Introduction to Segmentation with ITK

Josh Cates

Scientific Computing and Imaging Institute
University of Utah
Topics

Image Processing in ITK: Pipelines and Filters
ITK Segmentation Algorithms
Examples and Results
Acknowledgements

Stephen Aylward, Thomas Boettger, Celina Imielinska
Jisung Kim, Karl Krissian, Dimitris Metaxas, Lydia Ng, Sayan Pathak
Raul San-Jose, Jay Udupa, Paul Yushkevich, Carl-Fredrik Westin
Ross Whitaker, Tolga Tazdizen, Darby Van Uitert, Suyash Awate, Xinwei Xue

and (of course):
The National Library of Medicine
Segmentation: What do we have to offer?

Specific algorithms (from various groups)
  • Implementations in templated FILTER objects
  • Examples, simple and complex

Preprocessing
  • Denoising
  • Feature extraction
  • Resampling, cropping

Infrastructure
  • Readable and reusable modular code
  • Frameworks to enable range of strategies
  • ITK Features – event signaling, parallel (shared-memory) execution, user community
Where to go to learn to use the filters

http://www.itk.org
Manual Pages
Software Guide
• Algorithm descriptions
• Tutorials for using algorithms
Users Mailing List
CVS Repositories:
• Insight/Examples
• InsightApplications

*Insight into Images* (theory book) – coming soon
Roadmap: An Image Processing Pipeline

Raw Data

Filtering
linear
nonlinear

Feature Extraction
differential
differential
 geom.
edge detection

Segmentation
region growing
watersheds
level-sets

Dataflow architecture: ITK filters fit together to produce segmentation applications.

Moral: ITK segmentation filters are not complete applications – components in a pipeline.

Visualization
binary volume
meshes
labeled image
implicit surfaces
Notes on using ITK filters

Most filters are N dimensional

• Image = volume
• Pixel = voxel

Data type matters

• Different filters may have different requirements
• Cast if necessary

Watch spacing – resample if necessary

Read filter's documentation

– And send feedback!
<table>
<thead>
<tr>
<th>Edge-preserving Smoothing</th>
<th>Boundary</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Perona-malik</td>
<td>• Edge-detection</td>
</tr>
<tr>
<td>• Curvature flow and MCDE</td>
<td>• Watersheds</td>
</tr>
<tr>
<td>• Bilateral</td>
<td></td>
</tr>
<tr>
<td>Similarity/classifier</td>
<td>Deformable models</td>
</tr>
<tr>
<td>• Region Growing</td>
<td>• Level-sets</td>
</tr>
<tr>
<td>• Statistical</td>
<td>• Mesh</td>
</tr>
<tr>
<td></td>
<td>Hybrid methods</td>
</tr>
<tr>
<td></td>
<td>• Region growing + level-sets</td>
</tr>
<tr>
<td></td>
<td>• Fuzzy connect. + Voronoi</td>
</tr>
</tbody>
</table>
Linear vs. Nonlinear Diffusion

Gaussian Blurring
  • Convolution framework
  • Recursive Gaussian approximation

Nonlinear Diffusion (edge-preserving)
  • PDE Framework
  • Perona-Malik anisotropic diffusion, curvature flow and MCDE, bilateral
  • Scalar and vector


Linear Diffusion

Destroys and moves edges
Nonlinear Diffusion
Preserves Edges
Region Growing

Idea
• Start with set of *seed pixels* – *region*
• Iteratively include neighboring pixels that satisfy *membership criteria*

Membership criteria – similarity based metrics
• Intensity interval
• Regional statistics

Algorithms
• Simple to complex variations
• Easy to write using ND neighborhood tools
Confidence Connected Filter

Threshold based region growing
Mean and standard deviation of region determine upper and lower thresholds
Recomputes thresholds at intervals

1. Compute $\mu$ and $\sigma$ of region
2. Flood fill with threshold interval $[\mu-k\sigma, \mu+k\sigma]$
3. Repeat N times
Example: Confidence Connected Filter

Examples/Segmentation/ConfidenceConnected.cxx

- original
- white matter (60,116)
- ventricle (81,112)
- gray matter (107,69)

- smoothing iterations 5
- smoothing time step 0.125
- C.C. multiplier 2.5
- C.C. iterations 5
Statistical Pattern Classification Framework

Follows closely Duda & Hart, *Pattern Classification*

**Idea**
- Find disjoint regions in a feature space
- Classify image pixels according to feature vectors

**Classifier**
- Multiple membership functions (each represents one possible class) return scores from feature vectors
- Decision rule based on scores

**Examples**
- Insight/Examples/Statistics
Statistical Pattern Classification Framework

1. Measurement vectors are input to membership functions
2. Membership functions feed scores to decision rule
3. Decision rule compares scores and returns a class label
ND Segmentation via Fuzzy Connectedness

Estimates strength of connectivity among pixels
  • Relative position and intensity
Simple Fuzzy Connectedness
  • Connectivity to a seed point
  • Fuzzy map

Segmented regions can be extracted by thresholding the fuzzy map

Scalar and vector (RGB) implementations in ITK

Fuzzy Affinity

The affinity between any two pixels $c$ and $d$ depends on

- How near $c$ and $d$ are spatially
- How similar $c$ and $d$ are in their intensity-based features

$$\text{affinity}(c, d) = g(f(c), f(d), \|c - d\|, c, d)$$

where $f(.)$ is the image intensity.
Fuzzy Connectedness Relation

Each **path** between any $c$, $d$ has a **strength** which is the smallest affinity along the path (*thin segments in figure*).

The **connectivity value** between $c$ and $d$ is the strength of the strongest of all paths.
Fuzzy Connectedness Theorem

A fuzzy object $O$ of strength $\theta$ containing a point $o$ is a set of pixels such that, for any $c$ in this set, $\text{affinity}(o, c) \geq \theta$, and for any $e$ not in this pool, $\text{affinity}(o, e) < \theta$.

3D images can be segmented at interactive speeds on PCs via dynamic programming by using fuzzy connectedness.
Applications of FC Image Segmentation

1. Multiple Sclerosis lesion load quantification (MRI).
2. White matter lesion quantification in late life depression (MRI).
4. MR angiography and A/V separation.
5. CT angiography and suppression of bone.
7. CT Colonography.
10. Lung perfusion/ventilation analysis.
MR Angiography (CE-MRA)

CE-MRA

Vascular Tree

Artery/Vein separation
Brain Tumor (MRI)

Flair

Edema + Brain (SR)

T1

Enhancing tumor + brain (SR)

T1E

Edema + enhancing tumor (VR)
Edge Detection

Gradient Magnitude – scalar and vector
Canny edges
Convolution Framework
  • Sobel
  • Laplacian

See Insight/Examples directory and Software Guide.
Watershed Segmentation Algorithm

Edge-based, global and hierarchical labeling of image

N-dimensional

Example application

Evaluation work (user study)
The Watershed Transform in ITK

Image treated as a topological relief map – intensity represents height

Gradient descent defines *segmented regions*

- Set of all pixels whose paths of steepest descent terminate in same local minimum
- Bounded by image features

“No parameters”
ITK Watershed Transform

Image (filtered)  Feature Extraction “Edge Map”  Watershed Transform

Watershed Depth
The Oversegmentation Problem

Watershed transform produces too many regions
  • One per local minimum
  • Especially in noisy or highly detailed data
To alleviate oversegmentation

✓ Hierarchical approach – merge adjacent regions according to increasing \textit{watershed depth}

Watersheds Hierarchy

Enforce minimum watershed depths at successively higher levels.

Watershed Transform

Watershed Depth Threshold

Undersegmented

Oversegmented

= basin

Boolean Operations On Sub-trees (e.g. user interaction)

Initial Watershed Transform
Example: Watersheds GUI

InsightApplications/SegmentationEditor

Watershed transform
Data with overlay
Segmentation in progress
3D isosurface rendering

Sliders manipulate watershed depth and position in the hierarchy.
Example: Watersheds GUI

Algorithm Evaluation: User Study

Comparison of user-assisted hierarchical watersheds with hand contouring.

Hand contouring

- *De facto* standard
- General and reliable (?)

Issues

Can a general-purpose algorithm compete

Are our validation tools up to the task?

## User Study Overview

<table>
<thead>
<tr>
<th>Ground Truth Subjects (Slicer)</th>
<th>WS Segmentation Subjects</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MRI Brain Tumor</strong> (4 cases)</td>
<td><strong>Slice from HBW tumor database</strong> (4 per case)</td>
</tr>
<tr>
<td>Radiologists from U. of Utah (3)</td>
<td></td>
</tr>
<tr>
<td><strong>VHP Cryosection</strong> (Eyeball, optic nerve, Lateral rectus)</td>
<td><strong>3rd year med. students from HBW and Utah</strong> (EB-4, ON-3, LR-8)</td>
</tr>
<tr>
<td><strong>3rd year med. students U. of Utah</strong> (7)</td>
<td></td>
</tr>
</tbody>
</table>
Validation Strategy

Experts Hand Contouring

Benchmark Performance Parameters

Analysis Conclusions

Ground Truth

WS Performance Parameters

User Studies WS

STAPLE

Accuracy Precision Efficiency

(RR vs Full)
VHP Results

Data  Hand contour  Watersheds

Eyeball

Optic nerves
HBW Tumor Database Results

Hand contour Watersheds
Results

Visible Human cryosections

Brain tumor MRI
Summary of Evaluation Results

Accuracy
- Generally lower sensitivity (TPF) and higher specificity (TNF)
- Total correct fraction generally within variation of experts (better for tumor data)

Precision
- Significant improvement with watersheds

Efficiency
- Significant improvement with watersheds (minutes versus hours)
- Time/expertise to tune hidden parameters issue
Deformable Models

Level-set surface representation
PDE Framework
Example applications and Results
Mesh-based
Level-Set Surface Modeling Theory

$k^{th}$ Level Set: set of all points of value $k$

Embed $N$ dimensional surface as ZERO level set of $N+1$ dim. volume $\phi$

Model $N$ dim. surface movement as evolving wavefront – forward differences solution to PDE

$$\phi_t = -F |\nabla \phi|$$
Segmentation Using Level Sets

*Define speed term(s) to go to zero at edges – data fitting term*

Surface motion/speed based on intensity-based features

Solve the level-set equation where

\[ F = \text{Speed} - \text{Curvature} \]
Insight PDE Solver Framework

Purpose

- Nonlinear image processing – e.g. anisotropic diffusion
- Moving wave fronts – level set models
- Deformable registration

Generic framework

- Separate solvers from equations – *interchangeable* code objects
Constructing a PDE Filter
Level-Set Segmentation Framework

- Finite Difference Solver
- Sparse-Field Level-Set Solver
- Level-Set Segmentation Filter
  - Curvature Function
  - Level Set Function
- User-Defined LS Seg. Filter
- Curvature Function

- "Feature" Image
- Initial Model
- Output Model

- Shape Detection Function
- Canny Edges Function
- Active-Contours Function
- Laplacian Function
- Threshold Function
Level-Set Segmentation Algorithms in ITK

Fast marching
Geodesic active contours
CURVES (vessel segmentation)
Intensity interval (scalar and vector)
Canny edge distance
Laplacian edges

... and more.
Example: Threshold-based LS Segmentation

Speed function (positive inside object)
Similar to confidence connected filter

Points Inside: Points Outside

Low Threshold

High Threshold

Image Intensity

Model Speed

- Lefohn, Whitaker, Cates, “Interactive Level-Set Models for Brain Tumor Segmentation”, MICCAI 2003
Example: Level-Set Segmentation GUI

InsightApplications/ThresholdLevelSetFltkGui
CURves: Codimension-2 Active Contours

\[ \nu_t = |\nabla \nu| \lambda(\nabla \nu, \nabla^2 \nu) + \nabla \nu \cdot d \]

\( \nu \) is (positive) distance to curve

\( \lambda(\nabla \nu, \nabla^2 \nu) \) is smaller principal curvature of tube

\( d \) is some vector field in \( \mathbb{R}^3 \)

Lorigo et al., Medical Image Analysis, 2001.

Provided by L. Lorigo MIT AI Lab.
CURVES Applications: MR Angiography

Maximum Intensity Projection

Segmentation result

Isosurfaces 112, 60 and 40
CURVES: Interface Integration
### Multiscale Level-Set 3D Segmentation

<table>
<thead>
<tr>
<th>Scale</th>
<th>Seed surface</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/4</td>
<td><img src="image1.png" alt="Seed Surface" /></td>
<td><img src="image2.png" alt="Data" /></td>
</tr>
<tr>
<td>1/2</td>
<td><img src="image3.png" alt="Seed Surface" /></td>
<td><img src="image4.png" alt="Data" /></td>
</tr>
<tr>
<td>1/1</td>
<td><img src="image5.png" alt="Seed Surface" /></td>
<td><img src="image6.png" alt="Data" /></td>
</tr>
</tbody>
</table>
CURVES Applications: Visible Human Project

White Matter, 3D
800 x 1056 x 1211 sub-volume
- Pyramidal multiscale evolution
- 2 seed points

50x76x75 100x132x151 200x264x302

data provided by Peter Ratiu
Advanced Features in the PDE Framework

Parallel Solvers –
Narrowband, Sparse field

![Graph showing speedup vs. number of processors for SGI Origin 3000 64 600 Mhz Processors.](image-url)
4\textsuperscript{th} Order Flow Segmentation Framework

- **Finite Difference Solver**
- **Sparse-Field Level-Set Solver (Refitting)**
- **Level-Set Segmentation Filter**
  - Speed Function
  - Curvature Function
- **User-Defined LS Seg. Filter**
- **Manifold Solver (Normals Processing)**
- Input Image
- Initial Model
- Output Model

- Shape Detection Function
- Canny Edges Function
- Active-Contours Function
- LaPlacian Function
- Threshold Function
Segmentation Using $4^{\text{th}}$ Order Flows

Special $4^{\text{th}}$ order solver plugs into LS segmentation framework – no change in function objects.

- Speed term only
- Speed + Anisotropic $4^{\text{th}}$ order terms
- (not real-time)
SNAP Tool

Aimed at clinical users – easy to learn and use
Implements various ITK level set algorithms
Implements both edge-based and region competition
geodesic snake methodology

InsightApplications/Snap
SNAP User Interface

SNAP Segmentation of the Caudate Nuclei

SNAP Segmentation Wizard with VCR Controls

Manual Segmentation

Dialog for setting evolution parameters

Scalpel tool for editing segmentations

SNAP User Interface

Segmentation Pipeline

Step 1 of 3
Preprocessing
A. Choose the kind of image information used to drive snake evolution:
   - Intensity regions
   - Image edges
B. Use the buttons below to produce image regions or edges
   - Process Image
   - Load From File
C. Press Next to accept the unprocessed image
   - Next

Segmentation Pipeline

Step 2 of 3
Snake Initialization
A. Insert an initialization bubble and adjust its radius
   - Radius
   - Active bubbles
B. Use the buttons below to control snake evolution
   - Step size
   - Iteration
C. Press Next to begin segmentation
   - Next

Segmentation Pipeline

Step 3 of 3
Segmentation
A. Edit the parameters of the snake evolution:
   - Set Parameters
B. Use the buttons below to control snake evolution
   - Accept
   - Cancel
   - Help

InsightApplications/Snap
Deformable Models Framework

Deformable mesh filter with plug-in deformation forces – e.g. balloon force and/or gradient

Image → Gradient magnitude → Gradient → Vector Field → Deformable Model Filter → Output model (mesh)

Parameters:
- stiffness
- vector scaling
- time step
Deformable Model Example
3D Segmentation of Lung from CT

Seed Point

Slice 13  Slice 24  Slice 28

Segmentation Results \( (T = 1200) \)
Deformable Simplex Meshes

German Cancer Research Center, Heidelberg, Germany

For all mesh points $P_i$

$$P_i^{t+1} = P_i^t + (1 - \gamma)(P_i^t - P_i^{t-1}) + \alpha_i F_{int} + \beta_i F_{ext}$$

$F_{int}$: Internal Energy

$F_{ext}$: External Energy

Algorithm function (demonstrated on a cube):
Deformable Simplex Meshes

Advantages

- Effective curvature constraints
- Definition of additional interactive forces

Results of liver segmentation
“Hybrid” Segmentation Methods

Apply several algorithms in sequence
Utilize strengths of each

• Speed / accuracy tradeoffs — e.g. connected component vs. level-sets
• Maximize use of information in data — e.g. region based plus boundary based
Hybrid Method: Region Growing + Level Sets

Generate initial model using confidence connected filter

Fit to data using level-set methods – minimize distance to Canny edges
Confidence Connected + Level-Set Result

Initial confidence-connected result

LS Speed Term: distance from Canny edges

Post-processing with Canny LS segmenter

Hybrid Method: Fuzzy Connect. + Voronoi

Fuzzy Connectedness:
Mean = {201.0, 179.2, 127.6}
Var = {289.9, 389.8, 223.8}
Computed from a small sample region within the object.

Voronoi Diagram:
MeanPctErr = 0.35 for all channels.
VarPctErr = 2.0 for all channels.
Voronoi Diagram Segmentation Algorithm

1. Voronoi Diagram is computed from random seed points
2. Region boundaries are determined from homogeneity operator (i.e. confidence intervals computed from a binary mask)
3. Course-to-fine subdivision of boundary regions
4. Subgraph of Delaunay triangulation defines segmented boundaries

Visible Human Male: segmentation of temporalis muscle

Columbia University
Summary

ITK Segmentation Tools

- Selection of well-known algorithms – incl. preprocessing
- Tested and evaluated
- Examples and documentation
- Infrastructure – extend and combine methods
More Information …

enjoy ITK!

http://www.itk.org

http://www.sci.utah.edu/~cates