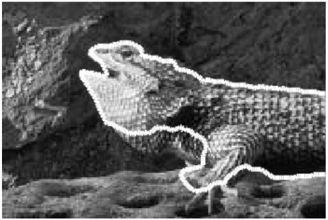
A Graph Cut Approach to Image Segmentation in Tensor Space

James Malcolm Yogesh Rathi Allen Tannenbaum

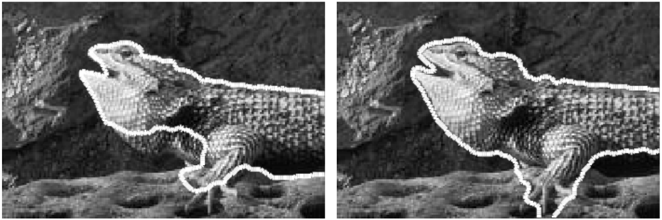
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, Georgia
malcolm@gatech.edu

The standard graph cut image segmentation technique naturally extends to richer tensor feature spaces for better segmentations than simply gray-scale intensity or color.

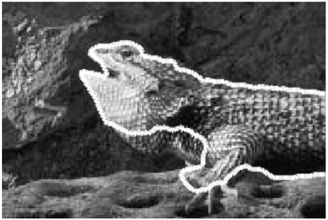
The standard graph cut image segmentation technique naturally extends to richer tensor feature spaces for better segmentations than simply gray-scale intensity or color.

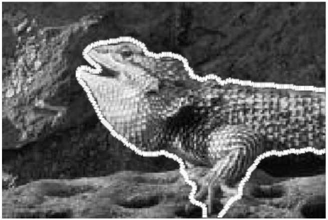


The standard graph cut image segmentation technique naturally extends to richer tensor feature spaces for better segmentations than simply gray-scale intensity or color.



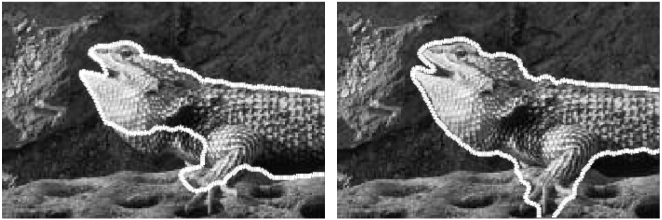
The standard graph cut image segmentation technique naturally extends to richer tensor feature spaces for better segmentations than simply gray-scale intensity or color.





- Tensor feature space and structure

The standard graph cut image segmentation technique naturally extends to richer tensor feature spaces for better segmentations than simply gray-scale intensity or color.



- Tensor feature space and structure
- Graph cut segmentation

Outline

So what are tensors?

So what are tensors?

- Symmetric positive-definite matrices

So what are tensors?

- Symmetric positive-definite matrices
- Capture structural information

Examples:

Examples:

- Structure tensor from smoothed image derivatives

$$\mathbf{T} = K_\rho * [\mathcal{I}_x \ \mathcal{I}_y]^T [\mathcal{I}_x \ \mathcal{I}_y] = \begin{pmatrix} K_\rho * \mathcal{I}_x^2 & K_\rho * \mathcal{I}_x \mathcal{I}_y \\ K_\rho * \mathcal{I}_x \mathcal{I}_y & K_\rho * \mathcal{I}_y^2 \end{pmatrix}$$

Examples:

- Structure tensor from smoothed image derivatives

$$\mathbf{T} = K_\rho * [\mathcal{I}_x \; \mathcal{I}_y]^T [\mathcal{I}_x \; \mathcal{I}_y] = \begin{pmatrix} K_\rho * \mathcal{I}_x^2 & K_\rho * \mathcal{I}_x \mathcal{I}_y \\ K_\rho * \mathcal{I}_x \mathcal{I}_y & K_\rho * \mathcal{I}_y^2 \end{pmatrix}$$

- Structure tensor across color channels

$$\mathbf{T} = \sum_{i=1}^{N} \left(K_\rho * [\mathcal{I}_x^i \; \mathcal{I}_y^i]^T [\mathcal{I}_x^i \; \mathcal{I}_y^i] \right)$$

Examples:

- Structure tensor from smoothed image derivatives

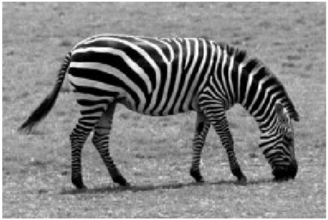
$$\mathbf{T} = K_\rho * [\mathcal{I}_x \ \mathcal{I}_y]^T [\mathcal{I}_x \ \mathcal{I}_y] = \begin{pmatrix} K_\rho * \mathcal{I}_x^2 & K_\rho * \mathcal{I}_x \mathcal{I}_y \\ K_\rho * \mathcal{I}_x \mathcal{I}_y & K_\rho * \mathcal{I}_y^2 \end{pmatrix}$$

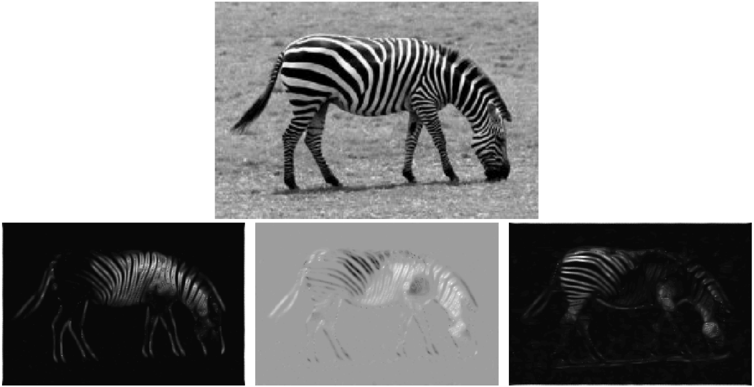
- Structure tensor across color channels

$$\mathbf{T} = \sum_{i=1}^{N} \left(K_\rho * [\mathcal{I}_x^i \ \mathcal{I}_y^i]^T [\mathcal{I}_x^i \ \mathcal{I}_y^i] \right)$$

- Incorporating intensity

$$\mathbf{T} = K_\rho * [\mathcal{I} \ \mathcal{I}_x \ \mathcal{I}_y]^T [\mathcal{I} \ \mathcal{I}_x \ \mathcal{I}_y].$$

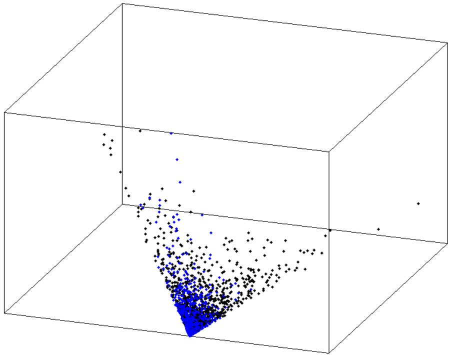


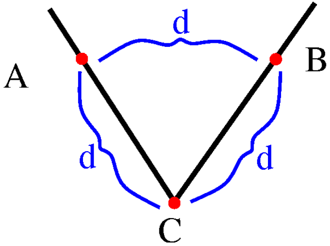


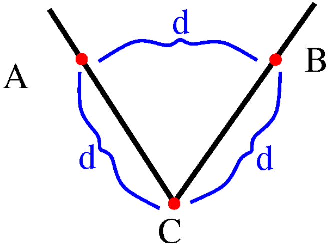
$$\mathcal{I}_x^2 \quad \mathcal{I}_x\mathcal{I}_y \quad \mathcal{I}_y^2$$

Outline

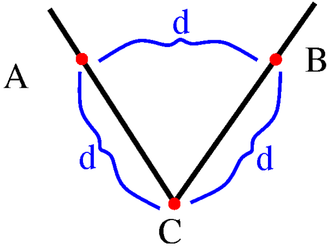
Conical surface structure



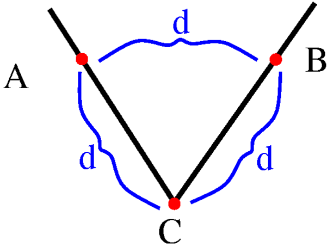

d
A
B
d
d
C



$$d(A, B) =$$



$$d(A, B) = d(A, C) + d(C, B) = 2d$$



$$d(A,B) = d(A,C) + d(C,B) = 2d \neq d$$

Why do we care?

Why do we care?

- Non-parametric estimates of $P(\mathcal{I}|\mathcal{F})$ and $P(\mathcal{I}|\mathcal{B})$

Why do we care?

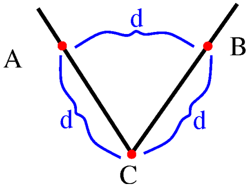
- Non-parametric estimates of $P(\mathcal{I}|\mathcal{F})$ and $P(\mathcal{I}|\mathcal{B})$
- Fast Gauss Transform uses Euclidean L_2 norm

Why do we care?

- Non-parametric estimates of $P(\mathcal{I}|\mathcal{F})$ and $P(\mathcal{I}|\mathcal{B})$
- Fast Gauss Transform uses Euclidean L_2 norm

Why do we care?

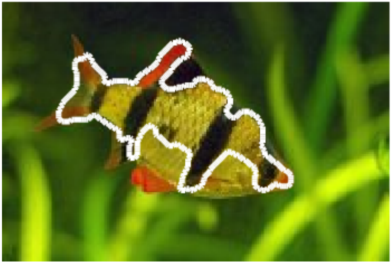
- Non-parametric estimates of $P(\mathcal{I}|\mathcal{F})$ and $P(\mathcal{I}|\mathcal{B})$
- Fast Gauss Transform uses Euclidean L_2 norm



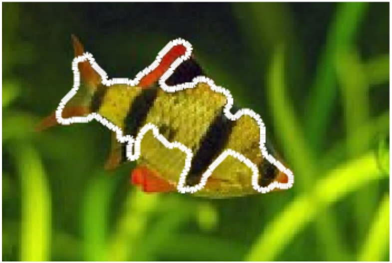
d(A,B) = d

What do we get?

What do we get?



What do we get?

Problem:

Problem:

- Want to use Fast Gauss Transform

Problem:

- Want to use Fast Gauss Transform
- Tensor norms computationally intensive

Problem:

- Want to use Fast Gauss Transform
- Tensor norms computationally intensive

Problem:

- Want to use Fast Gauss Transform
- Tensor norms computationally intensive

Solution:

Problem:

- Want to use Fast Gauss Transform
- Tensor norms computationally intensive

Solution: Map to Euclidean space

Problem:

- Want to use Fast Gauss Transform
- Tensor norms computationally intensive

Solution: Map to Euclidean space

Problem:

- Want to use Fast Gauss Transform
- Tensor norms computationally intensive

Solution: Map to Euclidean space

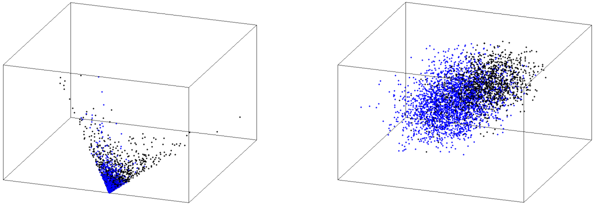
- Log map onto tangent space

Problem:

- Want to use Fast Gauss Transform
- Tensor norms computationally intensive

Solution: Map to Euclidean space

- Log map onto tangent space



Outline

Standard binary graph cut energy formulation:

$$E(A) = \sum_{p \in \mathcal{I}} R_p(A_p) + \lambda \sum_{(p,q) \in \mathcal{N}} B_{(p,q)}$$

Standard binary graph cut energy formulation:

$$E(A) = \sum_{p \in \mathcal{I}} R_p(A_p) + \lambda \sum_{(p,q) \in \mathcal{N}} B_{(p,q)}$$

$$R_p(\mathcal{F}) = -\ln P(\mathcal{I}_p|\mathcal{F}) \qquad R_p(\mathcal{B}) = -\ln P(\mathcal{I}_p|\mathcal{B})$$

Standard binary graph cut energy formulation:

$$E(A) = \sum_{p \in \mathcal{I}} R_p(A_p) + \lambda \sum_{(p,q) \in \mathcal{N}} B_{(p,q)}$$

$$R_p(\mathcal{F}) = -\ln P(\mathcal{I}_p | \mathcal{F}) \qquad R_p(\mathcal{B}) = -\ln P(\mathcal{I}_p | \mathcal{B})$$

$$B_{(p,q)} = \exp\left(\frac{-\|\mathcal{I}_p - \mathcal{I}_q\|^2}{2\sigma^2}\right) \frac{1}{\|p-q\|}$$

Tensor version:

$$E(A) = \sum_{p \in \mathcal{I}} R_p(A_p) + \lambda \sum_{(p,q) \in \mathcal{N}} B_{(p,q)}$$

$$R_p(\mathcal{F}) = -\ln P(\mathbf{T}_p | \mathcal{F}) \qquad R_p(\mathcal{B}) = -\ln P(\mathbf{T}_p | \mathcal{B})$$

$$B_{(p,q)} = \exp\left(\frac{-\|\mathbf{T}_p - \mathbf{T}_q\|^2}{2\sigma^2}\right) \frac{1}{\|p-q\|}$$

Outline

1. Form structure tensor

1. Form structure tensor
2. Map to Euclidean tangent plane

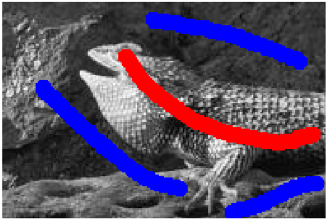
1. Form structure tensor
2. Map to Euclidean tangent plane
3. Estimate $P(\mathbf{T}|\mathcal{F})$ and $P(\mathbf{T}|\mathcal{B})$ from initialization

1. Form structure tensor
2. Map to Euclidean tangent plane
3. Estimate $P(\mathbf{T}|\mathcal{F})$ and $P(\mathbf{T}|\mathcal{B})$ from initialization
4. Graph cut segmentation

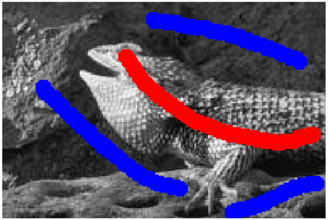
Outline

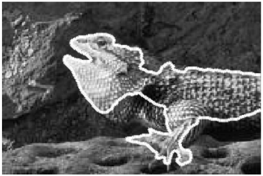
Intensity alone

Intensity alone

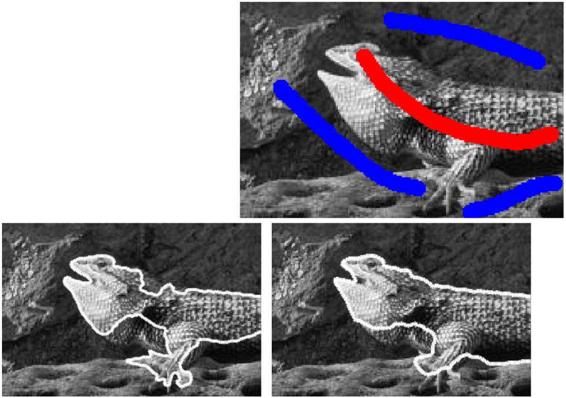


Intensity alone

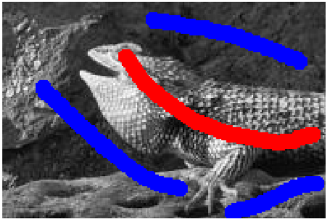


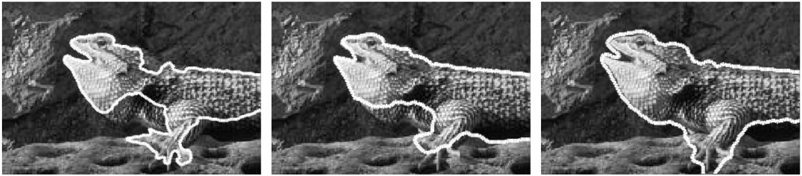


Intensity alone

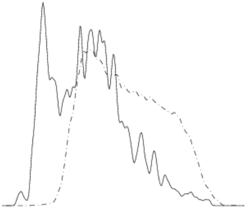


Intensity alone

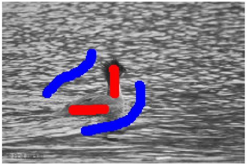


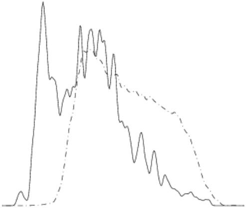


Intensity alone

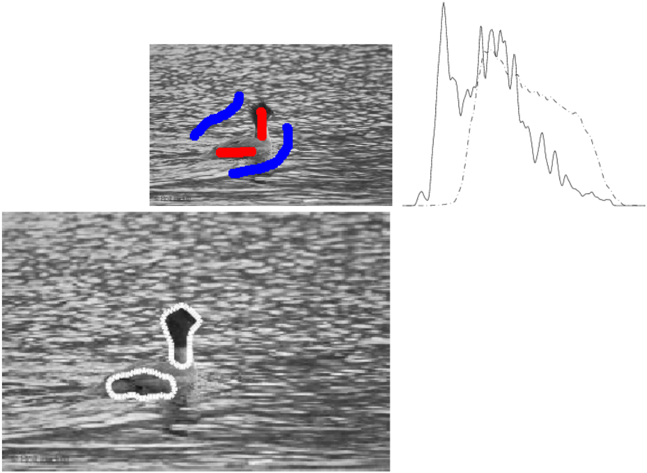


Intensity alone

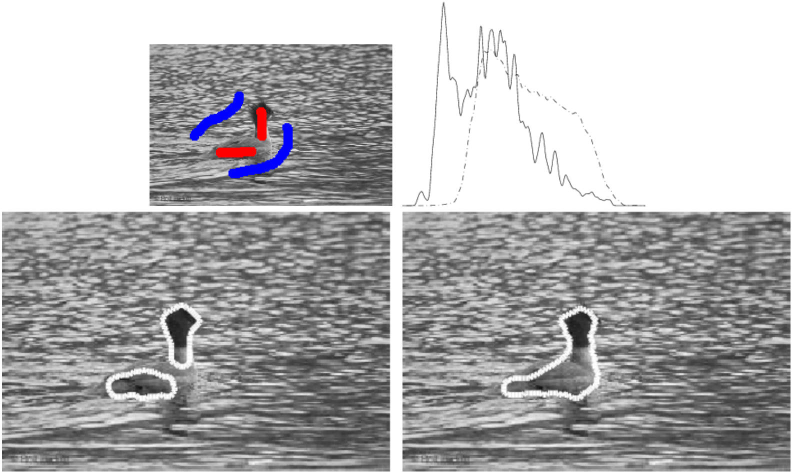




Intensity alone

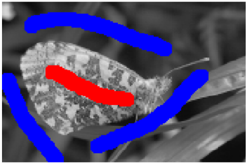


Intensity alone

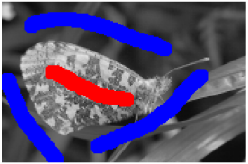


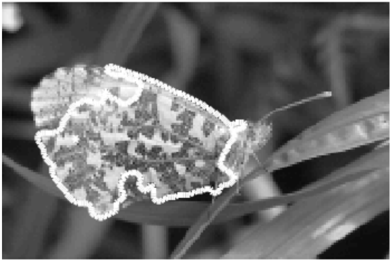
Ignoring conical structure

Ignoring conical structure

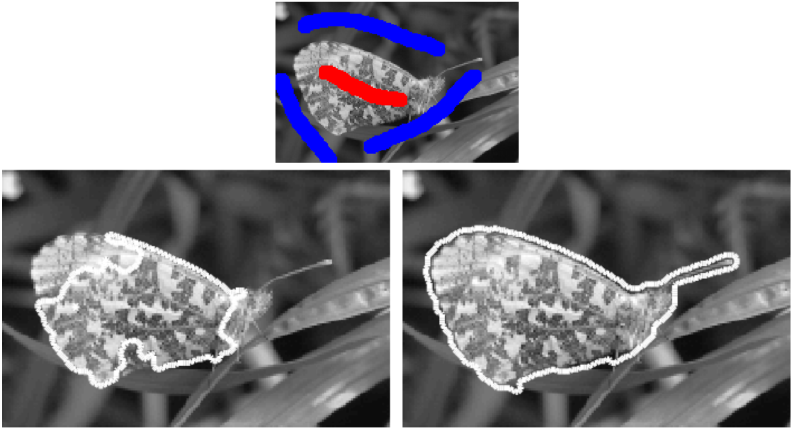


Ignoring conical structure



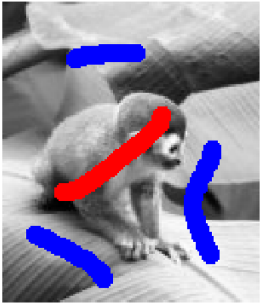


Ignoring conical structure

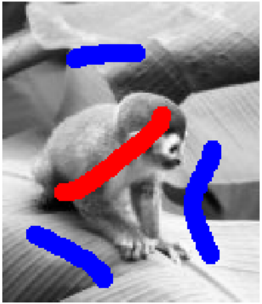


Ignoring conical structure

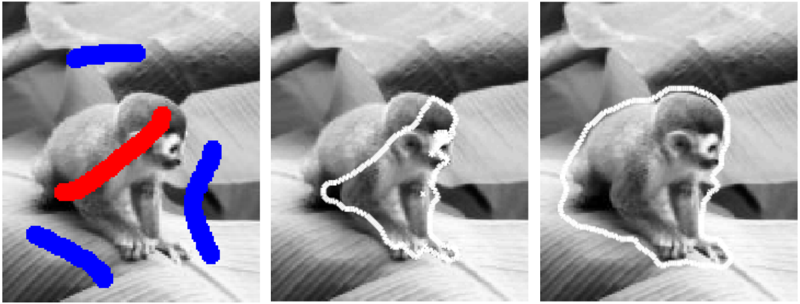
Ignoring conical structure



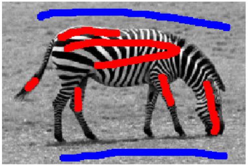
Ignoring conical structure



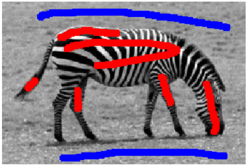
Ignoring conical structure

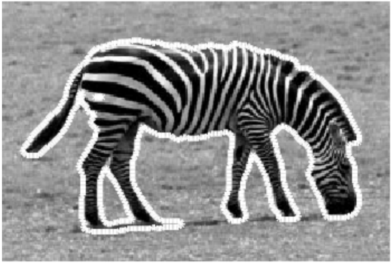


Validation, incorporate intensity

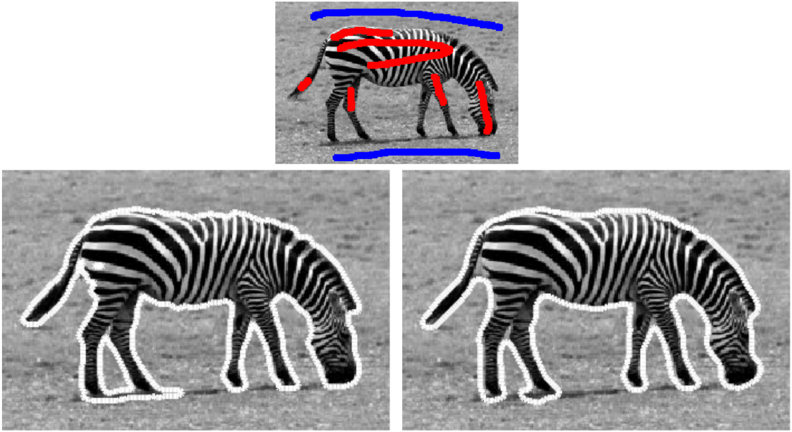


Validation, incorporate intensity

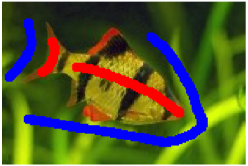




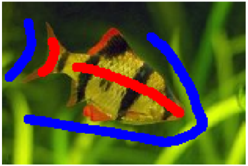
Validation, incorporate intensity

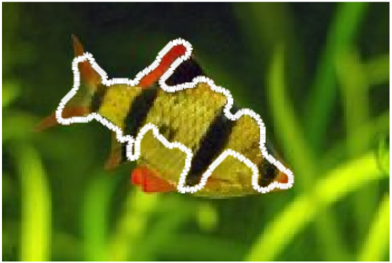


Using color

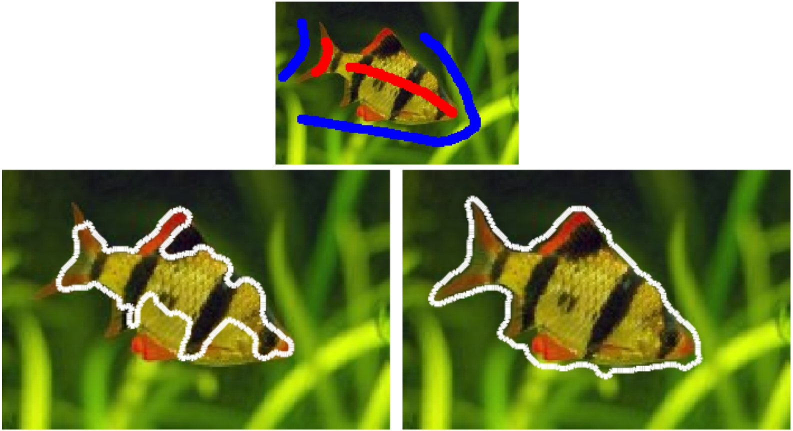


Using color





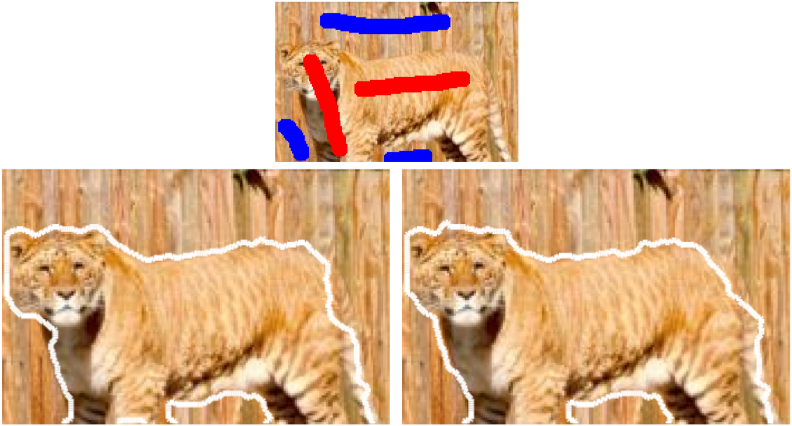
Using color



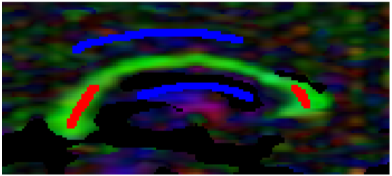
Map to identity (Log-Euclidean space)

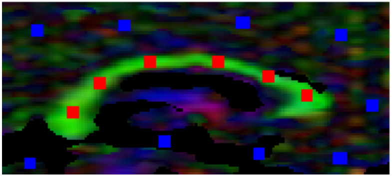
Map to identity (Log-Euclidean space)

Map to identity (Log-Euclidean space)

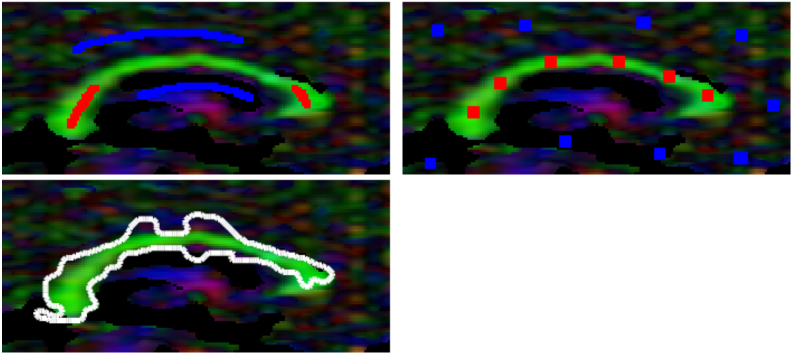


Diffusion tensor MRI

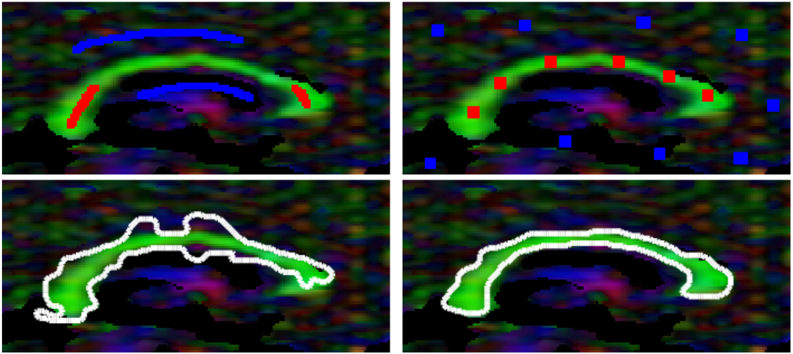




Diffusion tensor MRI



Diffusion tensor MRI



- Tensor space

- Tensor space
- Riemannian structure

- Tensor space
- Riemannian structure
- Map to Euclidean tangent space

- Tensor space
- Riemannian structure
- Map to Euclidean tangent space
- Estimate distributions

- Tensor space
- Riemannian structure
- Map to Euclidean tangent space
- Estimate distributions
- Graph cut segmentation

Questions?