

Automatic and Interactive Connectome Reconstruction by Finding an Adaptive Waterlevel in Watershed

Chao Chen

Computational Biomedicine Imaging and Modeling Center (CBIM),
Rutgers University

Sept. 2014

Outline

1 Optree

with Uzunbas and Metaxas MICCAI 2014

- Automatic
- Interactive

2 Potential Ideas

- Multiple Proposals
- Topological Priors

Outline

1 Optree

with Uzunbas and Metaxas MICCAI 2014

- Automatic
- Interactive

2 Potential Ideas

- Multiple Proposals
- Topological Priors

Outline

1 Optree

with Uzunbas and Metaxas MICCAI 2014

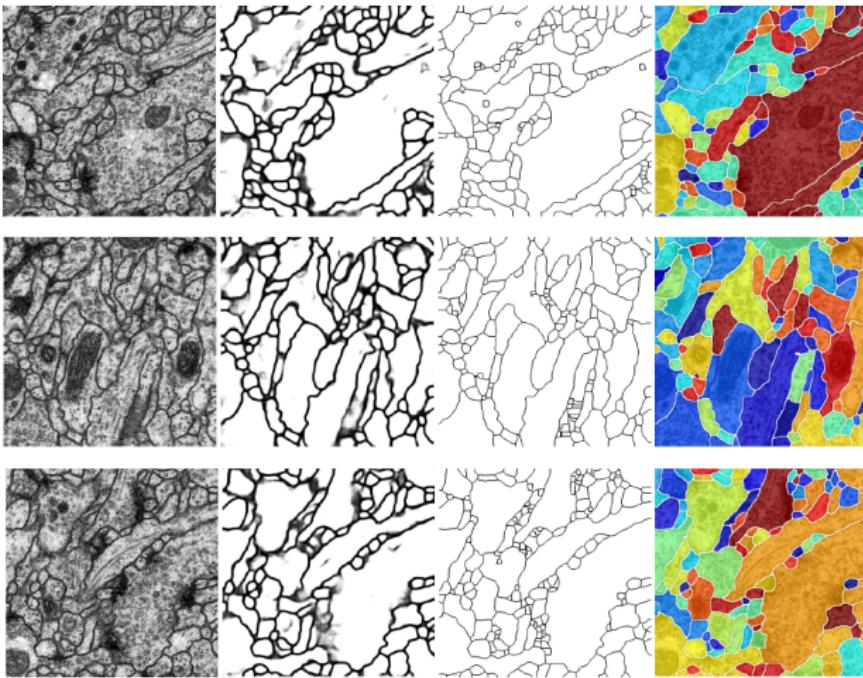
- Automatic
- Interactive

2 Potential Ideas

- Multiple Proposals
- Topological Priors

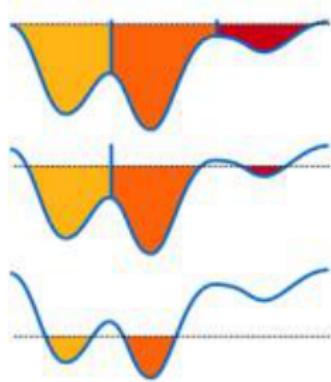
Problem

- Input: 2D or 3D EM images; boundary likelihood map
- Output: partitioning of the image



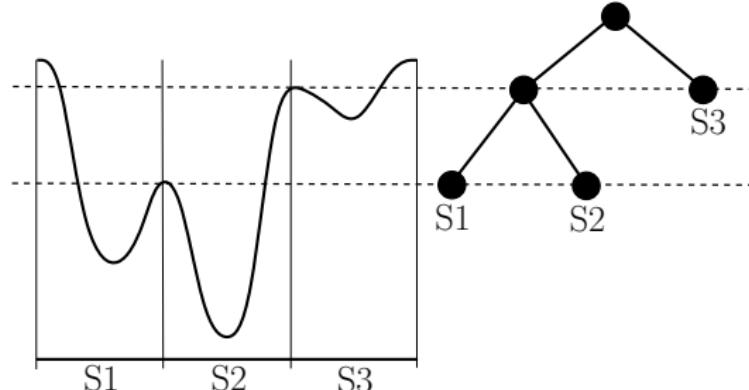
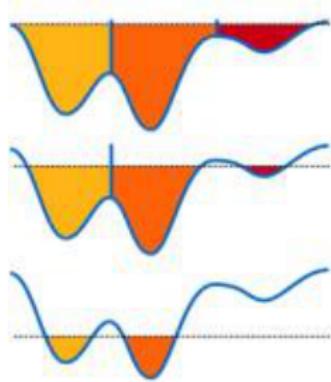
Watershed

- Watersheds Merging Tree



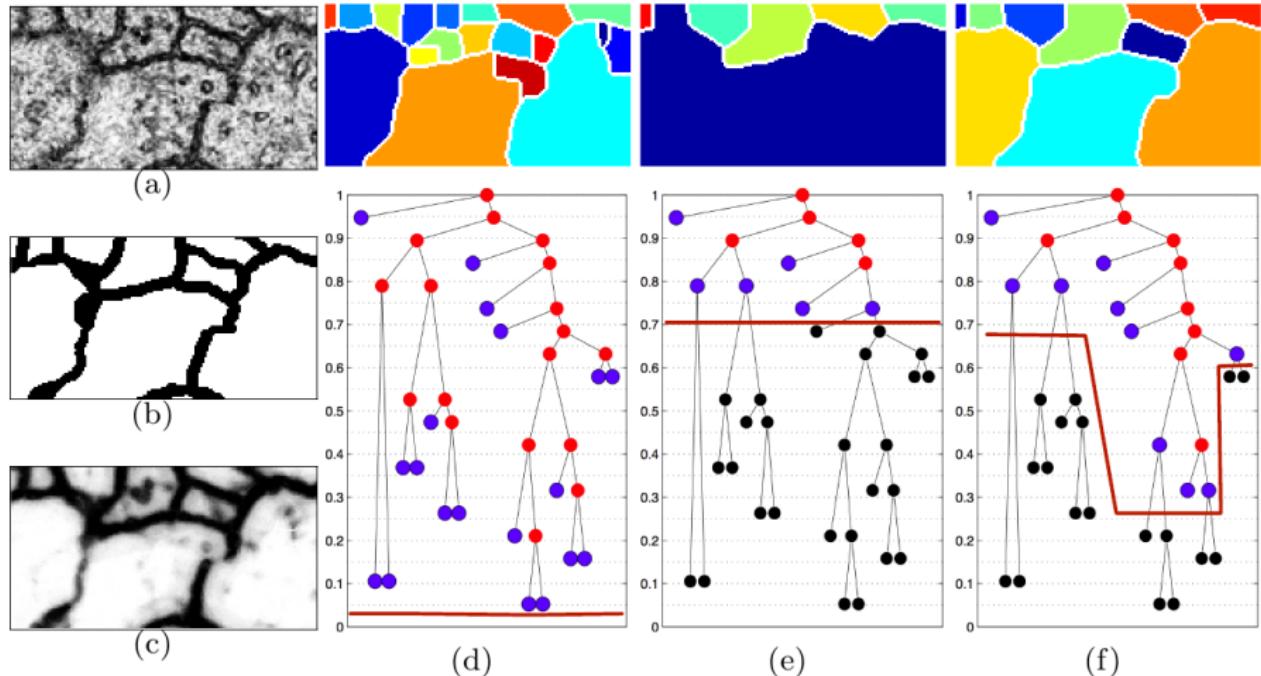
Watershed

- Watersheds Merging Tree



Picture from <http://www.svi.nl/watershed>

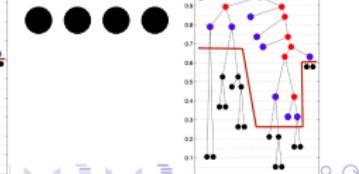
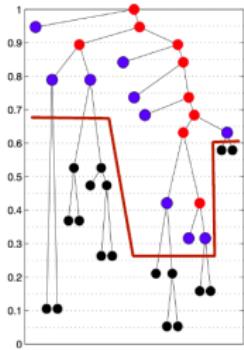
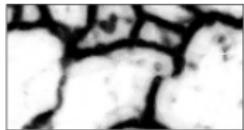
Goal: Adaptive Watersheds



White low, black high

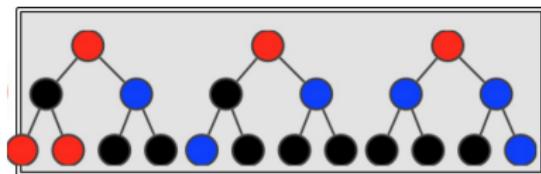
Tool: Conditional Random Field

- Given a graph $(\mathcal{V}, \mathcal{E})$
- A **labeling** assigns a label to each node, $\{1..L\}$
- CRF: models a **conditional** distribution over the space of all labelings, $P(\ell | I, w)$, $\ell \in \mathcal{L} = \{1..L\}^{|\mathcal{V}|}$
- Inference (prediction):
 - Maximum a posteriori (MAP),
 $\text{argmax}_{\ell \in \mathcal{L}} P(\ell | I, w)$
 - Marginals, $P(\text{Node } i = \text{Blue} | I, w)$
- Training:
 - Training Data $\mathcal{D} = \{(I, \ell)\}$
 - find w using maximum-likelihood estimation
$$w^* = \text{argmax}_w \prod_{(I, \ell) \in \mathcal{D}} P(\ell | I, w)$$

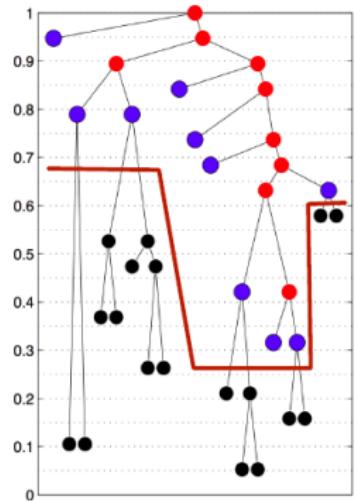


Implementation

- 3-Labeled Labelings of a tree
- $\mathcal{L} = \{\text{Legal labelings (valid partition)}\}$
 - For any path from a leaf to the root, Black, \dots , Black, Blue, Red, \dots , Red



- Testing: legal labeling \rightarrow partition
- Training: partition \rightarrow labeling ?



Why Optree

Previous Methods

- Learn a classifier to prune a tree [Liu *et al.* ICPR 2012]
- Learn to build merging trees [Jain *et al.* NIPS 2011]
- Graphical model on bottom level superpixels
[Andres *et al.* ECCV 2012]

Why Optree

Previous Methods

- Learn a classifier to prune a tree [Liu *et al.* ICPR 2012]
- Learn to build merging trees [Jain *et al.* NIPS 2011]
- Graphical model on bottom level superpixels
[Andres *et al.* ECCV 2012]

Our Advantage

- CRF provides a solid statistical model
less parameters to tune; easy to extend
- Tree model is efficient
- Nodes in the tree: features at different scales

Results

	Optree	Watershed	Optimal		training	testing
2D (RI)	0.023	0.050	0.015	2D (sec)	167	0.05
3D (RI)	0.0163		0.0077	3D (sec)	465	0.9

- **Optimal**

best labeling by enumeration and comparison to the groundtruth

- **2D (ISBI 2012 Challenge, ssTEM)**

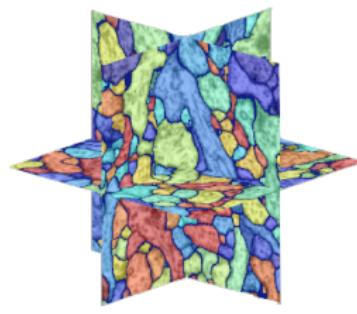
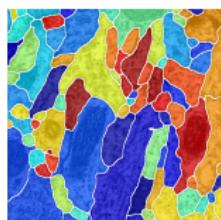
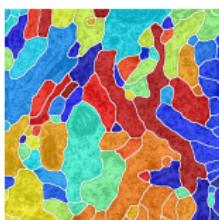
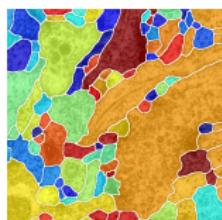
Among automatic methods: Rank 1 at submission, Rank 2 now

30 training, 30 testing, 512 by 512 pixels

- **3D (Drosophila Neuropil FIBSEM, Open Connectome)**

500 by 500 by 500, divide into 8 250 × 250 × 250, leave-one-out test;

Likelihood map from Ilastik



Outline

1 Optree

with Uzunbas and Metaxas MICCAI 2014

- Automatic
- Interactive

2 Potential Ideas

- Multiple Proposals
- Topological Priors

Problem & Solution

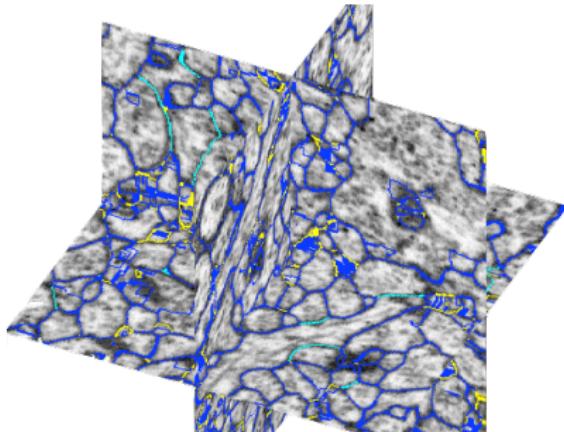
- Problem: proofreading
 - Huge data, labeling is difficult
 - Crowdsourcing (EyeWire), non-expert
- Smarter UI?

Basic Principle

Minimal User Effort for Maximal Model Improvement

- Focus on locations where the model has the least confidence
- Improve the model globally after each user click (tree modification)
Yellow: labeled split; Blue: labeled merge.
- Fast due to tree structure (2 seconds per click, matlab)

Play Demo



Basic Principle

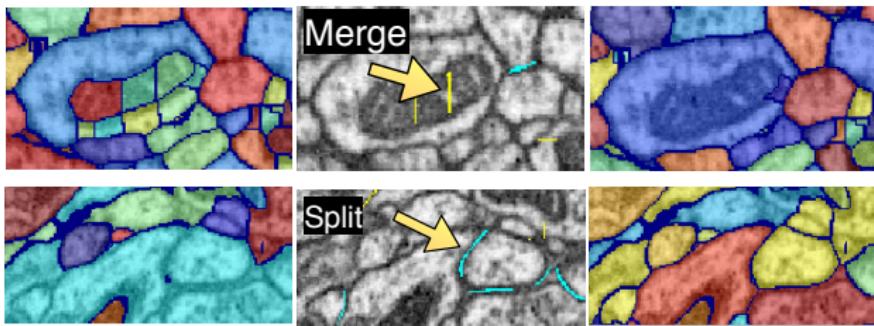
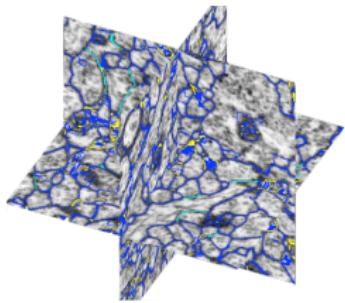
Minimal User Effort for Maximal Model Improvement

- Focus on locations where the model has the least confidence
- Improve the model globally after each user click (tree modification)

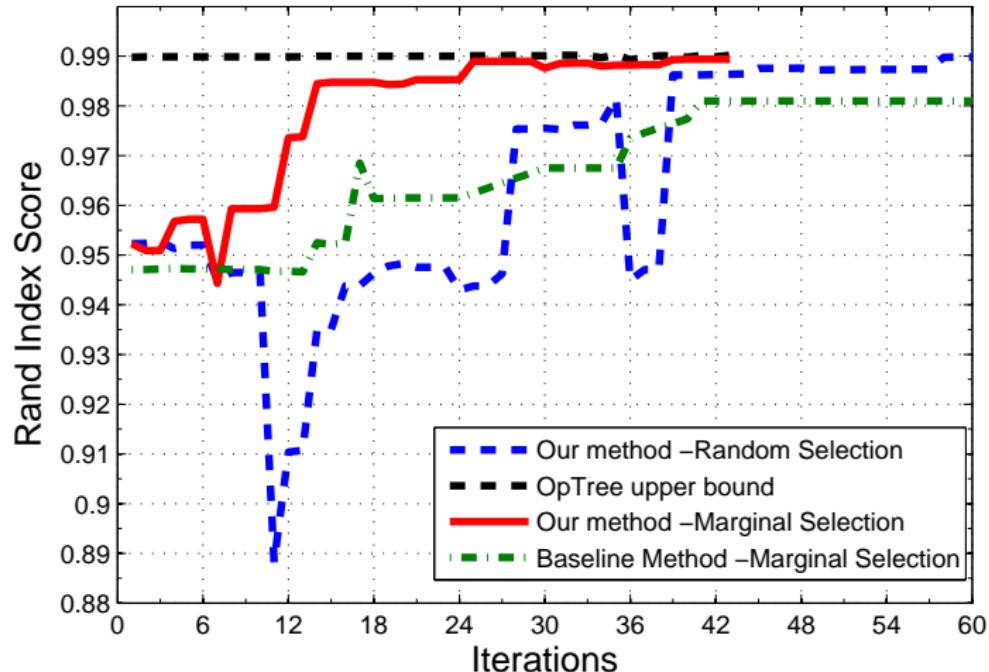
Yellow: labeled split; Blue: labeled merge.

- Fast due to tree structure (2 seconds per click, matlab)

Play Demo

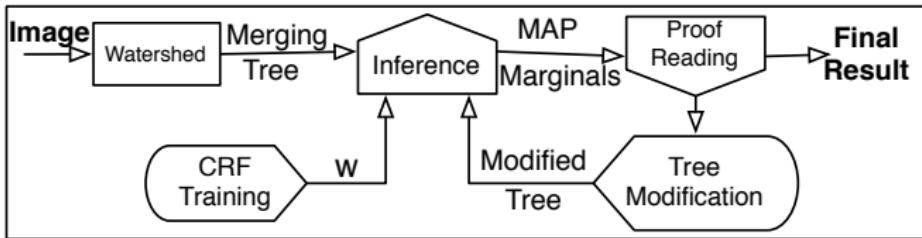


Results

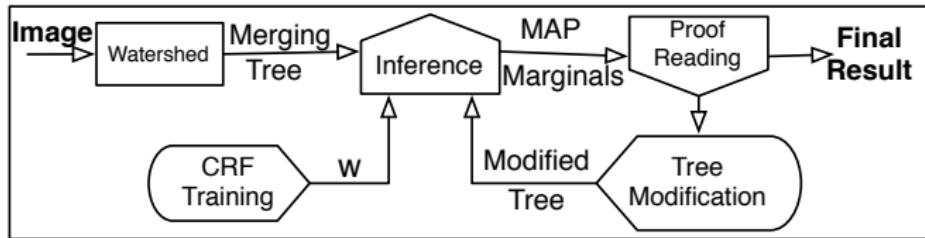


Black	Red	Blue	Green
Upperbound	Optree	No Marginal Highlight	No Global Improvement

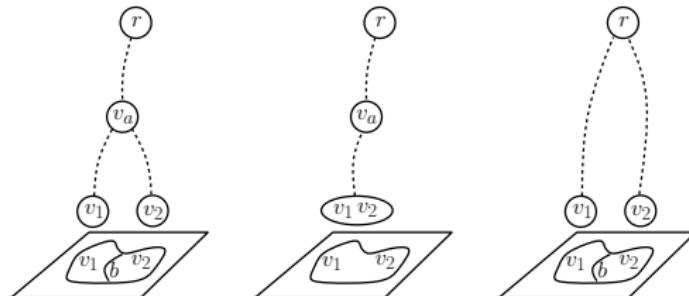
More Details



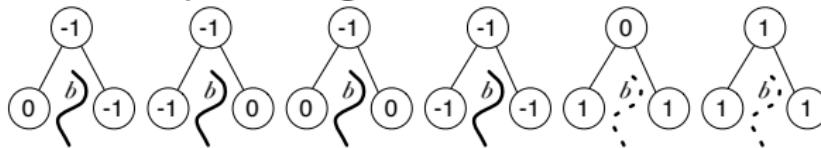
More Details



- Tree modification



- Focus: compute marginals



Open Questions

- Improve the model at each click (weights w)
- Adaptive to the domain
- Inconsistency (crowdsourcing)

Outline

1 Optree

with Uzunbas and Metaxas MICCAI 2014

- Automatic
- Interactive

2 Potential Ideas

- Multiple Proposals
- Topological Priors

Outline

1 Optree

with Uzunbas and Metaxas MICCAI 2014

- Automatic
- Interactive

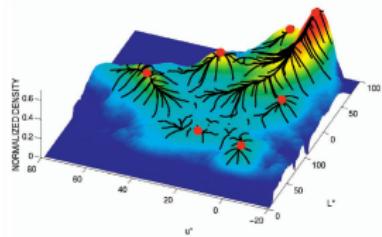
2 Potential Ideas

- Multiple Proposals
- Topological Priors

Multiple Predictions (AISTATS 2013; NIPS 2014)

- Multiple Predictions (Modes)

- High Probability
- Diverse



[Comaniciu and Meer, PAMI 2002]

Image Partitioning Task (Berkeley Dataset)

Ground Truth



1st Mode



2nd Mode



3rd Mode



Outline

1 Optree

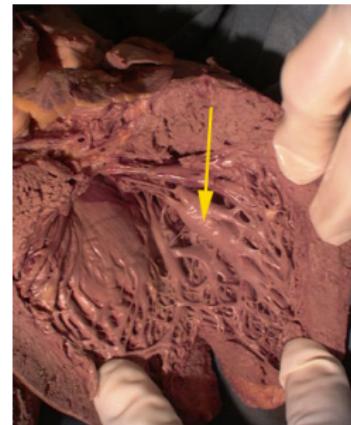
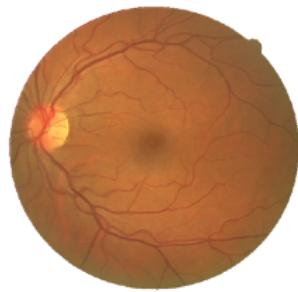
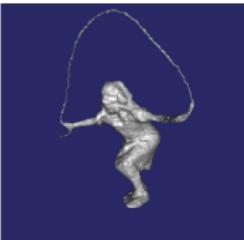
with Uzunbas and Metaxas MICCAI 2014

- Automatic
- Interactive

2 Potential Ideas

- Multiple Proposals
- Topological Priors

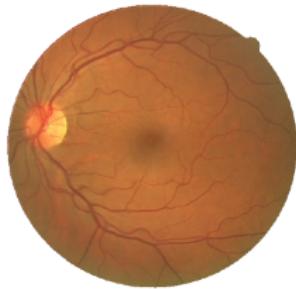
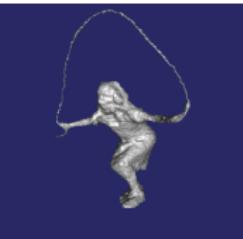
Motivation (Fine Structures)



Motivation (Fine Structures)

Idea: topological priors

topological structures; confidence

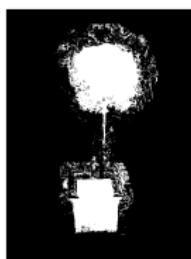


Segmentation: extracting y minimizing a given energy

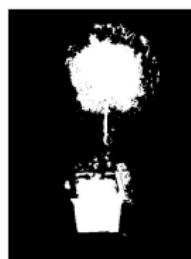
$$E(y) = E_{data}(y) + \omega E_{model}(y)$$



$$\omega = 0.0$$



$$\omega = 0.2$$



$$\omega = 0.8$$



$$\omega = 1.4$$

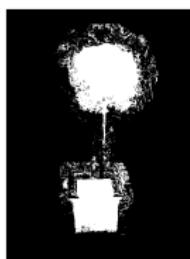


Segmentation: extracting y minimizing a given energy

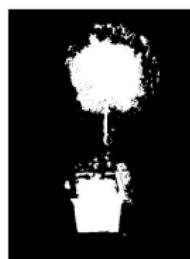
$$E(y) = E_{data}(y) + \omega E_{model}(y)$$



$$\omega = 0.0$$



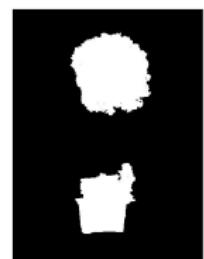
$$\omega = 0.2$$



$$\omega = 0.8$$



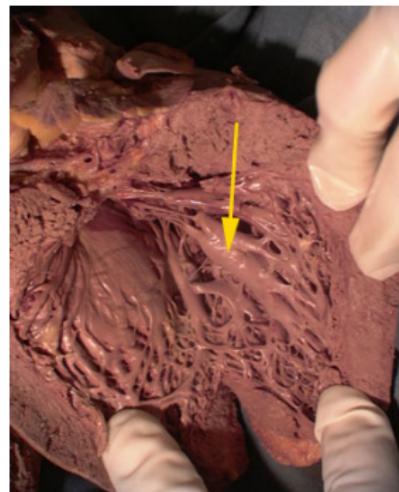
$$\omega = 1.4$$



Idea: incorporate the prior information of connectedness!

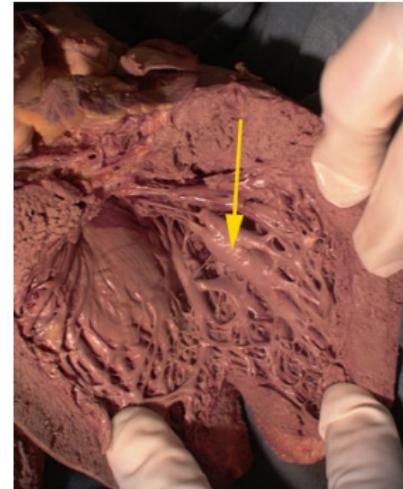
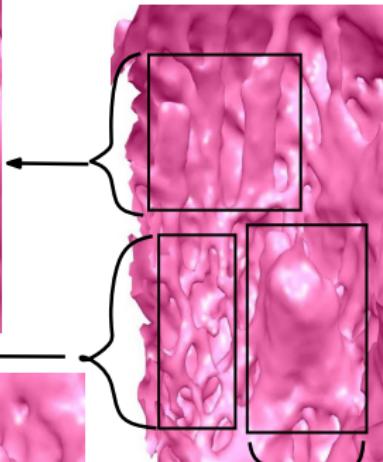
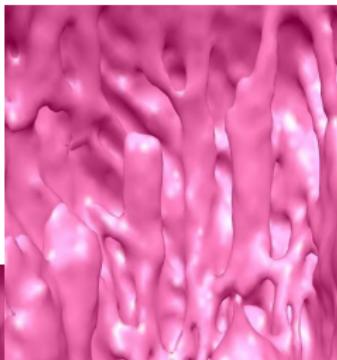


Cardiac Trabeculae, IPMI 2011



Cardiac Trabeculae, IPMI 2011

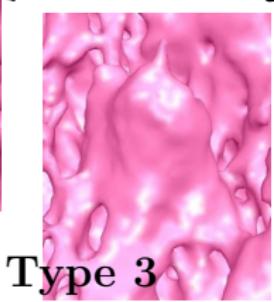
Type 1



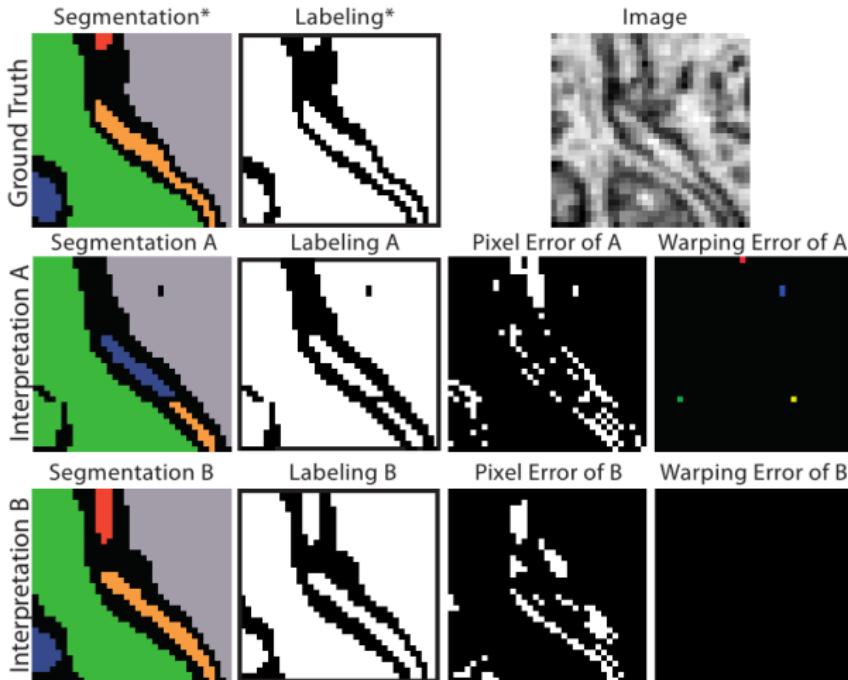
Type 2



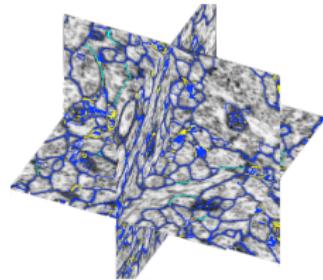
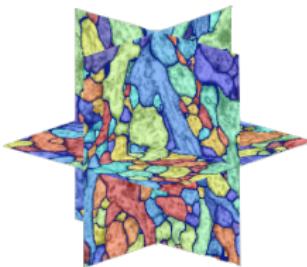
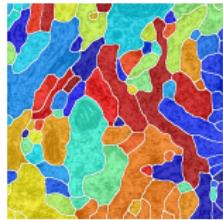
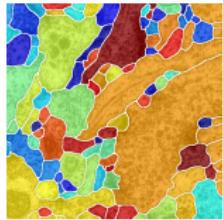
Type 3



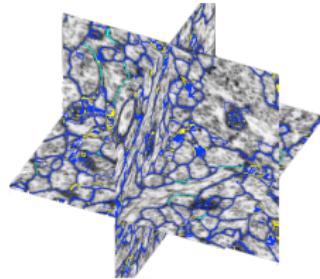
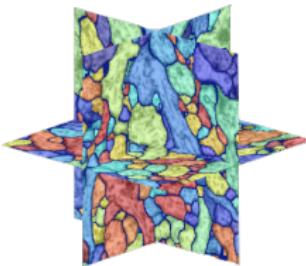
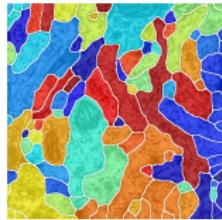
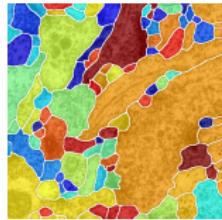
Revisiting Warping Error, Jain *et al.* CVPR 2010



The End



The End



Take Home Message

Interesting tools available. What is the right question to ask?

Thank you! Questions?

