



Published in final edited form as:

Proc IEEE Int Symp Biomed Imaging. 2013 ; 2013: 1304–1307. doi:10.1109/ISBI.2013.6556771.

Semi-Automatic Neuron Segmentation in Electron Microscopy Images Via Sparse Labeling

Cory Jones^{*}, Ting Liu^{*}, Mark Ellisman[†], and Tolga Tasdizen^{*}

^{*}Scientific Computing and Imaging Institute, University of Utah

[†]National Center for Microscopy and Imaging Research, University of California, San Diego

Abstract

We introduce a novel method for utilizing user input to sparsely label membranes in electron microscopy images. Using gridlines as guides, the user marks where the guides cross the membrane to generate a sparsely labeled image. We use a best path algorithm to connect each of the sparse membrane labels. The resulting segmentation has a significantly better Rand error than automatic methods while requiring as little as 2% of the image to be labeled.

Index Terms

connectomics; electron microscopy; semi-automatic segmentation; biological segmentation

1. Introduction

Machine assisted biological image segmentation methods range from manually labelled computer assisted methods [1, 2] to manually refined more automatic methods [3, 4, 5]. Manual methods require a significant time commitment to accurately label images such as is done for membrane detection in electron microscopy (EM) images of the brain in [1]. Active learning methods such as Ilastik [6] attempt to decrease the user input required for generation of training data used in supervised methods. In addition, automatic methods typically require a large dataset of manually labeled images to be used for learning.

In this paper, we propose a new method that seeks to utilize user input in an efficient way to produce highly accurate results with minimal user input. Different than [6] which uses learning on user input to segment the image, we utilize the user input as a starting point for best path finding to segment EM images. We sparsely label the dataset by using gridlines to guide the user in identifying membrane locations and then use a best path algorithm to identify the complete membrane structure. In EM images, cell membranes generally have complete connectivity with only a few exceptions per image so that finding the best path between all labeled membrane in an image results in the correct structure. This differs from manual methods that require membrane tracing [1]. In addition, we introduce a method of membrane labeling that replaces the binary label with an intensity label that allows further improvement through thresholding.

In Section 2, we outline the complete method for obtaining the sparse labeling, the best path algorithm used, and the method for label replacement. Following we will present results

from using this method on two different EM datasets in Section 3. In addition, we show the results of using a learning method for further refinement. Then in Section 4 we present our conclusions and propose future research.

2. Methods

To be able to take advantage of manual labeling and minimize the amount of effort required from the user, we propose using gridlines on an image as a guide to create a sparse sampling of the image. Along each gridline, the person doing the labeling will indicate the locations where the gridline crosses membrane pixels. The result gives us an image of squares where the membrane locations on the borders of each square are known and the labels of all pixels inside each square are entirely unknown. Figure 1 shows an example of the gridlines overlaid on a denoised image with the membrane labels in yellow and non-membrane labels in red.

In general, each of the cells in an EM image are immediately adjacent to another cell so that neighboring cells appear to share a membrane in the image. This is due to dehydration during sample preparation that shrinks the spacing between the cells. Once the membrane on the gridlines have been identified, we take advantage of this near complete connectivity between cell membranes to attempt to find a best path between all membranes on the borders of a single grid square. We then do this for every grid square so that we have complete connectivity of membranes throughout the image. To solve the best path problem, we use Dijkstra's algorithm computed separately using each membrane pixel along the border of the square as a starting node. Then the minimum path between the current starting node and every other membrane pixel on the border of the current grid is labeled as membrane. We use 8-connected neighbors as the edges of the graph and the edge costs are computed as the geometric distance between the current pixel node and its neighboring pixels times the cost (C) for each neighboring pixel where

$$C_n = e^{\lambda \left| \frac{I_n - \text{med}(M)}{\text{med}(M)} \right|}. \quad (1)$$

Here I_n is the intensity value at node n taken from a denoised version of the original image, M is the set of membrane pixels on the border of the current grid square, $\text{med}(M)$ is the median intensity value of M , and λ is a parameter that controls the penalty. The intensity values of I_n for all n are between 0 and 1 so the cost function at locations with intensity nearest to the intensity of the membrane pixels will have values close to 1 and locations with intensity furthest from the intensity of the membrane pixels will have values close to $\exp(\lambda)$. The absolute value is taken because the interior cell structures, such as mitochondria, can be darker than the membrane which would cause nodes in those structures to have a lower cost and create inaccurate paths through the middle of the cell. The resulting best path will primarily stay on membrane areas and only cross non-membrane areas when there is no other feasible path between two membrane points on the grid. Figure 2 shows the results of the paths connecting all of the membrane points in a single square with yellow showing the membrane gridline points, red showing the unlabeled gridline points, and blue showing the paths between them.

Once every grid square has been evaluated for the best path they are concatenated together into a full image where all pixels that have been visited by at least one path will be assigned a membrane label and all other pixels will be assigned a non-membrane label. Because paths between membrane pixels tend to get used multiple times, the membrane is often thin. In addition, the membrane pixels are frequently along the edges of the membrane since that tends to be the shortest distance between two points that has low cost. To clean up places where a single membrane has two separated paths going along it, we perform a morphological closing of the membrane pixels with a shaping element having a diameter similar to half the thickness of the membrane in the image.

The final result of this is a binary image with all of the membrane connected. However, some of these paths will cross non-membrane as a result of the forced requirement that all membrane pixels within a given square be connected. To allow a simple thresholding to be able to improve the resulting image even more by removing some of these paths, we replace the membrane paths with their corresponding denoised image intensity. The non-membrane locations are then assigned the value of 1. This resulting image will in general have lower values corresponding to areas with the highest probability of being membrane and higher values corresponding to areas with lower probability of being membrane.

3. Results

Here we present the results of this algorithm on two different datasets. The first dataset we used for these experiments is a stack of 60 images from a serial section Transmission Electron Microscopy (ssTEM) data set of the *Drosophila* first instar larva ventral nerve cord (VNC) [1]. It has a resolution of $4 \times 4 \times 50$ nm/pixel and each 2D section is 512×512 pixels. The corresponding binary labels were annotated by an expert neuroanatomist who marked membrane pixels with zero and the rest of pixels with one. During the ISBI Electron Microscopy Image Segmentation Challenge 30 images were used for training and the remaining images were used for testing. We have used the 30 images designated as training from this dataset to use for this experiment since there is no learning involved and the testing labels were not made available.

On the gridlines we used simulated user input to ensure that all membrane pixels on each gridline are complete and accurate to provide a best-case result for a given gridspacing. To do this we used the ground truth provided with the datasets and labeled as membrane everywhere that the gridlines crossed the correct membrane label in the ground truth. The parameters were grid-spacings of 25, 50, 75, and 100 pixels with $\lambda = 3$ and a square shaping element with diameter of 5 pixels. The images were denoised using a non-local means denoising algorithm [7] as this has shown to be effective at denoising textured images. The denoised image was used both as the input to the cost function and as the intensity values for replacement once the path finding was complete. To measure accuracy we use the 1 minus pair f-score ($1 - F$) metric as used in the ISBI 2012 segmentation challenge [8]. It is a measure of disagreement in the segmentation and is a useful measure for segmentation quality. Figure 3 shows the image results for 25 and 100 pixel grid spacing and the $1 - F$ results are presented in Table 1. For comparison the results on the same dataset from a

supervised learning method using a multi-scale artificial neural network [3] are also presented.

The second dataset used for these experiments is from the mouse neuropil. This set consists of 70 images size 700×700 manually annotated by an expert electron microscopist that have been separated into 5 bins with 14 images per bin. 1 bin of 14 images (bin 1) was used in training of the MSANN. The other 4 bins were used as different sets of testing images. Figure 4 shows the image results for 25 and 100 pixel grid spacing and the $1 - F$ results are presented in Table 2.

When training data is available, further improvement can be made by utilizing a region merging learning algorithm. Because of the requirement that all membranes be connected to every other membrane within a grid square, oversegmentation is a common reason for the errors. To see what kinds of additional improvement could be done by eliminating these errors, we used a watershed and tree merging learning algorithm [9] to learn and automatically remove some of these oversegmentations. The results are presented in Table 3. In some cases the region merging learning algorithm was able to remove large oversegmentation regions resulting in significant improvement as with bin 4. In the other cases the improvement is more modest, but still evident.

4. Discussion

By selectively labeling just $2/25^{th}$ of the image we were able to achieve $1 - F$ rates that were significantly better than a supervised learning algorithm, and even when reducing the labeling by a factor of 4 further to $2/100^{th}$ of the image we were still able to show a marked improvement. This method also has the advantage of being more immediate in the results than automatic methods. Whereas some supervised methods can take days [3] or even months [4] to train, within a few minutes a user should be able to mark the membrane crossings of the gridlines and get a high accuracy result. In addition, the creation of accurate training data requires complete manual labeling which is also very time intensive.

Using this method with a 25 pixel spacing between grid lines, we were able to achieve very high segmentation accuracy. We were also able to achieve further improvement by applying a region merging algorithm intended to correct oversegmentations. Currently we are working to be able to correct undersegmentations that occur as the grid spacing gets larger. Using adaptive sampling we hope to sample more sparsely in areas where it won't affect the segmentation and sample more fully areas where the accuracy can be improved. Even without these improvements, the positive results that we have been able to show using simulated user input indicate that this method of user interaction can become a viable alternative to both fully automatic and fully manual segmentation methods.

Acknowledgments

This work was supported by NSF IIS-1149299 (TT) and NIH 1R01NS075314-01 (TT,MHE). The mouse neuropil dataset was provided by NCMIR supported by NIH NCR (P41 GM103412) and is made available from the Cell Image Library/Cell Centered Database which is supported by NIH NIGMS (R01 GM082949). The drosophila VNC dataset was provided by the Cardona Lab at HHMI Janelia Farm.

References

1. Cardona A, Saalfeld S, Preibisch S, Schmid B, Cheng A, Pulokas J, Tomanák P, Hartenstein V. An integrated micro- and macroarchitectural analysis of the *Drosophila* brain by computer-assisted serial section electron microscopy. *PLoS Biol.* 2010; 8(no. 10):e1000502. 10. [PubMed: 20957184]
2. Helmstaedter M, Briggman KL, Denk W. High-accuracy neurite reconstruction for high-throughput neuroanatomy. *Nature neuroscience.* 2011; 14(no. 8):1081–1088.
3. Seyedhosseini M, Kumar R, Jurrus E, Guily R, Ellisman M, Pfister H, Tasdizen T. Detection of neuron membranes in electron microscopy images using multiscale context and radon-like features. *MICCAI 2011.* 2011; 6891:670–677. *Lecture Notes in Computer Science (LNCS).*
4. Ciresan DC, Giusti A, Gambardella LM, Schmidhuber J. Deep neural networks segment neuronal membranes in electron microscopy images. *NIPS.* 2012
5. Jain V, Murray JF, Roth F, Turaga S, Zhigulin V, Briggman KL, Helmstaedter MN, Denk W, Seung HS. Supervised learning of image restoration with convolutional networks. *ICCV 2007.* Oct.2007 : 1–8.
6. Sommer C, Straehle C, Koethe U, Hamprecht FA. *ilastik: Interactive learning and segmentation toolkit.* *ISBI.* 2011
7. Buades, A.; Coll, B.; Morel, JM. A non-local algorithm for image denoising. *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference;* june 2005; p. 60-65.vol. 2
8. [accessed on November 6, 2012] Segmentation of neuronal structures in EM stacks challenge - ISBI 2012. http://fiji.sc/Segmentation_of_neuronal_structures_in_EM_stacks_challenge_-_ISBI_2012
9. Liu T, Jurrus E, Seyedhosseini M, Ellisman M, Tasdizen T. Watershed merge tree classification for electron microscopy image segmentation. *ICPR.* 2012

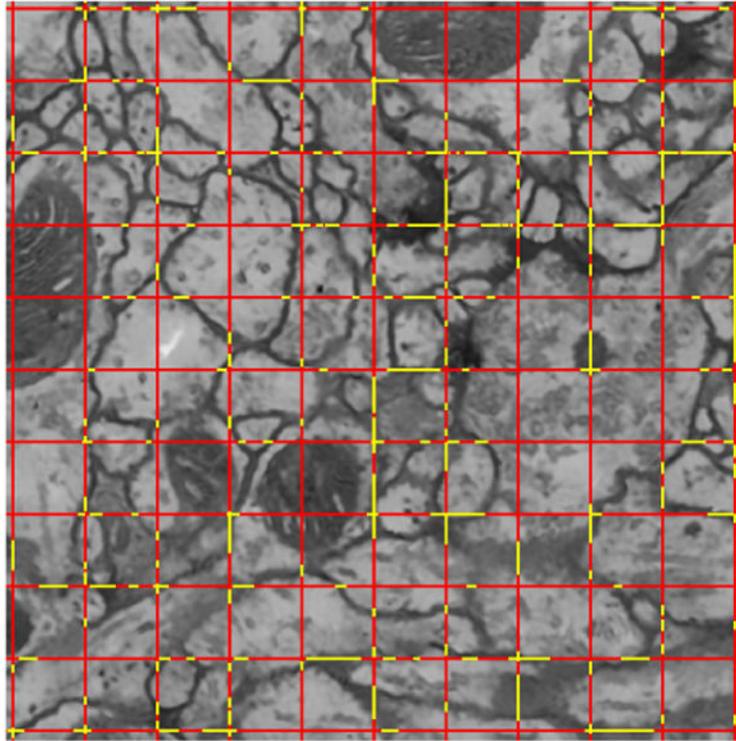


Fig. 1. Example of grid labeling of cell membranes. Red represents the non-membrane sections on the gridlines and yellow represents the membrane labeled gridlines.

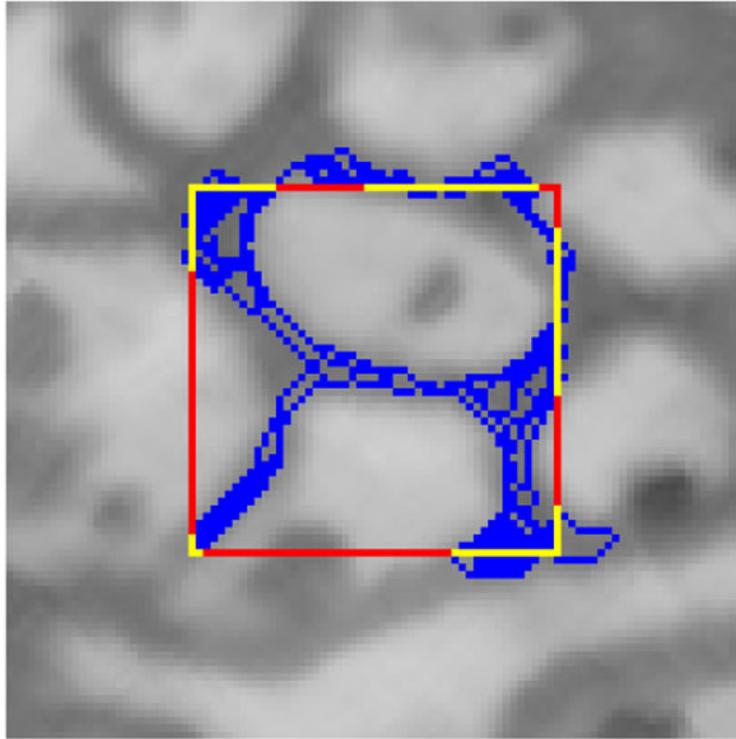


Fig. 2. Example of the paths found within a single grid square. Red and yellow represent non-membrane and membrane sections on the gridlines respectively. Blue represents the best paths connecting all membrane pixels on the gridlines.

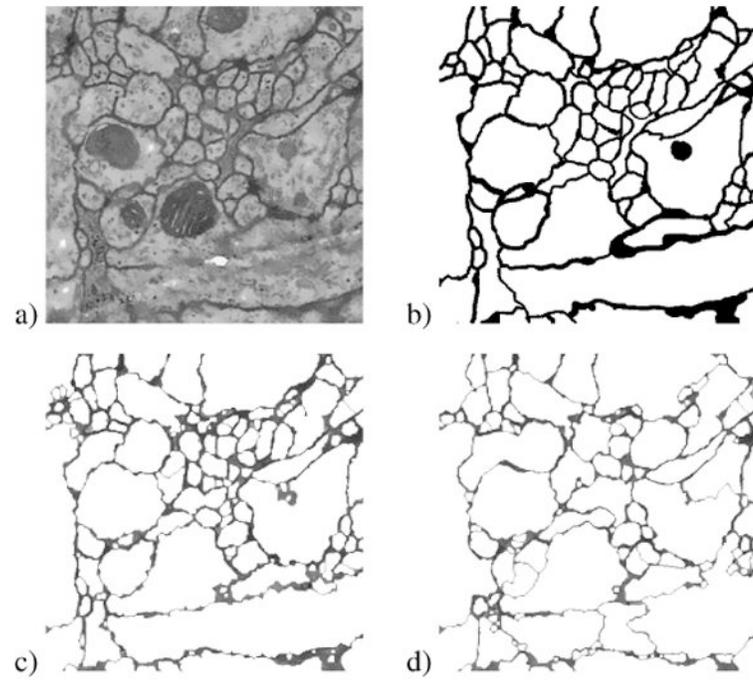


Fig. 3. Example of (a) original image, (b) ground truth, (c) 25, and (d) 100 pixel grid spacing on the *Drosophila* data set.

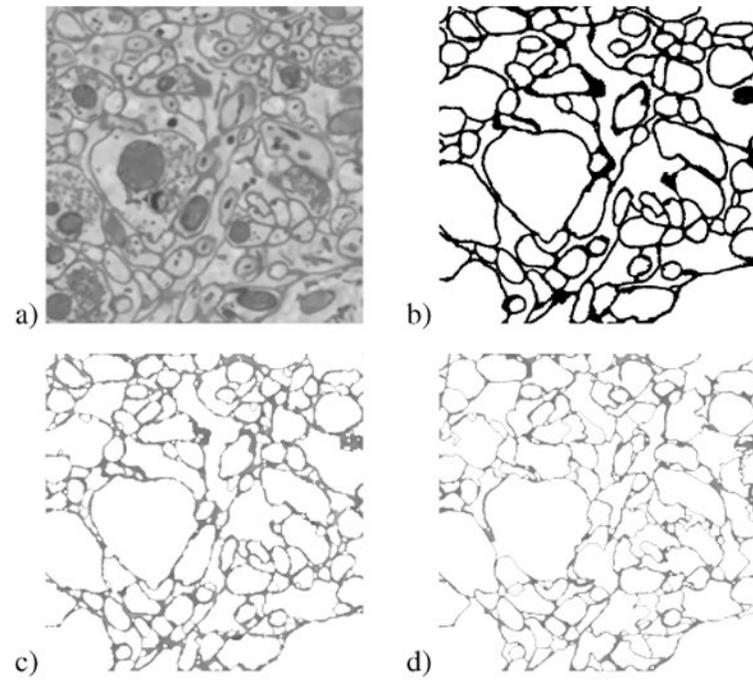


Fig. 4. Example of (a) original image, (b) ground truth, (c) 25, and (d) 100 pixel grid spacing on the mouse neuropil dataset.

Table 1

1 – F performance of the algorithm with different grid spacing using the Drosophila dataset.

	Method				
	25	50	75	100	
Error	0.208	0.049	0.088	0.120	0.169

Table 2

1 – F performance of the algorithm with different grid spacing using the mouse neuropil dataset. Bin 1 was the training bin for the MSANN learning.

	Method				
	MSANN	25	50	75	100
Bin 1	0.1626	0.0291	0.0875	0.1309	0.1591
Bin 2	0.2749	0.0651	0.0908	0.1145	0.1778
Bin 3	0.2419	0.0602	0.0964	0.1319	0.1487
Bin 4	0.2115	0.0328	0.0511	0.0689	0.1266
Bin 5	0.2717	0.0539	0.0819	0.1003	0.1399

Table 3

1 – F performance of the algorithm with different grid spacing followed by region merging using the mouse neuropil dataset. For each reported bin, that bin was used as the test data and the other 4 bins were used as the training data.

	Method			
	25	50	75	100
Bin 1	0.0217	0.0791	0.1162	0.1508
Bin 2	0.0537	0.0770	0.1028	0.1424
Bin 3	0.0454	0.0684	0.1031	0.1214
Bin 4	0.0229	0.0385	0.0629	0.0825
Bin 5	0.0436	0.0736	0.0841	0.1225