Segmentation in electron microscopy images

Aurelien Lucchi, Kevin Smith, Yunpeng Li Bohumil Maco, Graham Knott, Pascal Fua.

http://cvlab.epfl.ch/research/medical/neurons/
Outline

- Automated Approach to Segmentation of Mitochondria in EM Images
- Supervoxel-Based Segmentation of Mitochondria in EM Image Stacks
- Structured prediction for image segmentation (Structural SVM framework)
Understanding the brain

The Electromagnetic Spectrum

Penetrates Earth Atmosphere?

<table>
<thead>
<tr>
<th></th>
<th>Y</th>
<th>N</th>
</tr>
</thead>
</table>

Wavelength (meters)

<table>
<thead>
<tr>
<th>Wave Type</th>
<th>10^3</th>
<th>10^2</th>
<th>10^-3</th>
<th>10^-4</th>
<th>10^-5</th>
<th>10^-6</th>
<th>10^-7</th>
<th>10^-8</th>
<th>10^-9</th>
<th>10^-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radio</td>
<td></td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Microwave</td>
<td></td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Infrared</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Visible</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Ultraviolet</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>X-ray</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Gamma Ray</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

About the size of...

- Buildings
- Humans
- Honey Bee
- Pinpoint
- Protozoans
- Molecules
- Atoms
- Atomic Nuclei

| Size          | Image
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1 m</td>
<td>1 km</td>
</tr>
<tr>
<td>10 ^1 m</td>
<td>10 ^2 m</td>
</tr>
<tr>
<td>10 ^3 m</td>
<td>10 ^4 m</td>
</tr>
<tr>
<td>10 ^5 m</td>
<td>10 ^6 m</td>
</tr>
<tr>
<td>10 ^7 m</td>
<td>10 ^8 m</td>
</tr>
<tr>
<td>10 ^9 m</td>
<td>10 ^10 m</td>
</tr>
<tr>
<td>10 ^11 m</td>
<td>10 ^12 m</td>
</tr>
</tbody>
</table>

Eye

Light microscope

Electron microscope

- Hair thickness
- Hand
- Finger
- Cell
- Bacterium
- Virus
- Macromolecules
- Atoms
Electron Microscopy data

5 × 5 × 5 μm section taken from the CA1 hippocampus, corresponding to a 1024 × 1024 × 1000 volume (N ≈ 10^9 total voxels)

One estimate puts the human brain at about 100 billion (10^{11}) neurons and 100 trillion (10^{14}) synapses
Mitochondria segmentation

- Difficulties:
  - Vesicles and cell boundaries appear similar to mitochondria.
  - Assumptions about the shape are difficult.
Approach

(a) Original EM image

(b) Superpixels

(c) Superpixel graph

(d) SVM prediction

(e) Graph-cut segmentation

(f) Final segmentation

Approach

(a) Original EM image  (b) Superpixels  (c) Superpixel graph

(d) SVM prediction  (e) Graph-cut segmentation  (f) Final segmentation
SLIC superpixels

- Clusters pixels in the combined five-dimensional color and image plane space.
- Efficiently generate compact, nearly uniform superpixels.

SLIC Superpixels, EPFL, Technical Report, Nr. 149300, June 2010
R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua and S. Süsstrunk.
Source code available online.
SLIC superpixels

Under-segmentation error = error with respect to a known ground truth.

Boundary recall = fraction of ground truth edges fall within one pixel of a least one superpixel boundary.
SLIC superpixels

GKM = 10 iterations of k-means
Approach

(a) Original EM image
(b) Superpixels
(c) Superpixel graph

(d) SVM prediction
(e) Graph-cut segmentation
(f) Final segmentation
Large choice of features

1. Edge features
   (a) (b) (c) (d)

2. Line features
   (a) (b) (c) (d) (e) (f) (g) (h)

3. Center-surround features
   (a) (b)

The characteristic function returns the nearest edge point $c$ in direction $\theta$ given location $m$.

Texton Map → Training Histogram
Ray features

- Designed to consider image characteristics at distant contour points.
- Rays can characterize deformable or irregular shapes.
- Can be efficiently precomputed.
Ray features

Distance Feature
Considers absolute distance to nearest contour point in direction $\theta$.

$$f_{i}^{\text{dist}}(I) = \| c(I,m_i,\theta_i) - m_i \|$$
$$f_{i}^{\text{dist}}(I) = a$$

Orientation Feature
Considers the orientation of nearest contour point in direction $\theta$.

$$f_{i}^{\text{ori}}(I) = \frac{\nabla I(c(I,m_i,\theta_i))}{\| \nabla I(c(I,m_i,\theta_i)) \|} \cdot (\cos\theta_i, \sin\theta_i)^T$$
$$f_{i}^{\text{ori}}(I) = \alpha$$

Difference Feature
Compares relative distances to nearest contour points in directions $\theta$ and $\theta'$.

$$f_{i}^{\text{diff}}(I) = \frac{\| c(I,m_i,\theta_i) - m_i \| - \| c(I,m_i,\theta_i') - m_i \|}{\| c(I,m_i,\theta_i) - m_i \|}$$
$$f_{i}^{\text{diff}}(I) = \frac{a - b}{a}$$

Norm Feature
Considers gradient strength of nearest contour point in direction $\theta$.

$$f_{i}^{\text{norm}}(I) = \| \nabla I(c(I,m_i,\theta_i)) \|$$
$$f_{i}^{\text{norm}}(I) = \| \nabla I \|$$
## Ray features

<table>
<thead>
<tr>
<th>Edges / Rays</th>
<th>Ray descriptor</th>
<th>distance, $d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>original</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>rotation</td>
<td></td>
<td>0.27</td>
</tr>
<tr>
<td>scaled</td>
<td></td>
<td>0.80</td>
</tr>
<tr>
<td>affine</td>
<td></td>
<td>0.83</td>
</tr>
<tr>
<td>vesicles</td>
<td></td>
<td>4.73</td>
</tr>
<tr>
<td>puzzle</td>
<td></td>
<td>5.01</td>
</tr>
<tr>
<td>dendrite</td>
<td></td>
<td>5.64</td>
</tr>
</tbody>
</table>
Combining features

A feature vector \( f \) is extracted for each superpixel \( x_i \) in \( I \).

\[ f = [f_{\text{Ray}}, f_{\text{Rot}}, f_{\text{Hist}}] \]

- \( f_{\text{Ray}} \):
  - \( f_{\text{dist}} \)
  - \( f_{\text{diff}} \)
  - \( f_{\text{ori}} \)
  - \( f_{\text{norm}} \)

- \( f_{\text{Rot}} \):
  - \( G_x \)
  - \( G_{xx} \)
  - \( G_{yy} \)

- \( f_{\text{Hist}} \):
  - \( \mathcal{N} \cup \{x_i\} \)
Approach

(a) Original EM image  
(b) Superpixels  
(c) Superpixel graph  
(d) SVM prediction  
(e) Graph-cut segmentation  
(f) Final segmentation
Structured prediction models

\( X \)

\((x_1, \cdots, x_i, \cdots, x_n)\)

\( Y \)

\((y_1, \cdots, y_i, \cdots, y_n)\)

\( y_i = \{+1, -1\} \)
MRF for image segmentation

\[ E_w(X, Y) = \sum_{i \in V} D(y_i) + w \sum_{i, j \in \mathcal{E}} V(y_i, y_j) \]

Maximum-a-posteriori (MAP) solution:

\[ Y^* = \underset{Y \in \mathcal{Y}}{\arg \min} \ E_w(X, Y) \]

Data (D), Unary likelihood, Pair-wise Terms, MAP Solution

Boykov and Jolly [ICCV 2001], Blake et al. [ECCV 2004]
Potts model

- Pairwise potential usually has the form of a contrast sensitive Potts model

\[ V(y_i, y_j | x_i, x_j) = \begin{cases} 
\exp\left( - \frac{||I(x_i) - I(x_j)||^2}{2\sigma^2} \right), & \text{if } y_i \neq y_j \\
0, & \text{otherwise,}
\end{cases} \]
Energy minimization

- **Graph-cuts**
  - Optimal solution if energy function is submodular

- **Belief-propagation**
Learned pairwise term

\[ E_w(X, Y) = \sum_{i \in \mathcal{V}} D(y_i) + w \sum_{i,j \in \mathcal{E}} V(y_i, y_j) \]

- Pairwise term is now defined as the output of an SVM classifier classifying edges.
- How do we set \( w \) ?
Supervoxel-Based Segmentation of Mitochondria in EM Image Stacks with Learned Shape Features, IEEE Transactions on Medical Imaging, Vol. 30, Nr. 11, October 2011


Supervoxel-Based Segmentation
Approach

1. Image stack X

2. Extract SLIC supervoxels

3. Train SVM classifier using 3d rays + histograms

4. Final segmentation
Approach

- Given image stack $X$ and corresponding label $Y$
- Build an energy function $E_w(X, Y)$ defined on a graph $\mathcal{G} = \{V, E\}$ where nodes represent supervoxels.
- Look for a solution that minimizes it

$Y^* = \arg \min_{Y \in \mathcal{Y}} E_w(X, Y)$
SLIC supervoxels
3d Rays

(a) \( L \) Rays cast on a geodesic sphere.

(b) The \( f_{\text{dist}} \) descriptor ordered according to the canonical orientation defined by \( n_1 \) and \( n_2 \).

(c) A cropped EM image stack containing a mitochondrion. Edges appear in white.

(d) \( L \) Rays cast from \( c_i \) in the mitochondrion to the closest surface boundary. Principle axes \( e_1 \) and \( e_2 \) appear in green and red.

(e) The \( f_{\text{dist}} \) descriptor re-ordered into the canonical orientation defined by \( e_1 \) and \( e_2 \).
Results

- Striatum dataset (1536 × 872 × 318 voxels, with a 6 × 6 × 7.8 nm resolution)
Results

- Cubes vs Supervoxels
- Comparison to Ilastik
Results
Structured prediction

- Prediction of complex outputs
  - Structured outputs: multivariate, correlated, constrained

- Novel, general way to solve many learning problems
Structured prediction

$E_w(X, Y) = \sum_{i \in \mathcal{V}} D(y_i) + \sum_{i, j \in \mathcal{E}} V(y_i, y_j)$

• Efficient Learning/Training – need to efficiently learn parameters $w$ from training data

• Solution: use **Structural SVM framework**
  • Can also use Perceptrons, CRFs, MEMMs, M3Ns etc
Data term

- Data term can be defined as:
  - Output of a classifier (e.g. RBF-SVM trained with 3d ray features).
  - Weighted sum of features:

\[
D(Y_i) = w^U \cdot f(x_i)
\]
Structured SVM

\[ E_w(X, Y) = \sum_{i \in \mathcal{V}} D(y_i) + \sum_{i, j \in \mathcal{E}} V(y_i, y_j) \]

- Given a set of \(N\) training examples with ground truth labels \((Y^{(1)}, \ldots, Y^{(N)})\), we can write

\[ Y^* = \arg \min_{Y \in \mathcal{Y}} E_w(X, Y) \]

\[ \forall n, Y \in \mathcal{Y}_n \setminus Y^{(n)} \]

\[ E_w(Y^{(n)}) \leq E_w(Y) \]

Energy for the correct labeling at least as low as energy of any incorrect labeling.
Structured SVM

- Given a set of $N$ training examples with ground truth labellings $(Y^{(1)}, \ldots, Y^{(N)})$ we optimize:

$$\min_{w, \xi \geq 0} \frac{1}{2} \left\| w \right\|^2 + \frac{C}{N} \sum_{n=1}^{N} \xi_n$$

s.t. $\forall n, Y \in \mathcal{Y}_n \setminus Y^{(n)} : \delta E_w(Y) \geq \Delta(Y^{(n)}, Y) - \xi_n$

$$\delta E_w(Y) = E_w(Y) - E_w(Y^{(n)}) \quad \Delta(Y^{(n)}, Y) = \sum_{i \in \mathcal{V}} I(y_i \neq y_i^{(n)})$$
Structured SVM

- Since the SSVM operates by solving a quadratic program (QP), all the constraints must be linear.
- Energy function must be expressible as an inner product between the parameter vector $w$ and a feature map.

\[
E_w(X, Y) = \sum_{i \in \mathcal{V}} D(y_i) + \sum_{i, j \in \mathcal{E}} V(y_i, y_j)
\]

\[
E_w(Y) = \langle w, \Psi(Y) \rangle .
\]

\[
w = \left( (w^D)^T, (w^V)^T \right)^T
\]
Illustrative Example

Original SVM Problem

- Exponential constraints
- Most are dominated by a small set of “important” constraints

Structural SVM Approach

- Repeatedly finds the next most violated constraint…
- …until set of constraints is a good approximation.
Illustrative Example

Original SVM Problem
- Exponential constraints
- Most are dominated by a small set of “important” constraints

Structural SVM Approach
- Repeatedly finds the next most violated constraint…
- …until set of constraints is a good approximation.
Illustrative Example

Original SVM Problem
- Exponential constraints
- Most are dominated by a small set of “important” constraints

Structural SVM Approach
- Repeatedly finds the next most violated constraint…
- …until set of constraints is a good approximation.
Illustrative Example

Original SVM Problem
- Exponential constraints
- Most are dominated by a small set of “important” constraints

Structural SVM Approach
- Repeatedly finds the next most violated constraint…
- …until set of constraints is a good approximation.

*This is known as a “cutting plane” method.
## Results

<table>
<thead>
<tr>
<th>Unary term</th>
<th>Pairwise term</th>
<th>Score (Jaccard index)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear SVM</td>
<td>none</td>
<td>73 %</td>
</tr>
<tr>
<td>RBF SVM</td>
<td>none</td>
<td>75 %</td>
</tr>
<tr>
<td>RBF SVM</td>
<td>RBF SVM</td>
<td>79 %</td>
</tr>
<tr>
<td><strong>RBF SVM + learned pairwise term</strong></td>
<td></td>
<td><strong>83 %</strong></td>
</tr>
</tbody>
</table>
Future work

- 3d ray features: replace Canny edge detection with a membrane classifier.
- Introduce user interactivity.
- Segmentation of synapses, dendrites and axons.
- Define better energy functions, introduce higher order potentials...
Future work

- 3d ray features: replace Canny edge detection with a membrane classifier.
- Introduce user interactivity.
- Segmentation of synapses, dendrites and axons.
- Define better energy functions, introduce higher order potentials...
Credits

• Slides courtesy:
  • Yevgeny Doctor, IP Seminar 2008, IDC
  • Pushmeet Kohli (Efficiently Solving Dynamic Markov Random Fields using Graph Cuts)
  • Ben Taskar (Structured Prediction: A Large Margin Approach, NIPS tutorial)
  • Yisong Yue, Thorsten Joachims (An Introduction to Structured Output Learning Using Support Vector Machines)
Energy minimization

• In general terms, given some problem, we:
  • Formulate the known constraints
  • Build an “energy function” (aka “cost function”)
  • Look for a solution that minimizes it

• If we have no further knowledge:
  • The problem can be NP-Hard (requires exponential solution time)
  • Use slow, generic approximation algorithms for optimization problems (such as simulated annealing)
Focused Ion Beam Scanning Electron Microscopy

<table>
<thead>
<tr>
<th></th>
<th>2&lt;sup&gt;nd&lt;/sup&gt; order</th>
<th>3&lt;sup&gt;rd&lt;/sup&gt; order</th>
<th>4&lt;sup&gt;th&lt;/sup&gt; + order</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 labels</td>
<td>• [kolmogorov-pami-04]: Exact if submodular</td>
<td>• [kolmogorov-pami-04]: Exact if submodular</td>
<td>• [freedman-cvpr-05]: Exact if subclass of submodular</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• [kohli-cvpr-07]: Exact if P&lt;sup&gt;n&lt;/sup&gt; Potts</td>
</tr>
</tbody>
</table>
| 3+ labels        | • [veksler-phdthesis-99]: everywhere-smooth (P) piecewise-smooth (NP) piecewise-constant (NP) (in label difference)  
• [ishikawa-pami-03]: Exact if convex in label difference  
• [schlesinger-emmcvpr-07, ramalingam-cvpr-08]: Exact if submodular | • [ramalingam-cvpr-08]: Exact if submodular | • [kohli-cvpr-07, kohli-cvpr-08]: Approximate if P<sup>n</sup> Potts  
• [ramalingam-cvpr-08]: Submodular and P=NP |