This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIP.2013.2274388

Multi-class Multi-scale Series Contextual Model for Image Segmentation

Mojtaba Seyedhosseini, Tolga Tasdizen, *Senior Member, IEEE*

***Abstract*—Contextual information has been widely used as a rich source of information to segment multiple objects in an image. A contextual model utilizes the relationships between the objects in a scene to facilitate object detection and segmentation. However, using contextual information from different objects in an effective way for object segmentation remains a difficult problem. In this paper, we introduce a novel framework, called *multi-class multi-scale (MCMS)* series contextual model, which uses contextual information from multiple objects and at different scales for learning discriminative models in a supervised setting. The MCMS model incorporates cross-object and inter-object information into one probabilistic framework and thus is able to capture geometrical relationships and dependencies among multiple objects in addition to local information from each single object present in an image. We demonstrate that our MCMS model improves object segmentation performance in electron microscopy images and provides a coherent segmentation of multiple objects. By speeding up the segmentation process, the proposed method will allow neurobiologists to move beyond individual specimens and analyze populations paving the way for understanding neurodegenerative diseases at the microscopic level.**

***Index Terms*—Image segmentation, Contextual information, Artificial neural networks, Series classifier, Electron microscopy imaging, Neuroscience, Connectomics**

I. Introduction

SHAPE contexts are extremely rich descriptors [1] that have been used widely for solving high-level vision problems. Contextual information is interpreted as intra-object configurations and inter-object relationships [2]. These attributes play an important role in scene understanding [3], [4], [5]. For example, the existence of a keyboard in an image suggests that there is very likely a mouse near it [6]. To be precise, by contextual information we refer to the probability image map of the target object which can be used as prior information together with the original image information to solve the maximum aposteriori (MAP) pixel classification problem. Pixel classification is the problem of assigning an object label to each pixel.

There have been many methods that employ context for solving vision problems such as image segmentation or image classification. Markov random fields (MRF) [7] is one of the earliest and most widespread approaches. Lafferty *et al.* [8] showed that better results for discrimination problems can be obtained by modeling the conditional probability of labels given an observation sequence directly. This non-generative approach is called the conditional random field (CRF). He *et al.* [9] generalized the CRF approach for the pixel classification problem by learning features at different scales of the image. Jain *et al.* [10] showed MRF and CRF algorithms perform about the same as simple thresholding in pixel classification for binary-like images. They proposed a new single-scale version of the convolutional neural network [11] strategy for restoring membranes in electron microscopic (EM) images. Compared to other methods, convolutional networks take advantage of context information from larger regions, but need many hidden layers. In their model the back propagation has to go over multiple hidden layers for the training, which makes the training step computationally expensive. Tu and Bai [2] proposed the auto-context algorithm which integrates the original image features together with the contextual information by learning a series of classifiers. Similar to CRF, auto-context targets the posterior distribution directly without splitting it to likelihood and prior distributions. The advantage of auto-context over convolutional networks is its easier training due to treating each classifier in the series one at a time in sequential order. Although they used probabilistic boosting tree as classifier (PBT), auto-context is not restricted to any particular classifier and different type of classifiers can be used. Jurrus *et al.* [12] employed artificial neural networks (ANN) in a series classifier structure which learns a set of convolutional filters from the data instead of applying large filter banks to the input image.

Even though all the aforementioned approaches use contextual information together with the input image information to improve the accuracy of the achieved segmentation, they do not take contextual information from multiple objects into account and thus are not able to capture dependencies between the objects. Torralba *et al.* [6] introduced boosted random field (BRF) which uses boosting to learn the graph structure of CRFs for multi-class object detection and region labeling. Desai *et al.* [13] proposed a discriminative model for multi-class object recognition that can learn intra-class relationships between different categories. The cascaded classification model [14] is a scene understanding framework that combines object detection, multi-class segmentation, and 3D reconstruction. Choi *et al.* [15] introduced a tree-based context model which exploits dependencies among objects together with local features to improve the object detection accuracy.

While contextual models have been shown to be successful in several computer vision tasks, we propose a more effective way of extracting information from the context image, i.e., the classifier output. We develop a novel framework

Copyright (c) 2013 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

M. Seyedhosseini and T. Tasdizen are with the Electrical and Computer Engineering Department and the Scientific Computing and Imaging Institute (SCI), University of Utah, Salt Lake City, UT, 84112 USA. Email: {mseyed,tolga}@sci.utah.edu

Copyright (c) 2013 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIP.2013.2274388

that exploits contextual information from different scales and different objects to learn a discriminative model for object segmentation. To our knowledge, multi-class and multi-scale contextual information have not been previously used in a unified framework for object segmentation. The combination of multi-class and multi-scale schemes enables our method to make extensive use of contextual information and thus improves the segmentation accuracy.

We employ the series architecture in [12] and modify it in two important ways to provide more informative contextual information to the classifiers:

1) Multi-scale contextual model: We apply a series of simple linear filters to the context image consecutively to generate a scale-space representation of the context and give the classifier access to samples of the scale space. The samples of the coarser scales are more informative and robust against noise due to the averaging. Therefore, this framework provides more information from the context for the classifier in a similar number of features.

2) Multi-class contextual model: We also introduce the multi-class series architecture by allowing the classifier for each object type access to the contextual information from each object type of the previous stage. This flow of cross-object information is achieved by feeding neighborhoods from the output of each classifier in the current stage to each classifier in the next stage. The proposed multi-class framework is able to capture geometric relationships of objects and their dependencies which can be an important clue to their identity. For instance, the existence of mitochondria, i.e., the objects with green boundary in Figure 1, at a certain position in an electron microscopy image is a strong evidence that the existence of synapses, i.e. the objects with yellow boundary in Figure 1, is unlikely. Synapses are more likely in certain configurations and distances to cell membranes, i.e., the red objects in Figure 1.

We introduce a novel and powerful segmentation framework by employing multi-scale and multi-class contextual model in a series classifier architecture. The multi-class multi-scale (MCMS) series contextual model is able to leverage both the cross-object and the inter-object contextual information at multiple scales to give a coherent segmentation of multiple objects present in an image. The rich contextual information that the MCMS model extracts from the image helps the later classifiers to correct the mistakes of the early stages and thus improves the overall performance.

Our model is motivated by the problem of reconstruction of the connectome, i.e., the map of connectivity of all neurons in the mammalian nervous system [18], which is a challenge facing neuroscientists [12]. Electron microscopy (EM) is an image acquisition technique that can generate high resolution images with enough details for this problem [19]. However, the reconstruction of the connectome remains a challenging problem because of the noisy texture, irregular shapes, complex structures, and the large variations in the physical topologies of cells [10], [20]. Moreover, the sheer size of a typical EM dataset, often approaching tens of terabytes [21], makes manual analysis infeasible [22]. Hence, automated segmentation methods are required.

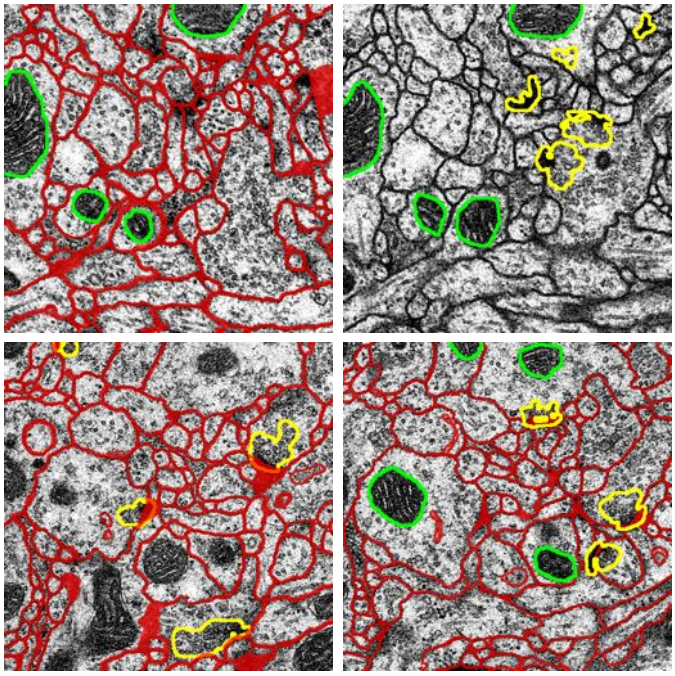


Fig. 1. Different objects appear in certain configurations to each other. For example synapses, i.e., objects with yellow boundary, are close to membrane, i.e., red objects, and usually overlap with them. Mitochondria, i.e., objects with green boundary, are far from membranes and never overlap with synapses. Using this information can improve the segmentation results for each of these objects. The images are from a serial section Transmission Electron Microscopy (ssTEM) dataset of the Drosophila first instar larva ventral nerve cord [16], [17].

General segmentation methods which have been proposed for natural image datasets yield poor results when applied to EM images [20]. Jain et.al. [23] showed that multi-scale normalized cut [24], boosted edge learning [25], and global probability boundary [26], which result in outstanding segmentation performance on natural images, perform poorly on EM datasets. Therefore, a powerful method for segmenting specific structures in EM images is required.

Many unsupervised techniques have been proposed to address this problem. Vu and Manjunath [27] proposed a graph-cut method that minimizes an energy function over the pixel intensity and flux of the gradient field for cell segmentation. However, their model might be confused by the complex intracellular structures and requires user interaction to correct segmentation errors. The contour propagation model [28] that minimizes an energy function for contour tracing of cell membranes can also get stuck in local minima due to complex intracellular structures. Kumar *et al.* [29] introduced a set of so-called Radon-like features (RLF), which take into account both texture and geometric information and overcome the problem of complex intracellular structures but only achieve modest accuracy levels due to the lack of a supervised classification scheme.

Several supervised methods also have been proposed for object segmentation in EM images such as convolutional neural networks [10] and series of ANN [12] for membrane detection or [20], [30] for mitochondria segmentation or [31], [32] for synapse segmentation. However, these frameworks

Copyright (c) 2013 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIP.2013.2274388

target only one object of interest and to our knowledge, they do not use intra-class information to give a coherent segmentation of multiple objects. One of the advantages of our proposed model is that it can segment multiple objects simultaneously. We show that the coherent segmentation improves the segmentation accuracy.

II. Multi-scale contextual model[1]

Let $X = (x(i,j))$ be the input image that comes with a ground truth $Y = (y(i,j))$ where $y(i,j) \in \{-1, 1\}$ is the class label for pixel (i,j). The training set is $T = \{(X_k, Y_k); k = 1, \ldots, M\}$ where M denotes the number of training images. Given an input image X, the MAP estimation of Y for each pixel is given by:

$$\hat{y}_{MAP}(i,j) = \arg\max_{y(i,j)} P(y(i,j)|X). \quad (1)$$

The local Markovianity assumption can be used to obtain a typical approximation of equation (1):

$$\hat{y}_{MAP}(i,j) = \arg\max_{y(i,j)} P(y(i,j)|X_{N(i,j)}), \quad (2)$$

where $N(i,j)$ denotes all the pixels in the neighborhood of pixel (i,j). $N(i,j)$ can be any arbitrary neighborhood lattice like 4-connected or 8-connected or sparse stencil [12] neighbors. This approximation decreases the computational complexity by giving the classifier access to a limited number of neighborhood pixels instead of the entire input image.

In auto-context [2] and series-ANN [12], a classifier is trained based on the neighborhood features at each pixel. We call the output image of this classifier the context image, i.e., $C = (c(i,j))$. The next classifier is trained not only on the neighborhood features of X but also on the neighborhood features of C. The MAP estimation formula for this classifier can be written as:

$$\hat{y}_{MAP}(i,j) = \arg\max_{y(i,j)} P(y(i,j)|X_{N(i,j)}, C_{N'(i,j)}), \quad (3)$$

where $N'(i,j)$ is the set of all neighborhood pixels of pixel (i,j) in the context image. Note that N and N' can be different neighborhood systems. The same procedure is repeated through several stages of the series classifier until convergence. It is worth mentioning that equation (3) is closely related to the CRF model; however, multiple models in series are learned which is an important difference from standard CRF approaches. It has been previously shown that this approach outperforms iterations with the same model [2].

According to equation (3), context provides prior information to solve the MAP problem. Even though the local Markovianity assumption is reasonable and makes the problem tractable, it still results in a significant loss of information from global context. However, it is not practical to sample every pixel in a very large neighborhood area of the context due to computational complexity problem and overfitting. Previous approaches [2], [12] have used a sparse sampling approach to cover large context areas. However, single pixel

[1]The preliminary version of this model was presented in MICCAI 2011 [33].

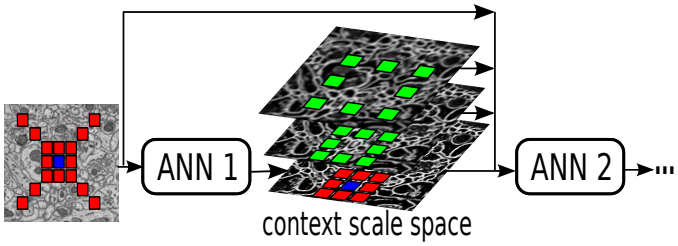


Fig. 2. Illustration of the multi-scale contextual model. Each context image is sampled at different scales (green squares). The blue squares represent the center pixel and the red squares show the selected locations at original scale.

contextual information in the finest scale conveys only partial information about its neighborhood pixels in a sparse sampling strategy while each pixel in the coarser scales contains more information about its surrounding area due to averaging filters used. In other words, while it is reasonable to sample context at the finest level a few pixels away, sampling context at the finest scale tens to hundreds of pixels away is error prone and presents a non-optimal summary of its local area. Conceptually, sampling from scale space representation increases the effective size of the neighborhood while keeping the number of samples small.

Figure 2 illustrates the multi-scale contextual model. In this model, a scale-space representation of the context image is created by applying a series of Gaussian filters. This results in a series feature maps with lower resolutions that are robust against the small variations in the location of features as well as noise. Unlike the auto-context structure that uses a sparse sampling approach to take samples from the context image, the multi-scale contextual model uses the samples of the scale space representation of context. Figure 3 shows the single-scale sampling strategy (Figure 3a) versus the multi-scale sampling strategy (Figure 3b). In Figure 3b the classifier can have as an input the center 3×3 patch at the original scale and a summary of 8 surrounding 3×3 patches at a coarser scale (The green circles denote the summaries of dashed squares). The green circles in Figure 3b are more informative and less noisy compared to their equivalent red circles in Figure 3a. The

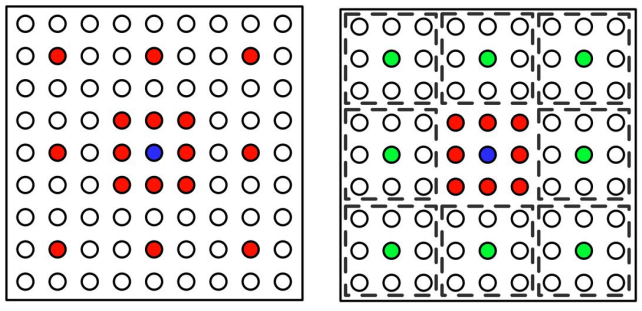


Fig. 3. Sampling strategy of context: (a) Sampling at a single scale (b) sampling at multiple scales. Green circles belong to a coarser scale and illustrate the summary of pixels in dashed squares. Green samples at the coarser scale are more informative than corresponding red samples at the original scale.

Copyright (c) 2013 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIP.2013.2274388

summaries become more informative as the number of scales increases. For example, in the second scale the summary is computed over 3×3 neighborhood of the first scale image, which is equivalent to 5×5 neighborhood of the original image. In practice, we use Gaussian averaging filters to create the summary (green circles). Other methods like maximum pooling can be used instead of Gaussian averaging [34]. The number of scales and Gaussian filter size are set according to the characteristics of the particular application. The size of the filter and number of scales should increase for larger objects.

From a mathematical point of view, equation (3) can be rewritten as:

$$\hat{y}_{MAP}(i,j) = \arg\max_{y(i,j)} P(y(i,j)|X_{N(i,j)}, C_{N_0'(i,j)}(0), C_{N_1'(i,j)}(1), \ldots, C_{N_l'(i,j)}(l)), \quad (4)$$

where $C(0), C(1), \ldots, C(l)$ denote the scale space representation of the context and $N_0'(i,j), N_1'(i,j), \ldots, N_l'(i,j)$ are corresponding neighborhood structures. Unlike equation (3) that uses the context in a single scale, equation (4) takes the advantage of multi-scale contextual information. Even though in equation (4), we still use the Markov assumption, the size of the neighborhood is larger and thus we lose less information compared to equation (3).

The series multi-scale contextual model updates the equation (4) iteratively:

$$\hat{y}^{k+1}_{MAP}(i,j) = \arg\max_{y(i,j)} P(y(i,j)|X_{N(i,j)}, C^k_{N_0'(i,j)}(0), C^k_{N_1'(i,j)}(1), \ldots, C^k_{N_l'(i,j)}(l)), \quad (5)$$

where $C^k(0), C^k(1), \ldots, C^k(l)$ are the scale space representation of the output of classifier stage k, $k = 1, \ldots, K-1$ and $\hat{y}^{k+1}_{MAP}(i,j)$ denotes the output of the stage $k+1$. In turn, the $k+1$'st classifier output as defined in equation (5) creates the context for the $k+2$'nd classifier. For $k = 0$ no prior information is used and the model only uses the input image for training. The model repeats equation (5) until the performance improvement between two consecutive stages becomes small. It must be emphasized that despite the iterative form of equation 5, multiple models are learned in the series separately and in sequential order which is an important difference from standard CRF models.

III. Multi-class multi-scale contextual model

While our multi-scale contextual model extracts a set of rich features from the context image of each object, it is unable to take into account the contextual information from multiple objects. We propose the multi-class multi-scale (MCMS) contextual model as a remedy to this problem as it is designed to leverage both the multi-scale and the multi-class contextual information. The proposed method can successfully capture long distance dependencies between objects and across different categories.

The multi-class contextual model is illustrated in Figure 4. In this figure, each classifier is a binary classifier, which is trained to segment only one object of interest. In other words,

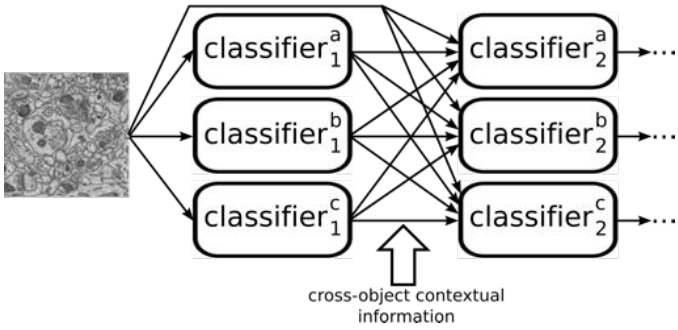


Fig. 4. Illustration of the multi-class contextual model. Each classifier is a binary classifier, which is trained for a specific object (a,b, and c are objects). Each classifier takes advantage of the context images of all objects from the previous stage. Superscripts show object type and subscripts show the classifier number in the series. Generalization to cases with more classes is straightforward.

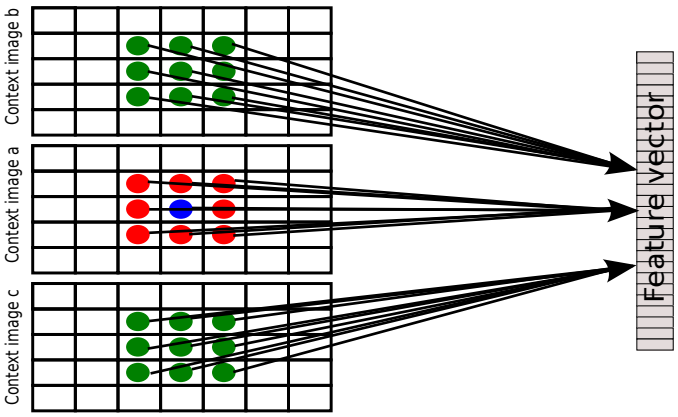


Fig. 5. The multi-class feature pooling scheme. The neighborhood samples of the center pixel (blue circle) in the context image "a", i.e., red circles, are used together with the neighborhood samples in the context images "b" and "c", i.e., green circles, to form the feature vector. The same feature vector together with the features of input image is used for all the classifiers. In the MCMS model the samples are pooled at multiple scales as well. The multi-scale sampling is not shown in this figure for the sake of clarity.

each classifier treats the pixels belonging to the object of interest as positive samples and all the other pixels including the background pixels as negative samples. The multi-class architecture allows the classifier of each object type access to the contextual information from each object type of the previous stage. This flow of information is achieved by feeding neighborhoods from the output of each classifier, i.e., the context image, in stage k to each classifier in stage $k+1$. The multi-class feature pooling scheme is shown in Figure 5. It extracts samples from the neighborhood of center pixel in all the context images from the previous stage. The extracted samples are used together with input image samples as the input to classifier. The same feature vectors are used for all the classifiers, nonetheless, each classifier is trained to segment a specific object. In other words, although the input feature vectors are the same, the target labels are different for each classifier. The propagation of contextual information among different categories enables the model to learn the geometrical relationships and object dependencies implicitly.

We describe the effectiveness of the multi-class model

Copyright (c) 2013 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIP.2013.2274388

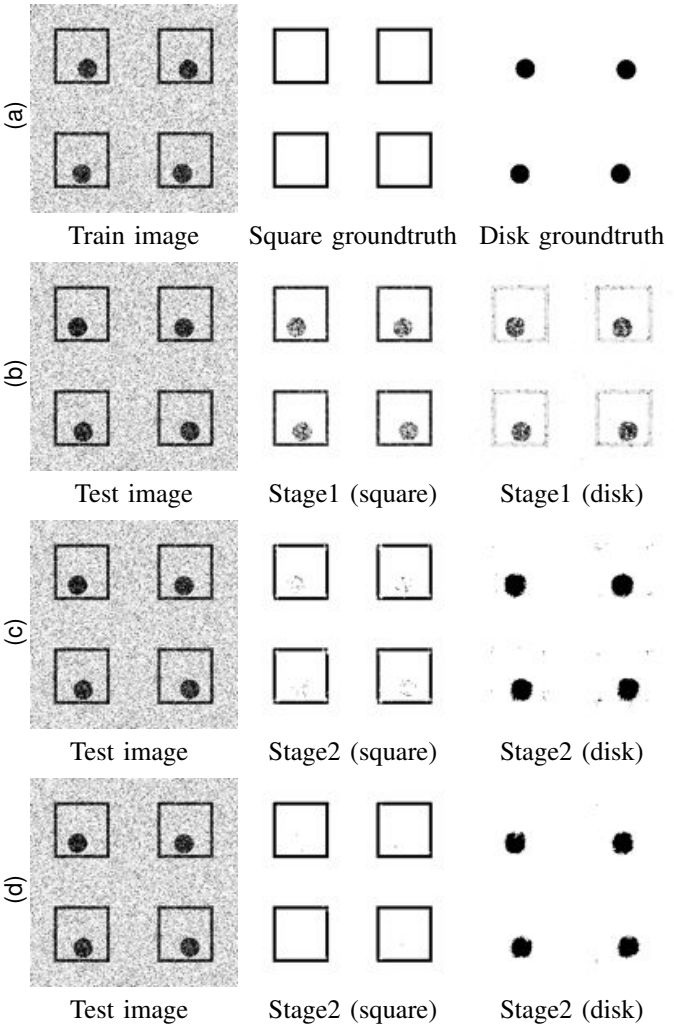


Fig. 6. A synthetic example that shows the effectiveness of the multi-class contextual model. (a) The input image and corresponding groundtruth images, (b) the outputs of the first stage classifiers, (c) the outputs of the second stage classifiers in the single-class model, and (d) the outputs of the second stage classifiers in the multi-class model. The multi-class model is more successful in removing the parts of the other object compared to the single-class model.

with a synthetic example. Consider the input image and the corresponding groundtruth images in Figure 6a. Two pixel classifiers are trained for the square and the disk classes separately. The outputs of these classifiers are shown in Figure 6b. The results are not perfect and each classifier misclassify some pixels of the other object as positive samples due to the noise and similarity between the textures. The single-class model that uses only the contextual information from the same object is not able to correct the wrong classified pixels completely (Figure 6c). By using the contextual information from both of the objects, the multi-class model will classify most of the previously misclassified pixels correctly as shown in Figure 6d. For example, the second stage square classifier exploits the information that those misclassified pixels from the previous stage are classified as disk by the first disk classifier and thus is able to correct them in the second stage. In this example we have two objects but this can be extended to any arbitrary number of objects.

The mathematical formulation of the multi-class contextual model for each classifier is obtained by incorporating the cross-contextual information in equation (3):

$$\hat{y}_{MAP}(i,j) = \arg\max_{y(i,j)} P(y(i,j)|X_{N(i,j)}, C^a_{N'(i,j)}, C^b_{N'(i,j)}, C^c_{N'(i,j)}), \quad (6)$$

where C^a, C^b, C^c denote the context images of different objects. We assume three objects in equation (6) for the sake of simplicity but the extension to more objects is straightforward.

By combining multi-class and multi-scale contextual models, the powerful MCMS model is obtained which is able to extract contextual information from large area and through different objects. The MCMS model is designed to make an extensive use of contextual information. This architecture allows the classifiers in the series to correct the errors of the previous stages by using the information from other classes and thus improves the segmentation performance. The update equation of the MCMS model can be derived by combining equation (3) and equation (4):

$$\begin{aligned}\hat{y}^{a,k+1}_{MAP}(i,j) = \arg\max_{y(i,j)} P(y(i,j)|X_{N(i,j)}, \\ C^{a,k}_{N'_0(i,j)}(0), C^{b,k}_{N'_0(i,j)}(0), C^{c,k}_{N'_0(i,j)}(0), \\ C^{a,k}_{N'_1(i,j)}(1), C^{b,k}_{N'_1(i,j)}(1), C^{c,k}_{N'_1(i,j)}(1), \\ \ldots, C^{a,k}_{N'_l(i,j)}(l), C^{b,k}_{N'_l(i,j)}(l), , C^{c,k}_{N'_l(i,j)}(l)), \end{aligned} \quad (7)$$

where $C^{a,k}(0), C^{a,k}(1), \ldots, C^{a,k}(l)$ are the scale space representation of the output of classifier stage k for object "a", $k = 1, \ldots, K-1$ and $\hat{y}^{a,k+1}_{MAP}(i,j)$ denotes the output of the stage $k+1$ for object "a". Similar equations are updated for objects "b" and "c". Each of these update equations are related to a row of classifiers in Figure 4. The main difference between equation (5) and equation (7) is that the former only pools contextual information from a single object while the latter takes advantage of contextual information from multiple objects. The overall training algorithm for the MCMS contextual model is described in Algorithm 1.

The time complexity of the MCMS model is almost the same as the multi-scale since the classifiers of each stage can be trained in parallel. Although this model has many parameters, the training is not complicated because the classifiers are trained separately through the stages and among the objects.

IV. Experimental Results

We perform experimental studies to evaluate the performance of both multi-scale and MCMS contextual models. We show the effectiveness of multi-scale contextual model for membrane detection in EM images and horse segmentation in a general computer vision dataset. We then show how membrane detection results can be used in MCMS model to improve mitochondria and synapse segmentation results.

A. Datasets

We used three different datasets in our experiments:

Copyright (c) 2013 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIP.2013.2274388

Algorithm 1 Training algorithm for the MCMS model

Input: A set of training images together with their binary groundtruth images for different objects, $T = \{(X_i, Y_i^s), i = 1, \ldots, M, s = 1, \ldots, N_{obj}\}$.

- For each input image X_i, generate non-informative probability maps, $C_i^{s,0}, s = 1, \ldots, N_{obj}$, with uniform distribution.
- $k = 0$

repeat

for $j = 1 : N_{obj}$ **do**

- Construct a new training set $T_j = \{((X_i, C_i^{s,k}), Y_i^j), i = 1, \ldots, M, s = 1, \ldots, N_{obj}\}$.
- Train a classifier, f_k^j, on features extracted from the input images and scale space representation of the context images (maximize equation (7) to obtain classifier parameters).

end for

for $j = 1 : N_{obj}$ **do**

- Use the trained classifier f_k^j to generate new context images $C_i^{j,k+1}$ (equation (7)).

end for

- $k = k + 1$

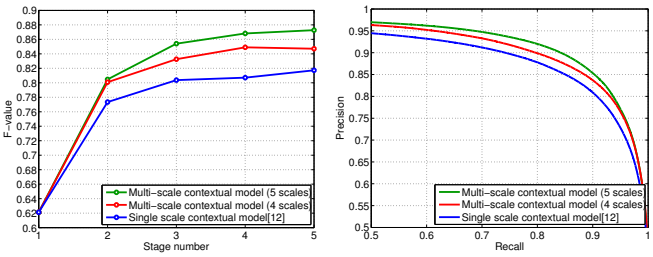
until convergence (improvement is negligible between two consecutive stages)

1) Weizmann horse dataset: The Weizmann dataset [35] contains 328 gray scale horse images with corresponding foreground/background truth maps. Similar to Tu *et al.* [2], we used half of the images for training and the remaining images were used for testing. There is only one object category, i.e., horse, in this dataset and thus we could only use it to test the multi-scale contextual model.

2) Mouse neuropil dataset: This dataset is a stack of 400 images from the mouse neuropil acquired using serial block face scanning electron microscopy (SBFSEM [19]). Each image is 4096 by 4096 pixels and the resolution is $10 \times 10 \times 50$ nm/pixel. To evaluate the segmentation performance, a subset of 70 images of size 700 by 700 pixels were selected. An expert anatomist annotated membranes and mitochondria in this subset with different labels. From those 70 images, 14 images were randomly selected and used for training and the 56 remaining images were used for testing.

3) Drosophila VNC dataset: This dataset contains 30 images from Drosophila first instar larva ventral nerve cord (VNC) [16], [17] acquired using serial-section transmission electron microscopy (ssTEM [36], [37]). It has a resolution of $4 \times 4 \times 50$ nm/pixel and each $2D$ section is 512 by 512 pixels. For this dataset, an expert annotated membranes, mitochondria, and synapses with different labels. We used 15 images for training and 15 images for testing.

The results presented in this paper were generated using a $HPDL980$ server containing 160, 2.40 GHz Intel CPUs and 750G of memory. The horse dataset, requires 19G of memory during training, while the mouse neuropil and Drosophila VNC datasets require 13G and 14G of memory, respectively. It took about 6, 5, and 3 days per stage to train the multi-scale contextual model on the horse, mouse neuropil, and Drosophila VNC datasets, respectively. As mentioned before, the training time of the MCMS model is almost the same as the multi-scale contextual model. Unlike the training, our model is relatively fast at the test time. Applying the classifiers weights on each input image takes less than one minute. Details regarding the parameters for each experiment are described in detail in the following sections.



(a) (b)

Fig. 7. Horse segmentation experiment on the Weizmann horse dataset. (a) The test F-value at different stages of the series for different methods with different number of scales. (b) The precision-recall curves for test images and for different methods (the last stage of the series). Using more scales improves the results.

B. Multi-scale contextual model (horse segmentation)

In this experiment, we test the multi-scale contextual model for horse segmentation. We used MLP-ANNs [38], [39] as the classifier in the series architecture, as in [12]. Each classifier in the series has one hidden layer with 30 nodes. Back-propagation was used to learn the weight vector and biases [38], [39].

Input image feature vectors were computed on a 31×31 sparse stencil [12] centered on each pixel. The size of the feature vector is 57. The context features were computed using 5×5 patches at five scales (one at original resolution and four at coarser scales). We used a Gaussian filter of size 7×7 to generate the scale space.

The average $F - value = \frac{2 \times Precision \times Recall}{Precision + Recall}$ at threshold 0.5, for different methods is shown in Figure 7(a). As we expected, the performance increases with the number of scales. The test F-value at stage 5 for multi-scale contextual model with 5 scales is 87.3%. This result outperforms the auto-context result which is 84% [2]. It must be emphasized that the improvement from the first stage to the last stage in our method is 25.2% while the improvement in the auto-context method is almost 5%. It is worth noting that we use a simple stencil to generate the input image feature vector instead of applying large filter banks to the input image as in [2] and our first stage F-value (62.1%) is less than auto-context first stage F-value (79%), but, our last stage result F-value is higher. This shows that multi-scale contextual model can compensate for the bad result of the first stage and improves the performance in later stages by using context in an effective manner. The precision-recall curves of the last stage results for the test set are shown in Figure 7(b).

Figure 8 shows some examples of our test images and their segmentation results using different methods with different

Copyright (c) 2013 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIP.2013.2274388

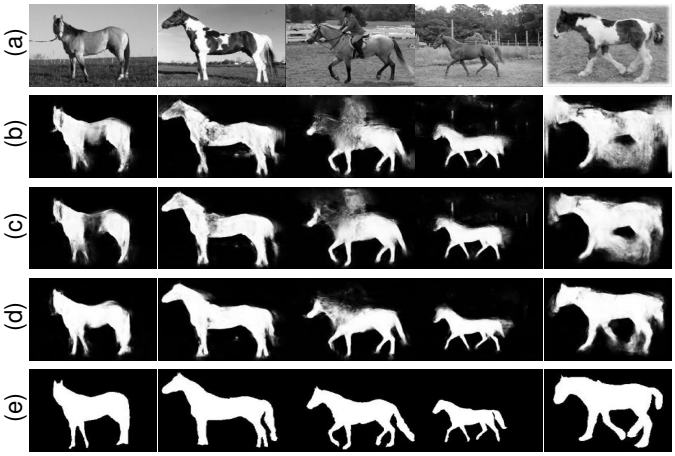


Fig. 8. Test results for the horse segmentation experiment. (a) Input images, (b) single-scale contextual model [12], (c) multi-scale contextual model with 4 scales, (d) multi-scale contextual model with 5 scales, (e) groundtruth images. The multi-scale contextual model is successful in removing the side effects of the cluttered background and filling the body of horses.

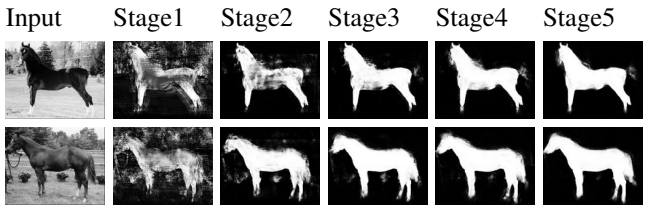


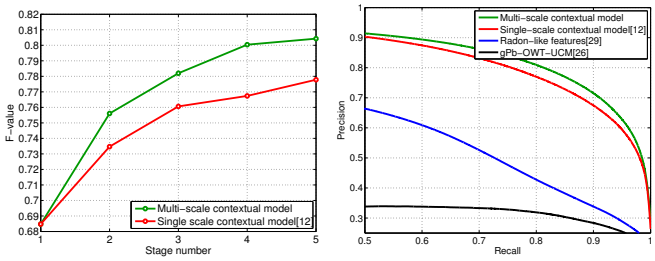
Fig. 9. Test results for the horse segmentation experiment. The first column shows the input image and the remaining columns show the output at different stages of multi-scale contextual model.

number of scales. As we can see, the multi-scale contextual model outperforms the single-scale contextual model in removing the side effects of the cluttered background and filling the body of horses. For example, in the middle column, the rider is removed by the multi-scale contextual model with 5 scales. Figure 9 shows two examples of test images and the corresponding segmentation results at different stages of the multi-scale contextual model. The converges of the model can be seen qualitatively in the results.

C. Multi-scale contextual model (membrane detection)

In this experiment, we show the performance of multi-scale contextual model for membrane detection on the mouse neuropil dataset. We used the same architecture as the previous experiment except that each MLP-ANN in the series had one hidden layer with 10 nodes.

This dataset is very imbalanced since the number of positive samples. i.e., membrane pixels, is much less than the negative samples, i.e., non-membrane pixels. To provide a relatively balanced dataset and optimize the MLP-ANN performance, 5.5 million samples were randomly selected from the training set to contain $\frac{1}{3}$ positive and $\frac{2}{3}$ negative examples, as in [12]. Input image feature vectors were computed on a 11×11 stencil. Context features were computed on 5×5 patches at four scales (one at original resolution and three at coarser scales). The classifier then gets as input the 5×5 patch at



(a) (b)

Fig. 10. Membrane detection experiment on the mouse neuropil dataset. (a) The test F-value at different stages of the series for different methods. The F-value for the RLF and gPb-OWT-UCM methods are 0.59 and 0.46, respectively. (b) The precision-recall curves for test images and for different methods (the last stage of the series).

the original resolution ($C_{N'_0(i,j)}(0)$ in equation 4) and 5×5 patches at three coarser scales ($C_{N'_l(i,j)}(l)$ in equation 4). We used a Gaussian filter of size 5×5 to generate the scale space.

We compared the performance of our methods with the RLF [29] and gPb-OWT-UCM (global probability of boundary followed by the oriented watershed transform and ultrametric contour maps) [26]. The average F-value for different stages of multi-scale contextual and MCMS models is shown in Figure 10(a). The performance of the multi-scale contextual model is 2.65% better than using a single-scale context [12]. The precision-recall curves for pixel-wise membrane detection are shown in Figure 10(b).

Figure 11 shows five examples of our test images and corresponding membrane detection results for different methods. As shown in our results, the multi-scale contextual model outperforms the methods in [12], [29], [26], and it is more successful in removing undesired parts from inside cells.

D. MCMS contextual model (mitochondria segmentation)

In this section, we show that MCMS model outperforms the multi-scale contextual model in mitochondria segmentation for the mouse neuropil dataset. For this dataset, the labels are only available for membrane and mitochondria, so, $N_{obj} = 2$ in Algorithm 1. We used MLP-ANNs with 10 hidden nodes for both membrane and mitochondria classifiers.

Input image feature vectors were computed on 11×11 and 15×15 stencils for membrane and mitochondria classifiers, respectively. For both of the categories, the context features were computed on 5×5 patches at four scales. To compare the performance, we used the same mitochondria classifiers with the same parameter settings in the multi-scale contextual model. The average F-value at different stages and for different methods is shown in Figure 12(a). The performance of the MCMS model is 2.42% better than the multi-scale contextual model. The precision-recall curves for pixel-wise mitochondria segmentation are shown in Figure 12(b). Figure 13 shows five test examples and corresponding mitochondria segmentation results for different methods. The MCMS model is more successful in correcting both false positive and false negative errors compared to the multi-scale contextual and RLF models.

Copyright (c) 2013 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIP.2013.2274388

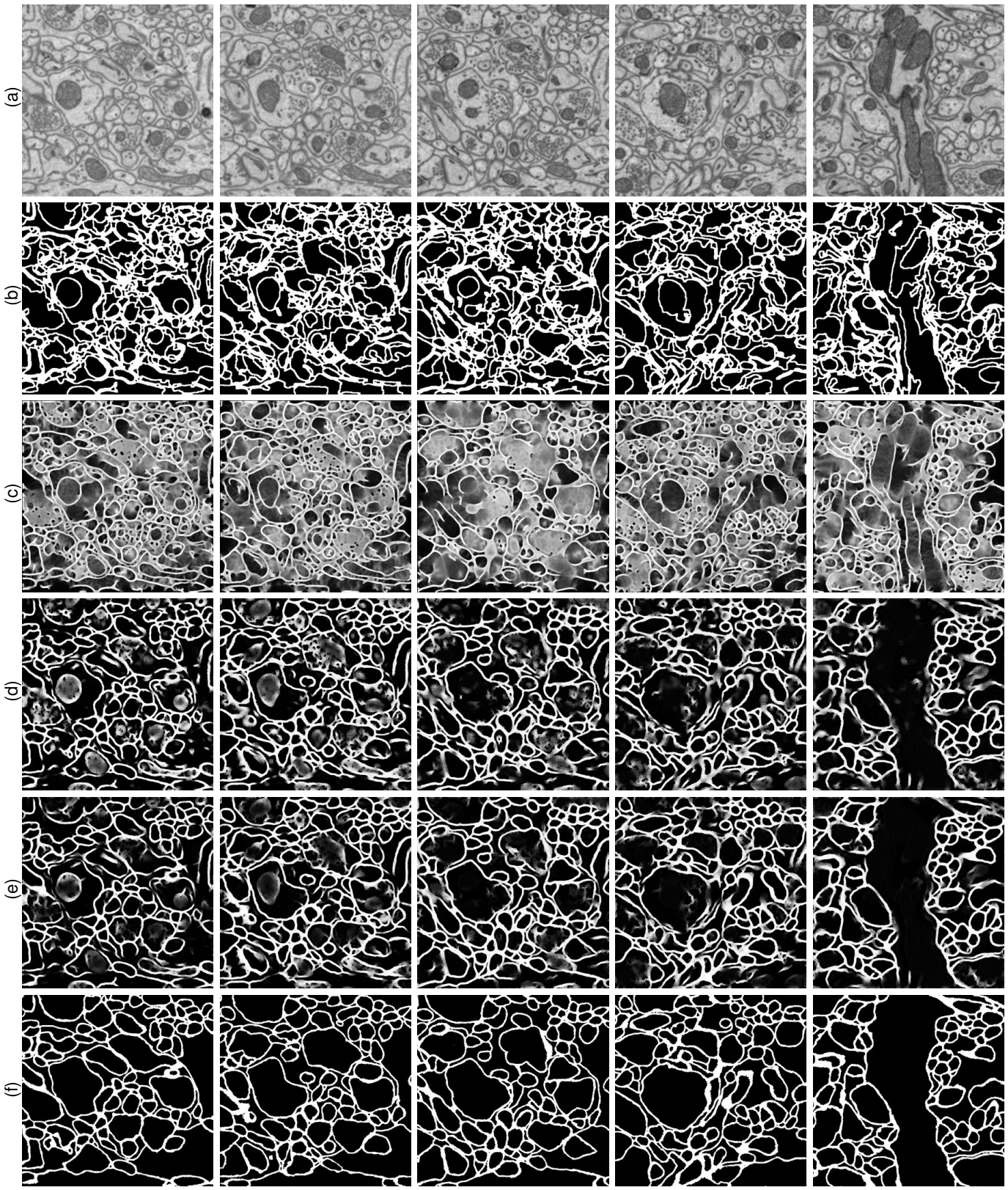
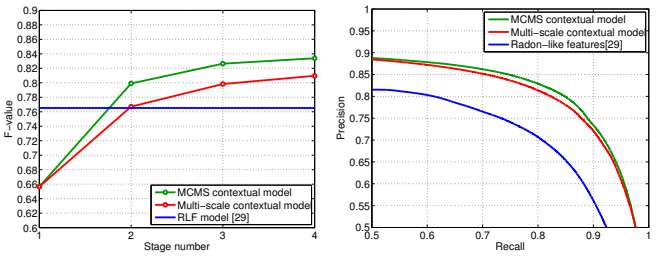


Fig. 11. Test results for the membrane detection experiment (mouse neuropil dataset). (a) Input images, (b) gPb-OWT-UCM method [26], (c) RLF method [29], (d) single-scale contextual model [12], (e) multi-scale contextual model, (f) groundtruth images. The multi-scale contextual model is more successful in removing undesired parts from inside cells than the algorithms proposed in [12], [29], [26]. For gPb-OWT-UCM method, the best threshold was picked and the edges were dilated to the true membrane thickness.

Copyright (c) 2013 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIP.2013.2274388



(a) (b)

Fig. 12. Mitochondria segmentation experiment on the mouse neuropil dataset. (a) The test F-value at different stages of the series for different methods. (b) The precision-recall curves for test images and for different methods (the last stage of the series).

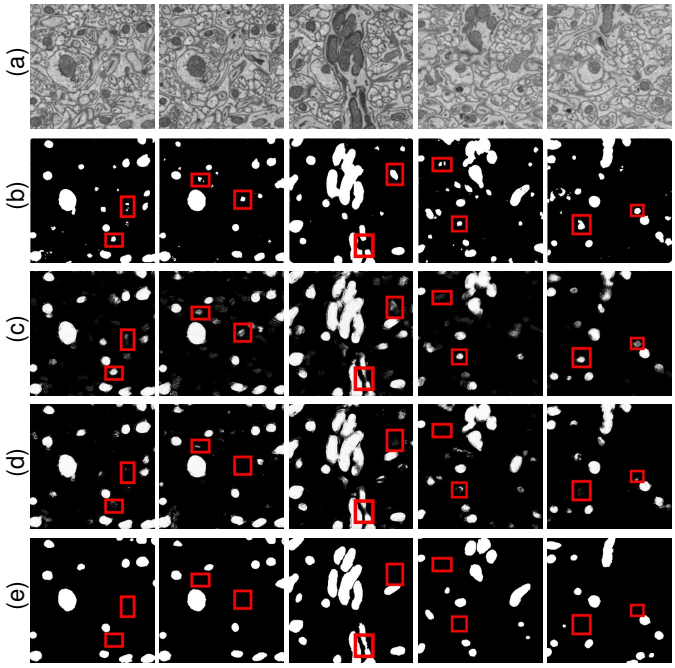


Fig. 13. Test results for the mitochondria segmentation experiment (mouse neuropil dataset). (a) Input images, (b) RLF method [29], (c) multi-scale contextual model, (d) MCMS contextual model, (e) groundtruth images. The MCMS contextual model is more successful in correcting both false positive and false negative errors compared to other methods. Some of the improvements are marked with red rectangles.

E. MCMS contextual model (mitochondria and synapse segmentation)

In this experiment, we test the MCMS model performance on the Drosophila VNC dataset with three object categories: membrane, mitochondria, and synapse. We used MLP-ANNs with 10 hidden nodes as classifier in the series.

Input image features were computed on 11×11, 15×15, and 15×15 for membrane, mitochondria, and synapse classifiers respectively. Similar to previous experiments, context features were computed on 5×5 patches at four scales. To compare with the multi-scale contextual model, we used classifiers with the same parameter settings for mitochondria and synapse segmentation. Figure 14 shows five test samples and corresponding mitochondria segmentation results for different methods. The MCMS model gives cleaner results compared to other methods. Figure 15 shows synapse segmentation

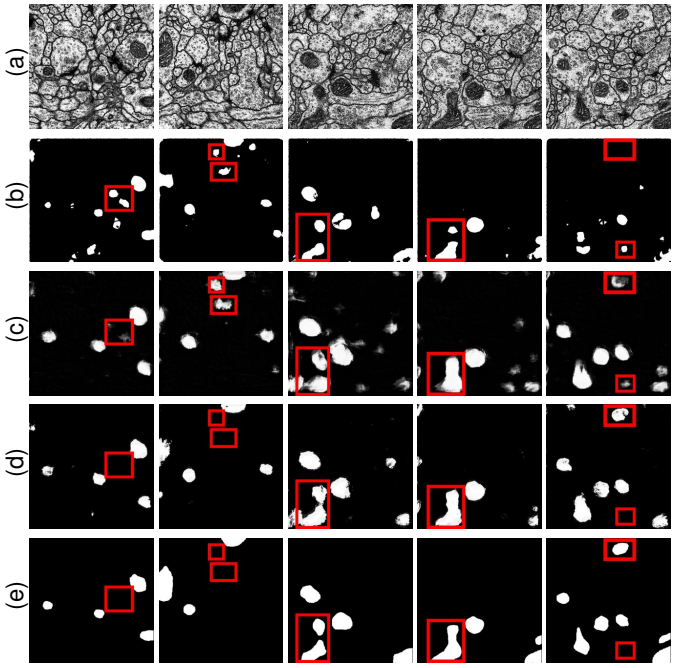


Fig. 14. Test results for the mitochondria segmentation experiment (Drosophila VNC dataset). (a) Input images, (b) RLF method [29], (c) multi-scale contextual model, (d) MCMS contextual model, (e) groundtruth images. The MCMS contextual model gives cleaner results compared to other methods. Some of the improvements are marked with red rectangles.

results for five test samples. The MCMS model is more successful in correcting false positive errors compared to the multi-scale contextual model. It must be emphasized that in this experiment we target four elements of synapses, i.e., synapstic cleft, postsynaptic density, T-band, and vesicles, simultaneously, which is a challenging task even for expert anatomists. That explains why the results are not as good as the membrane and mitochondria segmentation results.

The average F-value for the test set at different stages is shown in Figure 16. The MCMS model outperforms multi-scale contextual model with 2.9% and 2.92% in mitochondria and synapse segmentation respectively. The F-value of RLF method for mitochondria segmentation is 60% which is about 7% worse than the MCMS model.

F. Results discussion

In all of the above experiments, our goal was to study the effect of using rich contextual information in segmentation performance. We only used the samples of input images on a stencil structure as input image features. The overall performance can be improved by applying filter banks to input images and extract more informative features like what Tu *et al.* [2] did for horse segmentation. We previously showed [33] extracting Radon-like features from input images can improve the membrane detection results.

We noticed that in the MCMS model if a dataset is highly imbalanced then the effect of small classes on big classes is negligible. For example, the mitochondria contextual information in section IV-D and the synapse and mitochondria contextual information in section IV-E did not improve the

Copyright (c) 2013 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIP.2013.2274388

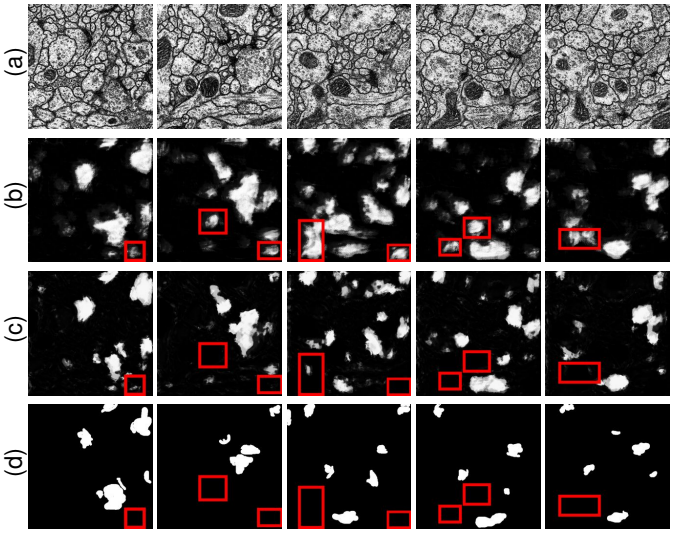


Fig. 15. Test results for the synapse segmentation experiment (Drosophila VNC dataset). (a) Input images, (b) multi-scale contextual model, (c) MCMS contextual model, (d) groundtruth images. The MCMS contextual model is more successful in correcting false-positives errors than multi-scale contextual model. Some of the improvements are marked with red rectangles.

membrane detection results. Nonetheless, big classes or same-size classes can improve the segmentation results of small classes as we showed in the experiments. In the mouse neuropil dataset the mitochondria class is 2.5 times smaller than the membrane class and in the Drosophila VNC dataset the mitochondria and synapse classes are 4.5 and 6 times smaller than the membrane class respectively.

In general image segmentation applications, other powerful techniques such as graph cuts and level sets can be applied to the results of the MCMS model to improve the segmentation accuracy. In segmentation of EM images, the final segmentation results can be improved further by applying appropriate post-processing techniques. For example, Andres *et al.* [40] propose a hierarchical method that uses over-segmented images obtained from membrane detection results and apply a classifier to merge regions. Funke *et al.* [41] and Liu *et al.* [42] use a tree structure to merge over-segmented regions for cell segmentation. These post-processing approaches can improve Rand error [43] for membrane detection. However, in our proposed method we target the pixel error and our method can be used for general computer vision datasets. The mitochondria and synapse segmentation results also can be improved by applying morphological post-processing, which removes tiny false positive errors. Our goal in the experiment section was to validate the multi-scale and the MCMS contextual models and study of post-processing approaches are beyond the scope of this paper.

V. Conclusion

We develop a supervised segmentation framework, which exploits contextual information from multiple objects and at different scales for learning discriminative models. Our multi-class multi-scale (MCMS) contextual model enables an implicit learning of geometrical relationships and dependencies among multiple objects present in an image. We applied

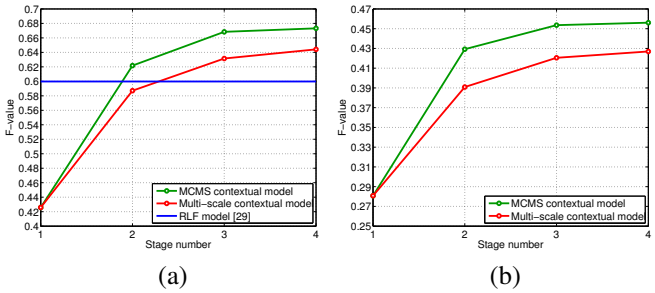


Fig. 16. Mitochondria and synapse segmentation experiment on the Drosophila VNC dataset. (a) The test F-value at different stages of the series for different methods (mitochondria segmentation). (b) The test F-value at different stages of the series for different methods (synapse segmentation).

our method to object segmentation in EM images. Results indicate that using multi-scale and cross-object contextual information can improve the segmentation results for each of the components present in EM images such as membrane, mitochondria, and synapse. It is worth noting that the proposed method is not restricted to this application and can be used in other image segmentation problems.

Even though our model has hundreds of parameters to learn, the complexity remains tractable since classifiers are trained one at a time separately. Our model can specially be useful in segmentation of imbalanced datasets that only a few samples of a particular object/class are available. In these datasets, large classes can improve the segmentation results of the small classes by providing informative contextual information.

We conclude by discussing a possible extension of the MCMS model presented in this paper. Our feature extraction model only exploits pixel intensities from input images and probabilities from context images. While this reduces the computational complexity and keeps the model simple, more complex features extracted from both input and context images can improve the results.

Acknowledgment

This work was supported by NIH 1R01NS075314-01 (TT,MHE) and NSF IIS-1149299 (TT). We thank the "National Center for Microscopy Imaging Research" and the "Cardona Lab at HHMI Janelia Farm" for providing the mouse neuropil and Drosophila VNC datasets. We like to acknowledge the support of the Utah Science Technology and Research Initiative (USTAR). We also thank the editor and reviewers whose comments helped greatly improve the paper.

References

[1] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. on PAMI*, vol. 24, no. 4, pp. 509–522, 2002.
[2] Z. Tu and X. Bai, "Auto-context and its application to high-level vision tasks and 3d brain image segmentation," *IEEE Trans. on PAMI*, vol. 32, no. 10, pp. 1744–1757, 2010.
[3] M. Fink and P. Perona, "Mutual boosting for contextual inference," *NIPS*, 2004.
[4] A. Singhal, J. Luo, and W. Zhu, "Probabilistic spatial context models for scene content understanding," *CVPR*, 2003.
[5] K. Murphy, A. Torralba, and W. T. Freeman, "Using the forest to see the trees: A graphical model relating features, objects, and scenes," *NIPS*, 2003.

Copyright (c) 2013 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIP.2013.2274388

[6] A. Torralba, K. P. Murphy, and W. T. Freeman, "Contextual models for object detection using boosted random fields," *NIPS*, 2004.
[7] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Trans. on PAMI*, vol. 6, no. 6, pp. 721–741, 1984.
[8] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," *Proc. of ICML*, pp. 282–289, 2001.
[9] X. He, R. Zemel, and M. Carreira-Perpinan, "Multiscale conditional random fields for image labeling," *CVPR*, 2004.
[10] V. Jain, J. F. Murray, F. Roth, S. Turaga, V. Zhigulin, K. L. Briggman, M. N. Helmstaedter, W. Denk, and H. S.Seung, "Supervised learning of image restoration with convolutional networks," *ICCV*, 2007.
[11] Y. LeCun, F. J. Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," *Proc. of CVPR*, vol. 2, pp. 97 –104, 2004.
[12] E. Jurrus, A. R. C. Paiva, S. Watanabe, J. R. Anderson, B. W. Jones, R. T. Whitaker, E. M. Jorgensen, R. E. Marc, and T. Tasdizen, "Detection of neuron membranes in electron microscopy images using a serial neural network architecture," *Medical Image Analysis*, vol. 14, no. 6, pp. 770–783, 2010.
[13] C. Desai, D. Ramanan, and C. Fowlkes, "Discriminative models for multi-class object layout," *Proc. of ICCV*, pp. 229 –236, 2009.
[14] G. Heitz, S. Gould, A. Saxena, and D. Koller, "Cascaded classication models: Combining models for holistic scene understanding," *Proc. of NIPS*, pp. 641–648, 2008.
[15] M. J. Choi, A. Torralba, and A. S. Willsky, "A tree-based context model for object recognition," *IEEE Trans. on PAMI*, vol. 34, no. 2, pp. 240–252, 2012.
[16] A. Cardona, S. Saalfeld, S. Preibisch, B. Schmid, A. Cheng, J. Pulokas, P. Tomančák, and V. Hartenstein, "An integrated micro- and macroarchitectural analysis of the *Drosophila* brain by computer-assisted serial section electron microscopy," *PLoS Biol*, vol. 8, no. 10, p. e1000502, 10 2010.
[17] A. Cardona, S. Saalfeld, J. Schindelin, I. Arganda-Carreras, S. Preibisch, M. Longair, P. Tomancak, V. Hartenstein, and R. J. Douglas, "Trakem2 software for neural circuit reconstruction," *PLoS ONE*, vol. 7, no. 6, p. e38011, 06 2012.
[18] O. Sporns, G. Tononi, and R. Ktter, "The human connectome: a structural description of the human brain," *PLoS Computational Biology*, vol. 1, p. e42, 2005.
[19] W. Denk and H. Horstmann, "Serial block-face scanning electron microscopy to reconstruct three-dimensional tissue nanostructure," *PLoS Biology*, vol. 2, p. e329, 2004.
[20] A. Lucchi, K. Smith, R. Achanta, V. Lepetit, and P. Fua, "A fully automated approach to segmentation of irregularly shaped cellular structures in em images," in *MICCAI (2)*, 2010, pp. 463–471.
[21] J. Anderson, B. Jones, C. Watt, M. Shaw, J.-H. Yang, D. DeMill, J. Lauritzen, Y. Lin, K. Rapp, D. Mastronarde, P. Koshevoy, B. Grimm, T. Tasdizen, R. Whitaker, and R. Marc, "Exploring the retinal connectome," *Molecular Vision*, no. 17, pp. 355–379, 2011.
[22] K. L. Briggman and W. Denk, "Towards neural circuit reconstruction with volume electron microscopy techniques," *Curr. Opin. in Neurobio.*, vol. 16, no. 5, pp. 562–570, 2006.
[23] V. Jain, B. Bollmann, M. Richardson, D. Berger, M. Helmstaedter, K. Briggman, W. Denk, J. Bowden, J. Mendenhall, W. Abraham, K. Harris, N. Kasthuri, K. Hayworth, R. Schalek, J. Tapia, J. Lichtman, and H. Seung, "Boundary learning by optimization with topological constraints," *CVPR*, pp. 2488 –2495, 2010.
[24] T. Cour, F. Benezit, and J. Shi, "Spectral segmentation with multiscale graph decomposition," *CVPR*, vol. 2, pp. 1124–1131, 2005.
[25] P. Dollar, Z. Tu, and S. Belongie, "Supervised learning of edges and object boundaries," *CVPR*, 2006.
[26] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "From contours to regions: An empirical evaluation," *CVPR*, vol. 0, pp. 2294–2301, 2009.
[27] N. Vu and B. S. Manjunath, "Graph cut segmentation of neuronal structures from transmission electron micrographs," in *Proceedings of ICIP*, 2008, pp. 725–728.
[28] J. H. Macke, N. Maack, R. Gupta, W. Denk, B. Schlkopf, and A. Borst, "Contour-propagation algorithms for semi-automated reconstruction of neural processes," *Journal of Neuroscience Methods*, vol. 167, no. 2, pp. 349–357, 2008.
[29] R. Kumar, A. Va andzquez Reina, and H. Pfister, "Radon-like features and their application to connectomics," in *CVPRW*, june 2010.
[30] R. Giuly, M. Martone, and M. Ellisman, "Method: automatic segmentation of mitochondria utilizing patch classification, contour pair classification, and automatically seeded level sets," *BMC Bioinformatics*, vol. 13, no. 1, p. 29, 2012.
[31] C. Becker, K. Ali, G. Knott, and P. Fua, "Learning context cues for synapse segmentation in em volumes," *MICCAI*, 2012.
[32] A. Kreshuk, C. N. Straehle, C. Sommer, U. Köthe, G. Knott, and F. A. Hamprecht, "Automated segmentation of synapses in 3d em data," in *ISBI*, 2011, pp. 220–223.
[33] M. Seyedhosseini, R. Kumar, E. Jurrus, R. Guily, M. Ellisman, H. Pfister, and T. Tasdizen, "Detection of neuron membranes in electron microscopy images using multi-scale context and radon-like features," in *MICCAI*, 2011.
[34] Y. LeCun, K. Kavukcuoglu, and C. Farabet, "Convolutional networks and applications in vision," *Proc. of ISCAS*, pp. 253–256, 2010.
[35] E. Borenstein, E. Sharon, and S. Ullman, "Combining top-down and bottom-up segmentation," *Proc. of CVPRW*, pp. 46 –46, 2004.
[36] J. R. Anderson, B. W. Jones, J.-H. Yang, M. V. Shaw, C. B. Watt, P. Koshevoy, J. Spaltenstein, E. Jurrus, K. UV, R. T. Whitaker, D. Mastronarde, T. Tasdizen, and R. E. Marc, "A computational framework for ultrastructural mapping of neural circuitry," *PLoS Biol*, vol. 7, no. 3, p. e1000074, 03 2009.
[37] D. B. Chklovskii, S. Vitaladevuni, and L. K. Scheffer, "Semi-automated reconstruction of neural circuits using electron microscopy," *Current Opinion in Neurobiology*, vol. 20, no. 5, pp. 667 – 675, 2010.
[38] S. Haykin, *Neural networks - A comprehensive foundation*, 2nd ed. Prentice-Hall, 1999.
[39] J. C. Principe, N. R. Euliano, and W. C. Lefebvre, *Neural and adaptive systems: Fundamentals through simulations*. Wiley, 2000.
[40] B. Andres, U. Kothe, M. Helmstaedter, W. Denk, and F. A. Hamprecht, "Segmentation of sbfsem volume data of neural tissue by hierarchical classification," in *Proceedings of the 30th DAGM symposium on Pattern Recognition*, 2008, pp. 142–152.
[41] J. Funke, B. Andres, F. A. Hamprecht, A. Cardona, and M. Cook, "Efficient automatic 3D-reconstruction of branching neurons from EM data," in *CVPR*, 2012.
[42] T. Liu, E. Jurrus, M. Seyedhosseini, M. Ellisman, and T. Tasdizen, "Watershed merge tree classification for electron microscopy image segmentation," *Proceedings of the 21st International Conference on Pattern Recognition (ICPR)*, p. (to appear), 2012.
[43] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical Association*, vol. 66, no. 336, pp. 846–850, 1971.

Mojtaba Seyedhosseini received the B.S. degree in Electrical Engineering from the University of Tehran in 2007, and the M.S. degree in Electrical Engineering from the Sharif University of Technology in 2009. He is currently working toward PhD degree at the Scientific Computing and Imaging (SCI) Institute at the University of Utah. His research interests include machine learning, statistical pattern recognition, and image analysis.

Tolga Tasdizen received the B.S. degree in electrical and electronics engineering from Bogazici University in 1995. He received his M.S. and Ph.D. degrees in engineering from Brown University in 1997 and 2001, respectively. After working as a postdoctoral researcher position at the Scientific Computing and Imaging (SCI) Institute at the University of Utah, he was a Research Assistant Professor in the School of Computing at the same institution. Since 2008, he has been with the Department of Electrical and Computer Engineering at the University of Utah where he is currently an Associate Professor. Dr. Tasdizen is also a Utah Science Technology and Research Initiative (USTAR) faculty member in the SCI Institute. His research interests are in image processing, computer vision and pattern recognition with a focus on applications in biological and medical image analysis. Dr. Tasdizen is a recipient of the National Science Foundation's CAREER award. He is a member of Bio Imaging and Signal Processing Technical Committee (BISP TC) of the IEEE Signal Processing Society and serves as an associate editor for the IEEE Signal Processing Letters and BMC Bioinformatics.

Copyright (c) 2013 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.