

Semantic Volume Segmentation with Iterative Context Integration for Bio-medical Image Stacks

Sven Sickert,^{1,*} Erik Rodner,^{1,**} and Joachim Denzler^{1,***}

¹*Computer Vision Group,
Friedrich Schiller University Jena,
07743 Jena, Germany
<http://www.inf-cv.uni-jena.de>*

Automatic recognition of biological structures like membranes or synapses is important to analyze organic processes and to understand their functional behavior. To achieve this, volumetric images taken by electron microscopy or computer tomography have to be segmented into meaningful semantic regions. We are extending iterative context forests which were developed for 2D image data to image stack segmentation. In particular, our method is able to learn high-order dependencies and import contextual information, which often can not be learned by conventional Markov random field approaches usually used for this task. Our method is tested on very different and challenging medical and biological segmentation tasks.

Keywords: volume segmentation, semantic segmentation, random decision forests

1. INTRODUCTION

Current imaging techniques in medicine and in general in biomedical research allow for capturing huge amounts of volume data and are a common tool to investigate biological processes or structures in organic and inorganic specimens. In particular, extracting specific regions in volume images is an important task in medical image processing and a prerequisite for quantitative analysis of biological data (Figure 1). However, manual annotation of 3D data is challenging and often requires a huge amount of time and expertise. Therefore, the

* Electronic address: sven.sickert@uni-jena.de

** Electronic address: erik.rodner@uni-jena.de

*** Electronic address: joachim.denzler@uni-jena.de

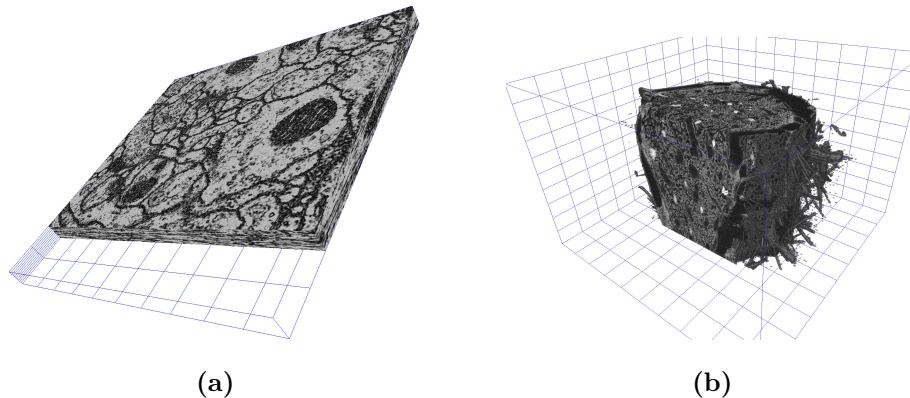


Figure 1. Volume images used in our experiments: (a) an electron microscopy stack of neuronal tissue and (b) computer tomography scan of a sponge

aim of the paper is to present a method, which is easy to implement, to use, and can be applied to generic volume data by learning from a small number of previously annotated volumes. Being able to automatically segment semantic regions in volume data does not only save time, but it also allows for quantitatively analyzing a large number of volumes, important for providing applied researchers with robust statistics.

Our proposed method labels every single voxel in the volume and is able to capture context information along every axis of the volume. We show that the use of relatively simple feature extraction methods on channels for color and gradient values is sufficient to obtain a decent segmentation of volume images.

The outline of the paper is as follows: In Section 2, we give an overview of related work. Section 3 and 4 continue with a description and discussion of the proposed method. In Section 5, we present segmentation results for different data sets. We conclude with a discussion of the results in Section 6.

2. RELATED WORK

The task of semantic volume segmentation for image data provided by computer tomography imagery or microscopy is of great interest in image analysis [15, 17, 24]. Related works most often propose segmentation techniques to find and reconstruct objects of a specific class, *e.g.*, synapses or membranes in neuronal tissue. A popular tool for segmentation are conditional Markov random fields and the graph-cut algorithm. The method in [17] uses a

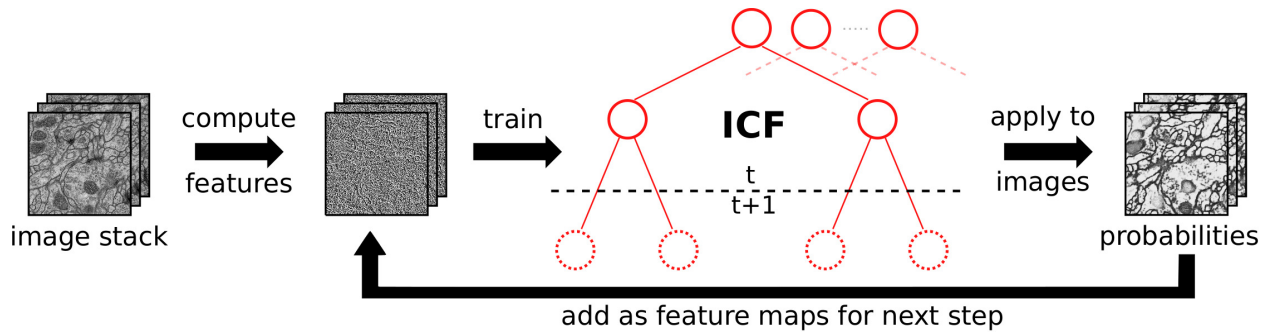


Figure 2. Framework of our method: First, slices of an image stack are processed in order to compute feature maps. Feature extraction methods applied to these maps are used to train the ICF by finding *good* splits at the current level t . The ICF learned so far is applied to the training slices in order to create class probabilities for each pixel. Then, these probability maps are add to the pool of feature maps in the training step $t + 1$.

gradient flux term in the energy function and incorporates information from different sections by applying the SIFT flow algorithm [18]. In [15], a modified energy function is presented which omits the gradient flux term. Instead, perceptual grouping constraints for contour completion are introduced, which can be useful for the segmentation of thin elongated structures. However, the authors argue that this term may also lead to false positive membrane segmentations due to textures. Therefore, they optimize a final energy function consisting of a data term, a directional energy term for smoothness, a penalty for discontinuities, and a term that incorporates information from adjacent sections.

In [24], the correspondence between nearby areas is transformed into a fusion problem. First, an random decision forest (RDF) classifier is trained to obtain the probability of each pixel belonging to the cell boundary. Watershed transformation is applied to obtain segmentations for each section. After that, 3D links between these sections should connect segments of different slices. Finally, a fusion problem of 2D segments and 3D links is defined that identifies each neuron.

The authors of [20] show that semantic segmentation can also help with localizing organs in a human body in computer tomography scans. They are using so called entangled-decision forests to model context between nearby organs which usually have a fixed composition.

In our approach, we do not model the connection between different sections explicitly, but use *auto-context* features [23] within an iterative pixel-wise labeling approach. Moreover,

we are using easy to compute features like gray values and gradients in a cubic neighborhood. Therefore, we extend the iterative context forests (ICF) of [11] to 3D data which was previously limited to 2D images and multi-channel images occurring in remote sensing applications.

More common applications for semantic segmentation are urban scenes and images of landscape, persons, or animals. In these settings typical constellations like *car on street*, *sky above building* or *sheep on grass* are observable as well. Many methods in this field of application are using either RDFs [4, 10, 11, 26] or conditional random fields (CRFs) [2, 12, 16] which is a common technique to model context. In contrast to CRFs, our approach is not restricted to pair-wise potentials and is able to learn high-order dependencies.

3. ITERATIVE CONTEXT FORESTS FOR IMAGE STACKS

The proposed approach for semantic segmentation consists of two fundamental parts: feature extraction and pixel classification. In this section, we focus on the classification framework which is based on the popular random decision forests. Details on the efficient computation of features will follow in the next section.

3.1. Random Decision Forests

The core of ICF [11] is a random decision forest (RDF) [3], which has been specifically adapted to semantic segmentation. The concept of decision trees is known for several decades already, so we will not elaborate on how they work and refer the interested reader to [3]. In order to explain iterative context forests, we give a short introduction to random decision forests. The RDF approach in general aims at overcoming drawbacks of original decision trees, like over-fitting and long training times, with two concepts of massive randomization during training.

The first concept is known as bagging and trains several decision trees individually with a random subset of the training data. Furthermore, a second randomization concept is applied determining binary splits in inner nodes. Instead of computing all available features in each inner node only a random subset is drawn from the set of all possible features, which we will refer to as *feature pool*. Among them the best feature and split is determined by maximizing the impurity criterion, which in our case is the information gain.

In our case, we are confronted with different types of features (see Sec. 4 for more details) with each of them having a number of parameters (*e.g.*, position, used feature map, *etc.*). To sample a specific feature, we first sample the feature type and we then sample the parameters in a second step. This guarantees that the learning is not biased towards feature types with a larger parameter space. The whole random selection process is exactly what renders learning in our case with millions of features tractable.

Although finding a good split in a single tree node makes it necessary to test various randomly chosen splits, the training of random decision forests is still fast. Learning an RDF is a matter of minutes, while convolutional neuronal networks for instance need several days on a GPU for such a task [7]. The classification of test images is even faster. Each pixel of an example is traversing the trees of the trained RDF until it reaches the leaf nodes. The empirical distributions in all leaves are then combined in order to estimate class-wise probabilities.

3.2. Incorporating Context Knowledge

Applying RDFs for pixel- or voxel-wise classification directly has two disadvantages: first, for each pixel the tree has to be traversed down to the leafs, and second, feature extraction is limited to a local neighborhood and is unable to integrate high-order dependencies.

To address the second issue, [11] proposed to sequentially traverse the tree level by level for all pixels in each image. This allows for using outputs of the previous level as an additional source for features. At each level of the random forest class probability maps for the current image (or volume) are computed. On a lower level these information allow the extraction of contextual features in order to model relational dependencies like one class is above another. This concept was introduced in [23] and is called *auto-context*. See Fig. 2 for an overview of our pixel-wise classification framework.

Furthermore, during testing time the structure of ICFs also addresses a trade-off between accuracy and time needed for inference. In some applications it might be sufficient to have a rough approximation of the best obtainable result after a short time. Since ICFs are built in breadth-first manner, a prediction on each level of the trees can be done by returning the empirical class distribution stored at inner nodes.

As observed in our experiments, each depth-level gives a more accurate classification

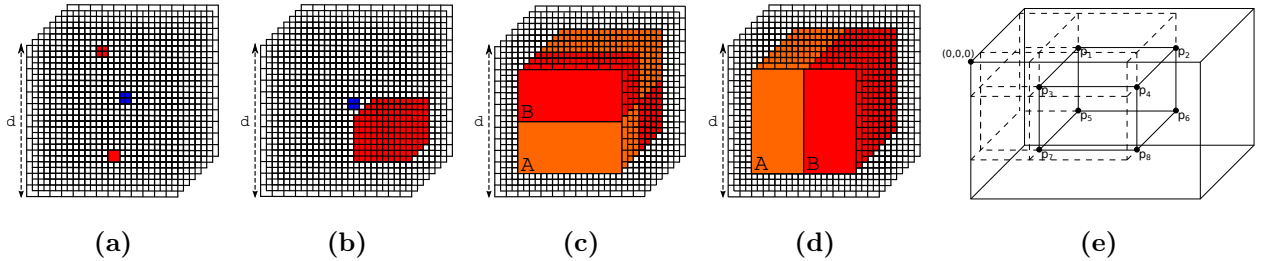


Figure 3. Feature extraction methods for a 3D neighborhood of side length d around a centered voxel: (a) pairs of voxels, (b) a smaller cuboid of arbitrary size or (c-d) different types of 3D Haar-like features where the feature value is the difference of the voxel intensity sums of darker and brighter regions. (e) Eight references are necessary to compute the sum of an arbitrary cuboid within the integral volume image.

result than the one before until saturation is reached. Stopping at an earlier level will give only rough results depending on fewer features but will also save time. The ability of an algorithm to allow for iterative refinements during testing is usually referred to as *anytime classification* [9].

4. EFFICIENT FEATURES IN VOLUMETRIC IMAGES WITH ICF3D

In this section, we describe our features for each pixel and their efficient computation with integral volumes.

4.1. Creating a Large Feature Pool

The ICF classifier allows inputs to have an arbitrary number of channels. Depending on the application, the algorithm can make use not only of raw color channels but additional layers like gradient images, resulting probabilities of other pre-learned classifiers, or unsupervised segmentation outputs. The relevance of these *feature maps* is automatically determined by the RDF classifier, since all binary splits are evaluated by an impurity measure and automatically selected.

Usually, for the application in the bio-medical field only gray-scale images are available as original input. There is no additional knowledge which can be incorporated. However, the gray-scale image can be used as a basis for the creation of further feature maps. In order to model textural properties of the desired classes we compute gradient maps for all

dimensions. Additionally, we incorporate feature maps from local binary patterns (LBP) [21].

These maps allow for the modeling of very small but meaningful difference between otherwise identically looking objects (e.g. membrane vs. mitochondria). As a drawback, they are prone to subtle changes in image noise, as well. In our case, however, images are either taken in a temporal sequence using one recording device or as for the data coming from CT scans image stacks are recorded at the same time, entirely. This leads to a very consistent image noise in the whole image stack and allows for the use of these textural feature maps.

The actual extraction of features from these maps is done in our case with simple operations performed in a neighborhood of the current center pixel: (a) Single values extracted from neighboring voxels, (b) sum or difference of two neighboring voxels, (c) sum of values within a cuboid and (d) 3D Haar-like features given by the difference of two or more cuboids. Visual examples can be found in Fig. 3.

With the size of the neighborhood the feature pool grows exponentially and it is not possible to compute all of the possible features in the training step. As a consequence, we only draw a fixed number of features. From this subset of features, the best split in each level of a decision tree is chosen by the impurity criterion.

4.2. Fast Computation of 3D Features

In order to compute 3D features in a fast manner, it is possible to use an intermediate representation. In the case of 2D images the so called *integral image* (or *summed area table*) is used [8, 25]. It contains at location (x, y) the sum of the pixel values above and to the left of the same location in the original image. The computation of rectangle features as for instance Haar-like features benefit from this representation. The sum of a rectangle area of arbitrary size can be computed by only four points in the integral image.

This idea can be easily extended to *integral volumes*. In consequence, an integral volume image \mathcal{V} contains at location (x, y, z) the sum of gray values from the voxels above, to the left and in front of its location in the input image \mathcal{I} as well as the gray value of the voxel (x, y, z) itself:

$$\mathcal{V}(x, y, z) = \sum_{x' \leq x, y' \leq y, z' \leq z} \mathcal{I}(x', y', z') . \quad (1)$$

With the integral volume of an image, any sum of values in a cuboid can be computed by eight array references. Analogously to [25], the values of the cuboids to the left, on top and in front of the currently processed volume have to be removed. Otherwise, these region would be incorporated twice in the final computation. Consider a cuboid \mathcal{C} with an arbitrary position inside a larger cuboid, *i.e.* the whole integral volume image V . The eight corners of \mathcal{C} are p_1, \dots, p_8 with $p_i = (x_i, y_i, z_i)$ being the locations in V . Location p_6 marks the lower right back corner and p_2 the upper right front corner of \mathcal{C} . Then, the integral volume $\mathcal{V}_{\mathcal{C}}$ of \mathcal{C} can be computed as:

$$\mathcal{V}_{\mathcal{C}} = p_1 + p_4 + p_6 + p_7 - p_2 - p_3 - p_5 - p_8 \quad . \quad (2)$$

A corresponding visualization showing the eight reference points can be found in Fig. 3(e). With this method cuboid features and especially the 3D Haar-like features can be computed very efficiently. Now that we have extended the ICF framework with 3D features and added the possibility to analyze volume images, we will henceforth call it ICF3D.

5. EXPERIMENTS

We evaluate our method on two different datasets. For a qualitative demonstration of ICF3D, we use data of a computer tomography scan. A quantitative comparison with other approaches using an EM stack of neuronal tissue is done after that.

5.1. Volume Data of a Sponge

We are using computer tomography scanning data of a sponge with an isotropic resolution of 3.4 micrometers per pixel. The resolution of each slice is 1536×1536 pixels. The task is to distinguish between the classes *canals*, *tissue*, and *background*. There is a stack of eleven labeled slices and a stack of 351 unlabeled slices. We used the former stack for training and the second one for testing. All images feature noise and artifacts from the recording procedure. The training slices were accurately labeled by a biologist. In Fig. 4 you can find a typical image. According to the expert, all dark gray structures belong to the class *canals*, even the tiny ones. A magnified part of another slice can also be seen in the second row of Fig. 4.

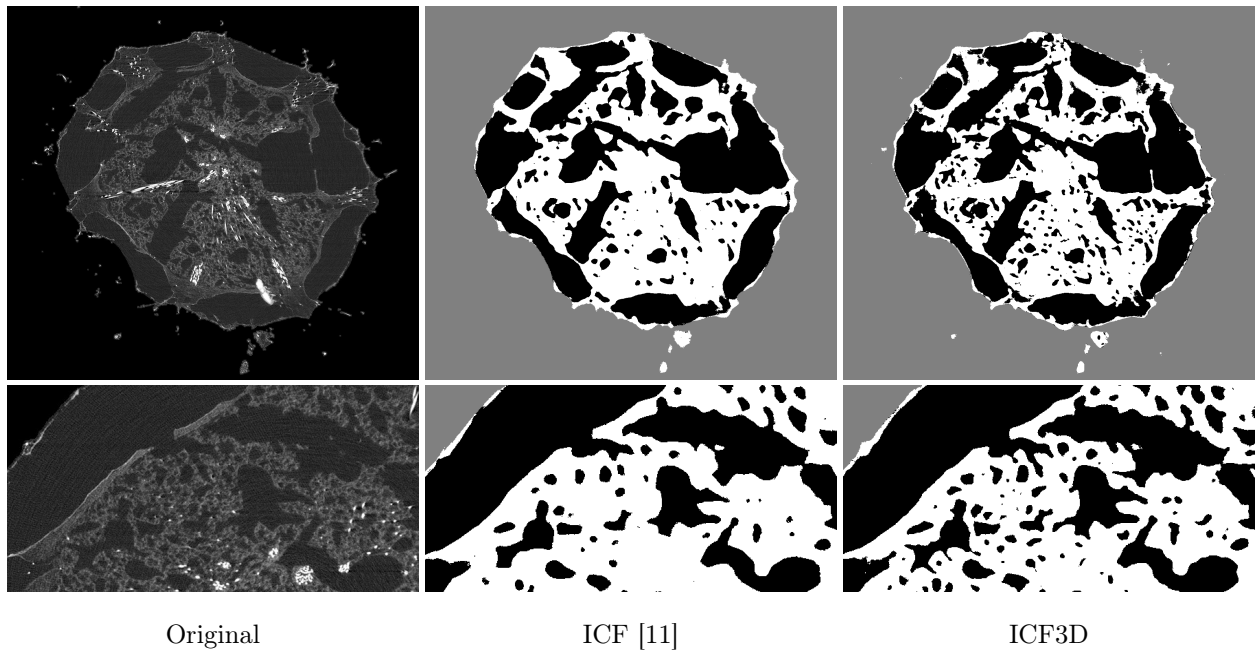


Figure 4. Results for the sponge volume data: The first row depicts slice #85 of the testing stack and corresponding segmentation results by ICF [11] and ICF3D with labeled classes *canal* (black), *tissue* (white) and *background* (gray). Second row shows an enhanced comparison of slice #259. Our ICF3D approach improves segmentation results of subtle canal structures. Figures are best viewed in color.

Because the resolution of this data is in all dimensions the same, a 2D window of the neighborhood for each pixel can be simply replaced by a cube with the same side length. In consequence, we can run the experiments with ICF [11] and ICF3D with comparable parameter configurations. We use feature maps originating from gray values and gradients. Color information is not available for this data.

Figure 4 demonstrates the potential of our volume segmentation method. Each of the images show a slice of the test stack and the voxel classification results of [11] and ICF3D. As can be seen our approach shows improvements when it comes to subtle structures. Without using 3D information from nearby slices the segmentation is more coarse and fine structures get lost.

Method	Rand error [$\cdot 10^{-2}$]	Pixel error [$\cdot 10^{-2}$]
CNN [7]	4.8	6.0
Dense correspondence [17]	6.4	8.3
Watershed Tree [19]	8.4	13.4
Perceptual Grouping [15]	8.4	15.7
CellProfiler [14]	9.0	10.0
Segment features [5]	13.9	10.2
Two-step class. [22]	15.3	8.8
Contextual Grouping [1]	16.2	10.9
Patch-based SVM [13]	23.0	15.0
ICF [11]	28.1	13.5
Ours: ICF3D w/o LBP	22.1	11.3
Ours: ICF3D w/ LBP	21.4	11.4

Table 1. Results of some competitors and our generic method on the ISBI 2012 challenge data [6].

5.2. EM Stack of Neuronal Tissue

We use the freely available *Drosophila* first instar larva ventral nerve cord data [6] of the ISBI 2012 challenge. A stack with 30 slices is labeled to distinguish between the classes *membrane* and *interior* of neurons. The resolution of these stacks is $4 \times 4 \times 50$ nanometers, which is rather coarse in *z*-direction.

To account for this setting, we adapt the neighborhood in *z*-direction in an analogous way. The length of the neighborhood cuboid in *z*-direction is only 10% of the length in the other two directions, which are of higher resolution. For instance, a neighborhood of size $d = 50$ represents a $50 \times 50 \times 5$ cuboid in the image volume. In consequence, only two slices in front of and behind the current slice are used in the process.

We report results for the measures *pixel error* and *rand error* and compare our performance with selected methods of the challenge. The former one is a common measure in binary classification of pixels and is also referred to as accuracy or overall recognition rate. The second measure is related to the well known F-score, which usually consists of precision and recall. However, in this specific task the so called *Rand index* is used for the computation. It is a measure of similarity between two clusters or segmentations. The most important

measure in the ISBI 2012 challenge is the rand error and we therefore also sort our results by this measure (see Table 1).

In order to model the textural differences between membrane and mitochondria in the data not visible in the gradients maps, LBPs (see Sect. 4.4.1) proved to be especially useful for this task. They are a powerful tool in texture classification and increased the performance slightly in our case.

As can be seen in the rand error column, we are not able to produce state-of-the-art results. When taking the actual pixel errors into consideration ICF3D performs decently. It is a typical binary segmentation task, where modeling of context is hardly possible. However, we achieve comparable results to [15] and the patch-based SVM of [13]. The convolutional network approach of [7] shows best performance for this data. Note that we are not using task specific knowledge (*e.g.* shape information) or applying any post-processing for smoothing.

5.3. Analysis of Feature Types

In a third set of experiments, we analyzed the influence of different feature types (see Sec. 4.4.1) for the segmentation of neuronal tissue. For this task, we only used the training data of the ISBI 2012 challenge data, because ground truth labels are available. We split the 30 slices of the training stack into six stacks of five slices each to do a 6-fold cross-validation. This allowed us to analyze performance without using the evaluation server.

In Fig. 5, we show a qualitative comparison of two slices taken from two different cross-validation runs. While the first two columns depict original input images and their corresponding ground truth annotation the other three columns show results with different feature extraction methods. It can be seen that the use of pixel pair differences is sufficient to segment cell membranes. However, synapses are also labeled as membrane and areas of cell interior are not very homogeneous. This is due to the limitation of single pixel values instead of average values in feature maps across whole regions.

When rectangle features and Haar-like features are added, some parts of the synapses are not labeled as membrane anymore. Furthermore, the inhomogeneity in the cell interiors is less. Both issues even improve when context features taken from probability maps are incorporated and some wrong labeled synapses even vanish.

For a quantitative evaluation, we used the provided script of the ISBI 2012 challenge.

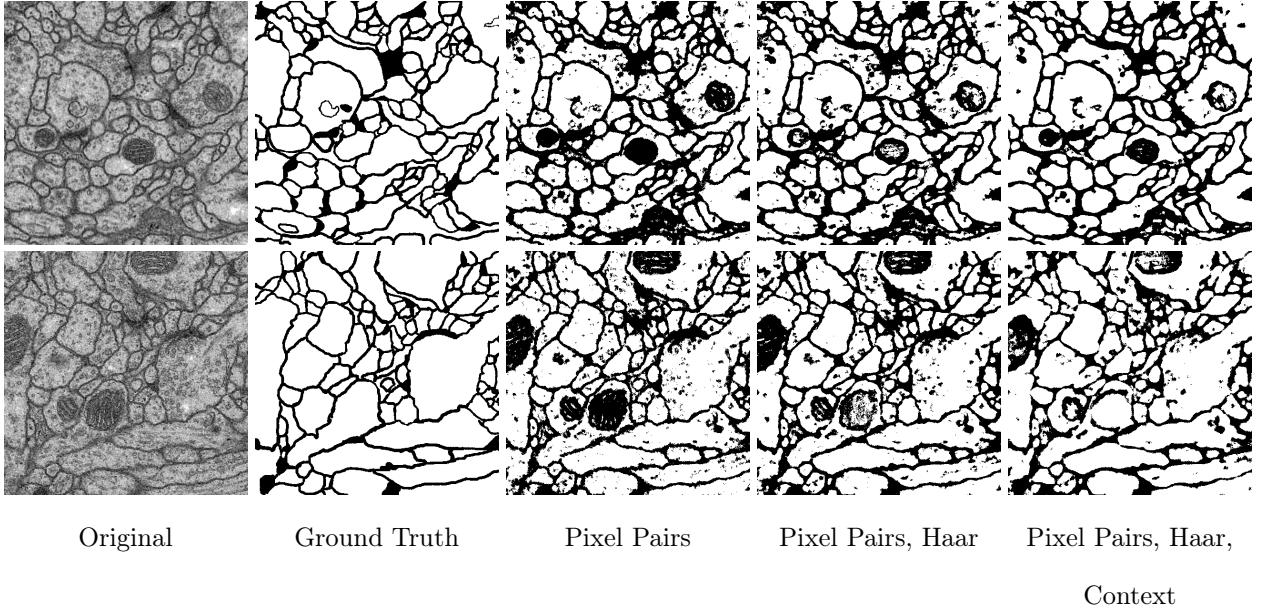


Figure 5. Qualitative comparison of the neuronal tissue dataset using different types for the feature extraction. When only using pixel pair features the results appear to be cluttered. Especially in the larger interior areas the labels of nearby pixels are not consistent. Furthermore, synapses are labeled as cell membrane. These issues lessen or even vanish when Haar-like features and context features are added.

Method	ORR	ARR	F1
ICF3D w/o context	75.03%	64.24%	57.97%
ICF3D w/ context	71.85%	75.09%	63.87%

Table 2. Quantitative results for the multi-class setting of the EM data of neuronal tissue [6].

When only pixel pair differences are allowed the pixel error is 15.6% (average of 6-fold cross-validation). With incorporated rectangle and Haar-like features the error decreases to 14.2%. The best result of 13.5% with respect to the pixel error can be achieved when context features are used.

5.4. Context in a Multi-class Setting

The authors of [6] also provide a multi-class labeling for the EM data of neuronal tissue. It includes additional annotations for the classes *mitochondria* and *synapses* together with the before mentioned classes *membrane* and *interior*. There is also a special background

class called *noise* which is used to exclude erroneous data from the evaluation process. We use this multi-class setting to emphasize on the importance of context in bio-medical image data.

For this evaluation we repeated the 6-fold cross-validation, because ground truth labels are limited to the 30 slices of the training stack. We report quantitative results for the overall recognition rate (ORR) and the average recognition rate (ARR) as they are common measures for multi-class semantic segmentation tasks [11]. Additionally, we provide results for the F1-score which is the harmonic mean of precision and recall.

The results for this set of experiments are listed in Table 2. Due to the fact that 70% of the data is labeled as *interior* the ARR is of special importance in this evaluation. It reports on the average performance for each of the labeled classes. As can be seen there is a huge performance boost when context features are used. Although, the ORR is slightly worse the ARR increases clearly by almost eleven percent points. The enhanced segmentation quality is also visible in the increased F1-score demonstrating that both precision and recall have improved.

6. CONCLUSIONS

In this paper, we presented a fast method for volume image segmentation. We extended an existing semantic segmentation method that incorporates context information. Our proposed algorithm is able to segment even subtle structures in volume data. The method is also applicable to EM stacks, as we have shown for the ISBI 2012 challenge data.

We also performed experiments with volume data from electron microscopy and computer tomography scans and discovered that incorporating information from nearby slices can also be misleading. Whether segmentation performance improves with these information highly depends on the quality of the data and on the recording method itself.

While computer tomography usually creates volume images with isotropic resolution and similar illumination at each slice, electron microscopy stacks often have a coarse resolution in z-dimension as well as illumination and contrast changes. On the other hand, computer tomography scans usually contain artifacts or noise. These things have to be considered when working with such stacks to attain improvements over sequential 2D slice segmentation.

Additionally, we were able to show how important the modeling of context for bio-medical

image data is. Although, classes like *synapses* and *mitochondria* have no fixed positions or constellations in the EM image stack there was a big performance boost visible when context knowledge is used.

ACKNOWLEDGMENT

The authors would like to thank Henry Jahn and Jörg U. Hammel of the *Porifera.net Lab* at the *Friedrich Schiller University Jena* for providing pixel-wise labeled CT imagery data of sponges.

-
1. Erhan Bas, Mustafa G. Uzunbas, Dimitris Metaxas, and Eugene Myers. Contextual grouping in a concept: a multistage decision strategy for em segmentation. In *Proceedings of ISBI 2012 EM Segmentation Challenge*, 2012.
 2. Xavier Boix, Joseph M. Gonfaus, Joost van de Weijer, Andrew D. Bagdanov, Joan Serrat, and Jordi Gonzalez. Harmony potentials - fusing global and local scale for semantic image segmentation. *International Journal of Computer Vision (IJCV)*, 96(1):83–102, 2012.
 3. Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.
 4. Gabriel .J. Brostow, Jamie Shotton, Julien Fauqueur, and Roberto Cipolla. Segmentation and recognition using structure from motion point clouds. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 44–57, 2008.
 5. Radim Burget, Vaclav Uher, and Jan Masek. Trainable segmentation based on local-level and segment-level feature extraction. In *Proceedings of ISBI 2012 EM Segmentation Challenge*, 2012.
 6. Albert Cardona, Stephan Saalfeld, Stephan Preibisch, Benjamin Schmid, Anchi Cheng, Jim Pulokas, Pavel Tomancak, and Volker Hartenstein. An integrated micro- and macroarchitectural analysis of the drosophila brain by computer-assisted serial section electron microscopy. *PLoS Biology*, 8(10), 2010.
 7. Dan C. Ciresan, Alessandro Giusti, Luca M. Gambardella, and Jürgen Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2852–2860, 2012.
 8. Franklin C. Crow. Summed-area tables for texture mapping. In *International Conference and Exhibition on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 207–212, 1984.
 9. Saher Esmeir and Shaul Markovitch. Anytime learning of anycost classifiers. *Machine Learning*, 82(3):445–473, 2011.
 10. Björn Fröhlich, Erik Rodner, and Joachim Denzler. A fast approach for pixelwise labeling of facade images. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, pages 3029–3032, 2010.
 11. Björn Fröhlich, Erik Rodner, and Joachim Denzler. Semantic segmentation with millions of

- features: Integrating multiple cues in a combined random forest approach. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 218–231, 2012.
12. Carolina Galleguillos, Andrew Rabinovich, and Serge Belongie. Object categorization using co-occurrence, location and appearance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.
 13. Saadia Iftikhar and Afzal Godil. The detection of neuronal structure using patch-based multi-features and support vector machines algorithm. In *Proceedings of ISBI 2012 EM Segmentation Challenge*, 2012.
 14. Lee Kamensky. Segmentation of em images of neuronal structures using cellprofiler. In *Proceedings of ISBI 2012 EM Segmentation Challenge*, 2012.
 15. Verena Kaynig, Thomas J. Fuchs, and Joachim M. Buhmann. Neuron geometry extraction by perceptual grouping in sstem images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2902–2909, 2010.
 16. Lubor Ladicky, Chris Russell, Pushmeet Kohli, and Philip H.S. Torr. Graph cut based inference with co-occurrence statistics. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 239–253, 2010.
 17. Dmitry Laptev, Alexander Vezhnevets, Sarvesh Dwivedi, and Joachim M. Buhmann. Anisotropic sstem image segmentation using dense correspondence across sections. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 323–330, 2012.
 18. Ce Liu, , Jenny Yuen, and Antonio Torralba. Sift flow: Dense correspondence across scenes and its applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 33:978–994, 2011.
 19. Ting Liu, Mojtaba Seyedhosseini, Elizabeth Jurrus, and Tolga Tasdizen. Neuron segmentation in em images using series of classifiers and watershed tree. In *Proceedings of ISBI 2012 EM Segmentation Challenge*, 2012.
 20. Albert Montillo, Jamie Shotton, John Winn, Juan E. Iglesias, Dimitry Metaxas, and Antonio Criminisi. Entangled decision forests and their application for semantic segmentation of ct images. In *Proceedings of the International Conference on Information Processing in Medical Imaging (IPMI)*, pages 184–196, 2011.
 21. Timo Ojala, Matti Pietikäinen, and Topi Mäenpää. Multiresolution gray-scale and rotation in-

- variant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 24(7):971–987, 2002.
22. Xiao Tan and Changming Sun. Membrane extraction using two-step classification and post-processing. In *Proceedings of ISBI 2012 EM Segmentation Challenge*, 2012.
 23. Zhuowen Tu and Xiang Bai. Auto-context and its application to high-level vision tasks and 3d brain image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32(10):1744–1757, 2010.
 24. Amelio Vazquez-Reina, Michael Gelbart, Daniel Huang, Jeff Lichtman, Eric Miller, and Hanspeter Pfister. Segmentation fusion for connectomics. In *Proceedings of the International Conference on Computer Vision (ICPR)*, pages 177–184, 2011.
 25. Paul Viola and Michael Jones. Robust real-time object detection. *International Journal of Computer Vision (IJCV)*, 57(2):137–154, July 2004.
 26. Chenxi Zhang, Liang Wang, and Ruigang Yang. Semantic segmentation of urban scenes using dense depth maps. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 708–721, 2010.



Sven Sickert, born in 1986, earned the degree 'Diplom-Informatiker' in 2012 from the Friedrich Schiller University of Jena, Germany. He is currently a PhD student and teaching assistant at the Computer Vision Group of Joachim Denzler at the University of Jena. His research interests are in the field of machine learning, local features and semantic image segmentation with a focus on applications in urban road scenarios and 3D bio-medical image data.



Erik Rodner, earned the Diploma degree in Computer Science with honours in 2007 from the University of Jena, Germany. He received his Ph.D. in 2011 with summa cum laude for his work on learning with few examples, which was done under supervision of Joachim Denzler at the computer vision group of the University of Jena. From 2012 to 2013, Erik joined UC Berkeley and the International Computer Science Institute as a postdoctoral researcher and he is now senior researcher and lecturer in the computer vision group at the University of Jena. His research interests include domain adaptation, deep learning, visual object discovery, active and continuous learning, and scene understanding.



Joachim Denzler, earned the degrees 'Diplom-Informatiker', 'Dr.-Ing.' and 'Habilitation' from the University of Erlangen in years 1992, 1997, and 2003, respectively. Currently, he holds a position of full professor for computer science and is head of the Computer Vision Group, Department of Mathematics and Computer Science, Friedrich Schiller University. He is also Director of the Michael Stifel Center for Data-Driven and Simulation Science, Jena. His research interests comprise the automatic analysis, fusion, and understanding of sensor data, especially development of methods for visual recognition tasks and dynamic scene analysis. He contributed in the area of active vision, 3D reconstruction, as well as object recognition and tracking. He is author and co-author of over 250 journal and conference papers as well as technical articles. He is a member of IEEE, IEEE computer society, DAGM, and GI.