3D Segmentation of SBFSEM Images of Neuropil by a Graphical Model over Supervoxel Boundaries

Bjoern Andres[1], Ullrich Koethe[1], Thorben Kroeger[1],
Moritz Helmstaedter[2], Kevin L. Briggman[2],
Winfried Denk[2], and Fred A. Hamprecht[1]

[1] HCI, University of Heidelberg, Speyerer Str. 6, D-69115 Heidelberg

[2] Max-Planck-Institute for Medical Research, Jahnstrasse 29, D-69120 Heidelberg

December 2011

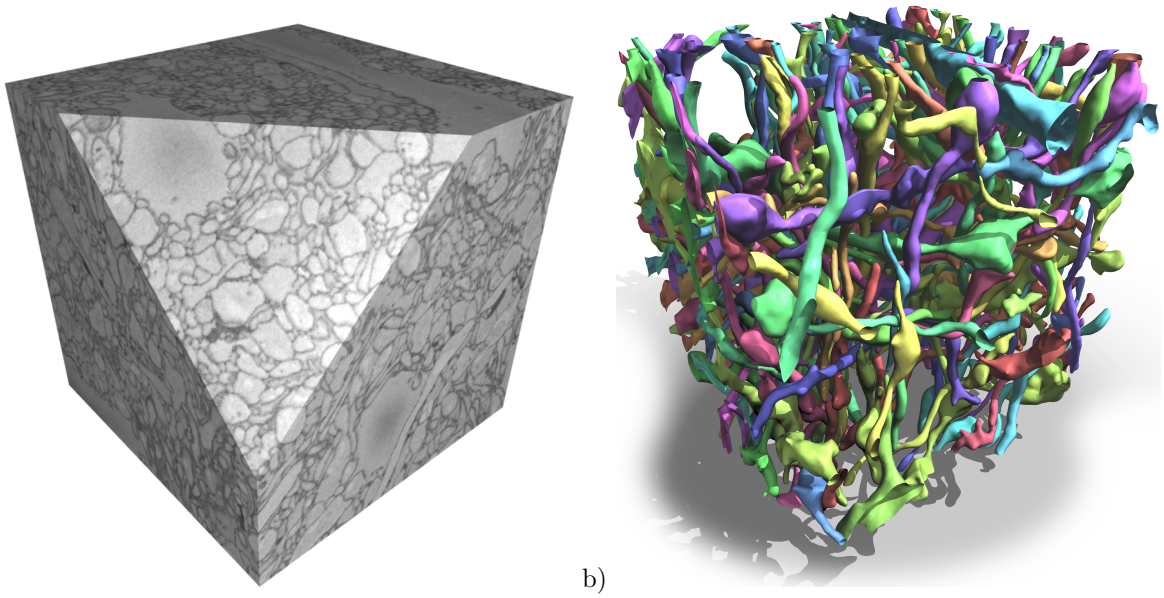
The segmentation of large volume images of neuropil acquired by serial sectioning electron microscopy is an important step towards the 3D reconstruction of neural circuits. The only cue provided by the data at hand are boundaries between otherwise indistinguishable objects. This indistinguishability, combined with the boundaries becoming very thin or faint in places, makes the large body of work on region based segmentation methods inapplicable. On the other hand, boundary-based methods that exploit purely local evidence do not reach the extremely high accuracy required by the application domain that cannot tolerate the global topological errors arising from false local decisions. As a consequence, we propose a supervoxel merging method which arrives at its decisions in a non-local fashion, by posing and approximately solving a *joint* combinatorial optimization problem over *all* faces between supervoxels. The use of supervoxels allows the extraction of expressive geometric features. These are used by the higher-order potentials in a graphical model that assimilate knowledge about the geometry of neural surfaces by automated training on a gold standard. The scope of this improvement is demonstrated on the benchmark dataset E1088 (Helmstaedter et al., 2011) of 7.5 billion voxels from the inner plexiform layer of rabbit retina. We provide C++ source code for annotation, geometry extraction, training and inference.

1 Introduction

Until recently, experimental techniques in neuroscience have either provided detailed information on a small fraction of all neurons (cell recordings, imaging of stochastically stained cells), or information averaged over relatively large regions (fMRI, DTI, EEG). However, detailed knowledge of the complete connectivity pattern of all neurons (Sporns et al., 2005) would be of tremendous value for the understanding of neural computation (Bock et al., 2011; Briggman et al., 2011; Seung, 2011). Obtaining this knowledge has become a realistic objective because serial block-face scanning electron microscopy (SBFSEM) (Briggman and Denk, 2006; Denk and Horstmann, 2004) makes it possible to acquire volume images of up to 1 mm^3 at nanoscopic and nearly isotropic resolution, without alignment problems[1]. A subset of 512^3 voxels from the SBFSEM benchmark dataset E1088 (Helmstaedter et al., 2011) of about 2000^3 voxels is depicted in Fig. 1. In this volume image, the brighter intra-cellular space makes up more than 90% and contrasts the stained extra-cellular space that forms thin membranous faces.

As decisive as the acquisition of these volume images is their semi- or fully automated analysis because a manual 3D reconstruction of neural circuits is costly in terms of labor (Helmstaedter et al., 2008, 2011). The task differs from medical segmentation problems where objects are distinguishable by color or texture, and their geometry can be captured by deformable shape models. In contrast, the SBFSEM

[1] An in-depth treatment of the image alignment problem in serial section *transmission* EM is given by Kaynig et al. (2010a). In SBFSEM, this problem is avoided ab initio because the sample remains stationary (Denk and Horstmann, 2004).



a) b)

Figure 1: a) A subset of 512^3 voxels from a volume image acquired by serial block-face scanning electron microscopy (SBFSEM) at nearly isotropic resolution. All segments are of equal interest. Their indistinguishability (but for the extra-cellular staining) rules out segmentation by segment classification. b) Randomly selected neuronal processes reconstructed by means of the proposed method.

volume image needs to be partitioned completely, into hundreds of cells which are indistinguishable by brightness and texture (separated only by thin layers of extra-cellular space), and whose diverse shape froms intricate branching structures. The extra-cellular space becomes so thin or faint in places that a classification of extra-cellular versus intra-cellular space based on local neighborhoods leads to massive under-segmentation (data not shown). In addition, the sheer size of this data poses a computational challenge.

The approach we pursue in order to segment volume images of at least eight gigavoxels is to first group adjacent voxels that clearly belong together into supervoxels, thus exploiting the redundancy of the volume image. From this initial over-segmentation that consists of many more supervoxels than cells, an explicit representation of faces between supervoxels is constructed. The remaining problem which of these faces to remove, i.e. which supervoxels to merge, is formulated as a combinatorial optimization problem whose objective function factorizes according to a graphical model (Koller and Friedman, 2009). This new formulation has two conceptual advantages which are our main contribution:

1. Decisions whether or not to merge supervoxels are coupled in the graphical model, thus avoiding the exponential risk of under-segmentation inherent to independent decisions[2].

2. Knowledge about the geometry of correct object surfaces and about the distribution of gray values over these surfaces is incorporated into the model in the form of higher-order potentials. These potentials are learned automatically from training data collected by one expert in three days for the application at hand.

2 Related Work

On the methodological side, the classification of voxels that we use to restore local structure in SBFSEM volume images (Section 3.1) is related to statistical approaches to edge detection in two-dimensional images. Various methods have been used to learn from manually segmented images which brightness, color and texture features indicate edges[3], and consistent improvements over traditional edge detectors have been reported (Alpert et al., 2010; Derivaux et al., 2007; Dollar et al., 2006; Konishi et al., 2003; Levner and Zhang, 2007; Martin et al., 2004). The use of learned edge probabilities as elevation maps for the watershed algorithm was proposed by Derivaux et al. (2007) and Levner and Zhang (2007).

Over-segmentations of images that capture all boundaries between objects at the cost of introducing

[2]The risk of incorrectly merging at least one out of $n \in \mathbb{N}$ pairs of adjacent supervoxels is $1-(1-p)^n$, that is, exponential in n, if these decisions are made independently, with an equal error rate $p \in [0, 1/2]$.

[3]Generalized linear models, k-means, support vector machines, (boosted) classification trees, likelihood ratio tests, combinatorial search and fuzzy classification.

additional splits have been studied by Ren and Malik (2003) who coined the term superpixel, and more recently by Gorelick and Basri (2009), Levinshtein et al. (2009), Moore et al. (2008), and Veksler et al. (2010). The advantage of superpixel segmentations as intermediate structures that support the extraction of statistical and geometric features has been demonstrated consistently by Alpert et al. (2007), Gorelick and Basri (2009), and Ren and Malik (2003). An application in 3D is shown by Armstrong et al. (2007) who extract features from volume segments and the faces between segments. We go beyond this approach by extracting features also from curves between faces, using an explicit representation of the geometry and topology of the supervoxel segmentation (Andres et al., 2010b).

Our model for the removal of excessive supervoxel faces builds on previous work on hierarchical segmentation by Alpert et al. (2007), Armstrong et al. (2007), Ren and Malik (2003), and Vanhamel et al. (2003). We extend these methods in which boundaries between segments are classified separately by considering *sets* of boundaries *jointly* and coupling the decisions to remove or preserve these boundaries in a single combinatorial optimization problem whose objective function factorizes according to a graphical model (cf. e.g. Koller and Friedman, 2009).

The majority of graphical models used for image segmentation are Markov Random Fields (MRFs) in which each pixel holds one variable to which a segment label is assigned. Univariate functions relate single variables to an observed image, and functions that depend on two (neighboring) variables penalize label transitions, (cf. e.g. Szeliski et al., 2008). MRFs of this class have been used with superpixels instead of pixels e.g. by He et al. (2006). These MRFs are only suitable for settings in which super-pixels are distinguishable by color or texture and each superpixel belongs to a specific class, e.g. foreground or background. They are, however, unsuitable for segmenting SBFSEM volume images of nervous systems where supervoxels are indistinguishable by brightness and texture and any assignment of labels to segments would be arbitrary. The graphical model we propose reflects this difference: It contains no variables associated with segments but instead, one binary variable for each face between adjacent supervoxels, indicating whether this face is to be removed or preserved. This representation was used before for segmenting photographs (Andres et al., 2011; Kappes et al., 2011).

Towards the 3D reconstruction of neural circuits, several paths are currently pursued, starting from different types of images. Manual reconstructions from nearly isotropic SBFSEM volume images in which the extra-cellular space is stained are possible by distributing (crowd-sourcing) the problem of tracing neuronal processes to non-experts (Helmstaedter et al., 2011). Attempts to solve the reconstruction problem automatically for the same type of data have been made by Jain et al. (2007) and Turaga et al. (2010) who use convolutional neural networks to segment the volume image into distinct neuronal processes, and by Andres et al. (2008) who uses random forests for this purpose. Both methods exploit the isotropy of the volume image by treating all three dimensions equally. In this article, we extend the work of Andres et al. (2008).

A different line of research starts from anisotropic electron microscopic volume images, SBFSEM (Glasner et al., 2011; Jurrus et al., 2009; Macke et al., 2008; Vazquez-Reina et al., 2009) and Serial Section Transmission EM (Chklovskii et al., 2010; Jurrus et al., 2010; Kaynig et al., 2010b; Vazquez-Reina et al., 2011). This anisotropic data is processed slice by slice before the results of all slices are consolidated.

These attempts to reconstruct the geometry of neural circuits are complemented by methods for identifying stained intra-cellular structure such as mitochondria (Kumar et al., 2010; Lucchi et al., 2010) and synapses (Kreshuk et al., 2011).

A semi-automated approach to reconstruct neuronal processes from fluorescent light rather than electron microscopic images is pursued by Lu et al. (2009).

3 Methods

The algorithm we propose for the automated segmentation of SBFSEM volume images consists of eight steps:

1. A set of rotation-invariant non-linear features is extracted from the raw data, describing the 3-dimensional neighborhood of each voxel.

2. A classifier C_{voxel}, trained to distinguish intra-cellular from extra-cellular tissue based on these features, predicts the probability of each voxel to belong to either class.

3. Based on these probabilities, the volume image is over-segmented into supervoxels using a marker-based watershed algorithm.

4. The faces between supervoxels and the curves that bound these faces are represented explicitly, assigned unique labels and encoded as lists of topological coordinates (Brice and Fennema, 1970). The neighborhood system of these objects is stored as a cellular complex (Andres et al., 2010b; Hatcher, 2002).

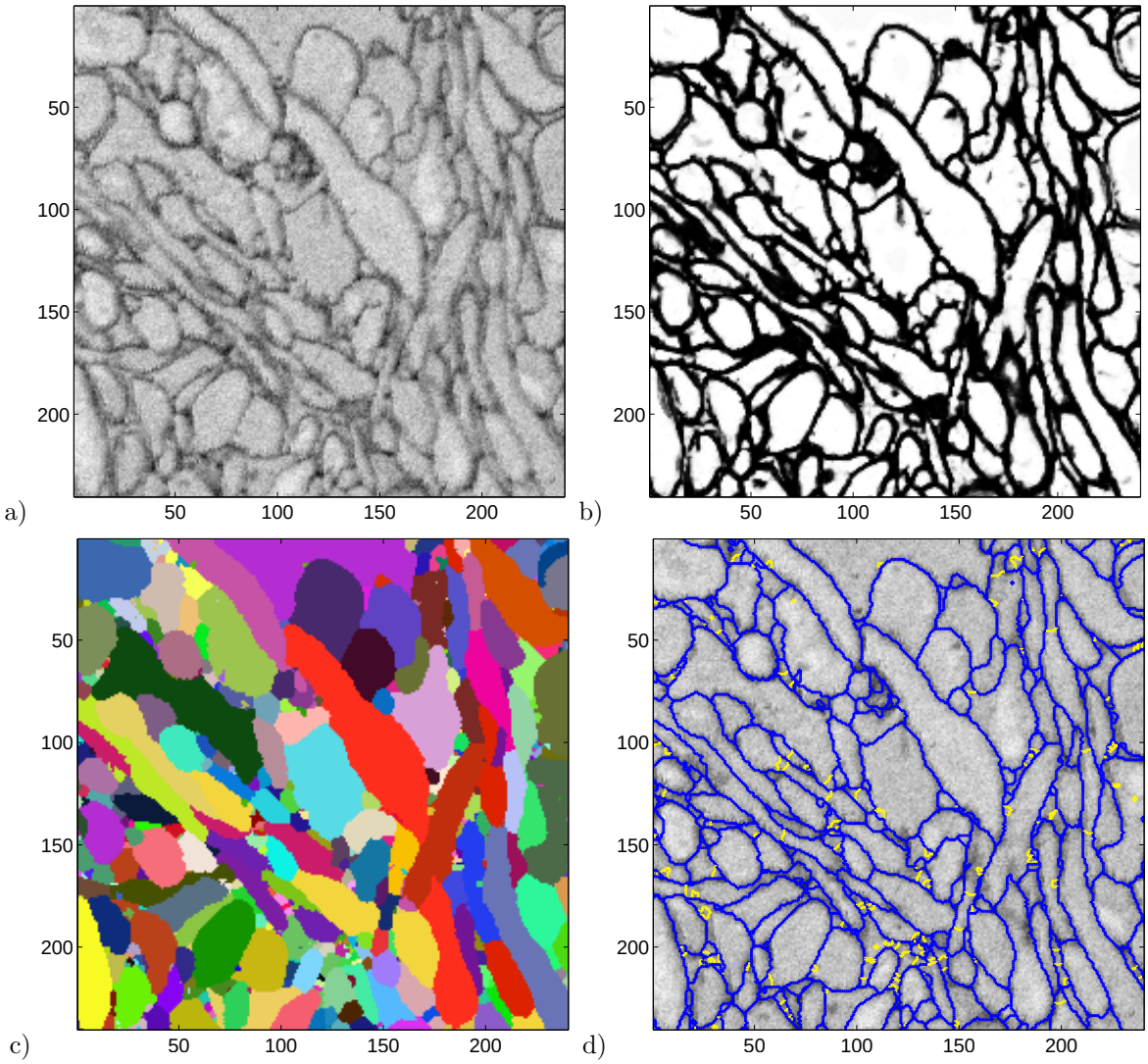


Figure 2: a) A yz-slice of 242^2 voxels from the SBFSEM volume image (Helmstaedter et al., 2011). b) Contrast enhancement between intra- and extra-cellular space obtained by the classifier C_{voxel}. c) Supervoxel segmentation. d) Faces between supervoxels which are preserved (blue) and removed (yellow), according to the solution of the combinatorial optimization problem (4), cf. Section 3.4.

5. From this explicit representation of the geometry and topology of the volume segmentation, features are extracted that describe faces between supervoxels and the distribution of angles between these faces.

6. For every supervoxel face, a second classifier C_{face} predicts the probability that this face should be preserved, given its features. Moreover, for every curve in which several faces meet, a third classifier C_{curve} predicts the probabilities of all configurations that can be attained by preserving or removing these faces, given the angles between them.

7. The predicted probabilities from C_{face} and C_{curve} are combined as first and third order potentials in a graphical model, i.e. an energy function that depends on as many binary variables as there are supervoxel faces, indicating whether these faces are to be preserved or removed.

8. A joint optimal decision to preserve or remove supervoxel faces is found by minimizing this energy function approximately using loopy belief propagation (Kschischang et al., 2001; Pearl, 1988) with message damping (Murphy et al., 1999), followed by lazy flipping (Andres et al., 2010a).

3.1 Restoration of Local Structure by Voxel Classification

Steps 1 and 2 serve to enhance the contrast between intra-cellular and extra-cellular space and to close local gaps in the extra-cellular space that have not been stained. Towards this goal, 28 rotation-invariant

non-linear features are extracted from the volume image, describing the distribution of gray values in a $31 \times 31 \times 31$ neighborhood around each voxel (cf. supplementary material).

The feature vector of each voxel is mapped to an estimated probability that this voxel belongs to the intra-cellular space. This mapping is learned from hand-labeled data (cf. Section 4) by means of a Random Forest C_{voxel} (Breiman, 2001). The learning itself is decribed in the supplementary material. The fraction of decision trees in C_{voxel} that classify a given feature vector as intra-cellular is an estimate of the probability of the corresponding voxel to belong to this class. The predicted probabilities of all voxels form a 3-dimensional probability map that is essentially a restored and contrast-enhanced version of the SBFSEM volume image (Fig. 2b).

Feature extraction and Random Forest prediction are operations whose runtime complexity is linear in the number of voxels. The construction of decision trees during learning has log-linear runtime in the number of training samples which is negligible in this application, compared to the prediction time on volumes much larger than those used for training. Absolute runtimes are summarized in Section 4. The C++ code of the Random Forest and the volume image features are made available as part of the image processing library Vigra (Koethe, 2010).

3.2 Supervoxel Segmentation

To ease the computational burden and to permit the extraction of expressive geometric features, the restored volume image is over-segmented into supervoxels such that no distinct neurons are incorrectly merged (cf. Section 4 for an experimental validation of this claim), at the cost of introducing superfluous splitting faces that do not correspond to cellular membranes. These excessive faces are removed later, in Step 8 (Section 3.4).

The initial supervoxel segmentation is found by means of the marker-based watershed algorithm described in detail in the supplementary material. Markers are defined as the connected components of those voxels that are classified as intra-cellular space by *all* decision trees of the Random Forest C_{voxel}, i.e. those voxels whose estimated probability to represent intra-cellular space is 1. These markers serve as supervoxel seeds. They are grown until the entire volume is occupied such that each voxel is assigned the label of a seed whose *min-max distance* to the given voxel is minimal (cf. Nguyen et al., 2003; Turaga et al., 2009). The min-max distance between two points a and b is the highest point on the lowest path $C(a, b)$ from a to b:

$$d(a, b) = \min_{C(a,b)} \max_{r \in C(a,b)} f(r) \ . \tag{1}$$

The elevation $f(r)$ at a voxel r is given by the number of decision trees that classify this voxel as extra-cellular. The runtime of this algorithm is linear in the number of voxels. Its output is a 3-dimensional *segment label map* that assigns a segment label to each voxel (Fig. 2c).

The segment label map encodes faces between supervoxels and the curves between these faces only implicitly, in the form of neighboring voxels whose segment labels differ. Moreover, it does not store explicitly which segments are adjacent, separated by which faces and in which curves adjacent faces meet. We have developed algorithms and data structures (Andres et al., 2010b) that encode every face between two supervoxels and every curve between supervoxel faces as a list of topological coordinates (Brice and Fennema, 1970) and store the adjacency of these objects as a cellular complex (Hatcher, 2002). A partition data structure allows us to construct this representation in a block-wise fashion, in a runtime that is linear in the number of voxels and log-linear in the number of faces and curves. This facilitates the extraction of geometric features from the supervoxel segmentation based on which excessive faces are identified and removed.

3.3 Extraction of Statistical and Geometric Features

In order to learn from hand-labeled training data (Section 4) what distinguishes essential from excessive supervoxel faces, 31 features of supervoxel faces and 21 features of curves between adjacent faces are extracted. The former describe the distribution of gray values on a given face; the latter characterize the arrangement of adjacent supervoxel faces in terms of the distribution of angles between these faces along the given curve. All features are described in detail in the supplementary material.

The motivation behind the extraction of angles is that humans can often distinguish excessive from essential supervoxel faces without seeing the SBFSEM volume image at all, merely from the configuration or gestalt of supervoxel faces. It is, for instance, obvious to humans that smooth continuations of supervoxel faces are more probable in neural tissue than sharp edges, the latter indicating possible over-segmentation (Fig. 3). The improvement achieved by learning and incorporating this information is shown in Section 4.

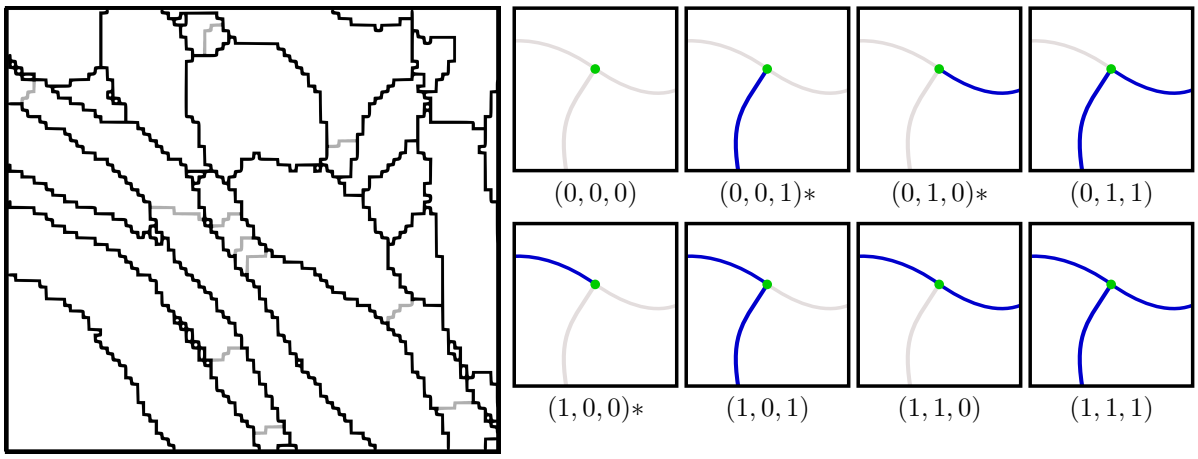


Figure 3: Left: Humans can often distinguish excessive supervoxel faces (gray) from essential ones (black) without seeing the SBFSEM volume image at all, merely from their gestalt and arrangement. Right: There are 8 possibilities to preserve or remove three supervoxel faces (blue) that meet in one curve (green). In the biological problem studied, sharp edges, $(0, 1, 1)$ and $(1, 0, 1)$, are unlikely, and smooth continuations, $(1, 1, 0)$, are more probable than junctions, $(1, 1, 1)$. Open surfaces (*) do not occur; all surfaces in SBFSEM volume images of neural tissue are closed. The probabilities of the remaining 5 configurations are learned from hand-labeled data by the Random Forest C_{curve}.

3.4 Removal of Over-Segmentation

The last step towards the construction of the final segmentation is to remove remaining over-segmentation, i.e. to identify and remove excessive supervoxel faces based on the features just described. Three alternatives of increasing complexity are considered in the following and compared experimentally in Section 4.

In all alternatives, a unique index is assigned to each of the $n_f \in \mathbb{N}$ supervoxel faces and to each of the $n_c \in \mathbb{N}$ curves between faces. Every supervoxel face $j \in \{1, \dots, n_f\}$ is associated with a binary variable $x_j \in \{0, 1\}$ that indicates whether this face is to be removed ($x_j = 0$) or preserved ($x_j = 1$). Due to the discrete structure of the voxel grid, curves have either three or, rarely, four adjacent faces (Andres et al., 2010b). The sets $N_3 \subseteq \{1, \dots, n_c\}$ and $N_4 \subseteq \{1, \dots, n_c\}$ contain the indices of the respective curves. The indices of the adjacent faces are stored in the rows of the matrices $R \in \mathbb{N}^{|N_3| \times 3}$ and $S \in \mathbb{N}^{|N_4| \times 4}$.

In **Alternative A**, a Random Forest C_{face} is trained on hand-labeled data (Section 4) to distinguish essential from excessive faces based exclusively on their 31 features, exactly as proposed in (Andres et al., 2008). The distribution of angles between faces is ignored in this approach. For every face j, C_{face} predicts a probability $p_j^{(1)}(0)$ that this face should be removed and the corresponding probability $p_j^{(1)}(1) = 1 - p_j^{(1)}(0)$ that this face should be preserved, i.e. one probability mass function $p_j^{(1)} : \{0, 1\} \to [0, 1]$ for every face j. The final segmentation is obtained by removing all faces for which $p_j^{(1)}(1)$ is below a threshold $\beta \in [0, 1]$. This threshold establishes a trade-off between false removals (under-segmentation) and false preservations (over-segmentation) (Section 4).

In **Alternative B**, the decision is based only on the gestalt of the over-segmentation, namely on the 21 features that describe the angles between supervoxel faces. The 31 features of supervoxel faces are ignored in this alternative. A Random Forest C_{curve} is trained on hand-labeled data (Section 4) to predict for every curve $k \in N_3$ between three adjacent faces r_{k1}, r_{k2}, r_{k3} the probability of each of the eight possibilities to remove or preserve these faces (Fig. 3). One probability mass function $p_k^{(3)} : \{0, 1\}^3 \to [0, 1]$ is thus obtained for every curve k (Tab. 1). It translates into an energy function $E_k^{(3)} := -\log p_k^{(3)}$ that is minimal where the probability $p_k^{(3)}$ is maximal (Tab. 1). Curves at which only one adjacent face is to be preserved (Fig. 3) do not occur in SBFSEM volume images of neural tissue that contain only closed surfaces. The energy of these configurations is therefore set to infinity. The same argument applies to curves $k \in N_4$ with four adjacent faces $s_{k1}, \dots, s_{k4}$ which motivates the introduction of additional 4th order energy functions $E_k^{(4)} : \{0, 1\}^4 \to \{0, \infty\}$ that suppress those 4 out of the 16 possible configurations

Table 1: Probabilities $p_k^{(3)}$ predicted by C_{curve} for configurations of three adjacent supervoxel faces, and corresponding energies $E_k^{(3)}$.

$x_{r_{k1}}$	$x_{r_{k2}}$	$x_{r_{k3}}$	$p_k^{(3)}(x_{r_{k1}}, x_{r_{k2}}, x_{r_{k3}})$	$E_k^{(3)}(x_{r_{k1}}, x_{r_{k2}}, x_{r_{k3}})$
0	0	0	$p_k(0,0,0)$	$-\log p_k(0,0,0)$
0	0	1	0	∞
0	1	0	0	∞
0	1	1	$p_k(0,1,1)$	$-\log p_k(0,1,1)$
1	0	0	0	∞
1	0	1	$p_k(1,0,1)$	$-\log p_k(1,0,1)$
1	1	0	$p_k(1,1,0)$	$-\log p_k(1,1,0)$
1	1	1	$p_k(1,1,1)$	$-\log p_k(1,1,1)$

in which only one face is preserved[4]: $\forall x_1, \ldots, x_4 \in \{0,1\}$:

$$E_k^{(4)}(x_1, \ldots, x_4) = \begin{cases} \infty & \text{if } x_1 + \ldots + x_4 = 1 \\ 0 & \text{otherwise .} \end{cases} \tag{2}$$

It happens in practice that predictions from different curves that delimit the same face lead to conflicting recommendations as to preserve or remove the given face. Moreover, the decision to preserve or remove one face has implications on the decisions for the neighboring faces, and these implications propagate across the adjacency graph of faces. The final segmentation can thus no longer be obtained by a separate classification of supervoxel faces but requires the *joint* minimization of the total energy comprising the 3rd order potentials of Tab. 1 and the 4th order potentials (2), i.e. the solution of the following combinatorial optimization problem:

$$\min_{x \in \{0,1\}^{n_f}} \left(\sum_{k \in N_3} E_k^{(3)}(x_{r_{k1}}, x_{r_{k2}}, x_{r_{k3}}) + \sum_{k \in N_4} E_k^{(4)}(x_{s_{k1}}, x_{s_{k2}}, x_{s_{k3}}, x_{s_{k4}}) \right). \tag{3}$$

Alternative C is the combination of A and B, and is our best proposal. For every supervoxel face j, the function $p_j^{(1)}$ obtained from C_{face} translates into a unary potential $E_j^{(1)} = -\log p_j^{(1)}$ that assigns a higher energy to the less likely decision. These unary potentials are added to the objective function in (3), leading to the joint optimization problem

$$\min_{x \in \{0,1\}^{n_f}} \left((1-\alpha) \sum_{j=1}^{n_f} E_j^{(1)}(x_j) + \alpha \sum_{k \in N_3} E_k^{(3)}(x_{r_{k1}}, x_{r_{k2}}, x_{r_{k3}}) \right.$$
$$\left. + \alpha \sum_{k \in N_4} E_k^{(4)}(x_{s_{k1}}, x_{s_{k2}}, x_{s_{k3}}, x_{s_{k4}}) \right). \tag{4}$$

The mixture parameter $\alpha \in [0,1]$ is an adjustable parameter that it is optimized on training data (Section 4). The optimization problem (4) specializes to Alternative A for $\alpha = 0$ and to Alternative B for $\alpha = 1$.

Finding solutions of (3) and (4), even approximately, is a formidable problem: While energy functions that decompose according to a graphical model can be minimized efficiently by dynamic programming if the graphical model is a tree (Pearl, 1988) and by solving a minimum s-t-cut problem if the energy function is (permuted) submodular (Boykov et al., 2001; Kolmogorov and Zabih, 2004; Schlesinger, 2007), the graphical model defined here has loops and its energy function is not permuted submodular[5]. The minimization of functions with non-submodular loopy graphical models is in general NP-hard (Kolmogorov and Zabih, 2004).

For small problems obtained from subsets of 150^3 voxels (less than 1/2000) of the SBFSEM benchmark dataset, global optima of (3) and (4) can be found by means of integer linear programming (ILP) (Schrijver, 2003) using branch-and-cut, as shown in (Andres et al., 2010a). Less than 8 GB of RAM and a few minutes of runtime are sufficient in this case. In contrast, 512 GB of RAM and four weeks of runtime are insufficient to solve the ILP for the supervoxel segmentation of the whole dataset of 7.5 billion voxels

[4]Learning to distinguish the remaining 12 configurations based on angles, similar as in Tab. 1, would require substantially more training data and increase the labeling time several times over, an effort that is not justified given that more than 99% of all curves have only three adjacent faces.

[5]In particular, the projections $E_k^{(3)}(1,\cdot,\cdot)$ are supermodular whereas the projections $E_k^{(3)}(0,\cdot,\cdot)$ are submodular (cf. Tab. 1).

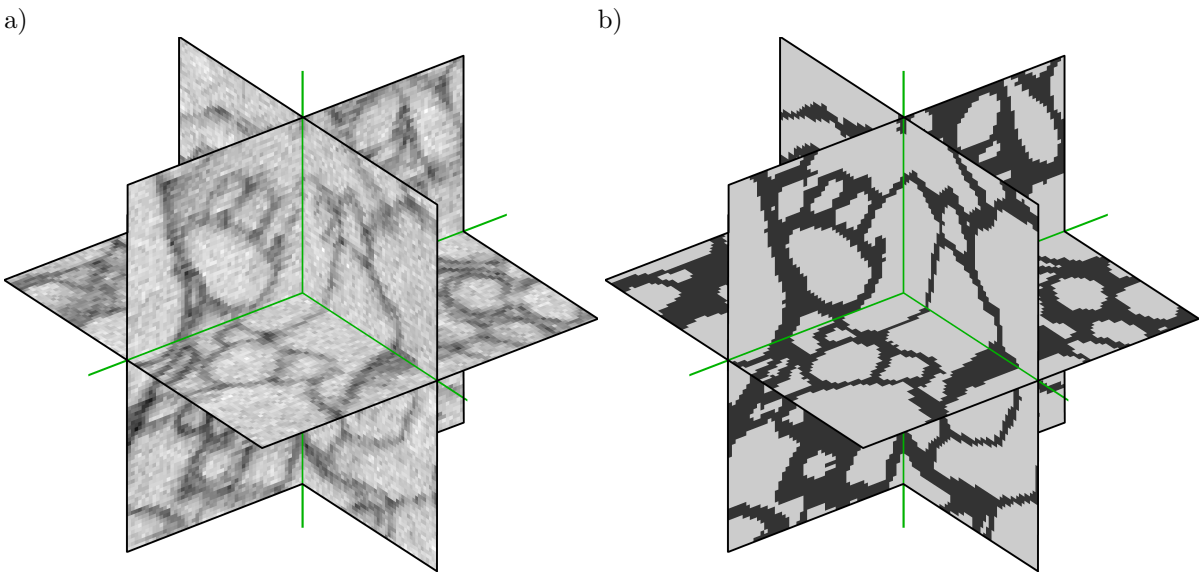


Figure 4: The individual steps and the final result of the segmentation procedure are evaluated on the gold standard validation dataset, a sub-block of 100^3 voxels (a) in which all intra-cellular voxels have been labeled manually (b), cf. Helmstaedter et al. (2008).

(Section 4) that consists of 1.8 million supervoxels, 15.1 million supervoxel faces (variables and unary potential functions) and 30.1 million curves between supervoxel faces (higher-order potential functions). The description of problem (4) alone takes 3.6 GB of space, excluding data structures that are needed to perform the optimization. We therefore need to settle for approximations.

In (Andres et al., 2010a), state-of-the-art alternatives are assessed on ten subsets of 150^3 voxels where the energies can be compared to the global optimum found by ILP. The experiments show that loopy belief propagation (LBP) (Kschischang et al., 2001; Pearl, 1988) with message damping (Murphy et al., 1999) performs exceptionally well, significantly outperforming both tree-reweighted belief propagation (Wainwright and Jordan, 2008) and a dual decomposition ansatz using sub-gradient descent methods (Komodakis et al., 2011) on this problem. The energy of configurations found by LBP deviates by only 0.4% on average from the global optimum. Starting from these, a local search via lazy flipping (Andres et al., 2010a) finds configurations whose energies deviate by only 0.1% from the global optimum. Here, we therefore use a combination of LBP and lazy flipping to solve (3) and (4) approximately.

4 Results

The procedure just described is applied to the SBFSEM benchmark dataset E1088 (Helmstaedter et al., 2011), a volume image of $2048 \times 1792 \times 2048 \approx 7.5 \cdot 10^9$ voxels that shows a part of the inner plexiform layer (IPL) of rabbit retina at a resolution of $22 \times 22 \times 30$ nm^3 (Fig. 1). The gold standard validation subset of 100^3 voxels in which all intra-cellular voxels have been labeled by hand (Fig. 4) is used to quantify the quality of computed segmentations and to compare these results to those of a previous approach (Andres et al., 2008). In addition, for a qualitative comparison with a non-trivial baseline method, 3D reconstructions of neurons from the entire volume image are compared to 3D reconstructions of the same neurons found by means of (Andres et al., 2008).

The Random Forests C_{voxel}, C_{face} and C_{curve} are learned from training data collected in two blocks of 150^3 voxels, one from the dense inside of the IPL where neuronal processes intertwine and the average gray value is low, the other from the clearer border of the IPL where the average gray value is higher. These blocks have no overlap with the validation set and make up less than 0.1% of the volume image.

For C_{voxel}, 3200 voxels are labeled as either intra-cellular or extra-cellular using the open source program Ilastik (Sommer et al., 2011). The labeling is performed incrementally, starting with 500 voxels per class that are placed at least 5 voxels away from each other. The classifier is then trained, and the predicted probability maps for the training volumes are displayed as overlays to the data. Another 500 voxels per class are labeled where the probability maps need improvement. This procedure is repeated another two times, labeling 300 voxels per class. Labeling according to this protocol takes an expert approximately one day.

The restoration of the validation volume image as achieved by the trained Random Forest C_{voxel} is depicted in Fig. 2b. On a maximal random stratified subset that contains as many intra-cellular voxels as extra-cellular voxels, 91.8% of all voxels are classified in agreement with the manual tracing. 8.2%

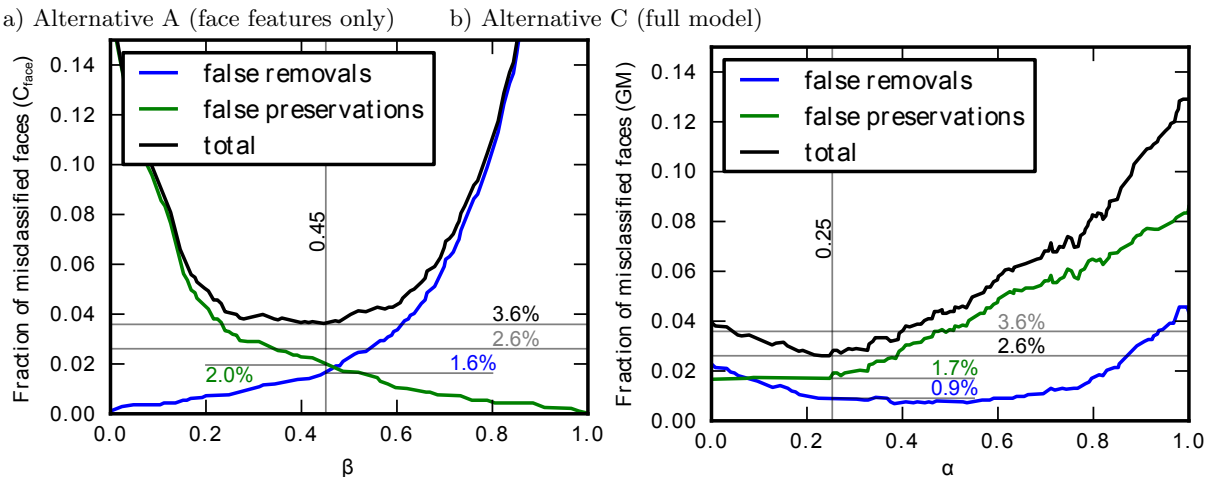


Figure 5: Comparison of models, and choice of parameters. The full graphical model (Alternative C) achieves lower error rates than the model using higher-order potentials only (special case $\alpha = 1$, Alternative B) and lower than a model treating all faces independently (Alternative A).

Table 2: Quality of automatic segmentations of the gold standard validation dataset (Fig. 4) in terms of the fraction of correctly preserved and correctly removed faces. F/C means false/correct. P/R means preservation/removal. All values are given in percent.

Alternative	**FR**	**FP**	**CR**	**CP**	**C**
A (face features only, $\beta = 0.45$)	1.6	2.0	17.2	79.2	96.4
B (curve features only)	4.4	8.6	10.6	76.4	87.0
C (full model, $\alpha = 0.25$)	**0.9**	**1.7**	**17.5**	**79.9**	**97.4**

are classified in disagreement with the manual tracing, 5.1% as extra-cellular space and 3.1% as intracellular space. While a hard thresholding of these predictions fails to give accurate segmentations, the topography of the probability map can still be exploited to obtain a proper over-segmentation by means of the watershed transform. It is crucial that no under-segmentation is introduced at this stage because false mergers could not be corrected in subsequent steps of the procedure as presented here. The following indicator of under-segmentation corroborates this assumption: For every connected component j of voxels labeled as intra-cellular in the validation set and every segment k in the watershed segmentation of the same volume, Q_{jk} denotes the number of voxels that belong at the same time to the connected component (true segment) j and to the watershed segment k. Let R be obtained from the overlap matrix Q by column normalization. R_{jk} is then the fraction of the watershed segment k overlapped by the true segment j. In an over-segmentation, all except very small segments have no relevant overlap with more than one true segment. This is quantified by the under-segmentation index, the second largest entry of the k-th column of R. In this application, *all* watershed segments that are larger than 100 voxels exhibit an under-segmentation index of less than 10%.

In order to train C_{face} and C_{curve}, 5000 faces between supervoxels are labeled by hand in the watershed segmentations of the training volumes, using an interactive 3D labeling tool (Kröger, 2010) based on the Visualization Toolkit[6]. Faces that are easy to label are labeled first because ambiguous cases become clear once the surrounding faces are labeled. It takes an expert roughly two days to collect a training set of the given size. The labeled faces and the features of these faces make up a training set for C_{face}. Triples of adjacent labeled faces and the angles between these faces are used to train C_{curve}. The training set for C_{curve} is extended by adding all permutations of adjacent supervoxel faces[7].

Over-segmentation is removed by identifying and removing excessive supervoxel faces. Alternatives A (face features only), B (curve features only) and C (full model) described in Section 3.4 are compared. Optimization is performed by loopy belief propagation (50 steps) with a message damping of 0.3, followed by lazy flipping (Andres et al., 2010a), using a maximum subgraph size of 3. The quality of the final segmentation is measured on the validation set in terms of the fraction of supervoxel faces that are falsely removed from the initial over-segmentation or falsely preserved, as judged by the dense manual segmentation that is the gold standard.

For Alternative A (face features only), the trade-off between false removals and false preservations

[6] http://www.vtk.org

[7] As an example, a triple where the first face is labeled as excessive and the last two faces are labeled as essential $(0, 1, 1)$ makes up three items labeled $(0, 1, 1)$, $(1, 0, 1)$ and $(1, 1, 0)$ in the training set for which the angle features are permuted accordingly.

Table 3: At every curve in which three supervoxel faces meet, the classifier C_{curve} predicts a probability for each of the eight possibilities to remove (0) or preserve (1) these faces (cf. Fig. 3). The three configurations $(0,0,1)$, $(0,1,0)$ and $(1,0,0)$ in which only one face is preserved do not occur in the SBFSEM volume image that contains only closed surfaces. The table below shows the confusion of the classifier C_{curve} for the remaining five configurations. Columns correspond to the predictions, rows to the truth. It can be seen from this table that the configurations $(0,1,1)$, $(1,0,1)$ and $(1,1,0)$ are well separated whereas it is not possible to distinguish these configurations from $(1,1,1)$ by C_{curve} alone. This motivates the combination of C_{face} and C_{curve} in the graphical model (4).

C_{curve}:	$(0,0,0)$	$(1,1,0)$	$(1,0,1)$	$(0,1,1)$	$(1,1,1)$
$(0,0,0)$	**1.9%**	0.7%	0.9%	0.5%	0.4%
$(1,1,0)$	1.7%	**6.9%**	0.9%	0.7%	2.9%
$(1,0,1)$	2.6%	0.9%	**6.7%**	1.1%	3.2%
$(0,1,1)$	2.3%	0.7%	1.6%	**5.3%**	2.9%
$(1,1,1)$	4.6%	5.0%	6.8%	5.2%	**33.8%**

Table 4: Absolute runtime, memory consumption and parallelization of the segmentation of 7.5 billion voxels. Numbers are given for an Intel 4×Quad Xeon equipped with 128 GB RAM, running at 2.4 GHz.

Computation	**Runtime**	**CPUs**	**RAM/CPU**	**HDD**
Voxel features	1d 11h 59m	10	$<$ 2 GB*	842 GB
Training of C_{voxel}	$<$ 10m	1	$<$ 2 GB	$<$ 1 GB
Voxel classification (C_{voxel})	11h 39m	10	$<$ 2 GB*	8 GB
Supervoxel segmentation	2h 13m	1	72 GB	61 GB
Geometry extraction	19h 22m	10	$<$ 2 GB*	273 GB
Supervoxel face features	2d 14h 28m	10	$<$ 2 GB	3 GB
Training of C_{face}	$<$ 10m	1	$<$ 2 GB	$<$ 1 GB
Face classification (C_{face})	$<$ 10m	10	$<$ 2 GB	$<$ 1 GB
Curve angle features	1h 25m	10	$<$ 2 GB	2 GB
Training of C_{curve}	$<$ 10m	1	$<$ 2 GB	$<$ 1 GB
Curve classification (C_{curve})	$<$ 10m	10	$<$ 2 GB	$<$ 1 GB
Loopy Belief Propagation	7h 41m	1	$<$ 2 GB	$<$ 1 GB
Lazy Flipper ($d = 3$)	2d 9h 38m	1	89 GB	$<$ 1 GB

*) adjustable by selecting a block size.

w.r.t. the threshold β is depicted in Fig. 5a. At the optimal $\beta = 0.45$, 96.4% of all faces are classified correctly (Tab. 2). Alternative B (curve features only) performs worse, classifying 84.1% of all faces correctly (Tab. 2). The reason can be seen in the confusion matrix of C_{curve} (Tab. 3): The configurations $(0,1,1)$, $(1,0,1)$ and $(1,1,0)$ are separated well by the curve features (angles) which demonstrates their predictive power. However, it is not possible to distinguish these configurations from $(1,1,1)$ based on the angles alone. In other words, it is hard to decide, based on the angles alone, whether or not faces should be removed. However, if there is evidence that a face should be removed, it is also clear which of the three this should be. Overall, this motivates the combination of the predictions from C_{face} and C_{curve} in Alternative C. Alternative C which takes into account evidence from both faces and curves yields the overall best results for all mixture parameters α between 0.05 and 0.4 (Fig. 5b). At the optimal $\alpha = 0.25$, 97.5% of all faces are classified correctly (Tab. 2). Compared to Alternative A, the fraction of falsely removed faces is reduced by more than 43% to 0.9%. At the same time, the fraction of false preservations is reduced by 15% to 1.7% (cf. Tab. 2).

Reconstructions of neurons from the entire volume image are possible using Alternatives A and C. The false removal rate of Alternative B is too high, leading to excessive under-segmentation in large scale reconstructions where neurons have between 1000 and 10000 supervoxel faces all of which need to be correctly preserved. In practice, a false removal rate below 1% is indispensable to prevent under-segmentation in large scale reconstructions. This can be achieved with Alternative A by setting $\beta = 0.2$ and by Alternative C with $\alpha = 0.25$. It can be seen from Fig. 6 that the full-featured Alternative C affords better reconstructions; less boundaries are falsely preserved and thus, larger parts of neuronal processes are correctly merged. A reconstruction of 100 randomly selected processes is shown in Fig. 7.

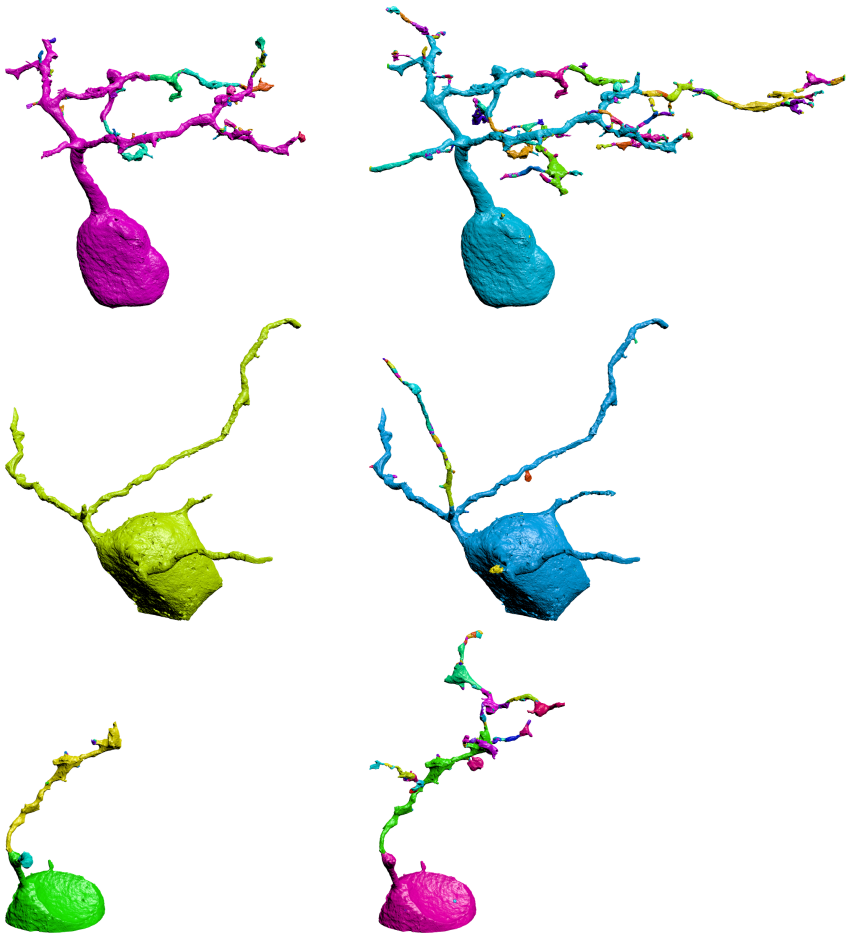


Figure 6: Comparison of automatic segmentations by Approach A with $\beta = 0.45$ (left) and the full graphical model (Approach C) with $\alpha = 0.25$ (right). In this example, Approach C reduces over-segmentation without introducing under-segmentation which is in accordance with the quantitative results in Tab. 2.

5 Conclusion

An automated procedure for segmenting SBFSEM volume images of neuropil is presented. It starts by over-segmenting the volume image into supervoxels and selectively merges supervoxels based on features of the over-segmentation. Particular to this approach is the coupling of the individual merging decisions in a joint optimization problem which allows clear evidence for the existence or absence of boundaries to propagate over long distances to ambiguous regions. The representation used and the training set provided allow the inference procedure to learn by itself gestalt laws that are most salient for the problem at hand.

Compared to a previous method (Andres et al., 2008) in which all decisions are made separately, the number of false mergers is reduced (by 43%) to 0.9% while the number of false splits is reduced (by 15%) to 1.7% at the same time. Although the thinnest neuronal processes are still falsely split and the fully automated and exhaustive 3D reconstruction of neuropil remains an unsolved problem, the accuracy achieved by the new method constitutes substantial progress over (Andres et al., 2008), and the resulting segmentations (Fig. 7) provide a basis for applying stitching procedures that are subject of future work. Advances in tissue preparation may alleviate this problem.

The procedure is not restricted in principle to SBFSEM volume images, and should carry over to other biological or medical images that bear resemblance with the data studied here. In general, the idea of segmenting images based on geometric features extracted from an initial over-segmentation is applicable in practice to volume images that consist of several billion voxels, provided that local evidence of boundaries exists between different objects. The C++ source code for geometry and topology extraction, classification, combinatorial optimization and a GUI for voxel labeling are provided[8].

[8] hci.iwr.uni-heidelberg.de/Software

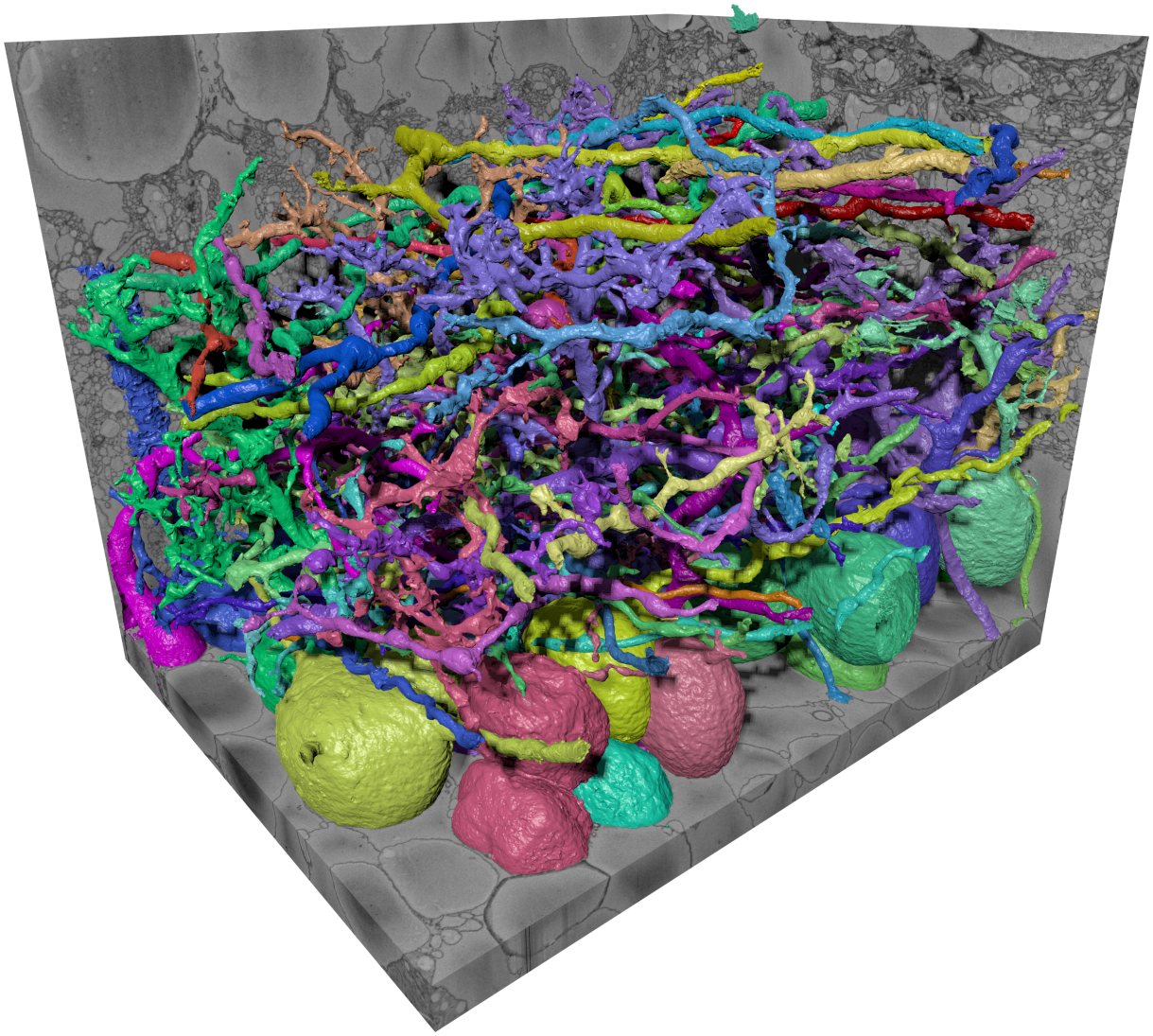


Figure 7: Automated 3D reconstruction of 100 randomly selected neuronal processes from the SBFSEM benchmark dataset E1088 (Helmstaedter et al., 2011) of the inner plexiform layer of rabbit retina, by means of the proposed method.

References

Alpert, S., Galun, M., Basri, R., Brandt, A., 2007. Image segmentation by probabilistic bottom-up aggregation and cue integration, in: CVPR.

Alpert, S., Galun, M., Nadler, B., Basri, R., 2010. Detecting faint curved edges in noisy images, in: Daniilidis, K., Maragos, P., Paragios, N. (Eds.), ECCV, Springer. pp. 750–763.

Andres, B., Kappes, J.H., Beier, T., Koethe, U., Hamprecht, F.A., 2011. Probabilistic image segmentation with closedness constraints, in: ICCV. (in print).

Andres, B., Kappes, J.H., Koethe, U., Hamprecht, F.A., 2010a. The Lazy Flipper: MAP inference in higher-order graphical models by depth-limited exhaustive search. ArXiv e-prints http://arxiv.org/abs/1009.4102.

Andres, B., Koethe, U., Kroeger, T., Hamprecht, F.A., 2010b. How to extract the geometry and topology from very large 3D segmentations. ArXiv e-prints http://arxiv.org/abs/1009.6215.

Andres, B., Köthe, U., Helmstaedter, M., Denk, W., Hamprecht, F.A., 2008. Segmentation of SBFSEM volume data of neural tissue by hierarchical classification, in: Rigoll, G. (Ed.), Pattern Recognition, Springer. pp. 142–152.

Armstrong, C.J., Price, B.L., Barrett, W.A., 2007. Interactive segmentation of image volumes with live surface. Computer Graphics 31, 212–229.

Bock, D.D., Lee, W.C.A., Kerlin, A.M., Andermann, M.L., Hood, G., Wetzel, A.W., Yurgenson, S., Soucy, E.R., Kim, H.S., Reid, R.C., 2011. Network anatomy and in vivo physiology of visual cortical neurons. Nature 471, 177–182.

Boykov, Y., Veksler, O., Zabih, R., 2001. Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. 23, 1222–1239.

Breiman, L., 2001. Random forests. Machine Learning 45, 5–32.

Brice, C.R., Fennema, C.L., 1970. Scene analysis using regions. Artificial Intelligence 1, 205–226.

Briggman, K.L., Denk, W., 2006. Towards neural circuit reconstruction with volume electron microscopy techniques. Current Opinion in Neurobiology 16, 562–570.

Briggman, K.L., Helmstaedter, M., Denk, W., 2011. Wiring specificity in the direction-selectivity circuit of the retina. Nature 471, 183–188.

Chklovskii, D.B., Vitaladevuni, S., Scheffer, L.K., 2010. Semi-automated reconstruction of neural circuits using electron microscopy. Current Opinion in Neurobiology (in press).

Denk, W., Horstmann, H., 2004. Serial block-face scanning electron microscopy to reconstruct three-dimensional tissue nanostructure. PLoS Biology 2, e329.

Derivaux, S., Lefevre, S., Wemmert, C., Korczak, J., 2007. On machine learning in watershed segmentation, in: IEEE Workshop Machine Learning for Signal Processing, pp. 187–192.

Dollar, P., Tu, Z., Belongie, S., 2006. Supervised learning of edges and object boundaries, in: CVPR, IEEE Computer Society. pp. 1964–1971.

Glasner, D., Hu, T., Chklovskii, D., Basri, R., 2011. High resolution segmentation of neuronal tissue from low depth-resolution em imagery, in: EMMCVPR.

Gorelick, L., Basri, R., 2009. Shape based detection and top-down delineation using image segments. Int. J. Comput. Vision 83, 211–232.

Hatcher, A., 2002. Algebraic topology. Cambridge University Press.

He, X., Zemel, R.S., Ray, D., 2006. Learning and incorporating top-down cues in image segmentation. ECCV , 338–351.

Helmstaedter, M., Briggman, K.L., Denk, W., 2008. 3D structural imaging of the brain with photons and electrons. Current Opinion in Neurobiology 18, 633–641.

Helmstaedter, M., Briggman, K.L., Denk, W., 2011. High-accuracy neurite reconstruction for high-throughput neuroanatomy. Nature Neuroscience 14, 1081–1088.

Jain, V., Murray, J.F., Roth, F., Turaga, S., Zhigulin, V., Briggman, K.L., Helmstaedter, M.N., Denk, W., Seung, H.S., 2007. Supervised learning of image restoration with convolutional networks, in: ICCV, IEEE Computer Society, Los Alamitos, CA, USA.

Jurrus, E., Hardy, M., Tasdizen, T., Fletcher, P.T., Koshevoy, P., Chien, C.B., Denk, W., Whitaker, R., 2009. Axon tracking in serial block-face scanning electron microscopy. Medical Image Analysis 13, 180–188.

Jurrus, E., Paiva, A.R., Watanabe, S., Anderson, J.R., Jones, B.W., Whitaker, R.T., Jorgensen, E.M., Marc, R.E., Tasdizen, T., 2010. Detection of neuron membranes in electron microscopy images using a serial neural network architecture. Medical Image Analysis 14, 770–783.

Kappes, J.H., Speth, M., Andres, B., Reinelt, G., Schnörr, C., 2011. Globally optimal image partitioning by multicuts, in: EMMCVPR.

Kaynig, V., Fischer, B., Müller, E., Buhmann, J.M., 2010a. Fully automatic stitching and distortion correction of transmission electron microscope images. Journal of Structural Biology 171, 163–173.

Kaynig, V., Fuchs, T., Buhmann, J.M., 2010b. Neuron geometry extraction by perceptual grouping in ssTEM images, in: CVPR.

Koethe, U., 2010. Computer vision library vigra. http://hci.iwr.uni-heidelberg.de/vigra.

Koller, D., Friedman, N., 2009. Probabilistic Graphical Models. MIT Press.

Kolmogorov, V., Zabih, R., 2004. What energy functions can be minimized via graph cuts? IEEE Trans. Pattern Anal. Mach. Intell. 26, 147–159.

Komodakis, N., Paragios, N., Tziritas, G., 2011. Mrf energy minimization and beyond via dual decomposition. IEEE Trans. Pattern Anal. Mach. Intell. 33, 531–552.

Konishi, S., Yuille, A., Coughlan, J., Zhu, S., 2003. Statistical edge detection: Learning and evaluating edge cues. IEEE Trans. Pattern Anal. Mach. Intell. 25, 57–74.

Kreshuk, A., Straehle, C.N., Sommer, C., Koethe, U., Cantoni, M., Knott, G., Hamprecht, F.A., 2011. Automated detection and segmentation of synaptic contacts in nearly isotropic serial electron microscopy images. PLoS ONE In print.

Kröger, T., 2010. Exploratory and Computational Analysis of Huge Volume Images for Connectomics. Master's thesis. University of Heidelberg. Germany.

Kschischang, F.R., Frey, B.J., Loeliger, H., 2001. Factor graphs and the sum-product algorithm. Transactions on Information Theory 47, 498–519.

Kumar, R., Reina, A.V., Pfister, H., 2010. Radon-like features and their application to connectomics, in: IEEE Computer Society Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA).

Levinshtein, A., Stere, A., Kutulakos, K., Fleet, D., Dickinson, S., Siddiqi, K., 2009. TurboPixels: Fast Superpixels Using Geometric Flows. TPAMI 31, 2290–2297.

Levner, I., Zhang, H., 2007. Classification-driven watershed segmentation. Transactions on Image Processing 16, 1437–1445.

Lu, J., Fiala, J.C., Lichtman, J.W., 2009. Semi-automated reconstruction of neural processes from large numbers of fluorescence images. PLoS ONE 4, e5655.

Lucchi, A., Smith, K., Achanta, R., Lepetit, V., Fua, P., 2010. A fully automated approach to segmentation of irregularly shaped cellular structures in EM images, in: International Conference on Medical Image Computing and Computer Assisted Intervention.

Macke, J.H., Maack, N., Gupta, R., Denk, W., Schölkopf, B., Borst, A., 2008. Contour-propagation algorithms for semi-automated reconstruction of neural processes. Journal of Neuroscience Methods 167, 349–357.

Martin, D.R., Fowlkes, C.C., Malik, J., 2004. Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Trans. Pattern Anal. Mach. Intell. 26, 530–549.

Moore, A., Prince, S., Warrell, J., Mohammed, U., Jones, G., 2008. Superpixel lattices, in: CVPR.

Murphy, K.P., Weiss, Y., Jordan, M.I., 1999. Loopy belief propagation for approximate inference: An empirical study, in: UAI, pp. 467–475.

Nguyen, H.T., Worring, M., van den Boomgaard, R., 2003. Watersnakes: Energy-driven watershed segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 25, 330–342.

Pearl, J., 1988. Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann, San Francisco, CA, USA.

Ren, X., Malik, J., 2003. Learning a classification model for segmentation, in: ICCV, pp. 10–16.

Schlesinger, D., 2007. Exact solution of permuted submodular MinSum problems, in: EMMCVPR.

Schrijver, A., 2003. Combinatorial Optimization: Polyhedra and Efficiency. Springer.

Seung, H.S., 2011. Neuroscience: Towards functional connectomics. Nature 471, 170–172.

Sommer, C., Straehle, C., Koethe, U., Hamprecht, F.A., 2011. ilastik: Interactive Learning and Segmentation Toolkit, in: 8th IEEE International Symposium on Biomedical Imaging. (www.ilastik.org).

Sporns, O., Tononi, G., Kötter, R., 2005. The human connectome: A structural description of the human brain. PLoS Computational Biology 1, e42.

Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., Rother, C., 2008. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. IEEE Trans. Pattern Anal. Mach. Intell. 30, 1068–1080.

Turaga, S.C., Briggman, K.L., Helmstaedter, M., Denk, W., Seung, H.S., 2009. Maximin affinity learning of image segmentation, in: Advances in Neural Information Processing Systems.

Turaga, S.C., Murray, J.F., Jain, V., Roth, F., Helmstaedter, M., Briggman, K., Denk, W., Seung, H.S., 2010. Convolutional networks can learn to generate affinity graphs for image segmentation. Neural Computation 22, 511–538.

Vanhamel, I., Pratikakis, I., Sahli, H., 2003. Multiscale gradient watersheds of color images. Transactions on Image Processing 12, 617–626.

Vazquez-Reina, A., Huang, D., Gelbart, M., Lichtman, J., Miller, E., Pfister, H., 2011. Segmentation fusion for connectomics. In print.

Vazquez-Reina, A., Miller, E., Pfister, H., 2009. Multiphase geometric couplings for the segmentation of neural processes, in: CVPR.

Veksler, O., Boykov, Y., Mehrani, P., 2010. Superpixels and supervoxels in an energy optimization framework, in: ECCV.

Wainwright, M.J., Jordan, M.I., 2008. Graphical Models, Exponential Families, and Variational Inference. Now Publishers Inc., Hanover, MA, USA.