Active Learning for Interactive 3D Image Segmentation

Rafeef Abugharbieh

Biomedical Signal and Image Computing Lab, University of British Columbia

Joint work with: Ghassan Hamarneh and Andrew Top

Medical Image Analysis Lab, Simon Fraser University

September 1, 2011

Biomedical Signal and Image Computing Laboratory BiSICL

MIAL Medical Image Analysis Lab

3D image segmentation categories

- **Manual Segmentation:** Time consuming, tedious, expensive, variability in results.

3D image segmentation categories

- **Manual Segmentation:** Time consuming, tedious, expensive, variability in results.
- **Fully Automatic Segmentation:** Either not robust or not accurate enough for many clinical settings. Needs to be tailored to specific applications.

3D image segmentation categories

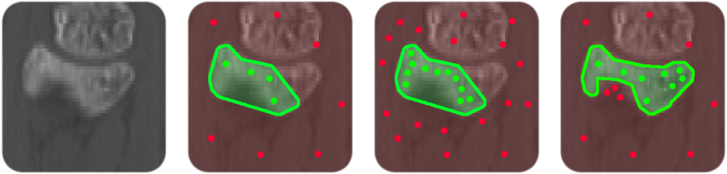
- **Manual Segmentation:** Time consuming, tedious, expensive, variability in results.
- **Fully Automatic Segmentation:** Either not robust or not accurate enough for many clinical settings. Needs to be tailored to specific applications.
- **Interactive Segmentation:** Good compromise between manual and automatic. Involves cycling between users providing input and the algorithm computing segmentation results until a high quality segmentation is achieved.

Interactive segmentation

- In interactive segmentation, some user input improves the segmentation faster than other.

Interactive segmentation

- In interactive segmentation, some user input improves the segmentation faster than other.

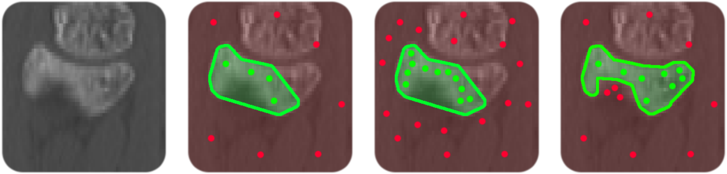


Base image | Initial input | Bad input | Good input

(Wrist CT) Green is a segmented region of interest and red is background. Dots represent user input: Region (green) and background (red) hard constraints.

Interactive segmentation

- In interactive segmentation, some user input improves the segmentation faster than other.



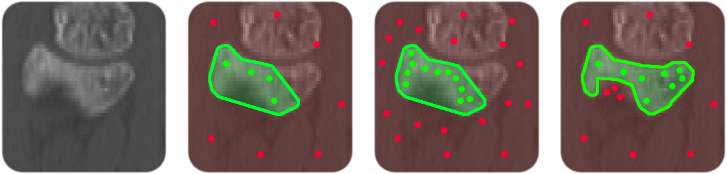
Base image   Initial input   Bad input   Good input

(Wrist CT) Green is a segmented region of interest and red is background. Dots represent user input: Region (green) and background (red) hard constraints.

- Problem: Where should the user provide input to most improve the segmentation?

Interactive segmentation

- In interactive segmentation, some user input improves the segmentation faster than other.



Base image | Initial input | Bad input | Good input

(Wrist CT) Green is a segmented region of interest and red is background. Dots represent user input: Region (green) and background (red) hard constraints.

- Problem: Where should the user provide input to most improve the segmentation? What about in 3D?

Active Learning in image segmentation

- Active learning guides the user to where input is most valuable.

Active Learning in image segmentation

- Active learning guides the user to where input is most valuable.
- Formulate segmentation as a supervised learning classification problem. Labelled voxels (features) represent training data.

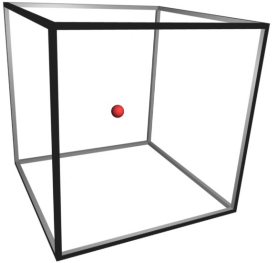
Active Learning in image segmentation

- Active learning guides the user to where input is most valuable.
- Formulate segmentation as a supervised learning classification problem. Labelled voxels (features) represent training data.
- **Query:** A feature chosen by the algorithm for the user to label.
- **Query strategy:** The method for choosing a query.

Active Learning in image segmentation

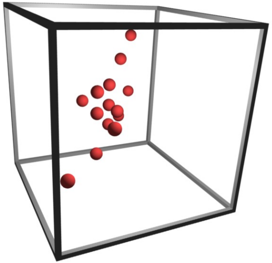
- Active learning guides the user to where input is most valuable.
- Formulate segmentation as a supervised learning classification problem. Labelled voxels (features) represent training data.
- **Query:** A feature chosen by the algorithm for the user to label.
- **Query strategy:** The method for choosing a query.
- Given a classifier, use a query strategy to produce a query that the user labels. Repeat until classifier is accurate.

Uncertainty sampling: Single query



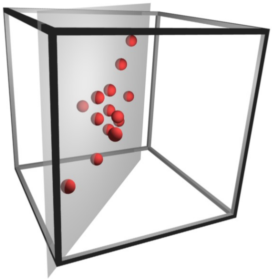
- Measure an intermediate segmentation's uncertainty.
- Query strategy: Uncertainty sampling (query is most uncertain voxel).
- Inefficient. Must re-segment after each query is labeled.

Uncertainty sampling: Naive batch query



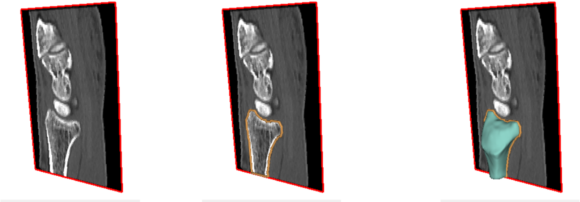
- Batch query: Query *n*-most uncertain voxels simultaneously.
- Inefficient in 3D. Cannot visualize image data at query voxels simultaneously.

Uncertainty sampling: Plane batch query

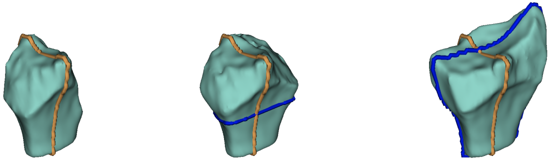


- Find plane that intersects maximal uncertainty.
- Image slice planes are intuitive, natural, familiar to clinicians.
- Efficient. User can label many uncertain voxels simultaneously by labelling the plane.
- Easy to segment, many options for 2D image segmentation.

Effects of interactive plane selection



First segmentation plane chosen by user in CT of radius bone.



Initial seg. Bad plane choice Better plane choice

Calculating 3D segmentation uncertainty

Segmentation uncertainty is calculated at every point in the image through a weighted sum of sub functions:

$$U(\mathbf{x}) = U_E(\mathbf{x}) + U_B(\mathbf{x}) + U_R(\mathbf{x}) + U_S(\mathbf{x}) \tag{1}$$

Calculating 3D segmentation uncertainty

Segmentation uncertainty is calculated at every point in the image through a weighted sum of sub functions:

$$U(\mathbf{x}) = U_E(\mathbf{x}) + U_B(\mathbf{x}) + U_R(\mathbf{x}) + U_S(\mathbf{x}) \tag{1}$$

- U_E: Entropy of probabilistic classification of voxels.

Calculating 3D segmentation uncertainty

Segmentation uncertainty is calculated at every point in the image through a weighted sum of sub functions:

$$U(\mathbf{x}) = U_E(\mathbf{x}) + U_B(\mathbf{x}) + U_R(\mathbf{x}) + U_S(\mathbf{x}) \tag{1}$$

- U_E: Entropy of probabilistic classification of voxels.
- U_B: Boundary term. U_B is high when image gradient is low near segmentation boundary.

Calculating 3D segmentation uncertainty

Segmentation uncertainty is calculated at every point in the image through a weighted sum of sub functions:

$$U(\mathbf{x}) = U_E(\mathbf{x}) + U_B(\mathbf{x}) + U_R(\mathbf{x}) + U_S(\mathbf{x}) \qquad (1)$$

- U_E: Entropy of probabilistic classification of voxels.
- U_B: Boundary term. U_B is high when image gradient is low near segmentation boundary.
- U_R: Regional term based on intensity priors for the foreground and background.

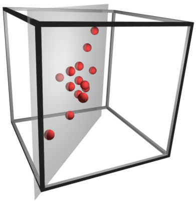
Calculating 3D segmentation uncertainty

Segmentation uncertainty is calculated at every point in the image through a weighted sum of sub functions:

$$U(\mathbf{x}) = U_E(\mathbf{x}) + U_B(\mathbf{x}) + U_R(\mathbf{x}) + U_S(\mathbf{x}) \tag{1}$$

- U_E: Entropy of probabilistic classification of voxels.
- U_B: Boundary term. U_B is high when image gradient is low near segmentation boundary.
- U_R: Regional term based on intensity priors for the foreground and background.
- U_S: Smoothness term based on reducing segmentation surface area.

Computing the most uncertain plane

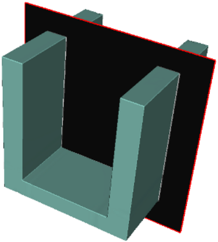


- We wish to find the plane, $\mathcal{P}_0$, that slices through the most uncertainty in the uncertainty field, U.

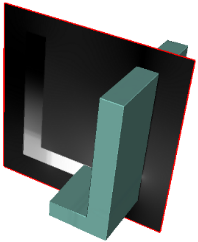
$$\mathcal{P}_0 = \mathrm{argmax}_{\mathcal{P}} \iint_{\mathcal{P}} U(\mathbf{x})\,\mathrm{d}A \tag{2}$$

Example computation of most uncertain plane

Gradient descent finding the plane intersecting with the most uncertainty.



Ground truth classification



Search for most uncertain plane

The final query plane matches our intuition for the best next plane.

Implementation Details

- Random walker algorithm (GPU implementation using NVIDIA's CUDA API) used for building 3D segmentation from user-labeled voxels.

Implementation Details

- Random walker algorithm (GPU implementation using NVIDIA's CUDA API) used for building 3D segmentation from user-labeled voxels.
- 2D Livewire used to label image planes. Each point on the plane is then used as a user-label (seedpoint) for random walker.

Implementation Details

- Random walker algorithm (GPU implementation using NVIDIA's CUDA API) used for building 3D segmentation from user-labeled voxels.
- 2D Livewire used to label image planes. Each point on the plane is then used as a user-label (seedpoint) for random walker.
- When computing the batch query plane, gradient descent is restarted 36 times, iterated 250 times each.

Example usage video

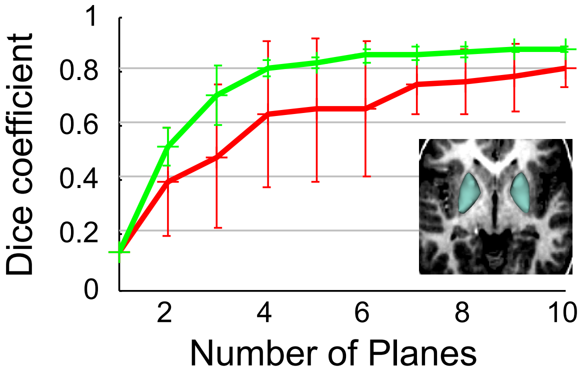
Active learning versus random plane selection

(Femur in MRI) Our active learning approach chooses planes on the left. Random selection is used on the right. Active learning converges to a better solution faster.

Active learning versus random plane selection

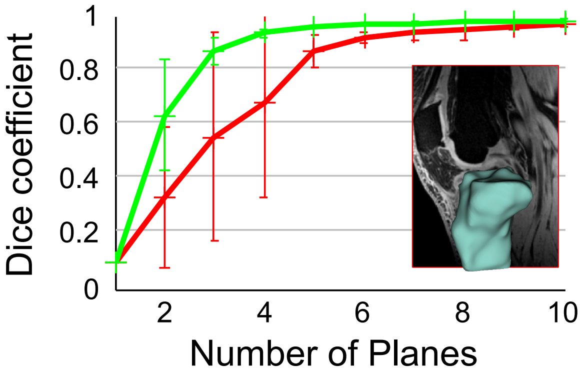
(Femur in MRI) Our active learning approach chooses planes on the left. Random selection is used on the right. Active learning converges to a better solution faster.

Putamen (brain) in MRI



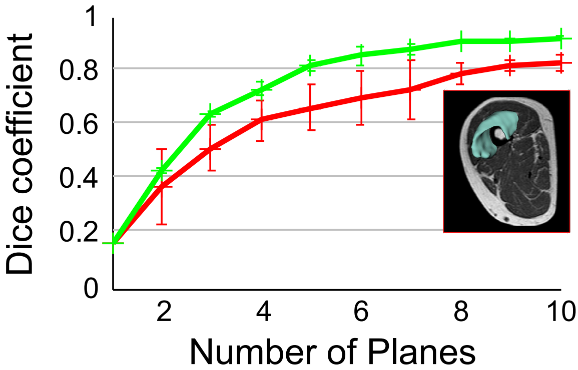
- Active learning plane suggestion strategy
- Random plane suggestion strategy

Tibia (leg bone) in MRI



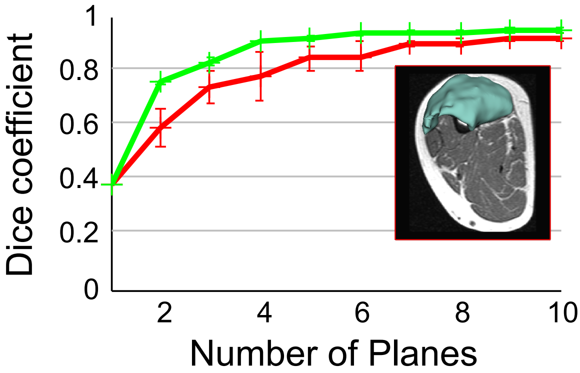
- Active learning plane suggestion strategy
- Random plane suggestion strategy

Vastus intermedius (thigh muscle) in MRI



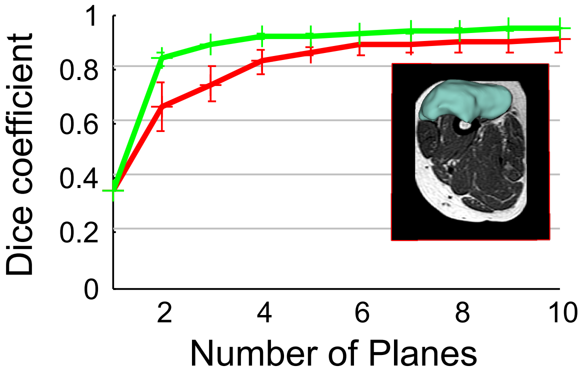
- Active learning plane suggestion strategy
- Random plane suggestion strategy

Vastus lateralis (thigh muscle) in MRI



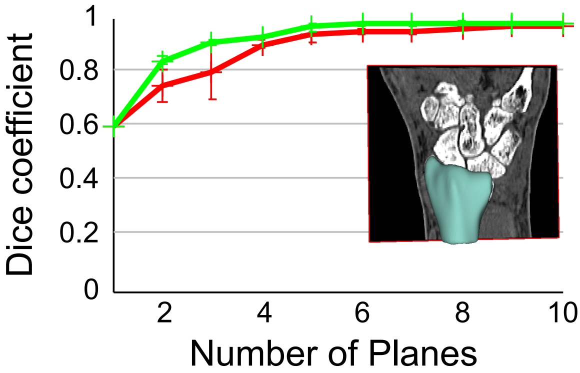
- Active learning plane suggestion strategy
- Random plane suggestion strategy

Vastus lateralis 2 (thigh muscle) in MRI



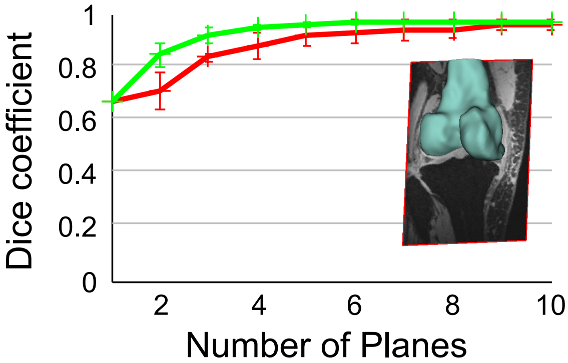
- Active learning plane suggestion strategy
- Random plane suggestion strategy

Radius (wrist bone) in CT



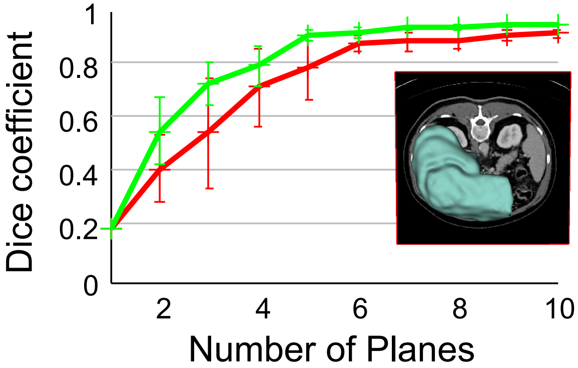
- Active learning plane suggestion strategy
- Random plane suggestion strategy

Femur (leg bone) in MRI



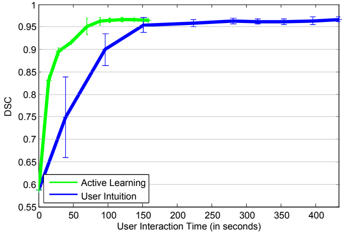
- Active learning plane suggestion strategy
- Random plane suggestion strategy

Liver in CT



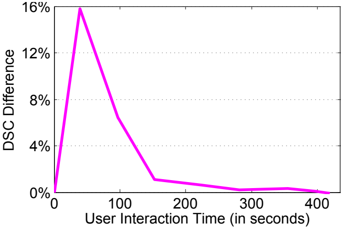
- Active learning plane suggestion strategy
- Random plane suggestion strategy

User study results



Active learning and user intuition.

First plane already chosen.



Percentage accuracy difference between green and blue curve on left versus time.

- Radius in CT image
- Active learning reduces user interaction time by 64%.

TurtleSeg

Interactive 3D Image Segmentation Software

http://www.turtleseg.org

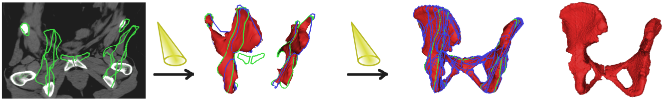
TurtleSeg

TurtleSeg

Interactive 3D Image Segmentation Software

http://www.turtleseg.org

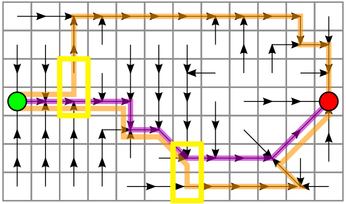
- Our active learning method is implemented for the 3D Livewire segmentation algorithm, titled **"Spotlight"**.
- Objects can be fully segmented by continuous invokation of Spotlight.



Spotlight: Contour instability

Other segmentation uncertainty terms are used in Spotlight, like contour instability.

- 3D Livewire is based on shortest path.
- We consider a given path uncertain if there are very different near-shortest paths.



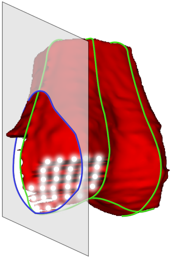
Points on paths are perturbed to evaluate their stability. Figure shows perturbed paths of the shortest path.

Spotlight: Uncertainty field

- Spotlight maintains an implicit uncertainty field.
- "Spotlight Attractors" are points in space that increase the uncertainty surrounding them.
- Spotlight attractors are placed at problem regions, attracting the suggestion plane.

Spotlight: Uncertainty field

- Spotlight maintains an implicit uncertainty field.
- “Spotlight Attractors” are points in space that increase the uncertainty surrounding them.
- Spotlight attractors are placed at problem regions, attracting the suggestion plane.



Spotlight attractors are shown in white around problem area. Suggested plane is shown with the new user contour within it.

Conclusion

- Active learning is applied to interactive 3D image segmentation.
- We demonstrated that active learning greatly reduces user time
- We plan to extend the method in the future by having it learn the uncertainty term weights via training data.

$$U(\mathbf{x}) = \lambda_E U_E(\mathbf{x}) + \lambda_E U_B(\mathbf{x}) + \lambda_R U_R(\mathbf{x}) + \lambda_S U_S(\mathbf{x}) \qquad (3)$$

- We would like to experiment with applying this method to other segmentation techniques.

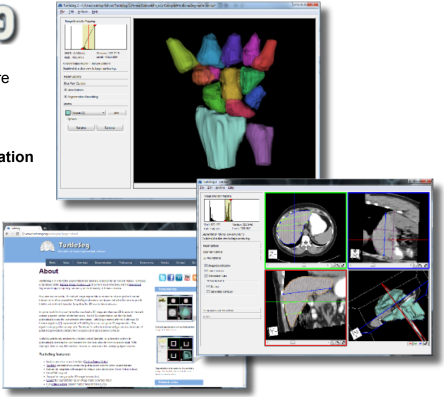
TurtleSeg Video

TurtleSeg

Interactive 3D Image Segmentation Software

- **Fully-operational active learning implementation**
- Intuitive and easy to use interface
- Full online manual and tutorial videos

TurtleSeg 2.0 coming soon with support for multiple segmentation labels



http://www.turtleseg.org

The end

Thank you.
Questions?

Prior works on active learning in image segmentation

- Ma, A. et al., 2006: Confidence based active learning for whole object image segmentation.
  Description: 2D image segmentation using the uncertainty sampling query strategy
- Li, J. et al., 2010: Supervised hyperspectral image segmentation using active learning.
  Description: 2D image segmentation of hyperspectral images. Batch queries let users label multiple pixels at once.
- Pavlopoulou, C. et al., 2003: Application of semi-supervised and active learning to interactive contour delineation.
  Description: Interactive (2D) segmentation, but their use of the term "active learning" differs from formal active learning.

None of these discuss **active learning** in the context of **3D** image segmentation.

Uncertainty term details

- $U_E(\mathbf{x}) = -p_1(\mathbf{x}) log_2\, p_1(\mathbf{x}) - (1 - p_1(\mathbf{x})) log_2(1 - p_1(\mathbf{x}))$
- $U_B(\mathbf{x}) = \delta(D_s(\mathbf{x}, y(\mathbf{x}))) \frac{1}{1+|\nabla I(\mathbf{x})|^\alpha}$
- $U_R(\mathbf{x}) = p(Y = y(\mathbf{x})|I(\mathbf{x})) = \frac{p(I(\mathbf{x})|Y=y(\mathbf{x}))}{p(I(\mathbf{x})|Y=0)+p(I(\mathbf{x})|Y=1)}$
- $U_S(\mathbf{x}) = \iiint_{N_\mathbf{x}} \delta(D_s(\mathbf{z}, y(\mathbf{x})))\, \mathrm{d}V$

Gradient descent details

- The uncertainty field can be arbitrary, so we cannot exploit its structure. Use gradient descent.
- There may be many local minima. Use gradient descent with multiple random restarts.
- Calculate gradient of $\iint\limits_{\mathcal{P}} U(\mathbf{x})\,\mathrm{d}A$ with respect to the plane $\mathcal{P}$'s parameters (i.e. its normal and a position on it).