Implicit Boundary Learning for Connectomics

Tobias Maier and Thomas Vetter(✉)

Department of Mathematics and Computer Science, University of Basel, Basel, Switzerland
{tobias.maier,thomas.vetter}@unibas.ch

Abstract. Segmentation of complete neurons in 3D electron microscopy images is an important task in Connectomics. A common approach for automatic segmentation is to detect membrane between neurons in a first step. This is often done with a random forest. We propose a new implicit boundary learning scheme that optimizes the segmentation error of neurons instead of the classification error of membrane. Given a segmentation, optimal labels for boundary between neurons and for non-boundary are found automatically and are used for training. In contrast to training random forests with labels for membrane and intracellular space, this novel training method does not require many labels for the difficult to label membrane and reduces the segmentation error significantly.

1 Introduction

Connectomics studies the structure and function of neuronal circuits. One approach to obtain these circuits is to reconstruct neuronal networks from 3D electron microscopy (3DEM) images. These 3D images are obtained by stacking 2D images. An example of such an image is given in Figure 1. With serial block face scanning electron microscopes [1] and focus ion beam scanning electron microscopes (FIB-SEM) the resolution in all three dimensions is high enough for an analysis in 3D. With FIB-SEM even isotropic resolution is possible. With these microscopes the acquisition is almost completely automated while for reconstruction the state of the art is still manual reconstruction. Skeleton tracings are made by hundreds of undergraduate students [2] or supervoxels are merged in the online game EyeWire[1] from more than 80 000 people.

Various approaches are proposed for the automatic reconstruction of 3DEM data. Some methods use segmentations from 2D slices and merge them to a consistent segmentation in 3D [3,4]. Laptev et al. [5] do a segmentation per slice and use neighboring slices to enrich the features for the slice. Andres et al. [6] presented a method that is working truly in 3D. All these methods have different approaches for the segmentation but all start with a random forest to detect membranes in 3DEM images.

Liu et al. proposed a hierarchical segmentation method [7] that uses learned membrane probability maps. In their experiments they use probability maps

[1] http://eyewire.org

V. Murino and E. Puppo (Eds.): ICIAP 2015, Part I, LNCS 9279, pp. 39–49, 2015.
DOI: 10.1007/978-3-319-23231-7_4

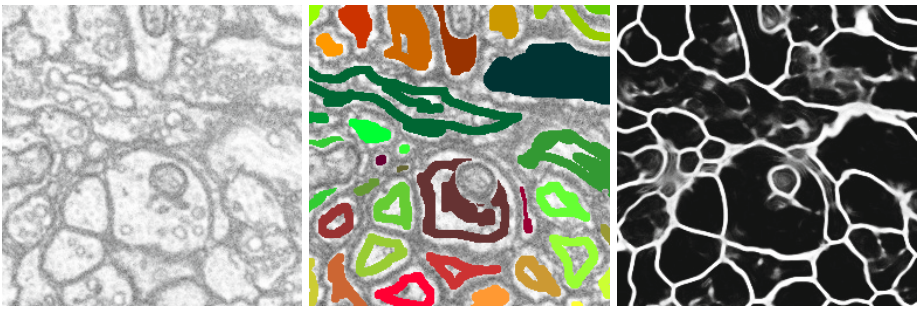


Fig. 1. From left to right: input image, rough labels of the intracellular regions overlayed on the input data, predicted probability map. Examples of the probability map on test data are shown in Figure 6.

learned with cascaded hierarchical models [8] and a deep neuronal network (DNN) from Ciresan et al. [9]. The drawback of DNNs is their requirement for much training data and long training time often on special hardware clusters. In contrast, random forests can be trained on normal workstations in a reasonable time.

Other approaches try to learn the segmentation directly. Turaga et al. [10] presented an online learning scheme for convolutional neuronal networks (CNN) that optimize the Rand index. Similarly Jain et al. [11] introduced the "warping error" and traind a CNN to optimize for this error. A supervised learning of image partitioning is proposed in [12]. A decision tree is learned in a greedy manner by selecting the split that maximizes the Matthews correlation coefficient.

When using a random forest for segmentation, the usual approach is to train a two class random forest with one class for membrane and one for intracellular space (including organelles like vesicles and mitochondria). The labels are given in advance or obtained interactively with the feedback of the prediction (e.g. with ilastik [13]). The random forest training optimizes the classification error given the training data. It is up to the user to select a set of labeled data that is appropriate for the segmentation. Therefore, labels where the output of the random forest leads to segmentation errors are important. These regions are also often difficult to label for a human. Cell membranes are thin elongated structures, sometimes only single pixels wide and occupy only a small fraction of the space. This makes it hard to get many accurate labels especially where the membrane is blurred, close to another membrane or touching organelles like mitochondria.

We introduce a new method to automatically select and label training data for a random forest. Instead of labels for membrane a rough segmentation of intracellular space as shown in Figure 1 is used. In an iterative process, a random forest is trained, the boundary prediction is analyzed in respect of the segmentation and optimal points for boundary and non-boundary are added to the training data. The segmentation error is reduced significantly compared to

training with dense membrane labels or ilastik. In Figure 1 an example of the input data, labels used for our method, and the predicted boundary is shown.

In Section 2 we describe the details of the implicit boundary learning scheme. An evaluation and comparison to training with other methods is given in Section 3. Conclusion follows in Section 4.

2 Method

To learn a random forest for predicting a boundary that optimizes a segmentation we go for a hybrid approach. Using some initial labels, a two-class random forest is trained to detect an initial boundary. After training, it is applied to the complete training volume. New training samples are selected where segmentation errors occur. Then the random forest is retrained with the additional training data. This is repeated until convergence. The most important and difficult task is to get new training samples that improve the segmentation. In active learning samples close to the decision boundary are selected and the user is asked to label them. But these samples are not necessarily critical for the segmentation. In ilastik [13], the user searches for regions where the prediction is unsatisfying and adds new labels there. This is where membrane has low membrane probabilities or intracellular space is detected as membrane. In difference to that, our method automatically identifies regions where segmentation errors occur and adds labels there.

2.1 Finding Labels to Optimize Segmentation

For the reconstruction of neuronal circuits there are two important types of segmentation errors: *false mergers* and *false splits*. A false merger erroneously connects two regions to a common segment. A false split erroneously divides a region in two segments. To evaluate these errors a segmentation model is needed to transform the predicted probability into a segmentation. We use a threshold segmentation. All pixels with a boundary probability above a threshold are removed. The remaining connected components build the segmentation.

We define a path γ from a pixel x_s to another pixel x_e on the pixel grid (2D or 3D) as

$$\gamma(x_s, x_e) = \{x_s, x_a, x_b, \ldots, x_e\}, \tag{1}$$

where $x_s, x_a, \ldots$ are unique neighboring pixels, i.e. the edges $(x_s, x_a), (x_a, x_b), \ldots$ are edges in the neighborhood graph. The path of all paths from x_s to x_e for which the maximum pixel value on the path is minimal we define as minimax path:

$$\gamma^*(x_s, x_e) = \arg \min_{\gamma \in \Gamma(x_s, x_e)} \left(\max_{x \in \gamma} p(x) \right), \tag{2}$$

where $\Gamma(x_s, x_e)$ is the set of all paths from x_s to x_e, and $p(x)$ is the pixel value.

Using the threshold segmentation model, two points are in the same segment if and only if there exists a path in the boundary probability map with all

pixels on the path below or equal a threshold. We can formulate the following conditions for false mergers and splits. A false merger occurs if

$$\max_{x \in \gamma^*(x_s, x_e)} p(x) \leq \text{threshold}, \tag{3}$$

where x_s and x_e are from different segments. A false split is induced if

$$\max_{x \in \gamma^*(x_s, x_e)} p(x) > \text{threshold}, \tag{4}$$

with x_s and x_e from the same segment.

The false split and false merge both happen at the maximum of the minimax path:

$$x^*(x_s, x_e) = \arg\max_{x \in \gamma^*} p(x). \tag{5}$$

This point can be found efficiently. We define the weight of the edges between neighboring pixels x_i, x_j as:

$$w_{ij} = \max(p(x_i), p(x_j)), \tag{6}$$

and use Kruskal's algorithm to find the minimum spanning tree (MST). From [14] it is known that for any edge (x_i, x_j) that is not part of the MST it is $w_{ij} \geq \max(w_{ia}, w_{ab}, \ldots, w_{dj})$. With our definition of the edge weight, the maximal edge weight on the path equals the maximal pixel value on the path. From that follows that $\gamma^*(x_s, x_e)$ is part of the MST. x^* can then be found by searching the maximum on the unique path between between x_s and x_e in the MST. This is illustrated in Figure 2.

To label new data and add it to the training data, pairs (x_s, x_e) are sampled. For x_s and x_e from different segments, x^* is labeled as "membrane". For x_s and x_e from the same segments, x^* is labeled as "intracellular". The points x^* are added to the training data regardless of their value $p(x^*)$. This means no threshold has to be specified for training.

For many pairs (x_s, x_e) the point x^* is the same. For example, in Figure 2 all points from the red and blue segment share the same maximum x^* on their minimax path. Sampling x_s and x_e spatially close and restricting the region of interest where the minimax path is searched gives more unique training data. To keep the labels and training balanced, the same number of new training data is added for boundary and non-boundary.

For the false mergers it can easily be seen that adding the maximum of the minimax path to the training data is reasonable. For false splits this is not that obvious. The path $\gamma^*(x_s, x_e)$ is the path with the lowest maximum between x_s and x_e, why should a new training label be taken from this path? Let's compare it to other methods to obtain a subset of training data from a segmentation.

(a) Randomly sample n points in the intracellular area. This is the simplest method but gives no control to select a subset of informative labels.
(b) Choose n worst classified points that are labeled as intracellular space. While this may reduce the classification error of the intracellular space, it is not known if a better classification of these points improves the segmentation.

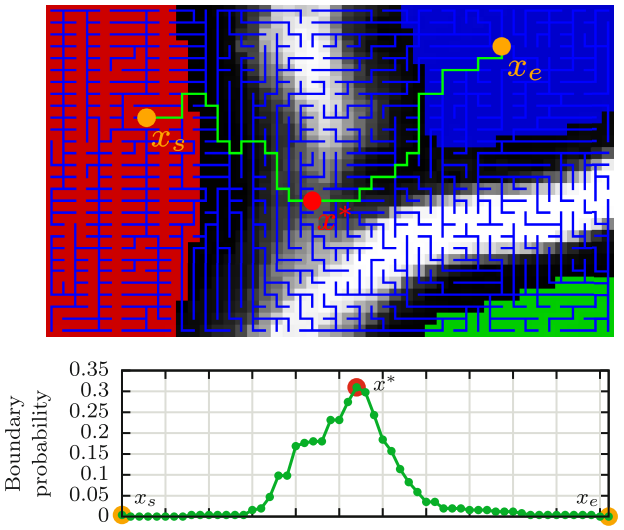


Fig. 2. Illustration of the minimax path $\gamma^*(x_s, x_e)$ (green). The corresponding probabilities for boundary on this path are plotted below. The point with the highest boundary probability is the minimax point x^*. All minimax paths in this region are part of the minimum spanning tree (blue). Segment labels are shown as red, green and blue regions. Note that for all paths between the red and blue segment, the shown minimax point x^* is the same.

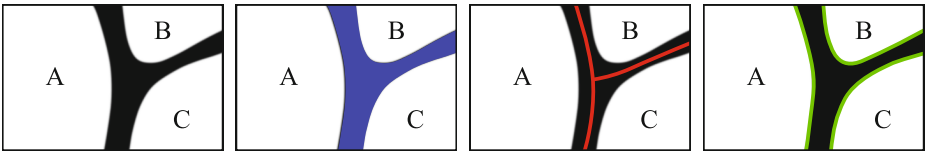
(c) Randomly sample n pairs of points, with each pair within the same segment, choose a point from the minimax path between the points. This selection gives the only points that are known to be critical for the segmentation. Choosing a point from another path, such as a random path, a minimal (spatial) path, gives no such information.

To initialize the method an initial prediction for boundary is required. For this we use a random forest that is trained with a few labeled pixels for membrane and non-membrane. The method is robust against the initialization. Even the intensity of the (inverted) input image can be used as initial boundary prediction.

After the selection of additional training data a new random forest is trained. The procedure is repeated until no more improvement in the segmentation is obtained or a maximum number of iterations is reached. The required number of iterations depends on the number of new training samples that are added in each iteration.

2.2 Difference to Learn Membrane

In neuronal tissue neurons are enclosed by a cell membrane that is visible in 3DEM images. Often a classifier is trained to detect this membrane and use it for segmentation. This is not the same as the boundary in our approach. We



(a) (b) (c) (d)

Fig. 3. (a) shows three segments A, B and C that are separated by a membrane. (b) shows a labeling for membrane in blue. It is also a valid boundary that separates the three segments. (c) and (d) show other possible boundaries that separate the segments correctly.

learn the boundary from the segmentation and do not enforce it to match the membrane. In Figure 3 an example for membrane and boundary is given. While the membrane is a valid boundary there exist other boundaries that also lead to a correct segmentation. By not enforcing the boundary to match the membrane a boundary that is easier for the classifier can be found.

3 Experiments and Results

We evaluate the proposed method on a FIB-SEM stack of the olfactory bulb from zebrafish larvae. The data is isotropic with a resolution of 6 nm. For training and testing we use two non overlapping regions of size $512 \times 512 \times 256$ and $300 \times 300 \times 300$ respectively. For both volumes we have a gold standard segmentation that is obtained by manual labeling with the interactive method from [15].

For all random forests the same implementation and the same set of features from the 3D rotation invariant Harmonic filters [16] in multiple scales are used. With these features we get good results. However, cell membranes and the membrane from mitochondria can not be differentiated. This results in mitochondria becoming segments within the neurons. Special designed features and segmentation methods for mitochondria [17,18] allow to detect and process them in a later step. We leave mitochondria and its membrane unlabeled for all experiments. This does neither enforce mitochondria to become segments of its own, nor being part of the neurons segment.

For anisotropic data the resolution in the z-direction is often insufficient leading to gaps between the membrane of consecutive slices. This allows an analysis only within the high resolution slice. With isotropic data like the FIB-SEM data the analysis can be done in 3D. We test if there is an advantage with this method working in 3D. Two random forests are trained with different sampling schemes. For the first, the pairs of points are sampled in a volume and the minimax paths between these points are calculated in 3D. For the second, the pairs of points are sampled within a slice parallel to the xy-, xz-, or yz-plane. The minimax path is then calculated only in this slice. In 2D we use a 4-connected neighborhood, in 3D a 6-connected neighborhood. While the training with the

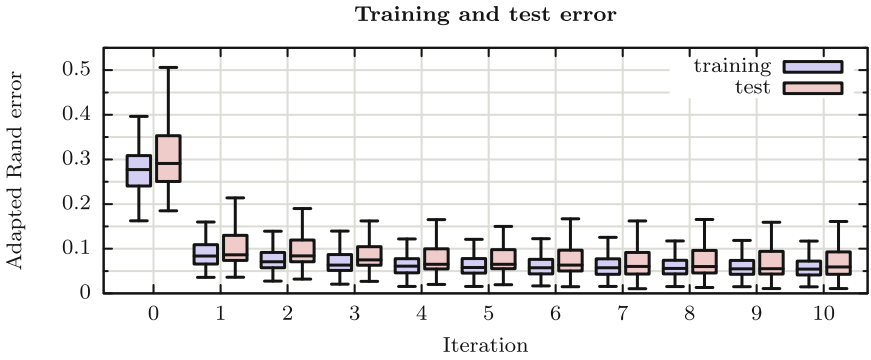


Fig. 4. Adapted Rand error for random forest trained with the given number of iterations. Iteration 0 is from the initial random forest trained with a few strokes for membrane and intracellular space. The segmentation error is evaluated for each slice and the distribution over the stack is shown.

Table 1. Segmentation errors on the test data. Each slice is evaluated with the same threshold and the error is calculated over all slices. For each method the optimal threshold is used.

	Pixel error	Warping error	Adapted Rand error
BEL[19]	0.123	0.00112	0.271
MSANN[20]	0.130	0.00111	0.176
ilastik	0.144	0.00128	0.227
dense labels	0.138	0.00081	0.177
implicit	**0.121**	**0.00050**	**0.106**

method working in 3D takes much longer than working only in-plane, there is no difference in the segmentation error. Therefore, we use in-plane sampling also for isotropic 3D data.

To analyze the convergence in training and the generalization to the test data we evaluate the segmentation in each iteration. In each iteration 5 000 new labels for boundary and non-boundary are added. This parameter depends on the size of the training stack but is not critical for the resulting random forest. More iterations are required for smaller values, for larger values more point pairs must be sampled to get new unique labels. The adapted Rand error is calculated for each slice individually and the distribution is shown in Figure 4. The largest improvement is gained in the 1st iteration. The next iterations show a further reduction of the adapted Rand error and the improvement converges rapidly. The test error is slightly higher than the training error with a larger variance. On the test data the adapted Rand error also converges after a few iterations. It is sufficient to train only a few iterations, what keeps the training time short. With further iterations the test error does not increase and no overfitting to the training data is visible.

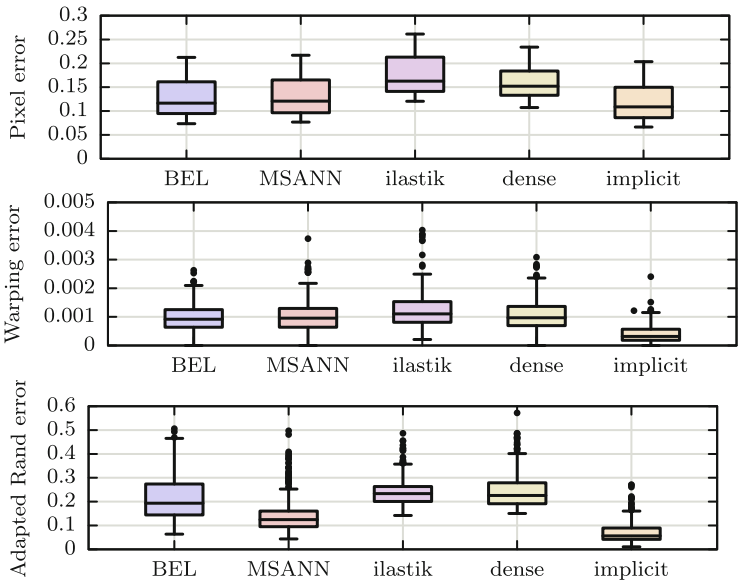


Fig. 5. Distribution of pixel error, warping error and adapted Rand error, evaluated for each slice in the test data. The methods used are BEL [19], MSANN [20], and random forests trained with labels obtained interactively with ilastik, dense labels and our proposed method.

We compare our method to random forests trained with labels from interactive labeling using ilastik and dense labels as well as boosted edge learning (BEL) [19] and multi-scale contextual model (MSANN) [20]. The dense membrane labels are obtained from the gold standard segmentation and used for BEL, MSANN and random forest training. The implicit boundary learning uses the segmentation labels directly. With ilastik an expert collected 9110 labels for membrane and 9104 labels for the intracellular space. For balanced training of the random forest with dense labels a subset of 50 000 labels per class is taken randomly. Training is done for 10 iterations with 5 000 new points for boundary and non-boundary in each iteration. If more than 50 000 labels per class are present only a subset, like for the training with dense labels, is used. The segmentation errors are calculated for the threshold that gives the lowest error. For BEL the output is filtered with a Gauss kernel ($\sigma = 1$) what gives better results than using it directly. The pixel error, warping error [11] and adapted Rand error of the test volume are given in Table 1. Additionally, the errors are calculated per slice (each with the optimal threshold) and the distribution is shown in Figure 5. Note that the pixel error is for the segment labels and not for classification of membrane. When evaluated per slice, BEL and MSANN give similar warping errors than the random forest with dense labels, but if the error is calculated with a common threshold over the volume the random forest

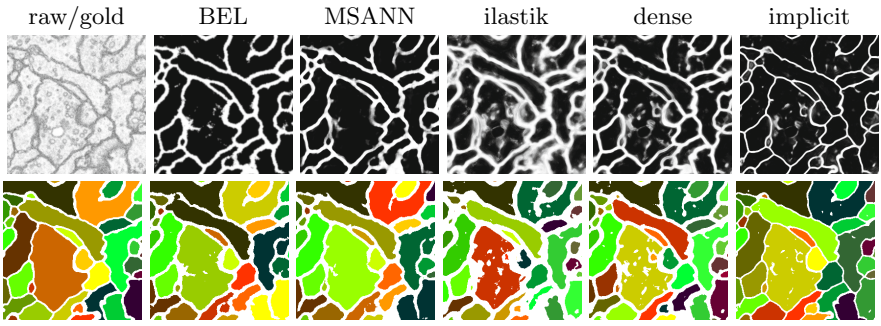


Fig. 6. The first column shows the input slice and the gold standard segmentation. In the next columns the probability maps for membrane / boundary and the corresponding segmentations are shown for the different methods.

gives better results. For the adapted Rand error MSANN seems better than the random forest with dense labels (but note that there are many outliers). If the error is evaluated over the volume they have almost the same error. With all error measures the proposed implicit boundary learning has the lowest error. For warping error and adapted Rand error the error was reduced about 40% compared to a random forest trained with dense labels and even more compared to a random forest trained with labels from ilastik. The interactive labeling with ilastik has the largest segmentation errors, but the labeling effort is the smallest.

We focus on the training of the random forests, not on segmentation. Therefore, only thresholding is used to get a segmentation. This approach is limited in practical use. Tiny holes in the boundary can produce huge errors in the segmentation. But we think that segmentation methods like [3,4,6] can also benefit of using random forests trained with our method. In Figure 6 the predicted probability maps for membrane / boundary and the resulting segmentation are shown. The output from our method shows sharp lines with high probability for boundary at the membranes and very low probabilities in the remaining regions. The output from the random forest trained with ilastik or dense labels are less distinct and blurry. The smaller regions in the upper left corner are hardly visible in the output from ilastik while they are clearly visible with the proposed method. BEL and MSANN have clear boundaries. But where the membrane is not detected correctly it gets almost deleted, leading to holes in the boundary.

4 Conclusion

We have presented a new method to train a random forest for segmentation where the boundaries are learned implicitly from a given segmentation. The segmentation error with a threshold segmentation of the boundary probability is reduced significantly compared to supervised learning with dense labels or labels from interactive labeling with ilastik. Methods that use membrane probability maps [3,4,6,7] can be used with the boundary probability maps and improve the segmentation accuracy further.

References

1. Denk, W., Horstmann, H.: Serial block-face scanning electron microscopy to reconstruct three-dimensional tissue nanostructure. PLoS Biol **2**(11) (2004)
2. Helmstaedter, M., Briggman, K.L., Turaga, S.C., Jain, V., Seung, H.S., Denk, W.: Connectomic reconstruction of the inner plexiform layer in the mouse retina. Nature **500**(7461) (2013)
3. Vazquez-Reina, A., Gelbart, M., Huang, D., Lichtman, J., Miller, E., Pfister, H.: Segmentation fusion for connectomics. In: ICCV (2011)
4. Funke, J., Andres, B., Hamprecht, F., Cardona, A., Cook, M.: Efficient automatic 3D-reconstruction of branching neurons from EM data. In: CVPR (2012)
5. Laptev, D., Vezhnevets, A., Dwivedi, S., Buhmann, J.M.: Anisotropic ssTEM image segmentation using dense correspondence across sections. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (eds.) MICCAI 2012, Part I. LNCS, vol. 7510, pp. 323–330. Springer, Heidelberg (2012)
6. Andres, B., Kroeger, T., Briggman, K.L., Denk, W., Korogod, N., Knott, G., Koethe, U., Hamprecht, F.A.: Globally optimal closed-surface segmentation for connectomics. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part III. LNCS, vol. 7574, pp. 778–791. Springer, Heidelberg (2012)
7. Liu, T., Jones, C., Seyedhosseini, M., Tasdizen, T.: A modular hierarchical approach to 3d electron microscopy image segmentation. Journal of Neuroscience Methods **226** (2014)
8. Seyedhosseini, M., Sajjadi, M., Tasdizen, T.: Image segmentation with cascaded hierarchical models and logistic disjunctive normal networks. In: 2013 IEEE International Conference on Computer Vision (ICCV), pp. 2168–2175, December 2013
9. Ciresan, D.C., Giusti, A., Gambardella, L., Schmidhuber, J.: Deep neural networks segment neuronal membranes in electron microscopy images. In: NIPS (2012)
10. Turaga, S.C., Briggman, K.L., Helmstaedter, M., Denk, W., Seung, H.S.: Maximin affinity learning of image segmentation. In: NIPS (2009)
11. Jain, V., Bollmann, B., Richardson, M., Berger, D., Helmstaedter, M., Briggman, K., Denk, W., Bowden, J., Mendenhall, J., Abraham, W., Harris, K., Kasthuri, N., Hayworth, K., Schalek, R., Tapia, J., Lichtman, J., Seung, H.: Boundary learning by optimization with topological constraints. In: CVPR (2010)
12. Straehle, C., Koethe, U., Hamprecht, F.A.: Weakly supervised learning of image partitioning using decision trees with structured split criteria. In: ICCV (2013)
13. Sommer, C., Straehle, C., Koethe, U., Hamprecht, F.A.: Ilastik: Interactive learning and segmentation toolkit. In: ISBI (2011)
14. Hu, T.C.: The Maximum Capacity Route Problem. Operations Research **9**(6) (1961)
15. Straehle, C.N., Köthe, U., Knott, G., Hamprecht, F.A.: Carving: scalable interactive segmentation of neural volume electron microscopy images. In: Fichtinger, G., Martel, A., Peters, T. (eds.) MICCAI 2011, Part I. LNCS, vol. 6891, pp. 653–660. Springer, Heidelberg (2011)
16. Schlachter, M., Reisert, M., Herz, C., Schlurmann, F., Lassmann, S., Werner, M., Burkhardt, H., Ronneberger, O.: Harmonic filters for 3d multichannel data: Rotation invariant detection of mitoses in colorectal cancer. IEEE Transactions on Medical Imaging **29**(8) (2010)
17. Smith, K., Carleton, A., Lepetit, V.: Fast ray features for learning irregular shapes. In: ICCV (2009)

18. Lucchi, A., Smith, K., Achanta, R., Knott, G., Fua, P.: Supervoxel-Based Segmentation of Mitochondria in EM Image Stacks With Learned Shape Features. IEEE Transactions on Medical Imaging **31**(2), 474–486 (2012)
19. Dollar, P., Tu, Z., Belongie, S.: Supervised learning of edges and object boundaries. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2 (2006)
20. Seyedhosseini, M., Kumar, R., Jurrus, E., Giuly, R., Ellisman, M., Pfister, H., Tasdizen, T.: Detection of neuron membranes in electron microscopy images using multi-scale context and radon-like features. In: Fichtinger, G., Martel, A., Peters, T. (eds.) MICCAI 2011, Part I. LNCS, vol. 6891, pp. 670–677. Springer, Heidelberg (2011)