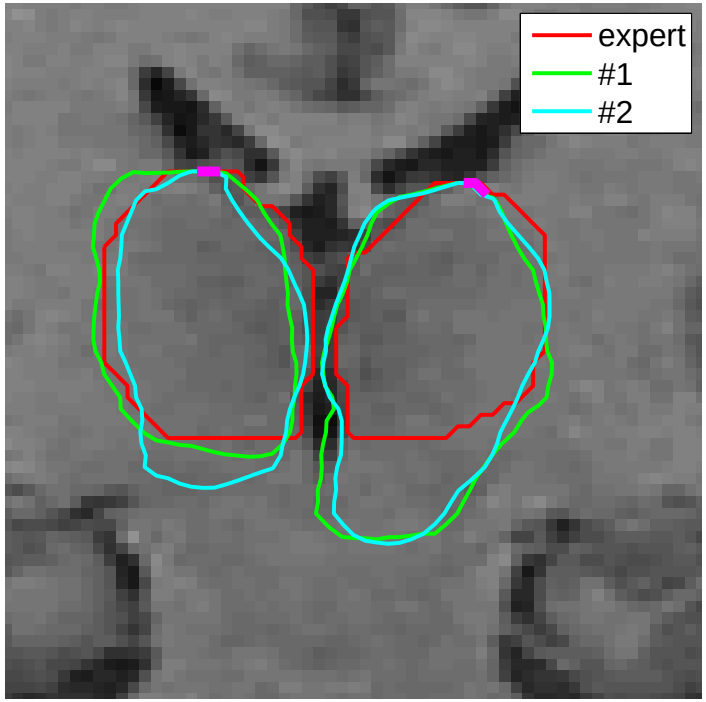
- Learn histograms from band of pixels within a distance d_0 of the expert-segmented boundary.
- Resulting data likelihood:

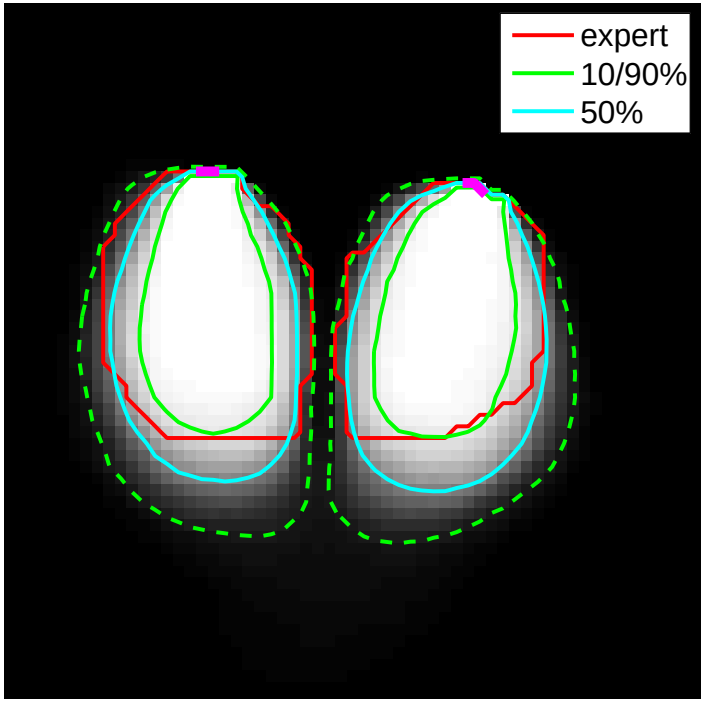
$$\mathrm{p}(I \,|\, \boldsymbol{C}) = \prod_{\substack{\{\boldsymbol{x} \,|\, \exists\, i \text{ s.t.} \\ |\tilde{\Psi}_{\vec{C}_i}(\boldsymbol{x})| \leq d_0\}}} \mathrm{p}\left(I(\boldsymbol{x}) \,|\, \lambda_{\boldsymbol{C}}(\boldsymbol{x})\right) \quad .$$

- This leads to an overall target distribution of:

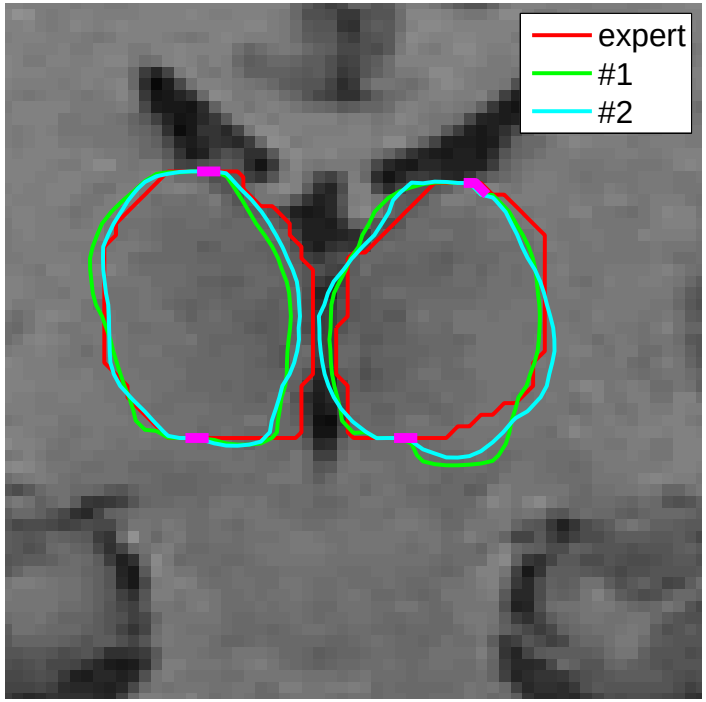
$$\pi(\boldsymbol{C}) \propto \mathrm{p}(I \,|\, \boldsymbol{C}) \exp\left(-\alpha \sum_i \oint_{\vec{C}_i} \mathrm{d}s\right) \quad .$$

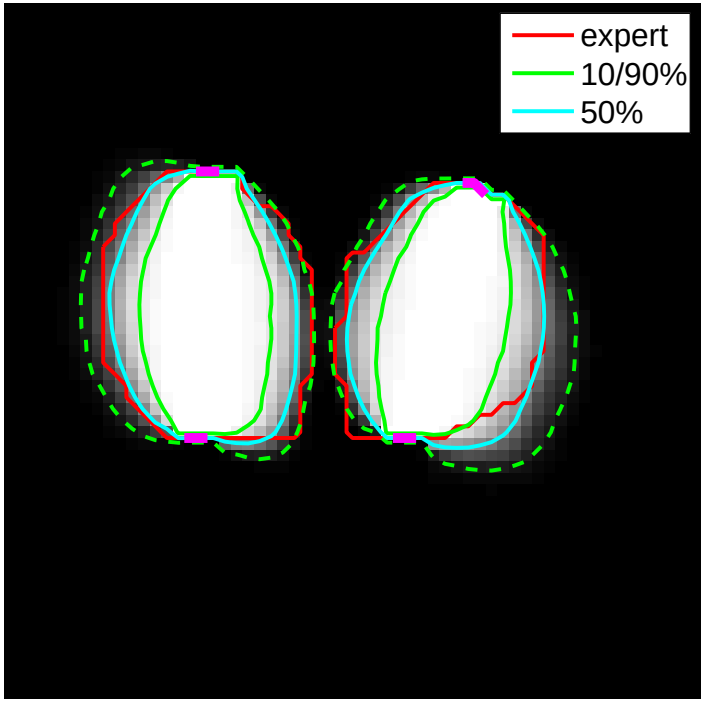
Top Points Fixed



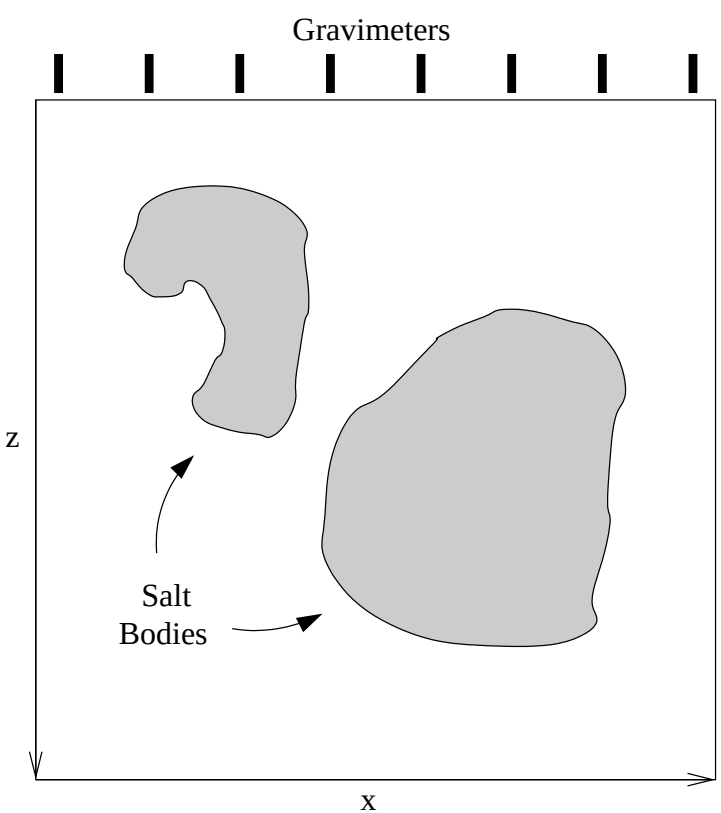


Adding Constraints at the Bottom





Gravity Inversion



- The goal is to find salt body boundaries using an array of surface gravimeters.
- Difficult to image below salt without knowing bottom salt. Salt bodies also act as liquid traps (*e.g.*, gas, oil).
- Data are processed to remove base effects (*e.g.*, the geoid, centrifugal force) to leave residual gravity effects from differing salt density:

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

- Shape priors difficult to apply.

Gravity Inversion Model

- To construct a curve-based gravity model, we assume constant density inside and outside salt:

$$\rho(\boldsymbol{x}; \vec{C}) = \Delta\rho\mathcal{H}(-\Psi_{\vec{C}}(\boldsymbol{x})) \ .$$

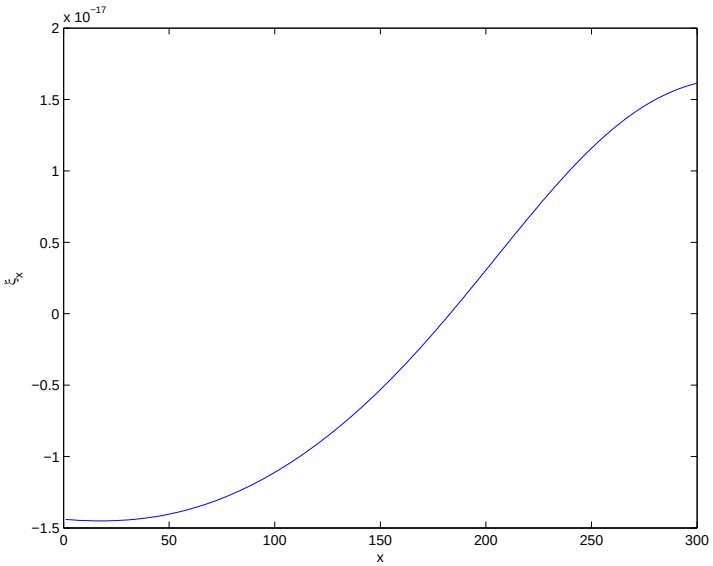
- This leads to the following forward model to translate a curve into a gravity measurement:

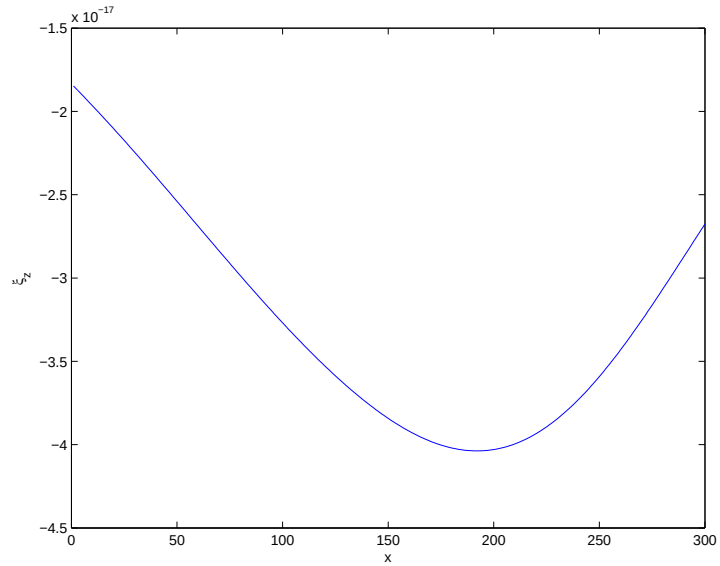
$$\vec{g}_i(\vec{C}) = \Delta\rho G \int_{\mathcal{R}_{\vec{C}}} \frac{(\boldsymbol{x} - \boldsymbol{x}_i)}{\|\boldsymbol{x} - \boldsymbol{x}_i\|^3} \mathrm{d}\boldsymbol{x} \ .$$

- We construct an energy functional that penalizes the L2 error between the observed gravity and the forward model plus a regularization penalty:

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

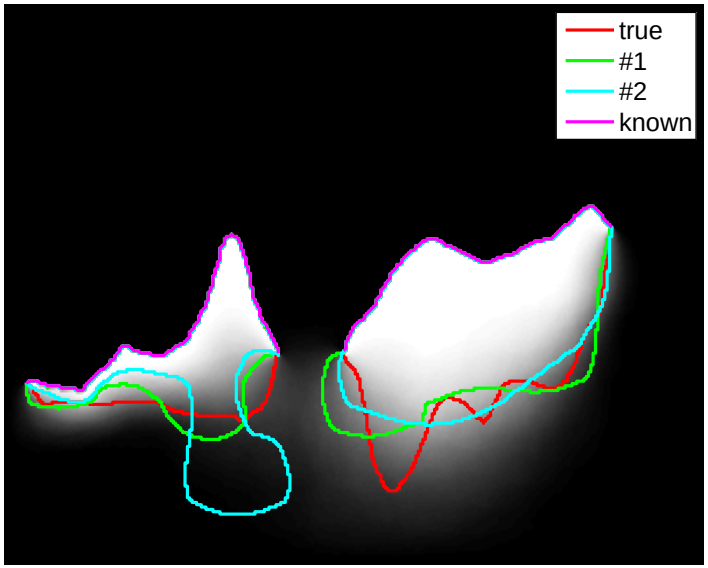
Real Geometry: Gravity Profile

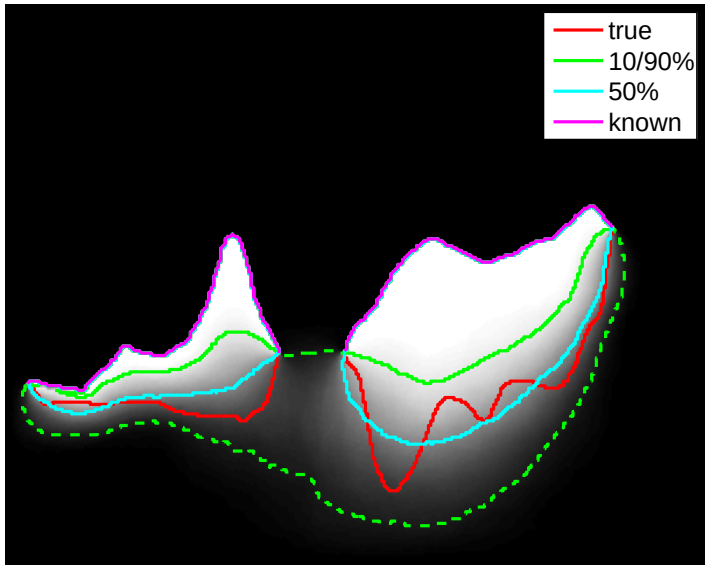




- 600 measurements, 300×240 image (72,000 pixels).
- Synthetic salt body constructed from expert-segmented seismic image.

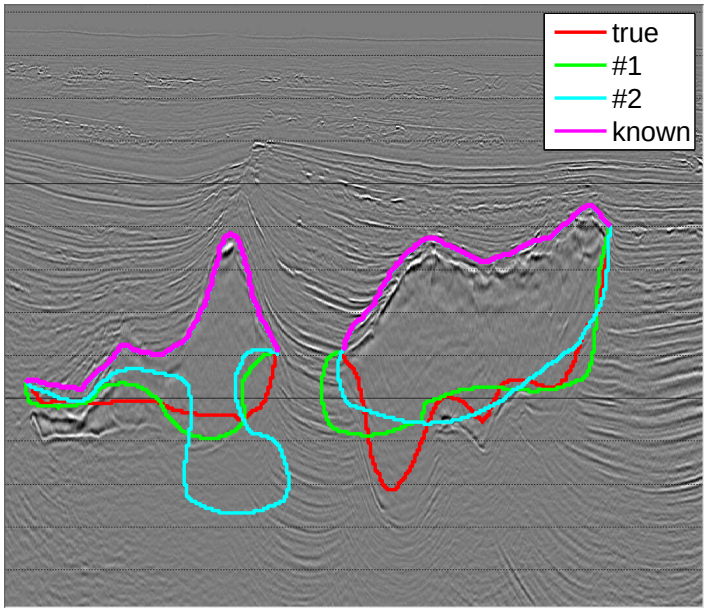
Real Geometry: Most Probable Samples & Confidence Bounds

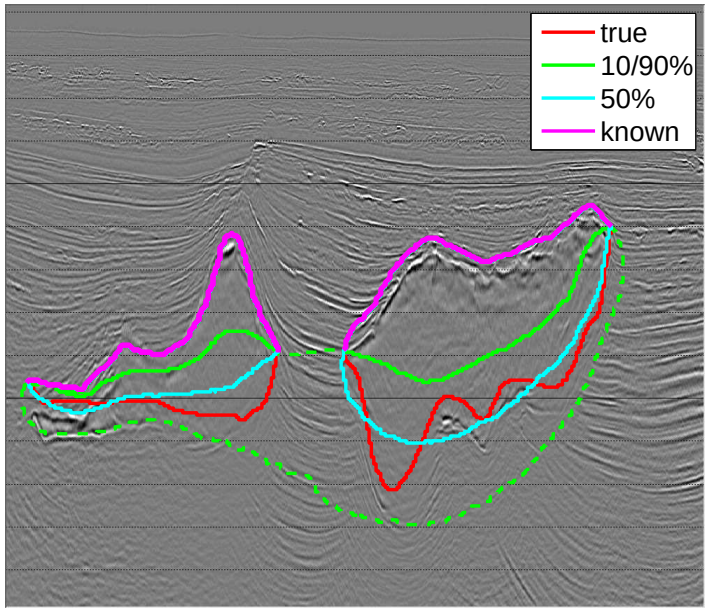




Samples generated with top salt fixed.

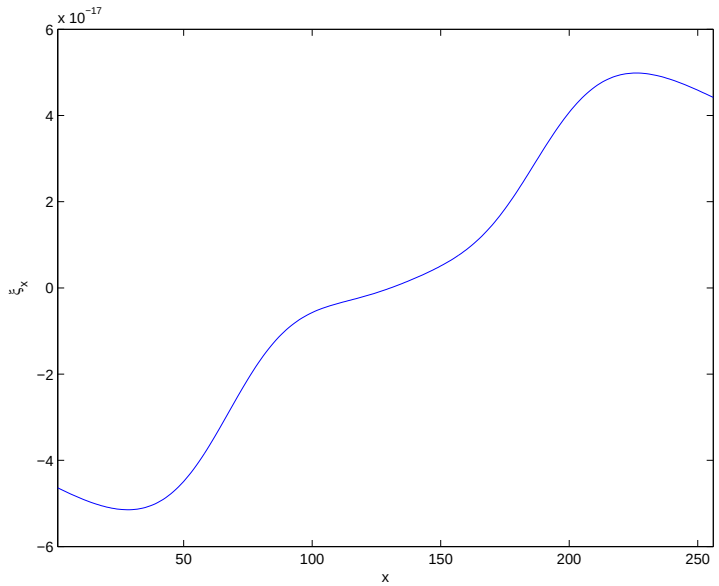
Real Geometry: Results Overlaid on Seismic

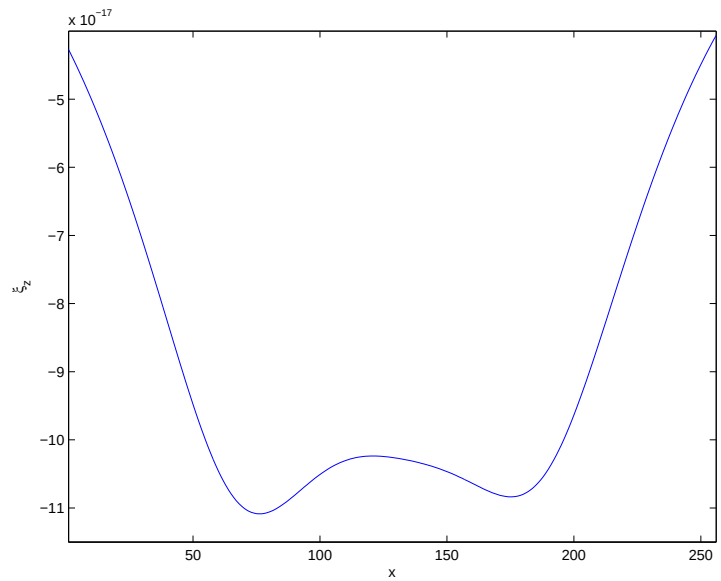




(in real life, this creates a registration problem)

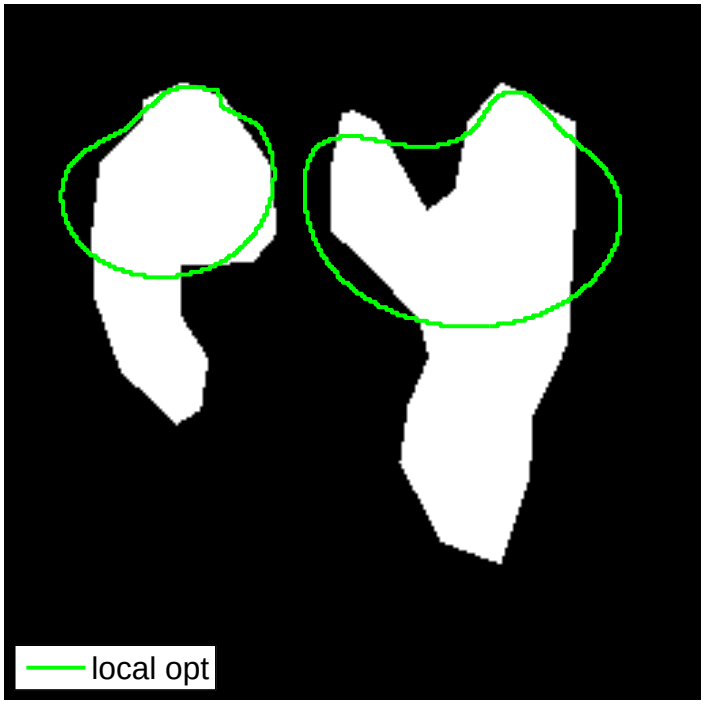
Synthetic Two Body: Gravity Profile





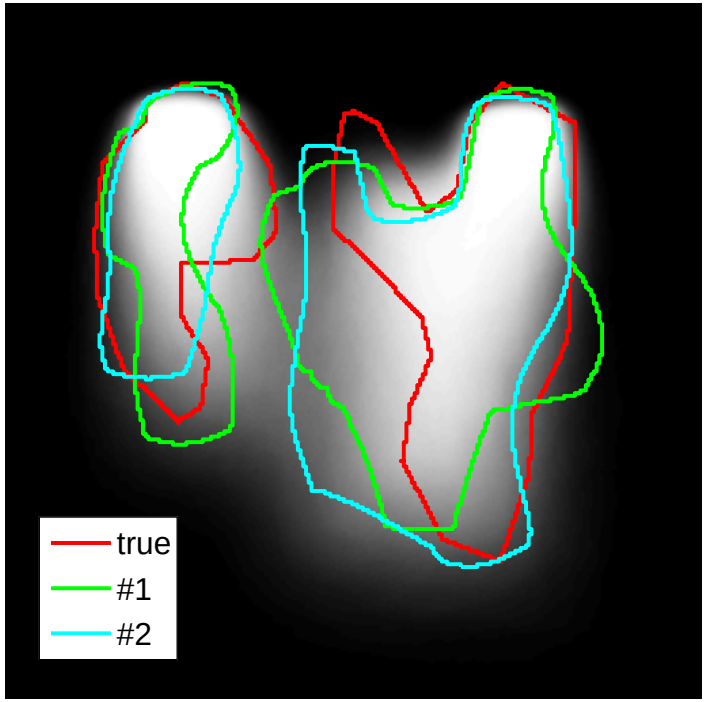
- 512 measurements, 256×256 image (65,536 pixels).
- Purely synthetic salt body.

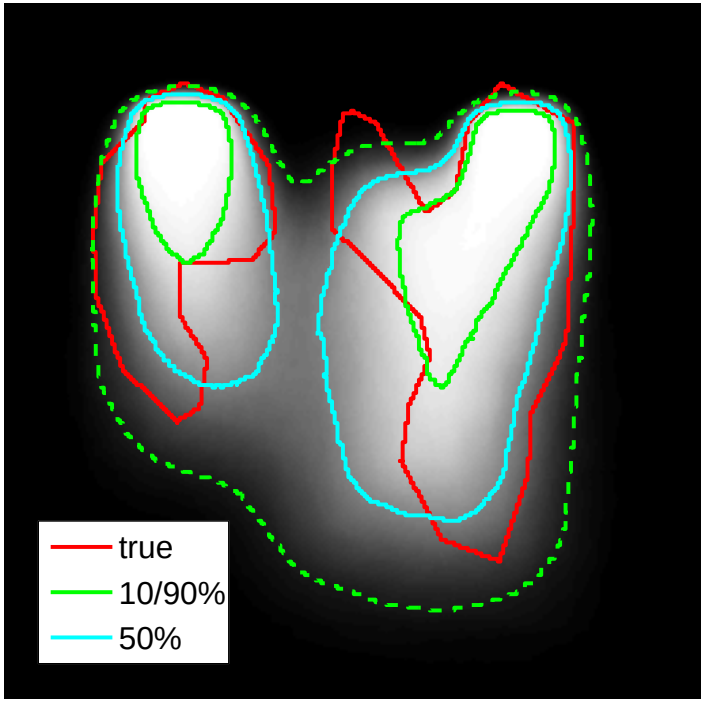
Synthetic Two Body: Optimization



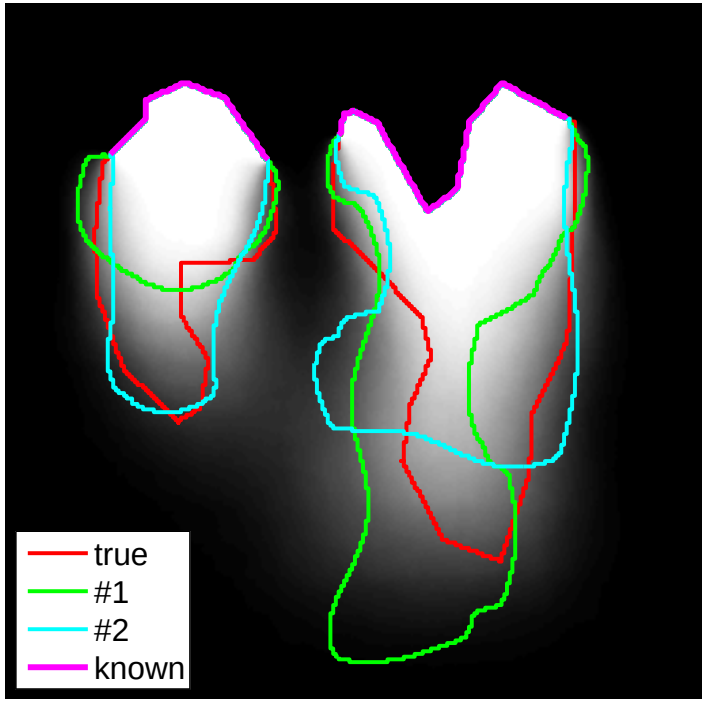
- We can form a curve evolution flow for our energy functional (not given here).
- After running multiple experiments with a varying regularization parameter α, we show here the best result.
- Location of bottom is more uncertain than location of top due to weaker gravitational effects. This results in the regularization having a stronger effect at the bottom.

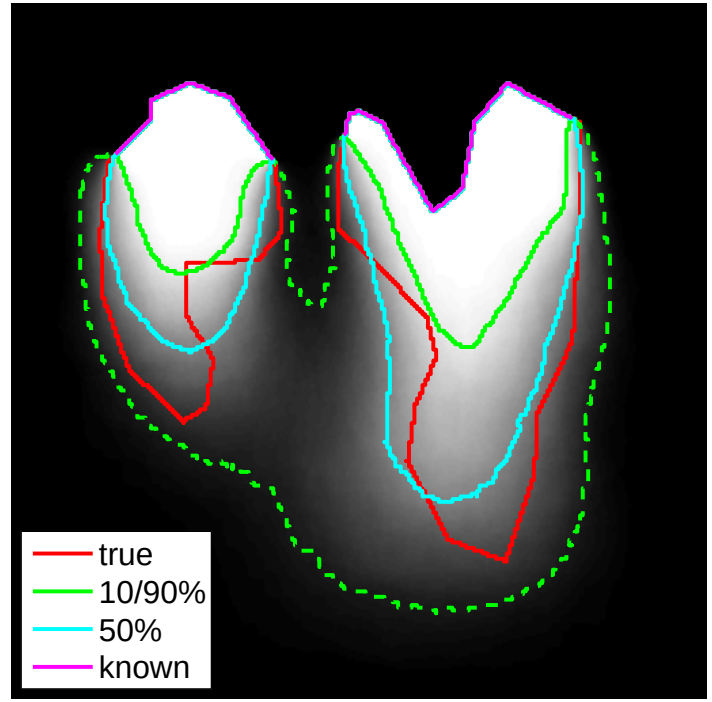
Synthetic Two Body: Unconstrained



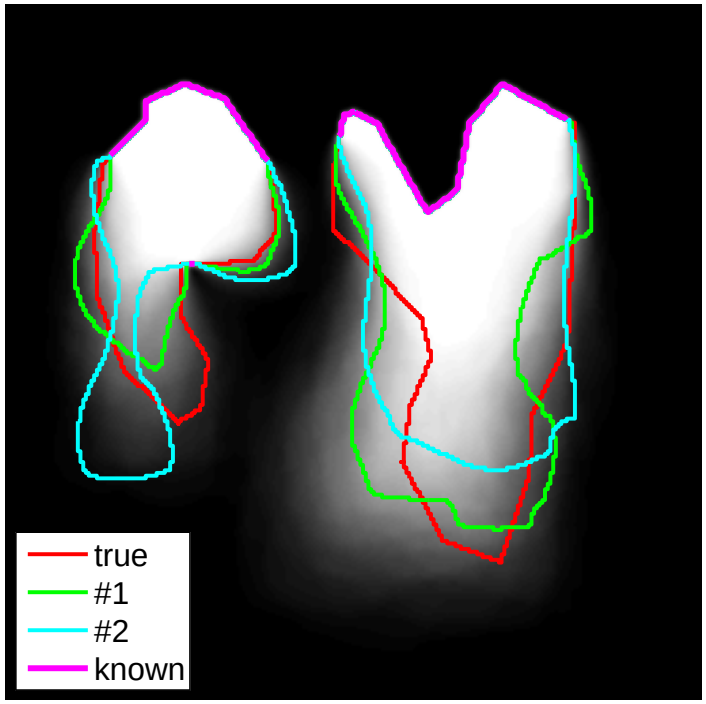


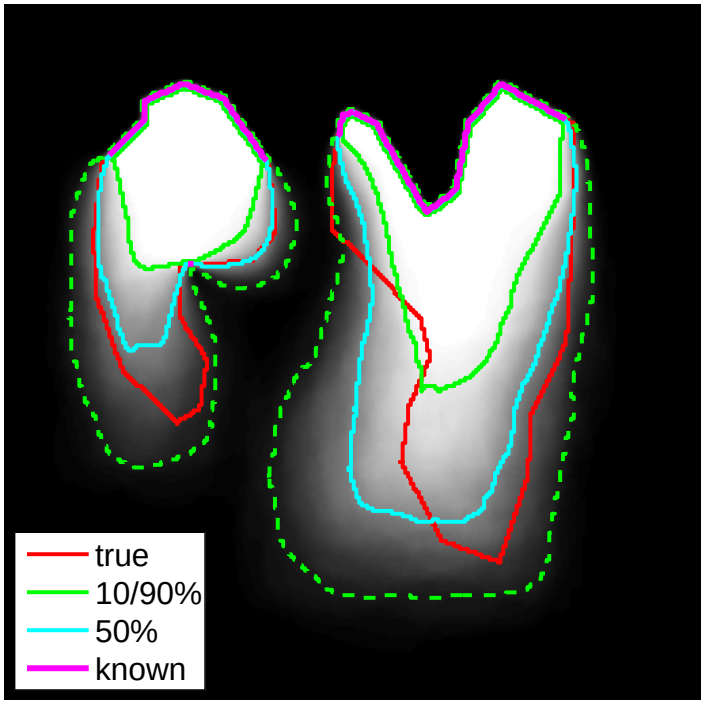
Synthetic Two Body: Top Salt





Synthetic Two Body: Top Salt and Recumbency





PCA Formulation

- Take a set of K curves $\{\vec{C}_i\}_{i=1}^{K}$, and compute the signed distance function Ψ_i for each $\vec{C}_i$.
- The mean level set function $\bar{\Psi}(\boldsymbol{x})$ is then computed and subtracted from each level set function:

$$\tilde{\Psi}_i(\boldsymbol{x}) = \Psi_i(\boldsymbol{x}) - \bar{\Psi}(\boldsymbol{x}) \ .$$

- Each level set is converted to a vector $\boldsymbol{a}_i$. These vectors then combine to form a matrix:

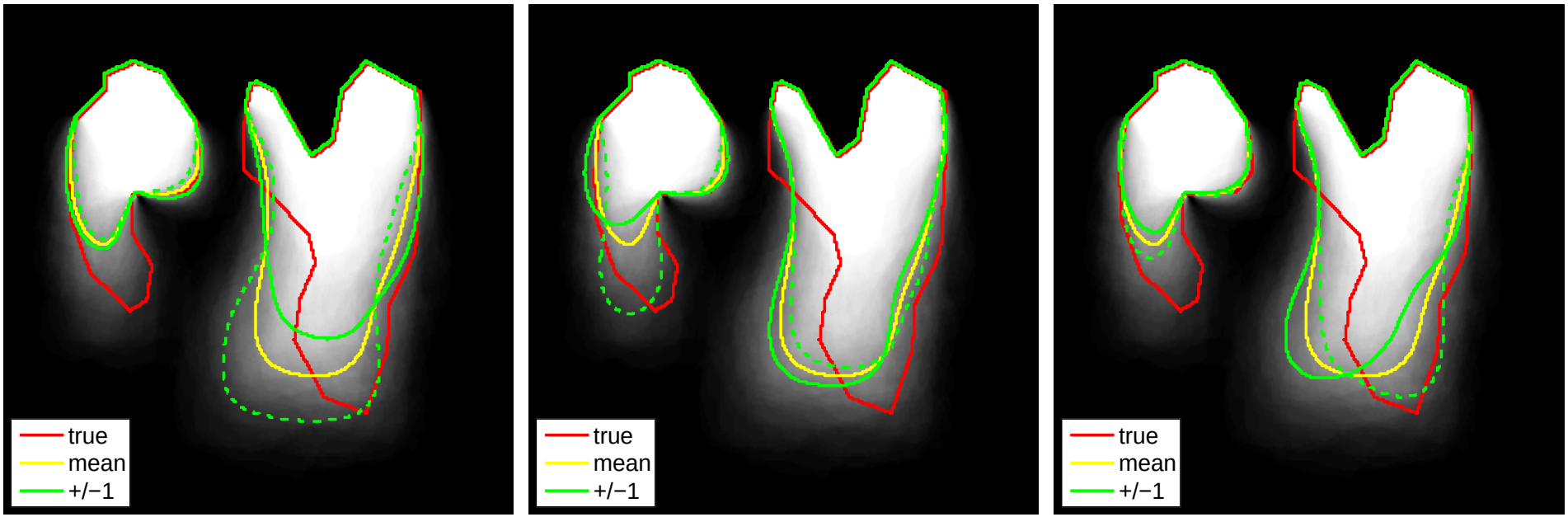
$$\boldsymbol{A} = \left(\ \boldsymbol{a}_1 \ \middle| \ \boldsymbol{a}_2 \ \middle| \ \ldots \ \middle| \ \boldsymbol{a}_K \ \right) \ .$$

- The singular value decomposition of $\boldsymbol{A}$ is:

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}} \ . \tag{1}$$

$\boldsymbol{U}$ contains the eigenvectors, and $\boldsymbol{\Sigma}$ has the variances on its a diagonal.

PCA Eigenvectors



- We show the zero level sets of the three principal eigenvectors at one standard deviation ($\Psi = \bar{\Psi} \pm 1 \cdot \sigma_i \Psi_i$).
- The second mode captures the left salt body nearly perfectly.

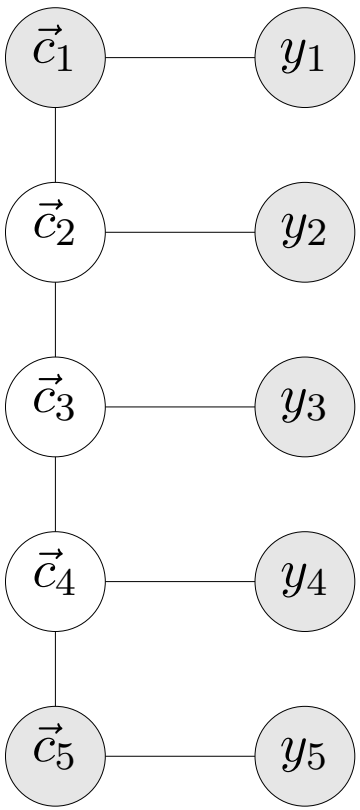
Outline of the talk

1. Motivation and problem statement.
2. Curve evolution and MCMC methods.
3. MCMC curve sampling.
4. 2D curve sampling results.
5. Conditional simulation.
6. **Hybrid 2D/3D models.**
7. Conclusions and future work.

Sampling Surfaces

- Extending our 2D curve sampling formulation to three dimensions is not straightforward, primarily because a canonical parameterization does not exist.
- Let $\vec{S} : [0,1] \times [0,1] \rightarrow \mathbb{R}^3$ be a surface and Y be the observed 3D volume.
- We construct a collection of curves on equally-spaced parallel slices $\mathcal{S} = \{\vec{c}_1, \vec{c}_2, \ldots, \vec{c}_N\}$.
- We then approximate the process of sampling from $\pi(\vec{S}|Y)$ by sampling from $\pi(\mathcal{S}|Y)$.

Hybrid 2D/3D Markov Model



- Construct target distribution as undirected Markov chain.
- This leads to the following factorization of $\pi(\mathcal{S} \,|\, Y)$ in terms of potential functions:

$$\pi(\mathcal{S} \,|\, Y) \propto \prod_{i=1}^{N} \Phi_i(\vec{c}_i) \prod_{i=1}^{N} \Lambda_i(\vec{c}_i, y_i) \prod_{i=1}^{N-1} \Phi_{i,i+1}(\vec{c}_i, \vec{c}_{i+1})$$

- $\Phi_i(\vec{c}_i)$ and $\Lambda_i(\vec{c}_i, y_i)$ involve intra-slice interactions. $\Phi_{i,i+1}(\vec{c}_i, \vec{c}_{i+1})$ models inter-slice interactions (*e.g.*, dynamic shape models).

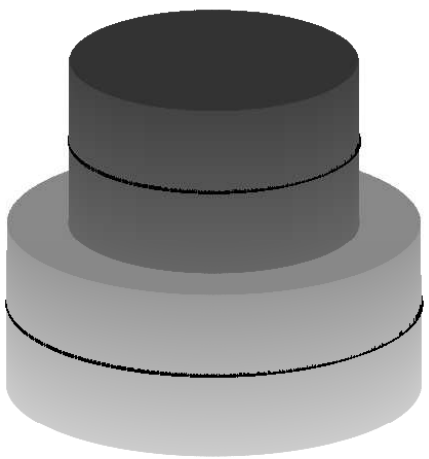
Slice-based Surface Area Model

- The standard regularizing term for 2D curve evolution is curve length. The analogous quantity in 3D is surface area.
- Consider a slice-based approximation to surface area:

$$\oiint_{\vec{S}} \mathrm{d}A \approx \iint_{\mathbf{0}\oplus\vec{c}_1} \mathrm{d}A + \sum_{i=1}^{N-1} \iint_{\vec{c}_i\oplus\vec{c}_{i+1}} \mathrm{d}A + \iint_{\vec{c}_N\oplus\mathbf{0}} \mathrm{d}A \ .$$

- We need to define a natural surface construction method to connect curves in adjoining slices (minimal surfaces are not geometrically accurate as they bow inwards).

Template Metric as a Coupling Term



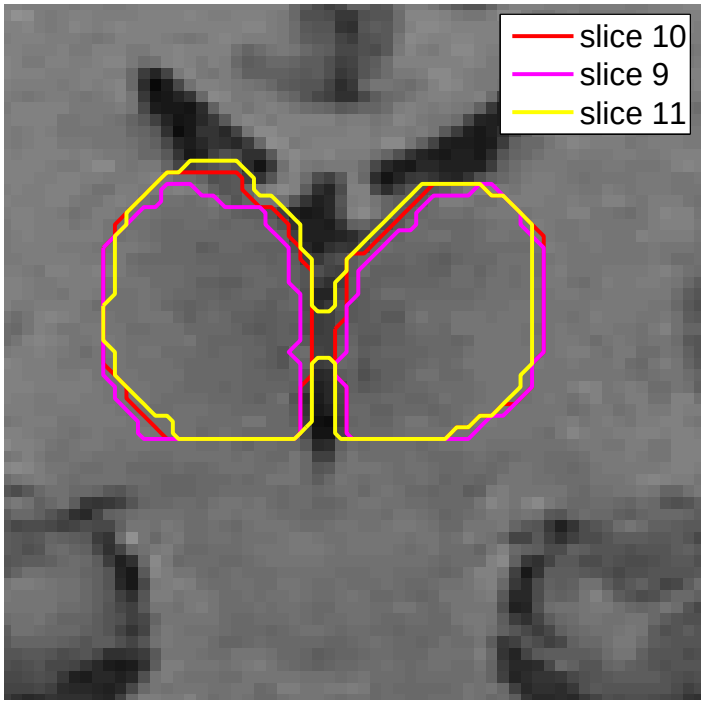
- If we approximate the coupling areas as piecewise-constant (in the z-direction), we get a stacked cylinder approximation.
- Again, the template metric (or symmetric area difference) is:

$$\mathrm{d}_{\mathrm{SAD}}(\vec{C}_1, \vec{C}_2) = \iint\limits_{\mathcal{R}_{\vec{C}_1} \setminus \mathcal{R}_{\vec{C}_2}} \mathrm{d}\boldsymbol{x} \quad + \iint\limits_{\mathcal{R}_{\vec{C}_2} \setminus \mathcal{R}_{\vec{C}_1}} \mathrm{d}\boldsymbol{x} \ .$$

- Using symmetric area difference, we can write the slice-coupling surface area as:

$$\iint\limits_{\vec{c}_i \oplus \vec{c}_{i+1}} \mathrm{d}A = \frac{\Delta z}{2} \oint_{\vec{c}_i} \mathrm{d}s + \frac{\Delta z}{2} \oint_{\vec{c}_{i+1}} \mathrm{d}s + \mathrm{d}_{\mathrm{SAD}}(\vec{c}_i, \vec{c}_{i+1})$$

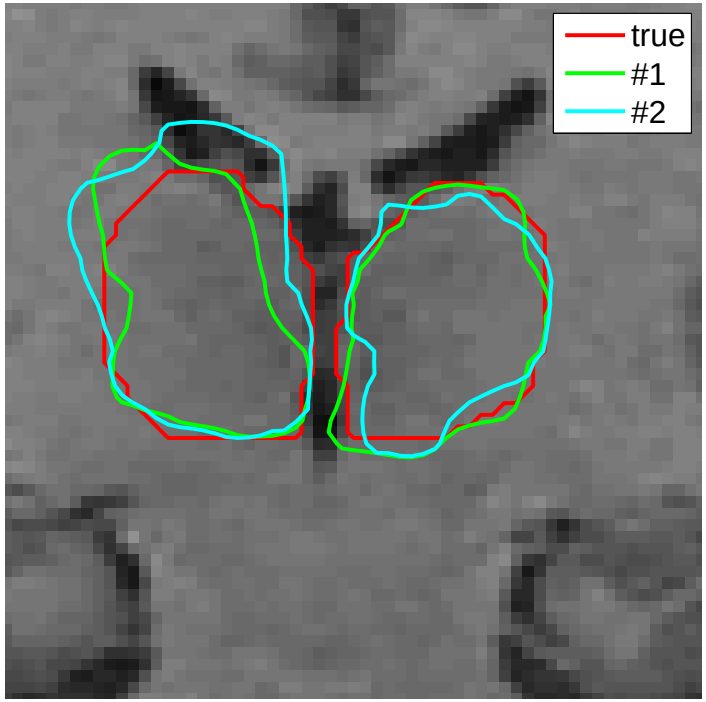
Neighbor Slice Constraints

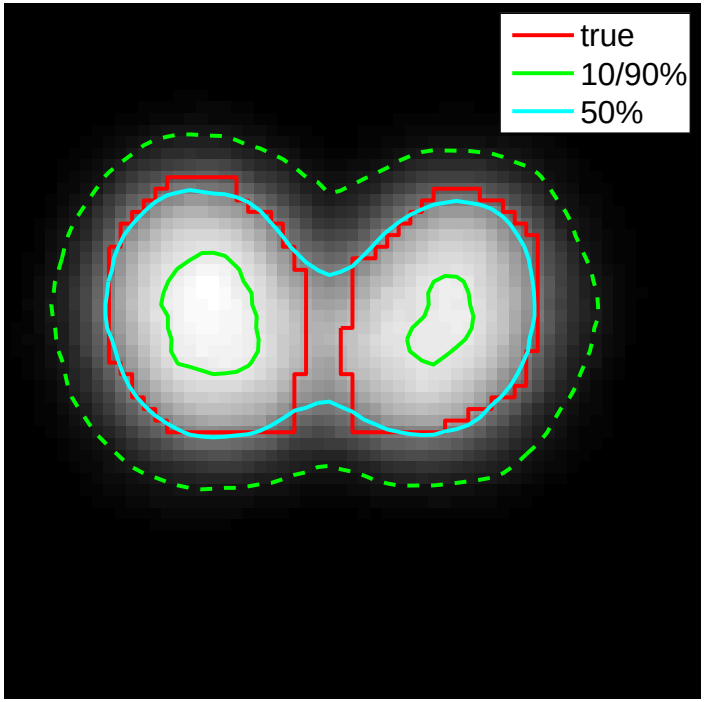


- Consider the situation where we are given $\vec{c}_{n-1}$ and $\vec{c}_{n+1}$ and we wish to $\vec{c}_n$.
- Due to Markov nature of model, $\vec{c}_i$ is conditionally independent of all other slices, so we simply have a 2D curve sampling problem with additional terms in the energy (can view as shape priors):

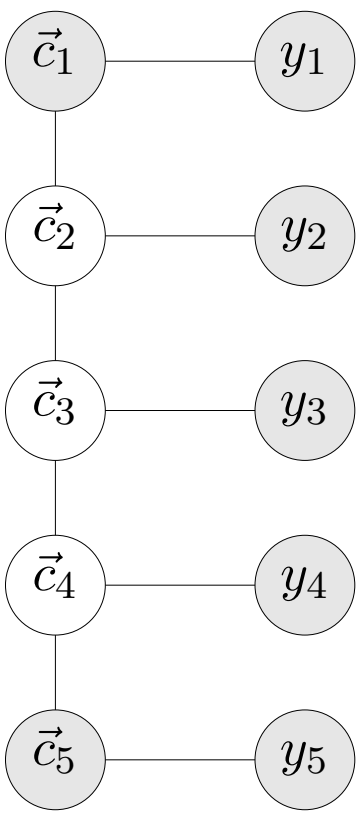
$$\pi(\vec{c}_n \,|\, \mathcal{S} \setminus \vec{c}_n, Y) \propto \Phi_n(\vec{c}_n)\Lambda_n(\vec{c}_n, y_n) \\ \Phi_{i-1,i}(\vec{c}_{i-1}, \vec{c}_i)\Phi_{i,i+1}(\vec{c}_i, \vec{c}_{i+1})$$

Thalamus: Neighbor Slice Results



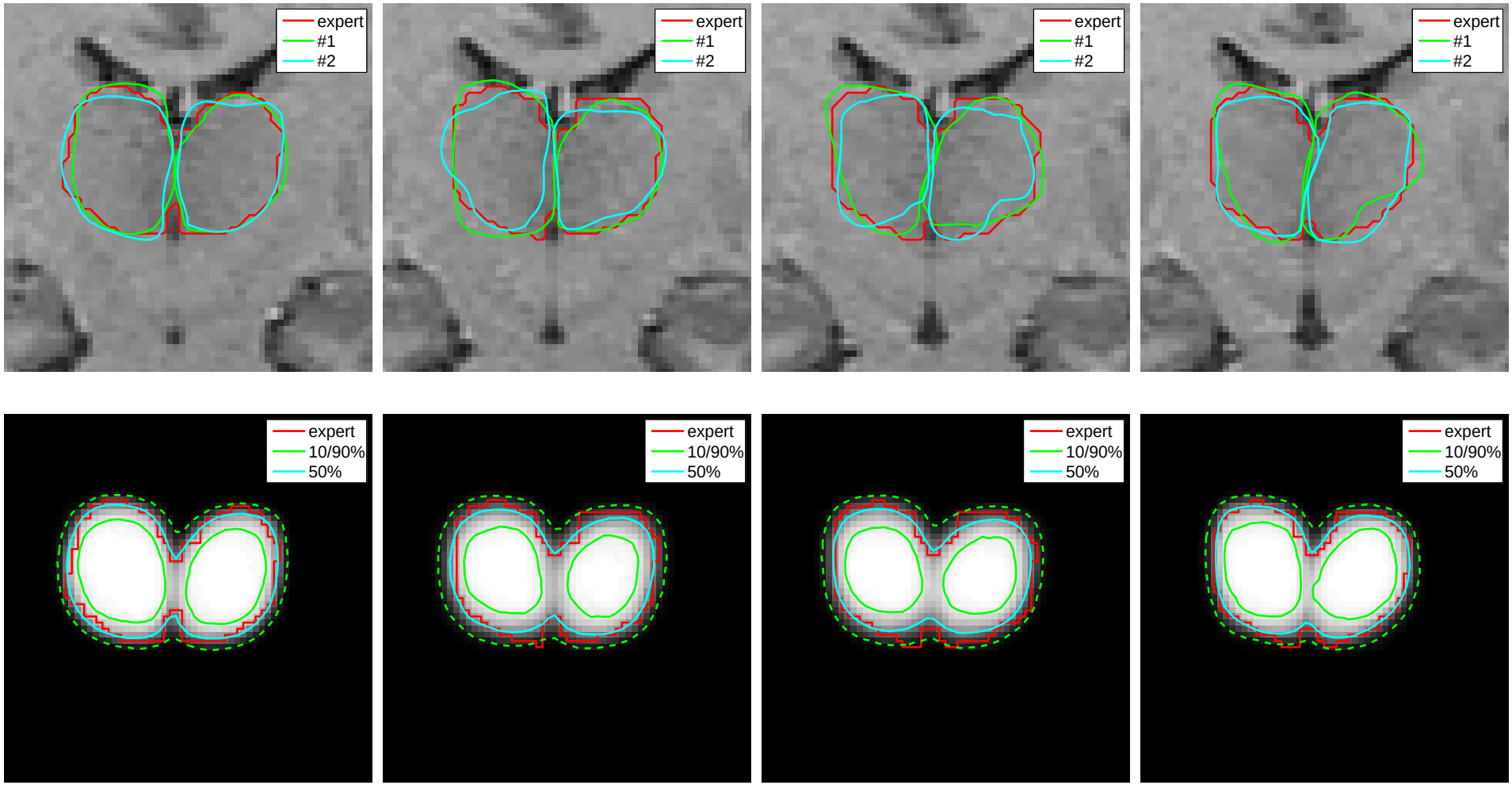


Local Metropolis-Hastings Sampling

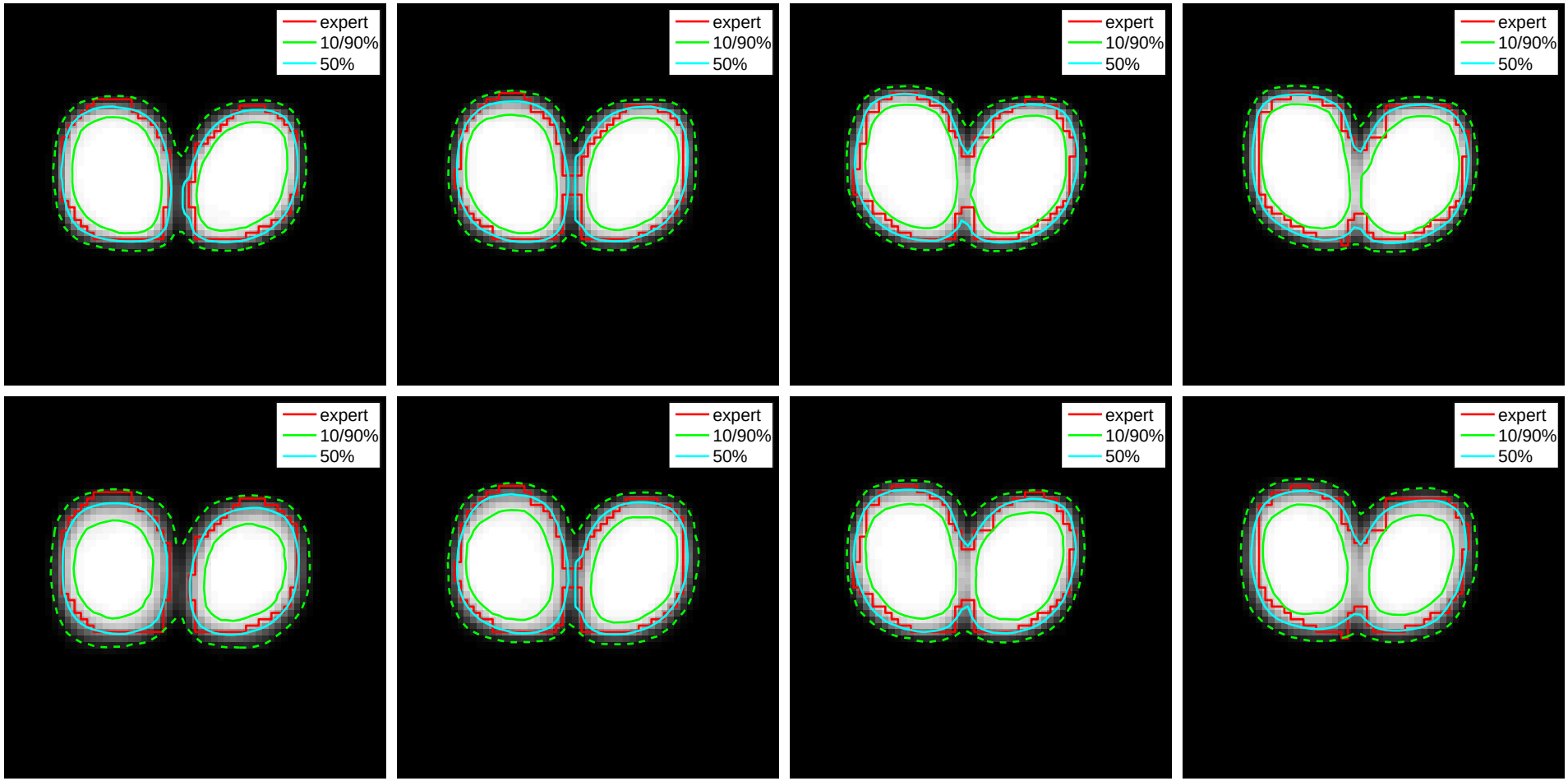


- If there are contiguous unknown slices, we need to be able to sample from the joint distribution over those slices (discontiguous groups of slices may still be processed independently).
- One option is to do Gibbs sampling and iteratively sample from $\mathrm{p}(\vec{c}_i|\vec{c}_{i-1}, \vec{c}_{i+1}, y_i)$ where i can be changed randomly or deterministically.
- We do not know how to sample from p directly. Instead, we can do N Metropolis-Hastings steps (same formulation as for the 2D case) and still have detailed balance hold.

Slices 13-16 Unknown

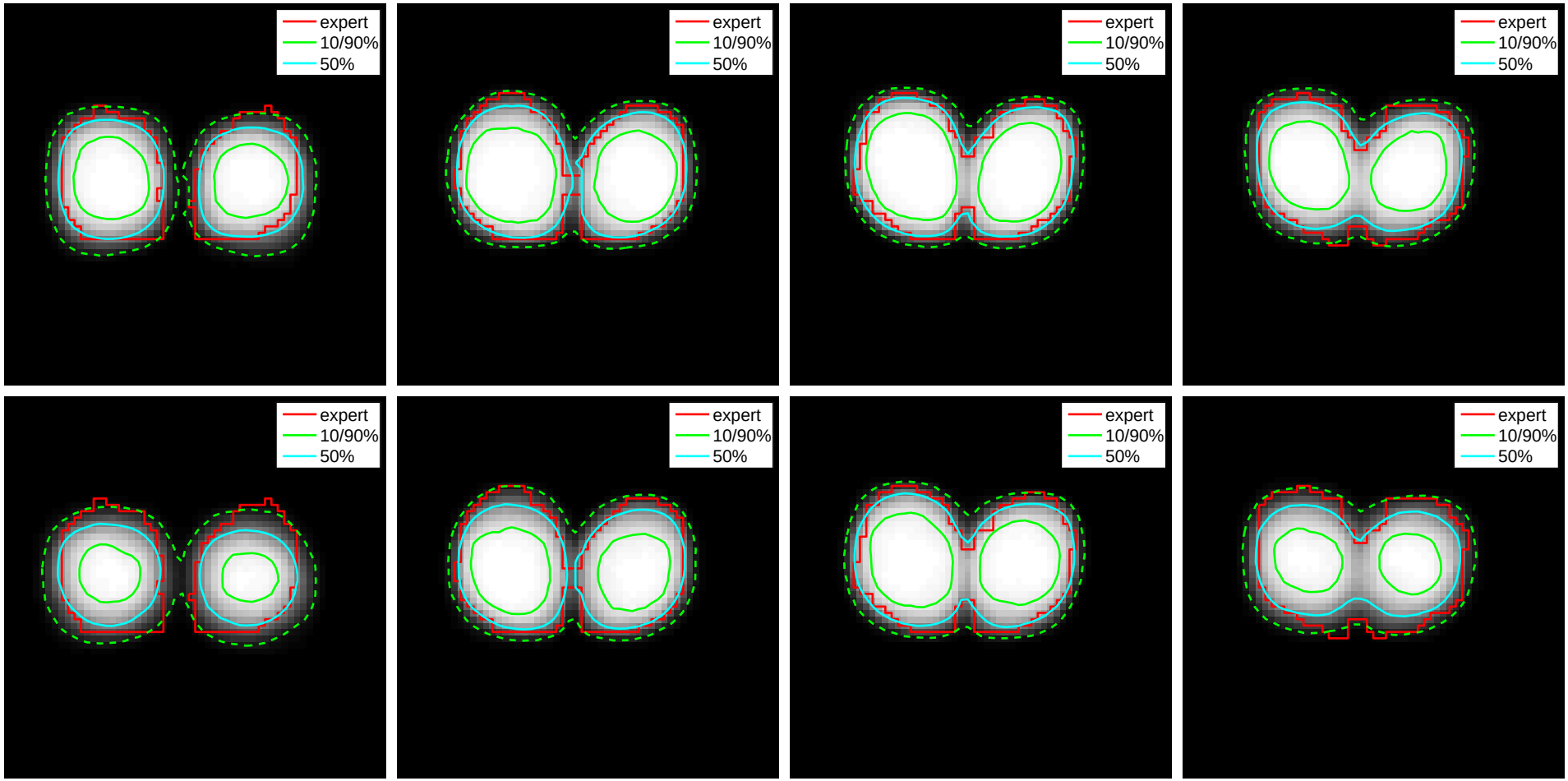


Variable Unknown Gaps (2 or 3)



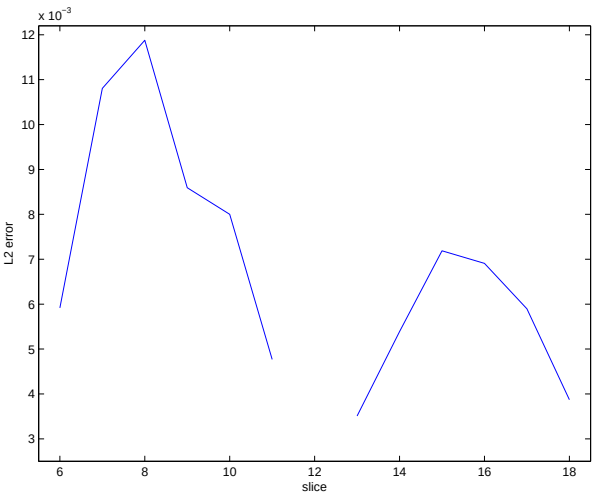
10 11 13 14

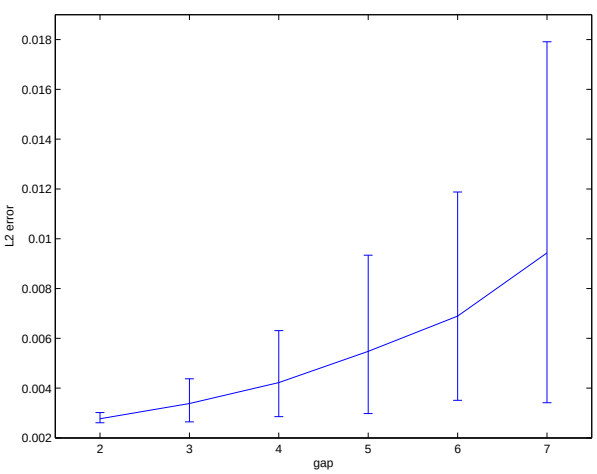
Variable Unknown Gaps (4 or 5)



9 11 13 15

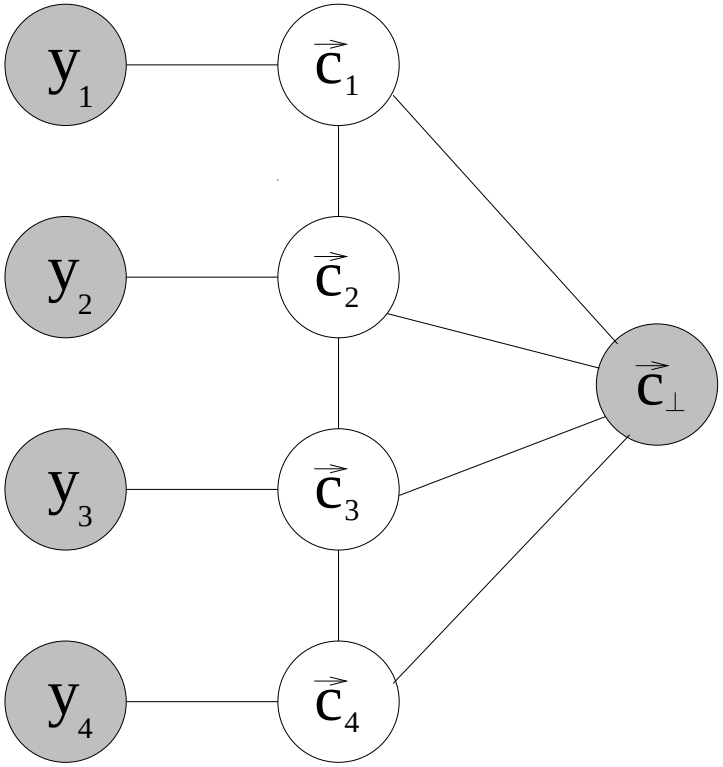
Error Per Slice





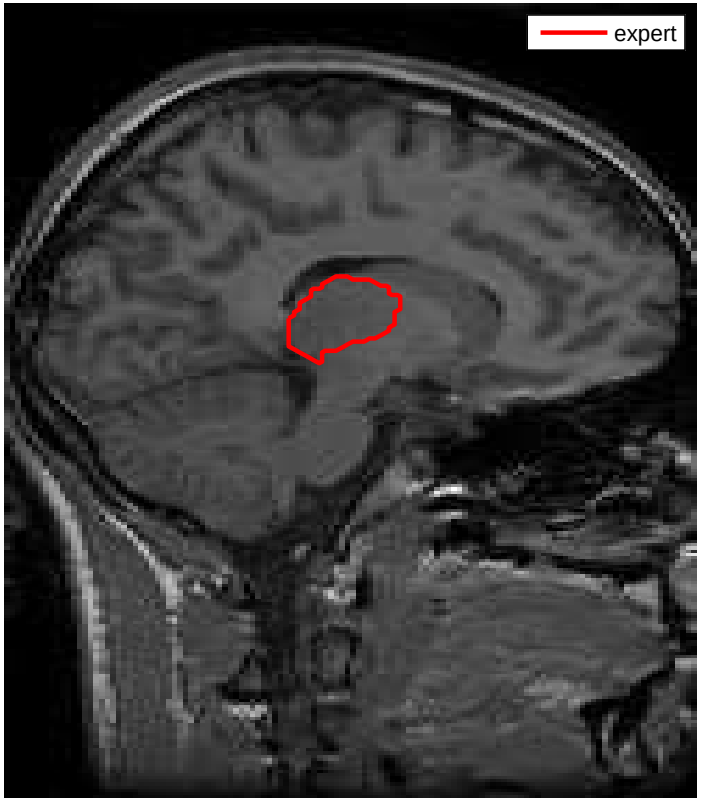
- We can use a number of methods to compare our sampling results with the expert segmentations:
  - Symmetric area difference (SAD) or Dice measure between median contour and expert contour.
  - L2 distance between histogram image and binary 0/1 expert label map.
- The upper-left figure shows the L2 error per slice for an example with a gap of 6.
- The lower-left figure shows average, minimum, and maximum L2 errors per slice for a varying gap size.

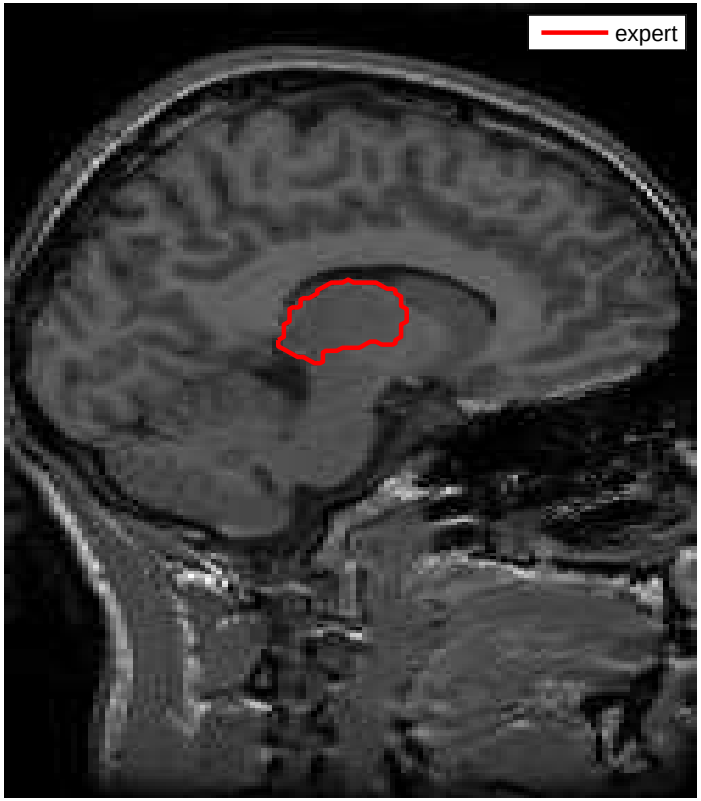
Orthogonal Slice Information



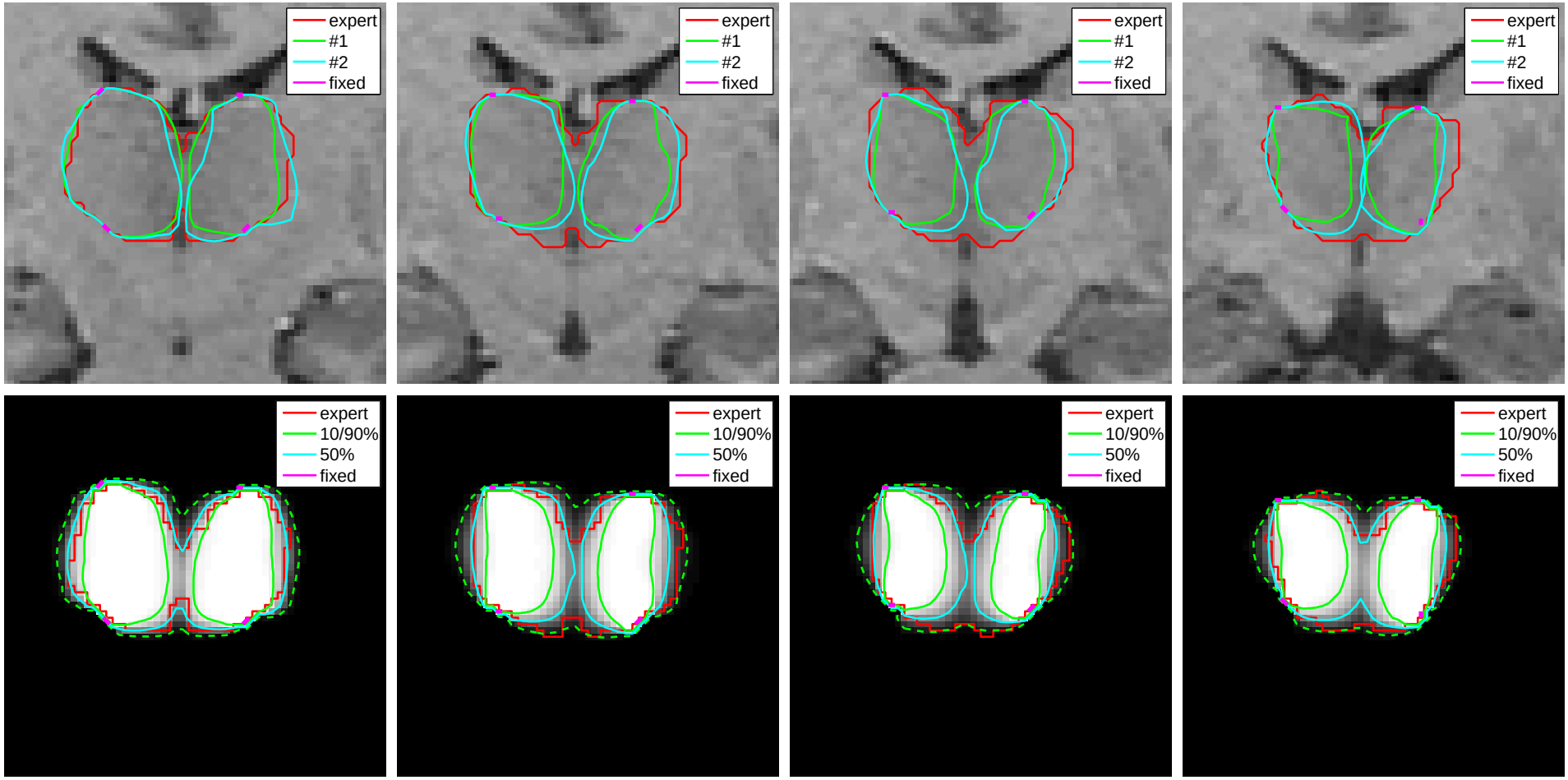
- An alternative model is to consider having slices oriented in orthogonal directions.
- If $\vec{c}_\perp$ is known and fixed, this is equivalent to fixing the segmentation values along a row or column in the original slices.
- Thus we can incorporate this information simply using our 2D conditional sampling framework.

Sagittal Expert Segmentations





Axial & Sagittal Constraints, Axial Slices



13 15 17 19

Summary

- We constructed a sampling method using Gaussian perturbations and showed how to maintain detailed balance.
- We demonstrated the benefits of sampling over standard optimization-based techniques on a number of examples.
  - Avoids local minima and handles multi-modal distributions.
  - Confidence bounds and PCA modes.
  - Do not need access to gradient of energy functional.
  - Robust to model error.
  - Conditional simulation and interactive segmentation.
- We extended our framework to a hybrid 2D/3D model to sample surfaces.

Future Work

- Sampling:
  - Faster numerical implementations.
  - Better perturbations (*e.g.*, multiresolution, feature-generating).
  - Jump-diffusion for topological change.
- Modeling:
  - Region-based user inputs.
  - Incorporate uncertainty into user information.
  - Time-based Markov chain problems.
- Visualization:
  - Interactive PCA mode exploration.
  - Conditional simulation using PCA approximations.
  - Manifold-based representations.

Gibbs Sampling

- MCMC method developed by Geman and Geman (1984). Most easily applied to models which have a Markov structure.
- Begin by dividing the variables into two subsets x_S and $x_{\setminus S}$. $x_{\setminus S}$ remains unchanged (so $y_{\setminus S} = x_{\setminus S}$).
- The proposal distribution $\mathtt{q}(y_S \,|\, x)$ is defined to be the conditional probability of y_S given the remaining variables: $\mathtt{q}(y_S \,|\, x) = \pi(y_S \,|\, x_{\setminus S})$. The resulting sample is always accepted, so $\mathtt{a}(y \,|\, x) = 1$.
- If the model is defined by a Markov graph structure, $\pi(y_S \,|\, x_{\setminus S}) = \pi(y_S \,|\, x_{\mathcal{N}(S)})$ where $\mathcal{N}(S)$ is the neighborhood of S.
- The subset S changes over time. This can be done randomly or according to some deterministic sequence.

Multi-modal Shape Model

- Here we construct a shape model using non-parametric Parzen density distributions. [Kim *et al.* 2007]
- We define the symmetric area difference (SAD) as:

$$\mathrm{d}_{\mathrm{SAD}}(\vec{C}_1, \vec{C}_2) = \iint\limits_{\mathcal{R}_{\vec{C}_1} \setminus \mathcal{R}_{\vec{C}_2}} \mathrm{d}\boldsymbol{x} \quad + \iint\limits_{\mathcal{R}_{\vec{C}_2} \setminus \mathcal{R}_{\vec{C}_1}} \mathrm{d}\boldsymbol{x} \ .$$

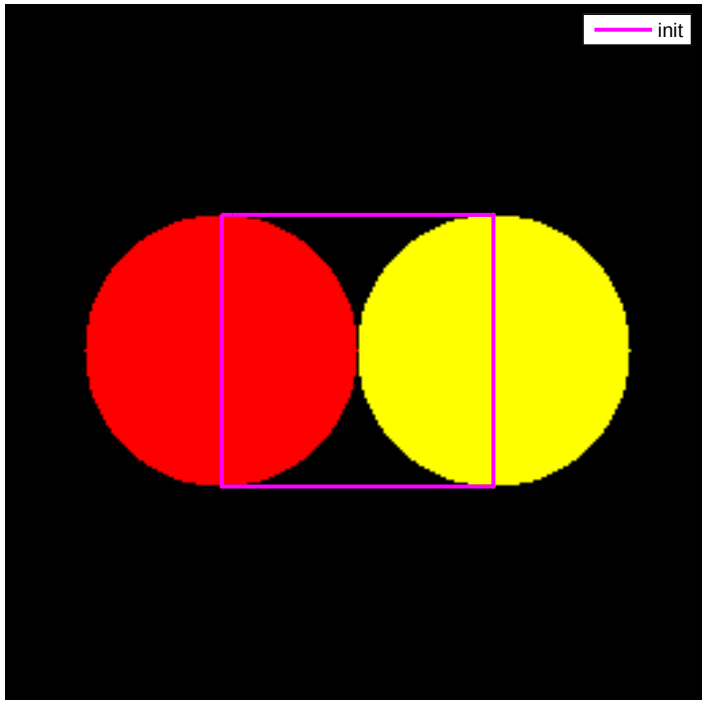
- A Parzen density is constructed from exemplars $\{\vec{C}_i\}$ and a kernel function K:

$$\hat{\mathrm{p}}(\vec{C}) \propto \frac{1}{N} \sum_{i=1}^{N} \mathrm{K}(\vec{C}, \vec{C}_i) \ .$$

- We construct our kernel function as exponentiated negative distance:

$$\mathrm{K}(\vec{C}, \vec{C}_i) = \exp(-\beta \, \mathrm{d}_{\mathrm{SAD}}(\vec{C}, \vec{C}_i)) \ .$$

Two Target Curves

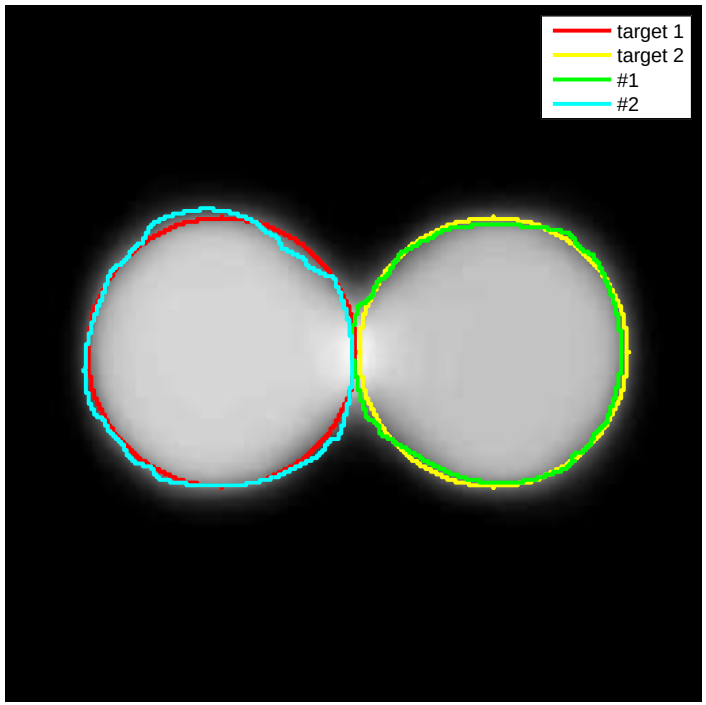


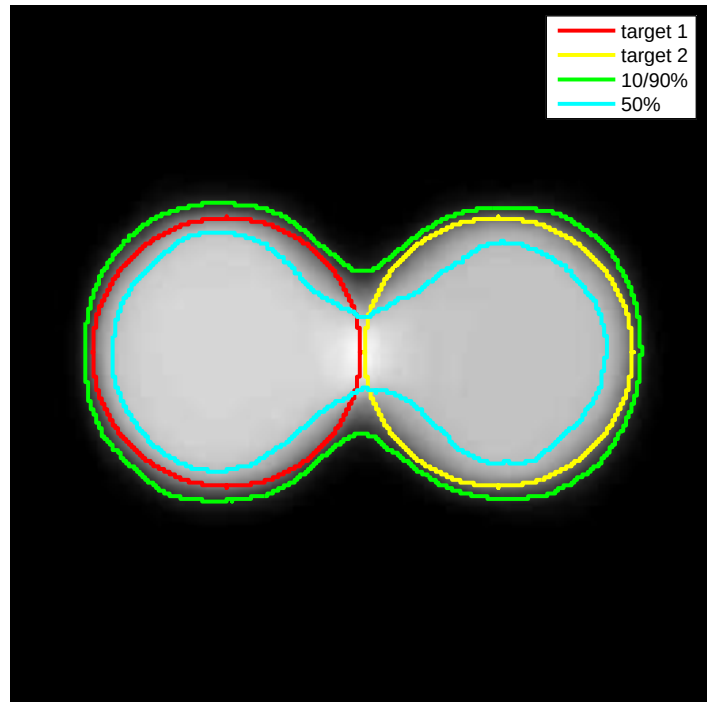
- We construct a target distribution using the shape model and a regularizing term:

$$\pi(\vec{C};\, \{\vec{C}_i\}_{i=1}^{M}) \propto \left(\sum_{i=1}^{M} e^{-\beta\, \mathrm{d}_{\mathrm{SAD}}(\vec{C},\vec{C}_i)} \right) e^{(-\alpha \oint_{\vec{C}} \mathrm{d}s)} \quad .$$

- In this example, we have two target curves $\vec{C}_1$ and $\vec{C}_2$ which are circles horizontally offset from each other.

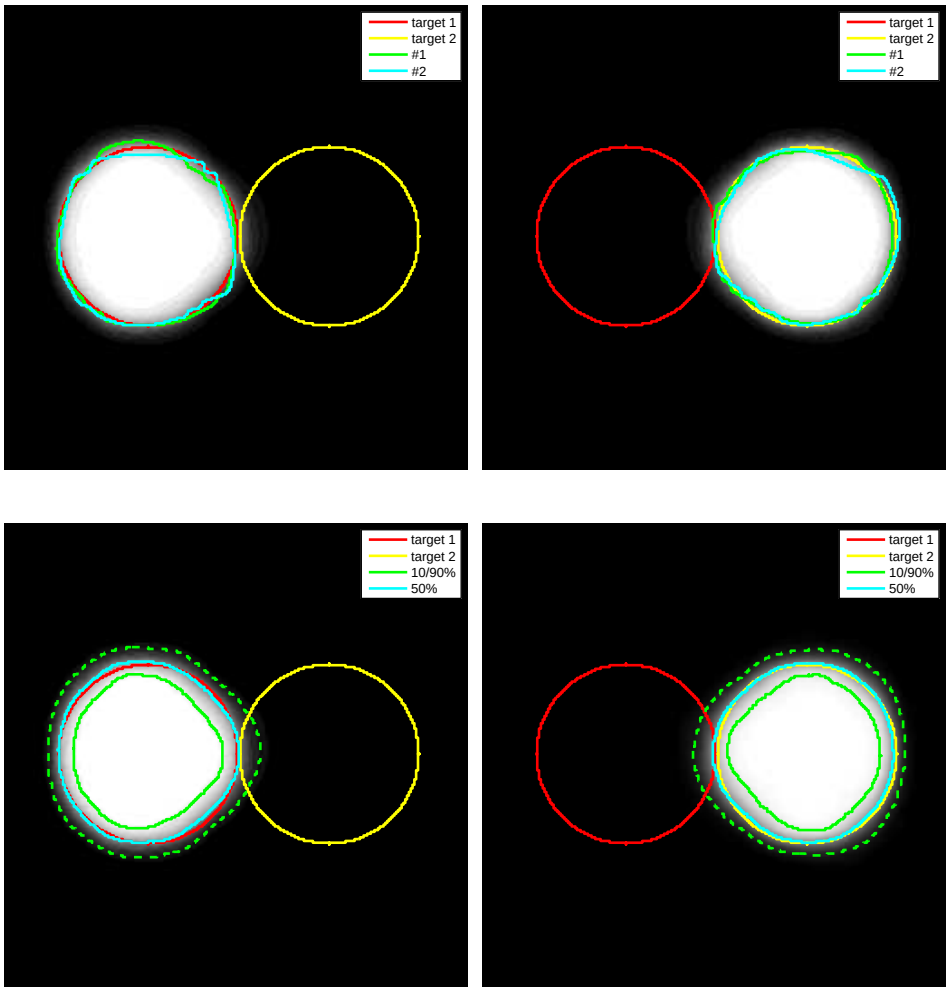
Sampling Results





- Most probable samples find each mode.
- Marginal confidence bounds are not as informative due to multi-modality (analogous to mean of Gaussian mixture).

Clustering the Samples



- The samples from the two modes are very different, so they are easy to cluster using a variety of techniques.
- Here we cluster the samples into groups corresponding to the left and right target curves.
- We display the resulting most probable samples, histogram images, and marginal confidence bounds for each cluster.