NeuroResource

Neuron

SegEM: Efficient Image Analysis for High-Resolution Connectomics

Highlights

- SegEM: semi-automated reconstruction workflow for connectomics image data
- Reconstruction speed about 10-times higher than that of alternative methods
- Applied to 3D electron microscopy data from mouse retina and cortex
- Ready-to-use analysis code for novel datasets

Authors

Manuel Berning, Kevin M. Boergens, Moritz Helmstaedter

Correspondence

manuel.berning@brain.mpg.de (M.B.), mh@brain.mpg.de (M.H.)

In Brief

Berning et al. developed SegEM, an image analysis workflow for efficiently extracting synaptic connectivity maps (connectomes) from large 3D-EM image datasets. The method is about 10-times more efficient than alternative methods.

Berning et al., 2015, Neuron *87*, 1193–1206
September 23, 2015
http://dx.doi.org/10.1016/j.neuron.2015.09.003

CellPress

Neuron
NeuroResource

SegEM: Efficient Image Analysis for High-Resolution Connectomics

Manuel Berning,[1,*] Kevin M. Boergens,[1] and Moritz Helmstaedter[1,*]
[1]Department of Connectomics, Max Planck Institute for Brain Research, Max-von-Laue-Strasse 4, 60438 Frankfurt, Germany
*Correspondence: manuel.berning@brain.mpg.de (M.B.), mh@brain.mpg.de (M.H.)
http://dx.doi.org/10.1016/j.neuron.2015.09.003

SUMMARY

Progress in electron microscopy-based high-resolution connectomics is limited by data analysis throughput. Here, we present SegEM, a toolset for efficient semi-automated analysis of large-scale fully stained 3D-EM datasets for the reconstruction of neuronal circuits. By combining skeleton reconstructions of neurons with automated volume segmentations, SegEM allows the reconstruction of neuronal circuits at a work hour consumption rate of about 100-fold less than manual analysis and about 10-fold less than existing segmentation tools. SegEM provides a robust classifier selection procedure for finding the best automated image classifier for different types of nerve tissue. We applied these methods to a volume of 44 × 60 × 141 μm^3 SBEM data from mouse retina and a volume of 93 × 60 × 93 μm^3 from mouse cortex, and performed exemplary synaptic circuit reconstruction. SegEM resolves the tradeoff between synapse detection and semi-automated reconstruction performance in high-resolution connectomics and makes efficient circuit reconstruction in fully-stained EM datasets a ready-to-use technique for neuroscience.

INTRODUCTION

Mapping neuronal circuits at single-cell resolution is the goal of high-resolution connectomics (Helmstaedter, 2013, Denk et al., 2012, Lichtman and Denk, 2011). While 3D-EM imaging methods have progressed substantially (Denk and Horstmann, 2004, Knott et al., 2008, Hayworth et al., 2006, Helmstaedter et al., 2013, Takemura et al., 2013; for a review, see Briggman and Bock, 2012), and 3D-EM imaging setups are being installed in many laboratories worldwide, the reconstruction speed of such data is lagging behind by at least 3 orders of magnitude (Helmstaedter, 2013). Fully automated reconstruction approaches have not provided the required reconstruction accuracy to date (Funke et al., 2012; Andres et al., 2012a, 2012b; Vazquez-Reina et al., 2011; Seyedhosseini et al., 2011, 2013; Jain et al., 2007, 2010a, 2011; Turaga et al., 2009, 2010; Ciresan et al., 2012; Kaynig et al., 2015; Nunez-Iglesias et al., 2014; Sommer et al., 2011; Liu et al., 2014). Instead, combinations of massive manual annotation with automated analysis methods have yielded first substantial connectivity maps in the fly optical system (Takemura et al., 2013) and mouse retina (Helmstaedter et al., 2013).

However, these approaches were limited either by analysis speed when proofreading pre-segmented data (Takemura et al., 2013; 14,400 hr investment for 105 mm circuit path length) or by the lack of direct synapse identification when using special cell-membrane-enhanced EM staining (Briggman et al., 2011; Helmstaedter, 2013; ~20,000 hr investment for 640 mm circuit path length).

In fact, even small blocks of neuronal tissue (Figures 1A and 1B) contain enormous amounts of local neuronal circuitry: for example, a block of mouse retina sized 44 μm × 60 μm × 141 μm on a side contains about 400 cell bodies, 1.5 m of neuronal wires, and 0.5 million chemical synapses (Figure 1A; Briggman et al., 2011; Helmstaedter et al., 2013). To reconstruct neuronal circuits from such a sizeable volume of neuronal tissue, dendrites and axons of all neurons have to be followed through the dataset and all synapses identified. Unequivocal synapse identification requires the staining of synaptic vesicles and potential postsynaptic structures. Such conventionally stained 3D-EM data (Figures 1C and 1D), however, also stains mitochondria and other intracellular structures, which results in highly overlapping single-voxel gray value distributions (Figure 1E).

The fully manual volume reconstruction of neuronal circuits in such a 3D-EM dataset (Figures 1A and 1B) would consume enormous amounts of human work hours even for medium-sized circuits. The reconstruction of neuronal circuits between, for example, 100 input axons and 100 postsynaptic neurons in such a dataset from layer 4 of cerebral cortex (Figure 1B) would consume about 200,000–500,000 work hours, amounting to $2–$5 million resource investment, which makes such analysis prohibitive in most settings. It is therefore essential to develop automated classifiers for large-scale 3D-EM data that have been stained for all relevant structures, including synaptic vesicles and post-synaptic densities, and to integrate such automated classifiers into a reconstruction workflow that provides full-neuron volume reconstructions at a tolerable investment of manual labor.

To allow the analysis of large-scale EM data, several software tools have been developed that either focus on the fully manual annotation of neurites and synapses (KNOSSOS, Helmstaedter et al., 2011; TrakEM2, Cardona et al., 2012; CATMAID, Saalfeld et al., 2009) or provide a combination of automated analysis and proof-reading capabilities (rhoANA, Kaynig et al., 2015; ilastik, Sommer et al., 2011). While the fully manual tools cannot relieve

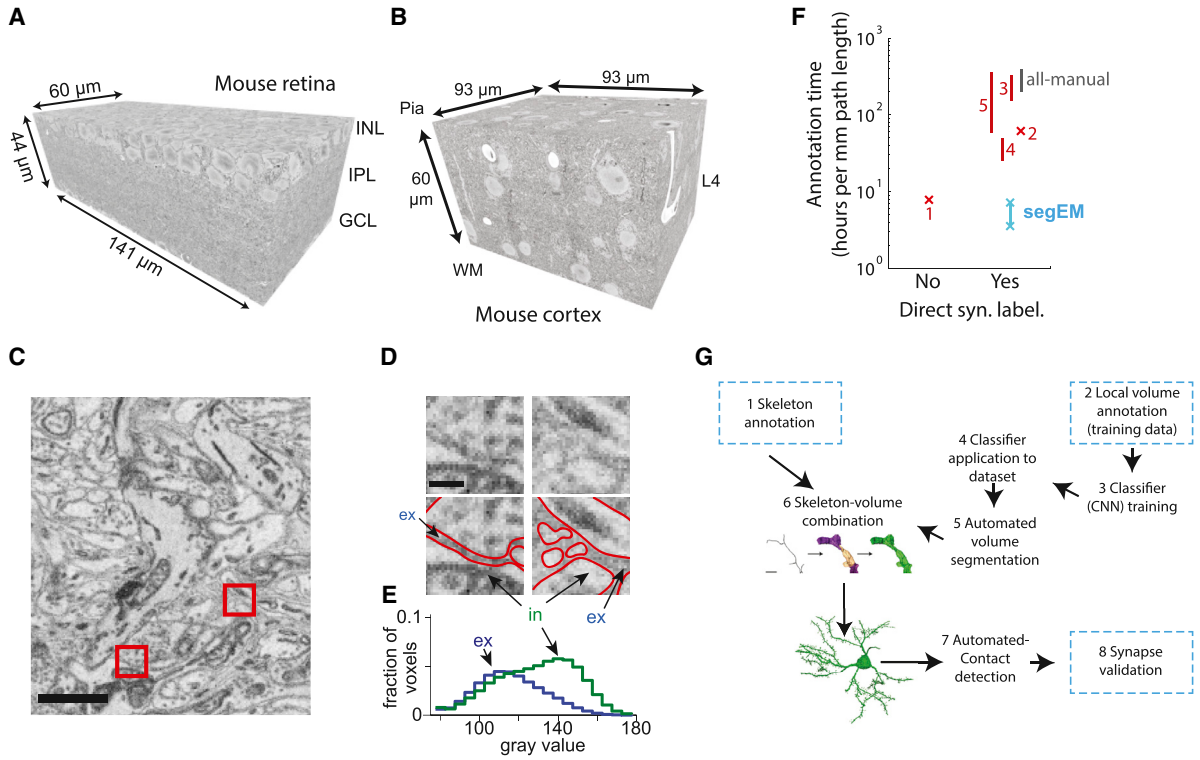

Figure 1. SegEM-Based Connectomic Analysis of Fully Stained 3D-EM Datasets

(A) Dataset boundaries of a 44 µm × 60 µm × 141 µm sized stack acquired by SBEM in mouse retina (dataset ek563; Briggman et al., 2011) extending from the ganglion cell layer (GCL) via the inner plexiform layer (IPL) to parts of the inner nuclear layer (INL).

(B) 93 µm × 60 µm × 93 µm sized SBEM dataset from mouse somatosensory cortex layer 4 (dataset 2012-09-28_ex145_07x2; K.M.B. and M.H., unpublished data). Orientation with respect to pial surface (Pia) and white matter (WM) is indicated.

(C) Neuropil was "fully stained" contrasting all membranes (retina).

(D) Magnified excerpts from (C) illustrate difficulty of distinguishing plasma membranes from vesicular and mitochondrial membranes. Outlines of plasma membrane boundaries from human annotation separating intracellular (in) and extracellular (ex) space are indicated in red. Excerpts sized (35 vx)^2 each.

(E) Gray value distributions of extra- (blue) and intracellular (green) image voxels are highly overlapping, making automated analysis difficult.

(F) Annotation time estimates per mm neurite path length as a measure of circuit size (see also Figure 6A) shown for all-manual volume annotation, finished large-scale connectomic reconstructions (1: Helmstaedter et al., 2013; 2: Takemura et al., 2013), methods descriptions (3: RhoANA, Kaynig et al., 2015; 4: RhoANA applied to cerebral cortex data, Kasthuri et al., 2015; 5: Jones et al., 2015), and SegEM (reconstruction throughput for retina [top marker] and cortex [bottom marker]). See Supplemental Experimental Procedures for details of calculation.

(G) SegEM flowchart for the reconstruction of neuronal circuits. Blue dashed boxes: steps involving manual annotation. Note that neurite reconstruction (step 1) is by far the most time-consuming step (see also Figure 6A).

the enormous work load of reconstructing even small circuits, the computer-supported tools require semi-automated manual annotation or proof reading. The required residual proof-reading efforts are substantial, amounting to about 30–300 work hours per mm path length (for a single annotator, Figure 1F). This is up to 10 times faster than fully manual annotation, but still an enormous burden for circuit reconstruction for most laboratories.

If automated classifiers for fully stained 3D-EM data were available that could be directly combined with efficient skeleton reconstruction, an additional reconstruction efficiency gain of about 10-fold would be possible, which can propel the field into a realistic regime of reconstructing local neuronal circuits sized on the scale of 0.1–1 m integrated path length (Figures 1F and 6A).

Here, we report the development of such automated machine-learning-based classifiers and segmentation procedures that operate on fully stained synapse-labeled large-scale volume EM data. We used convolutional neural network (CNN) classifiers since they are known to perform well in settings with little prior algorithmic knowledge about the classification task and since they profit directly from increased training set size. We applied semi-automated classifier selection routines that enable classifier training with little prior knowledge in a setting targeted to large-scale EM data applications.

Our classifiers achieve a volume segmentation quality that readily provides full-volume reconstructions when combined with skeleton-annotated neuron reconstructions, at an effective resource consumption of 4–8 hr/mm path length per annotator. We exemplify efficient circuit reconstruction for the bipolar cell inputs to one amacrine cell in an EM dataset from mouse retina, and local circuit reconstructions of spiny stellate cells and innervating axons in a dataset from mouse somatosensory cortex

layer 4. From these example reconstructions, we can predict that even circuits between hundreds of neurons are reconstructable using SegEM at resource investments that are realistic in most laboratories (Figure 6A).

SegEM (Data S8; segem.io) integrates all tools required to perform circuit analysis in these datasets and the toolset for efficiently analyzing novel 3D-EM datasets of neuronal circuits. In addition, SegEM comprises all training data and test metrics (Data S8, segem.io) to establish a large-scale 3D segmentation challenge for machine learning in connectomics.

In the following, the SegEM workflow and its application to two large 3D-EM datasets from mouse retina and cortex is described. Then, circuit reconstruction examples are presented for these two datasets, followed by general analysis guidelines for the application of SegEM to 3D-EM datasets. Finally, segmentation error rates are reported and discussed.

RESULTS

We used SegEM to analyze two large 3D-EM datasets in which the neuronal tissue had been fully stained to resolve neurites and synapses. Both datasets were imaged using serial blockface scanning electron microscopy (SBEM; Denk and Horstmann, 2004): one from mouse retina IPL (ek563; Briggman et al., 2011; Helmstaedter et al., 2013; voxel size: 12 × 12 × 25 nm^3, dataset size: 44 × 60 × 141 μm^3; Figure 1A) and one from mouse S1 neocortex (2012-09-28_ex145_07x2, K.M.B. and M.H., unpublished data; voxel size: 11.24 × 11.24 × 28 nm^3, dataset size: 93 × 60 × 93 μm^3; Figure 1B).

SegEM Workflow

The SegEM tools are aimed at replacing the labor-intensive manual volume segmentation of fully stained 3D-EM data by an efficient combination of skeleton annotation with automated volume segmentations (Figure 1F). The first branch of the workflow (Figure 1G) is the skeletonizing of axons and dendrites of interest using efficient 3D skeletonization software (such as KNOSSOS, Helmstaedter et al., 2011; www.knossostool.org, step 1 in Figure 1G). This is the key step that consumes manual labor. The consumption is about 4–8 hr per mm neurite path length for a single skeleton annotator, thus 25- to 100-fold faster than fully manual volume annotation (Helmstaedter et al., 2011; see Figure 1F).

To make use of this skeletonized data for volume segmentation, contact detection, and further circuit analysis, the second branch of the SegEM workflow is required (steps 2–5 in Figure 1G). First, small example volumes need to be manually labeled for training the automated classifiers (step 2 in Figure 1G). This step consumes about 1,000–2,000 work hours, but only once per dataset, which is only a fraction of the work hour investment required for neurite skeletonization in most settings. Next, the automated image classifiers have to be trained (step 3 in Figure 1G). Then, the trained classifier is applied to the 3D-EM image data, followed by an automated segmentation step (steps 4 and 5 in Figure 1G). The result of these SegEM steps is a piecewise volume segmentation, which is then combined with the skeleton reconstructions to yield full-volume reconstructions of neurons.

These full-neuron reconstructions are then fed into the SegEM-contact detection routines (step 7 in Figure 1G), which output contact area matrices and pointers to potential synapse locations. These can be used to finally validate synapses in the circuit of interest (step 8 in Figure 1G).

The SegEM workflow is rather modular: for example, the classifier training (steps 2 and 3 in Figure 1G) can be omitted if an existing classifier is to be applied to the data (for example, the classifiers developed here can be directly applied to novel 3D-EM datasets, see Data S2).

The logic of the SegEM workflow is described in the following as it was applied to the two large example datasets from retina (Figure 1A) and cortex (Figure 1B). A detailed step-by-step instruction for the application of SegEM can be found in the Supplemental Experimental Procedures.

Training Data

Machine-learning-based image analysis requires substantial amounts of labeled training data. For the retina dataset, we used manually segmented image volumes in the surroundings of ribbon synapses, which consisted of bipolar, amacrine, and ganglion cell neurites (Figure S1A) as training data. These annotations had been used in a previous study for the calibration of synapse probability based on neurite-to-neurite contact area (Helmstaedter et al., 2013). The 215 manual segmentations were generated by contouring of neurites by 33 trained undergraduate students using a custom-written software (KLEE, implemented in MATLAB; other available tools are, for example, ilastik, Sommer et al., 2011; CATMAID, Saalfeld et al., 2009; VAST, Kasthuri et al., 2015) and consumed a total of about 1,500 hr annotation time (thus, about $15,000 salary expense, which is on the scale of investments in chemicals for 3D-EM projects). This volume segmentation was split into a training and test set ($\sim 5 \times 10^8$ and $\sim 1 \times 10^6$ voxels, respectively).

For the cortex dataset, we volume-annotated locally dense data cubes of size $(100 \text{ voxel})^3$. We found training set size to be of critical importance. We initially trained with 10 of such volumes sampled evenly from the entire dataset (thus, about 10^7 training samples). However, classification results were much poorer than when training on the final 279 volumes (2.8×10^8 samples). Thus, the factor 30 in annotation cost and effort was likely required (total of 2,000 hr manual labeling time).

Classifier Training

We trained convolutional neural network (CNN) classifiers to convert the raw 3D-EM image data into 3D maps of intracellular continuity between face-to-face adjacent voxels (one map for each cardinal direction; Figures S1A–S1C; Jain et al., 2010b; Turaga et al., 2010). For the cortex data, we found it sufficient to train the CNNs to output a single 3D map, thus representing the probability of each image voxel to be intra- or extracellular. Training the CNN to output one map is the default setting in SegEM but may be switched to three output maps for lower-resolution and lower-contrast image datasets.

To find optimal learning rates and architectural parameters for our CNN training, we implemented a simple hyperparameter search. This relieved us of the need to hand-design training parameters, and it made direct use of the availability of GPU

compute clusters without the need to explicitly parallelize the training computations. Instead, we used a GPU cluster for parallel training of different CNN architectures. This was essential for the success of the classifiers. While more sophisticated hyperparameter search algorithms have been proposed (Bergstra et al., 2013; Snoek et al., 2012), the simple evolutionary approach as described in the following was robust and successful and was aimed at making CNN training for connectomics widely applicable. When applying SegEM to a novel 3D-EM dataset, we recommend to first test our best-performing classifiers (Data S2) on the novel dataset. This is especially recommended if the high-resolution ultrastructure is comparable to cortex tissue, such as is the case for hippocampus, thalamus, and many other subcortical regions, and the imaging resolution is similar. Then, SegEM can be used to refine the classifier by training it using the best training parameters from our search (reported in Table 1). For novel datasets that have different ultrastructure and geometry, we recommend using the entire SegEM hyperparameter search procedure as described in the following.

The cost function for CNN classifier training was squared voxel-based error between the adjacency maps and the graded classifier output; the cost function was masked for unlabeled image regions. While this voxel-based metric is known to be conceptually inferior to segmentation-based metrics, we profited from its more efficient implementation yielding training times on the scale of days, not months (see below). Then we randomly selected training batches out of that masked volume. These training batches had to fulfil the criteria that at least 1/3 of voxels in the batch were labeled and that at least 1/3 of these labeled voxels were from each of the two label classes. These criteria were necessary to reduce training time and to avoid activity map saturation (such saturation could occur because the two label classes were highly unbalanced in regions containing large dendrites or somata).

For classifier training, two modifications of the training data proved crucial for classifier convergence and performance. (1) Since the detection of neurite borders is essential for avoiding incorrect neurite mergers, we reduced the bias toward intracellular regions by enlarging neurite-to-neurite walls in the labeled data. This was implemented by eroding the volume objects in the training data with a spherical structuring element of size 3^3 voxel, and thus a broadening of neurite-to-neurite walls by about 20–80 nm (Figures S1E and S1F). (2) We excluded those training volumes that yielded substantial classification performance decreases (examples in Figure S1J; 11 of 215 training volumes were excluded, Table S1; see Supplemental Experimental Procedures for a quantitative guideline on which training data to exclude). This is an unusual approach in machine learning, but proved critical for classifier convergence. The training volumes that were excluded mostly contained significant errors in the manual annotations—caused by insufficient annotator attention or artifacts in the image data. We recommend this training data co-optimization step in settings where intense curation of training data is not feasible because it would consume substantial additional annotation time.

The CNNs were implemented on a GPU compute cluster, which accelerated the parallel screening of CNN architectures. CNN architectures and training procedures comprised a total of 17 varied parameters (most relevant: number of hidden layers, number of feature maps per hidden layer, filter size, learning rates for weights and biases and their respective learning rate decay, and batch size; see Table 1 and Figure S1D). Exhaustive parameter screening was not possible, since training convergence took at least about 2 days on one GPU card. We therefore first performed a qualitative manual network architecture selection. After an initial screening of about 200 network parameter sets, we started a semi-automated network selection procedure (Figures 2A–2C; shown for the retina dataset) with the best parameter set from the initial broad search (these parameters ranges are reported in Table 1; see Figure S2B for typical examples of rejected classifier outputs).

When we applied SegEM to the cortex dataset, we made a few improvements to the classifier search procedure, which we recommend for applying SegEM to novel datasets. First, we made use of a larger GPU cluster (28 GPUs), but CNNs were trained for only 1 day per iteration (Figure S2F). Then, the best half of the CNNs were selected based on averaged training error over the most recent 200 batch iterations, and the parameters for these were varied again (see Table 1 for best-performing network parameters and Supplemental Experimental Procedures for details). This procedure was iterated.

When applying the SegEM classifier search to a novel EM dataset, we recommend first trying to use the best-performing SegEM classifiers directly. When we applied the retina-trained classifier to cortex data, we obtained rather poor results (Figures S2D and S2E) and therefore applied the entire SegEM parameter search to the cortex dataset again. However, for datasets resembling cortex data or retina data in ultrastructure, the transfer of classifiers between datasets may be feasible.

After testing the existing classifiers on novel datasets, we recommend then performing SegEM hyperparameter search by initializing with our most successful hyperparameters (Table 1). This can provide a successful classifier within only about 10 such search iterations.

After classifier optimization, the best 2–5 output networks were used as input to the segmentation optimization (see below).

Automated Volume Segmentation

The three adjacency maps obtained as classifier output for the retina dataset or the one output map obtained for the cortex dataset were then used in a watershed-based segmentation procedure to generate a space-filling volume segmentation (Figure 3). The segmentation procedure consisted of the following steps (Figure 3). (1) Morphological opening and closing by reconstruction using a spherical structuring element of radius r_{se}. This was intended to suppress mergers in the segmentation by removing small connecting bridges between otherwise unconnected image regions. This step provided improvements for the retina, but not the cortex data (see Table 2). (2) Generation of markers for watershed using a threshold or local minimum operation on the classifier output (parameters: θ_{mg} and θ_{hm}), followed by connected components. In contrast to the retina classifier, for the cortex data, the h_{min} operation for marker generation performed better than thresholding. (3) Exclude markers smaller than θ_{ms} voxels. (4) Marker-based watershed on the result of

Table 1. Search Ranges and Best Parameters for SegEM-CNN Training

	Retina Initial Screening (ranges)		Retina CNN Selection Iteration 1		Retina CNN Selection Iteration 2			Retina CNN Selection Iteration 3			Cortex Automated CNN Selection		
	Min	Max	CNN1,j (ranges)	CNN1,1	CNN2,j (ranges)	CNN2,1	CNN2,2	CNN3,j (ranges)	CNN3,1	CNN3,2 (best class)	CNN (ranges)	$20130516T204040_{8,3}$	$20131012T234219_{41,3}$
n_{HL}	1	8	4	4	4	4	4	4	4	4	3–4	4	4
n_{fm}	5	30	10	10	10	10	10	10	10	10	[15,15,10,10], [10,10,10,10], [10 10 10]	[10,10,10,10]	[10,10,10,10]
n_{om}	1	3	3	3	3	3	3	3	3	3	1	1	1
$S_{Fx/y}$, S_{Fz}	5,5–20,10[a]		8,4	8,4	8,4	8,4	8,4	8,4	8,4	8,4	5,3–21,11	11,5	11,5
d_b	1	5	2	2	2	2	2	2	2	2	4	10	4+
$S_{Bxy,z}$	1,1–100,50[b]		12,6	12,6	12,6	12,6	12,6	4,2–100,50[c]	12,6	4,2	100,100	100,100	100,100
η_{w0}	10^{-8}	10^{3}	10^{-5}–10^{-1}	10^{-4}	10^{-6}–10^{-3}	10^{-6}	10^{-6}	10^{-7}–1	10^{-7}	10^{-6}	10^{-7}–10^{-13}	–	–
η_{b0}	10^{-6}	100	10-100	10	0.01-10	0.1	10	0.01-1	0.1	0.1	10^{-7}–10^{-13}	–	–
τ_w, τ_b	10^{7}–9×10^{12} (l,e)		5×10^{8}(l,e)	5×10^{8}(e)	5×10^{8}(l)	5×10^{8}(l)	5×10^{8}(l)	10^{7}–5×10^{8}(l)	5×10^{8}(l)	5×10^{8}(l)	(l)	–	–
$N_{it,cum}$ ($\times 10^6$)	$5^{*}10^{-4}$	3.7665	1.4355–1.75	1.7475	2.806–2.8095	2.8075	2.8085	2.8295–3.3855	3.2055	3.3855	3.5–7 $\times 10^{-4}$ / it.	–	–

Network $CNN_{3,2}$ was best performing on retina data and used for Figures 3 and 4. Network $20130516T204040_{8,3}$ was best performing on cortex data and used for Figures 3 and 5. n_{HL}: number of hidden layers; n_{fm}: number of feature maps per hidden layer; n_{om}: number of output maps; $S_{Fx/y}$, S_{Fz}: filter size in x/y and z direction in vx, respectively; d_b: mask border size in vx; $S_{Bxy,z}$: batch size in x/y and z direction, respectively; η_{b0}: weight learning rate initialization; η_{b0}: bias learning rate initialization; τ_w, τ_b: decay constants for weight and bias learning rates; $N_{it,cum}$: cumulative number of training iterations (number of batch learning iterations). l,e: linear and exponential learning rate decay, respectively.

[a] 8,4; 20,10; 12,6; 10,5; 5,5; 7,7.

[b] {[5,5,5], [10,10,10], [2,2,2], [20,20,20], [1,1,1], [30,30,30], [12,12,6], [8,8,4], [4,4,2], [20,20,10], [40,40,20], [100,100,50]}.

[c] {[12,12,6],[40 40 20],[20 20 10],[4 4 2],[100,100,50]}.

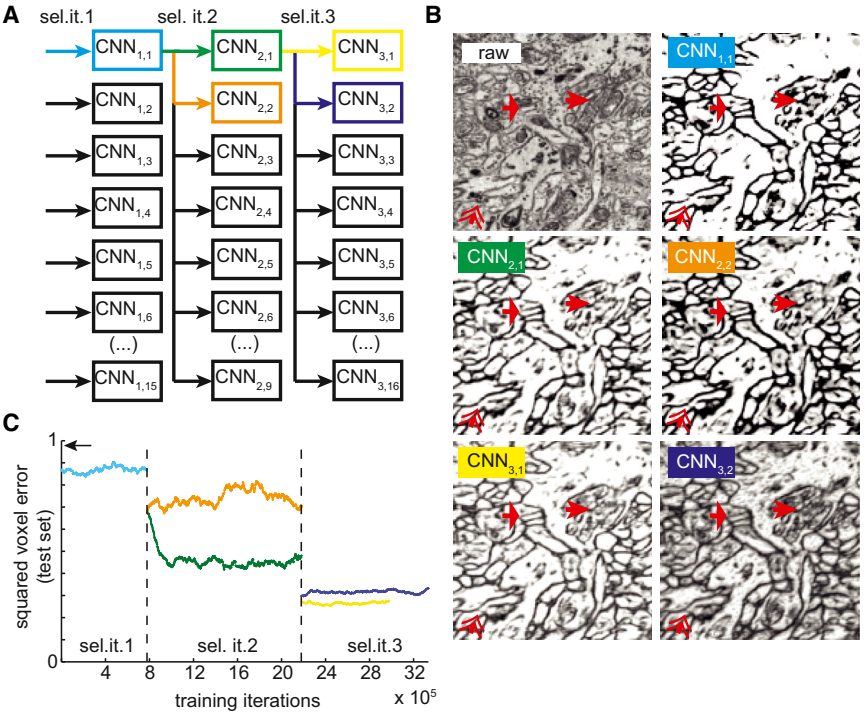

Figure 2. Classifier Training and Semi-automated CNN Hyperparameter Selection Procedure

(A) Sketch of CNN variation and selection procedure (Sel.it: selection iteration). See Table 1 for CNN and training parameters.

(B) Example classification results (average affinity maps) for classifiers developed via procedures in (A). Arrows indicate typical classification challenges and their improvement over selection iterations (mitochondria close to plasma membrane, broad arrow; intracellular clustered large mitochondria, narrow arrow; intracellular staining precipitate close to plasma membrane and proximity to small diameter neurites, double arrow).

(C) Normalized test error development during selection procedure shown in (A). Colors match CNNs in (A) and (B). Error was calculated every 500 batch iterations and averaged over 101 of those errors using a sliding window. Arrow: initial error after random initialization of CNN.

segmentation step 1 (see Supplemental Experimental Procedures for details and Data S8 for all required routines).

Thus, the segmentation procedure had a total of three relevant parameters ($[r_{se}, \theta_{mg}, \theta_{ms}]$ and $[r_{se}, \theta_{hm}, \theta_{ms}]$, respectively; Table 2). We optimized these segmentation parameters by an iterated parameter grid search (see Table 2 for parameter ranges and best-performing parameters). Segmentations were judged by the average inter-error distance (i.e., the combined split-merger inter-error distance, see below), the average segmentation object size, merger suppression, and avoidance of longitudinal neurite splits (see Experimental Procedures for details).

Combining Skeletons and Automated Segmentations

We then used our automated volume segmentations for whole-neuron reconstruction (step 6 in Figure 1F). We skeletonized the axons of 276 bipolar cells and the dendritic tree of one amacrine cell from the mouse retina dataset ek563 (Figure 4). 37 reconstructions were identified to be type 5 cone bipolar cell axons based on their lamination in the inner plexiform layer (dendrites could not be reconstructed since ek563 only spanned from the GCL to the beginning of the INL, see Figure 1A; each axon was traced by one experienced student). Of these skeletons, 10 were randomly chosen (Figures 4A and 4B), proofread by one additional tracer, and all volume segmentation objects overlapping with at least one skeleton node were collected (inset in Figure 1F) for each bipolar cell axon. Skeletons were traced at an average consumption of 7.2 ± 4.1 hr/mm, which is similar to the speed achieved in the surface-enhanced contrast sample from mouse retina (e2006; Helmstaedter et al., 2011, 2013; see Figure 1F for comparisons).

The resulting whole-cell volume reconstructions (Figures 4A and 4B) contained 3 obvious merge errors in 10 bipolar cells with a total path length of 4.13 mm, i.e., 1.38 mm distance between mergers (see below for a more detailed quantification of error rates). This was comparable to the error estimates in the retinal circuit reconstruction based on special cell-membrane enhanced staining.

Automated Contact Detection and Synaptic Circuits in Mouse Retina

We then used the volume reconstructions of type 5, 6, and 7 bipolar cells (CBCs 5, 6, 7; n = 37, 37, 22), rod bipolar cells (RBCs, n = 117), yet unclassified cells (n = 23), and a peculiar amacrine cell with wide-field ramifications (Figure 4C) to measure the synaptic bipolar-cell innervation profile of this amacrine cell. A total of 243 contacts between these neurons were automatically detected (step 6 in Figure 1F). Since we had previously shown that contact area predicts synaptic contact in bipolar-to-ganglion cell and amacrine cell synapses in mouse retina (Helmstaedter et al., 2013), we only validated three of these automatically detected contacts (of which two were ribbon synapses; Figure 4D). Synapse validation was performed using either KNOSSOS (Helmstaedter et al., 2011; knossostool.org) or webKNOSSOS (K.M.B. M.B., T. Bocklisch, and M.H., unpublished data; webknossos.brain.mpg.de); SegEM outputs skeleton files that point directly to the contact locations of interest, thus facilitating synapse validation for the human annotator (step 7 in Figure 1F). Synapse validation consumed about 3–5 min inspection time per contact. Figures 4C and 4D show the resulting innervation pattern of the investigated amacrine cell, providing evidence for a spatially segregated CBC innervation (Figure 4D; total annotation time 280 hr: 1 hr for each bipolar cell axon, 3 hr for AC, and 3 min per inspected synapse).

Synaptic Circuits in Cortical Layer 4

To perform local circuit analysis in cortex, we skeleton-reconstructed the dendrites of 4 spiny stellate neurons (52 hr of

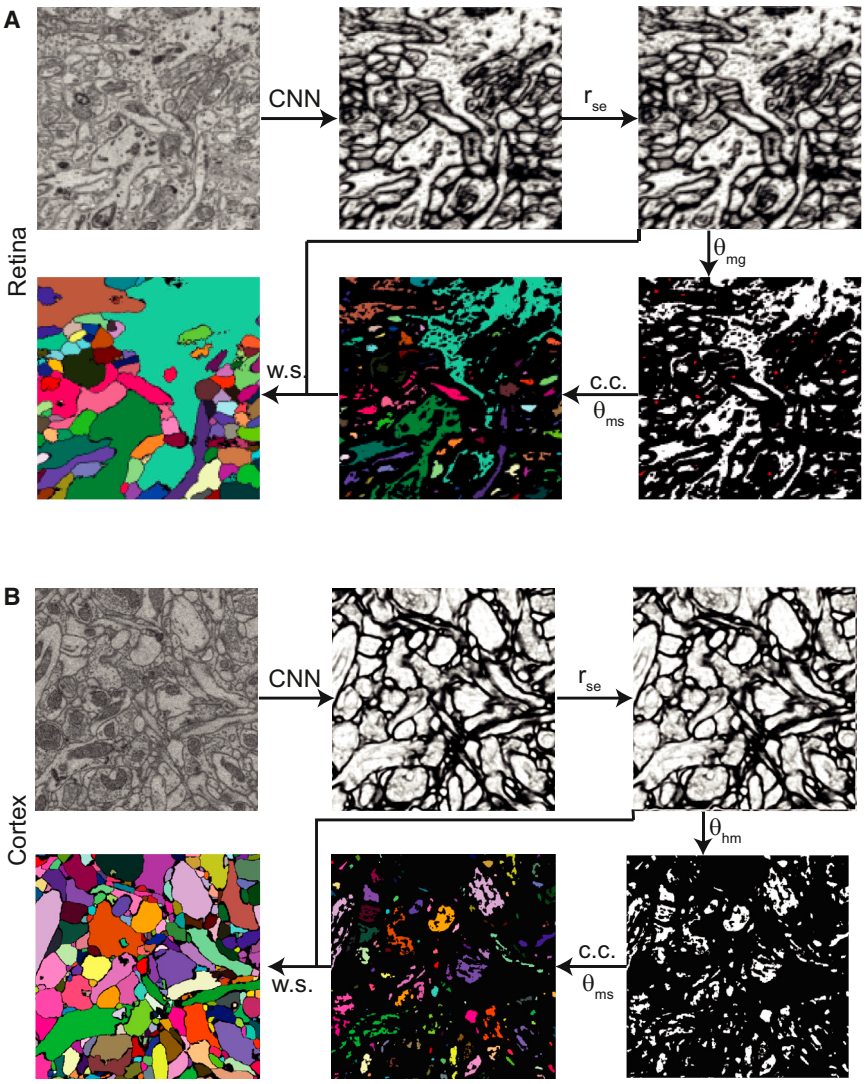

Figure 3. Segmentation Procedure Shown for Retina and Cortex Data

(A) Raw data forward-passed through a CNN classifier (retina; classifier $CNN_{3,2}$, see Table 1) yielding three voxel-to-voxel affinity maps (only x-output [o_x] shown, see Figures S1A and S1C), followed by image inversion, morphological reconstruction with radial structuring element of radius r_{se} (shown is result for r_{se} = 1, see Table 2), map binarization at threshold θ_{mg} (shown θ_{mg} = 0.3), application of connected component (c.c.) segmentation (26-surround) for marker generation, exclusion of markers with volume < θ_{ms} *voxels* (θ_{ms} = 150 shown), and marker-based watershed (w.s.) on inverted morphologically reconstructed affinity maps using the shown markers.

(B) Classification and segmentation for cortex dataset; steps are as for retina data (A) except for local minimum operation with minimal depth parameter θ_{hm} = 0.39 (r_{se} = 0, θ_{ms} = 50). Note that maps were inverted before watershed marker generation but are shown in non-inverted form here. Image sizes: (6.2 μm)^2 (A) and (5.8 μm)^2 (B).

tracing), combined them with the SegEM-generated volume segmentations (Figures 5A and 5B), and then identified the axons providing synaptic input to three directly adjacent spines on one of the neurons' dendrites (cell 1, primary dendrite 1; Figure 5C). We then reconstructed the three corresponding presynaptic axons and used SegEM to compute the local high-resolution contact matrix (Figure 5D) reporting the size of all contacts between these three axons and all of the spiny neuron dendrites (total of 33 contacts). For neuron-to-neuron contacts in cortex, a direct prediction of the existence of a chemical synapse based on the size of an individual neuron-to-neuron contact is not possible. We therefore inspected each of the detected contacts and determined the local synaptic connectome (Figure 5D, bottom; 8 of 33 contacts were confirmed to be synaptic; examples in Figure 5E). Synapses were validated by visual inspection using direct links to contact locations in our web-based annotation tool, webKNOSSOS (K.M.B., M.B., T. Bocklisch, and M.H., unpublished data; webknossos.brain.mpg.de), which consumed less than 1 min annotation time per suspected synaptic contact. The resulting local high-resolution synaptic connectome (Figure 5D, bottom) thus required a total manual annotation effort of 53 hr; skeleton tracing was possible at only 3.7 hr per mm path length, which we attribute to the better data quality in our cortex dataset and the skill level of the annotators. We finally searched all pairs of axons and dendrites for additional synaptic contacts to estimate the rate of missed synapses by SegEM. We found no additional synaptic contact (0 missed contacts for 33 detected contacts), which provides an additional validation of the quality of the SegEM volume segmentations.

Circuit Reconstruction Time Estimates

How realistic are synaptic circuit reconstructions at the measured throughput using SegEM, based on the small example circuits reported for the retina and cortex? Take, for example, the innervation of excitatory neurons in cortical layer 4 by thalamocortical axons (Figure 6A). While single synaptic innervations have been studied, the pattern of local target selectivity (or the lack thereof) of the thalamocortical innervation is not known. Therefore, a study to reconstruct the connectome between say 10 thalamocortical axons and 30 postsynaptic L4 neurons would be highly relevant. What investment would such a circuit reconstruction require using SegEM? Each of the involved neurites has a couple of millimeters of path length; let's assume that a redundancy of 4 tracings per dendrite and 2 per axon is about sufficient (tracing redundancy can be scaled to the required circuit accuracy, which depends

Table 2. Best-Performing Segmentation Parameters for Retina and Cortex Segmentations

	Segmentation Parameters Retina[a]				Segmentation Parameters Cortex[b]			
	Parameter ranges for split-merger metrics (Figure 7F)	Optimum segmentation (Figure 7F) θ_n = 1;2	Whole-cell segmentations (Figure 4)	All else shown	Parameter ranges for split-merger metrics (Figure 7F)	Optimum segmentation (Figure 7F) θ_n = 1;2	Whole-cell segmentations (Figure 5)	All else shown
r_{se}	{0,1}	1; 1	1	1	0	0; 0	0	0
θ_{mg}	{0.2,0.21,..0.5}	0.5; 0.38	0.31	0.3	–	–	–	–
θ_{ms}	{0,20,40,60, 100,150}	100; 0	20	150	{0,50,100}	50; 0	10	50
θ_{hm}	–	–	–	–	{0.02,0.04,.. 0.7}	0.58; 0.04	0.25	0.39
IED (μm)[c]	–	3.85; 7.91	2.20; 7.04	2.21; 6.24	–	1.58; 4.93	0.93; 4.48	1.26; 3.86
Split (μm)[c]	–	7.71; 9.76	2.37; 7.38	2.56; 6.65	–	2.89; 5.03	0.96; 4.56	1.46; 4.47
Merge (μm)[c]	–	7.71; 41.73	30.26; 151.28	16.36; 110.02	–	3.48; 214.52	28.41; 241.52	9.29; 28.41

r_{se} = radius of spherical structuring element for morphological operations; θ_{mg}: threshold used for marker generation; θ_{hm}: depth parameter of H-Minima operation; θ_{ms}: voxel threshold for watershed marker size. IED: average inter-error distance; split: average distance between splits; merge: average distance between mergers (see Experimental Procedures for calculation).

[a]Using $CNN_{3,2}$ (see Table 1).

[b]Using CNN $20130516T204040_{8,3}$ (Table 1).

[c]Reported for node overlap thresholds θ_n = {1;2}, see Figure 7F.

on the type of circuit that is being studied; see Helmstaedter et al., 2011, 2013), then we end up with about 3,300 work hours for neurite tracing. If distributed between, for example, 15 undergraduate students, who each work about 40 hr per month, such an analysis would take about 5.5 months and consume about $33,000 (assuming about $10 per hour salary expenses). Then, the about 1,000 contacts between the axons and dendrites need to be inspected, which would require about 15–20 work hours, thus only a fraction of the neurite reconstruction effort. Together, such a project is realistic in resource and work

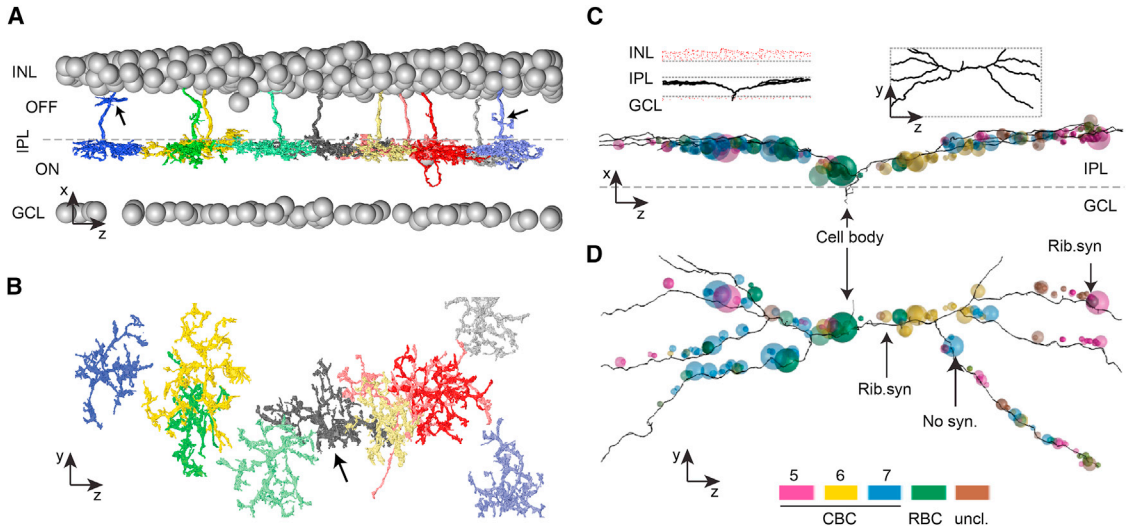

Figure 4. Connectomic Analysis in Fully Stained Retina Data

(A) Volume reconstruction of 10 randomly selected cone bipolar cell type 5 (CBC5) axons in dataset ek563 from mouse retina (see Figure 1) obtained by relabeling the computer-generated segmentation objects (classifier $CNN_{3,2}$, r_{se} = 1, θ_{mg} = 0.31, θ_{ms} = 20, see Tables 1 and 2) according to human-generated skeleton annotations. Tracing time was about 1 hr per bipolar cell axon. Arrows point to obvious merge errors (3 obvious errors, total skeleton path length: 4.13 mm, i.e., inter-merger distance ~ = 1.38 mm). Gray spheres indicate somata.

(B) Same reconstruction as in (A) viewed in the IPL plane.

(C) Automated contact detection of CBC inputs to a displaced amacrine cell (insets in C). Contacts with CBCs of type 5–7, RBCs, and unclassified neurons are shown as spheres; sphere surface proportional to contact size. Since this dataset was fully stained, automated contacts can be visually inspected and existence and directionality of synapses confirmed (Rib. syn.: ribbon synapse CBC→AC confirmed; No syn: accidental contact without evidence for synaptic contact). CBCs were annotated twice, AC once by an expert annotator. Annotation consumption: 7.2 hr/mm path length for a single annotator.

(D) Same reconstruction as in (C) viewed in the IPL plane. All images show the whole dataset with dimensions 141 μm (z), 44 μm (x), 60 μm (y).

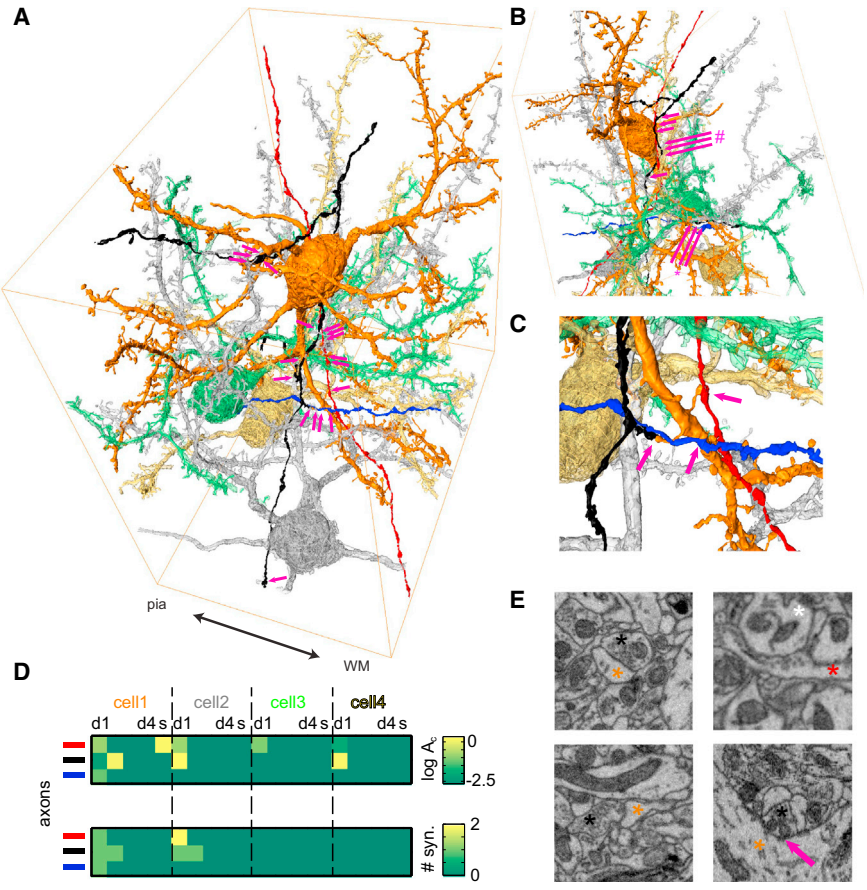

Figure 5. Connectomic Analysis in Fully Stained Cortex Data

(A) Volume reconstruction of 4 spiny neurons (orange, gray, green, yellow, respectively) by combination of skeleton tracing with SegEM classifier (see Figure 4, CNN 20130516T204040$_{8,3}$; see Tables 1 and 2 for classification and segmentation parameters).

(B) Same reconstruction as in (A) viewed from a different angle.

(C) The presynaptic axons from three neighboring spines of one neuron (orange, C) were also volume reconstructed (black, blue, red axon). Arrows in (A), (B), and (C) point to all detected contacts between these axons and spiny neurons (* and #: 5 and 6 contacts, respectively).

(D) The local connectome between the 3 axons (red, black, blue, respectively) and the primary dendrites (d1–d4) and somata (s) of the 4 spiny neurons. Top panel shows the total size of contact areas A_c (see magenta arrows in A–C). Bottom panel shows the result of visual inspection of each contact to determine the presence of chemical synapses.

(E) Examples of synapses and one accidental contact (bottom right). Annotation consumption: 3.7 hr/mm path length for a single annotator.

hour consumption and would provide unprecedented circuit data that currently cannot be obtained with other techniques. Notably, without SegEM, this reconstruction would have consumed about 10 times more work hours and resources, which would make it not feasible in most settings.

Similarly, one can project the required investments for other circuit reconstructions of similar or larger scale in cortex and retina (for example, the connectome between 300 bipolar and 300 small-field amacrine cells would consume about 6,500 hr total; see Supplemental Experimental Procedures and Figure 6A for details of the calculation and further examples).

Analysis Guideline for Novel Large-Volume EM Datasets

The analysis steps needed to perform circuit reconstruction in large-scale 3D-EM datasets (Figure 6B) can thus be summarized as a methodological guideline based on the SegEM analysis package. As a prototypical example, assume a SBEM dataset from mouse barrel cortex covering one layer 4 barrel from one cortical column at the required resolution (450 μm^3 volume, 12 × 12 × 25 nm^3 voxel size, i.e., 23 TB image data, about 60 days of experiment for 16,000 successive image layers). Manual analysis for training label generation can be done in parallel to data acquisition (Figure 6B), such that the CNN selection procedures are finished when data acquisition is. Dataset classification and segmentation will then approximately equal the imaging time (assuming a compute cluster of ~60 GPUs and ~75 CPUs). Annotation investment (for training data acquisition) would be about 2,000 work hours distributed to about 30 annotators, if de novo training is required. This step can be omitted if the existing SegEM classifiers are already successful without additional parameter search (see above). Circuit reconstruction by skeleton reconstruction is then possible at a consumption of about 4–8 hr/mm path length. Within 2 months (as proposed in Figure 6B), a team of 30 annotators working at 40 hr/month can provide 2,400 work hours—sufficient for reconstructing the circuits between 30 TC axons and 30 spiny stellate neurons, or 100 bipolar cells and 30 amacrine cells (Figure 6A; 80–180 mm circuit size) at moderate cost. Of course, this analysis phase can be extended for larger circuits. All processing code required for such an analysis setup is contained in SegEM (segem.io, Data S8).

Segmentation Metrics

The main contribution of SegEM is to enable efficient circuit reconstruction in fully stained 3D-EM datasets by combination of skeleton reconstructions with automated volume segmentations. For this, it was crucial to make 3D-EM classifiers sufficiently reliable such that they can be combined into full-neuron reconstructions by assembly along the center line of neurites as shown above.

In addition to this concrete application, it was however desirable to compare the SegEM volume segmentations to the performance of other segmentation benchmarks (Figure 7). For this, we computed voxel, rand, and warping error using the routines made available in the ISBI EM challenge (see Supplemental Experimental Procedures; Figure 7H). SegEM results according

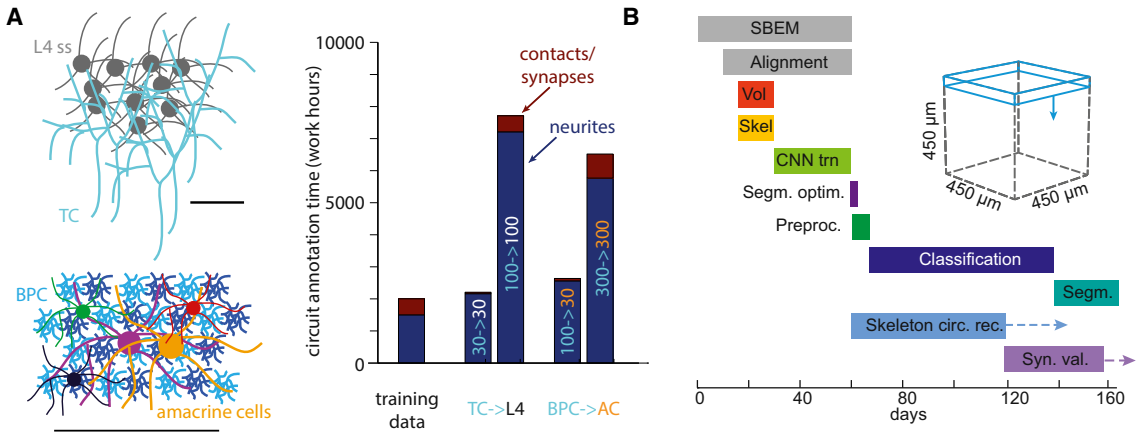

Figure 6. SegEM Circuit Reconstruction Time Estimates

(A) Projected circuit reconstruction cost measured in required work hour investment for two example circuits. Top left, circuit within layer 4 of cerebral cortex comprising thalamocortical axons (TC) and postsynaptic spiny neurons (L4 ss). Reconstructing a circuit of 30 TC axons and 30 spiny neurons would consume about 2,160 work hours for neurite tracing (at 2- to 4-fold tracing redundancy) and about 50 hr for synapse validation. Even a circuit of 100 axons and 100 dendrites is realistic (total effort: 7,700 work hours, i.e., less than $80,000 salary investment). Bottom right, examples for circuits between bipolar neurons and amacrine cells in mouse retina. Note that for novel dataset, an additional volume labeling investment of 1,500–2,000 hr may be required. See Supplemental Experimental Procedures for details of calculations.

(B) SegEM analysis procedure for large-scale 3D-EM datasets, exemplified by the analysis timeline for a SBEM dataset from mouse somatosensory cortex, where a volume of (450 µm)^3 contains layer 4 of one cortical column and the required voxel size is about (12 × 12 × 25) nm^3 (dataset size ~25 TB). All numbers are approximate. SBEM: SBEM imaging (60 days) assumes an effective imaging speed of ~5 × 10^6 vx/s (including cutting, motor movements, etc.). Vol.: volume training data annotation for CNN training; ~200 dense volume labelings of (100 vx)^3 consuming ~1,600 work hours (wh) distributed between 30–40 students over 2 weeks. Skel.: skeleton training data annotation for segmentation optimization; 5 students, ~300 work hours (wh) over 2 weeks. CNN train.: classifier training/selection procedure (see Figure 4) over approximately 28 days (28 selection iterations as in Figure 4B). Segm. optim.: segmentation optimization (Table 2) using skeleton training data; about 2–5 days of computation. Alignment: alignment of SBEM images and conversion to Knossos data format (3D). Preproc.: dataset preprocessing (after entire dataset acquired); gray scale equalization; blood vessel masking, nucleus masking, potential correction of tiling effects (about 7 days of computation). Classification: dataset classification (forward pass of best CNN from selection procedure): 77 days on 56-GPU cluster. Segm.: local dataset segmentation in (128 vx)^3 cubes with 10 vx margin on all sides on a 70-80-CPU cluster. Skeleton Circ. Rec: skeleton-based circuit reconstruction (at 4–8 hr per mm path length). Syn. Val.: synapse validation (at about 0.5–1 min per contact).

to the ISBI challenge metrics were among the best with respect to the warping error (aimed at directly optimizing topology of neurites) and rand error (Figure 7I). This is notable since the image data used in our (larger-volume) SegEM challenge was at resolution about 10-fold lower than that in the ISBI small-volume challenge (see Figure 7H, left panels).

However, such comparisons should be treated with caution, since the ISBI challenge is evaluated only on 2D-EM data slices (Figure 7H), and voxel based metrics can be highly biased by the concrete shape and size of local objects contained in the test volume.

We therefore used an additional metric to measure SegEM segmentation error rates: we compared the volume segmentations to manually annotated skeleton (center-line) neurite reconstructions (Figures 7A–7G). This metric (skeleton-based split-merger metric; Figures 7C–7F) evaluates neurite continuity along the main axis of the neurites. Since the center line path length of neurons is on the scale of millimeters, such a metric therefore evaluates the most challenging aspect of neurite reconstructions: volume pieces as small as 50 nm in diameter, but at millimeters path distance, have to be correctly assigned to each other. The skeleton-based split-merger metric measures the rate of splits, i.e., breaks in the volume segmentation along the neurite center line axis (Figure 7E), and the rate of mergers, i.e., volume segmentation objects that link together two separate neurite skeletons (Figure 7D; see Experimental Procedures for details of the calculation or evaluateSeg.m in Data S8 or SegEM package on github). Both measures are expressed as their inverse, i.e., the average distance between splits d_s and the average distance between mergers d_m.

Figure 7F shows the split-merger distances in the retina dataset for 372 combinations of segmentation parameters using $CNN_{3,2}$ and for 99 combinations of segmentation parameters using CNN $20130516T204040_{8,3}$ for cortex. Error distances are reported for two node thresholds θ_n indicating the minimum number of skeleton nodes that had to overlap with a segmentation object to be considered a match; higher values of θ_n result in more resistance to noise from the imprecise manual placement of skeleton nodes.

The optimum inter-error distance $1/(1/d_s + 1/d_m)$ for $\theta_n = 1,2$ was 3.9 µm, 7.9 µm, respectively (retina), and 1.6 µm, 4.9 µm (cortex; see Table 2 for the corresponding optimal segmentation parameter sets). This optimum inter-error distance assumes splits and mergers in the automated segmentation to be of equally detrimental effect.

However, the key goal of our automated segmentation was to enable the efficient combination of skeleton reconstructions and volume segmentations for whole-neuron reconstructions. For this application, the optimal segmentation is one with maximal merger distances under the constraint that objects

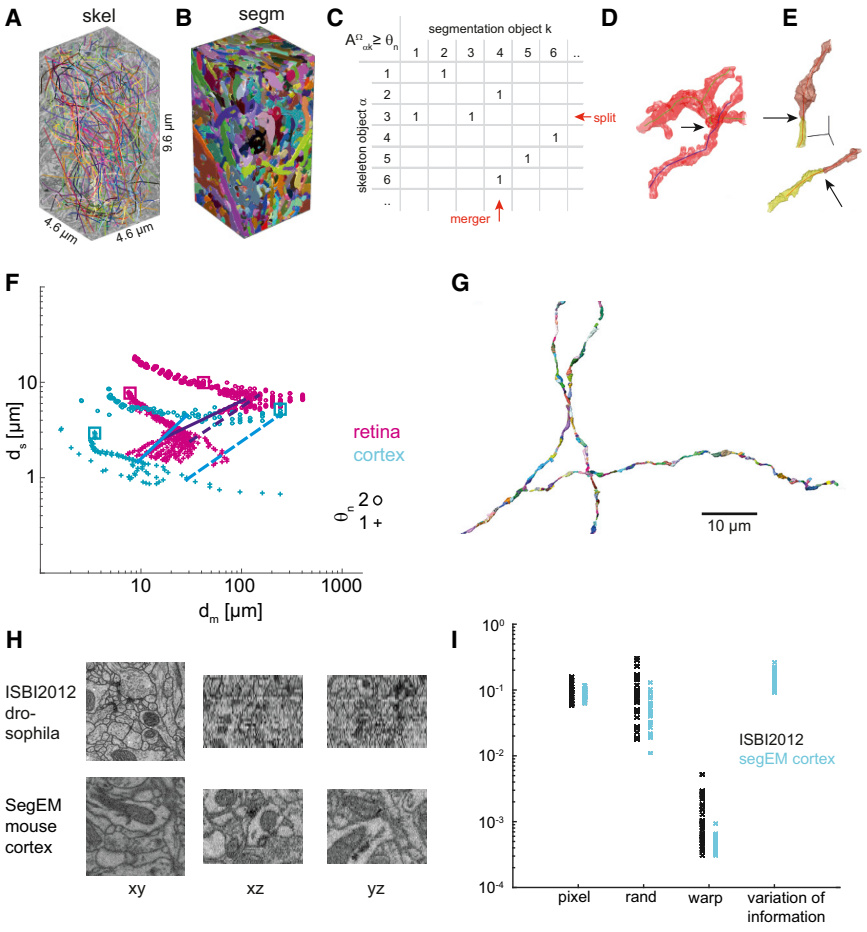

Figure 7. Segmentation Metrics and SegEM 3D Segmentation Challenge for Connectomics

(A–C) Quantification of segmentation accuracy by comparison of automated volume segmentation Ω (shown in B, classifier CNN3,2, see Table 2; r_{se} = 1, θ_{mg} = 0.3, θ_{ms} = 150) with dense human-labeled skeleton annotation (A) yields overlap matrix $A^{\Omega}_{\alpha k}$ between skeleton objects α and segmentation objects k, which is binarized at node-overlap threshold Θ_n (C, see Experimental Procedures). Row sums exceeding 1 integrate to total split count; column sums exceeding 1 integrate to total merger count.

(D and E) Examples of mergers (D) and splits (E) (see Data S4 and S5 for complete gallery).

(F) Skeleton-based split-merger metric: average distance between splits (d_s) and mergers (d_m) for 372 segmentations obtained from varying the segmentation parameters as shown in Figure 2D for retina (magenta) and 99 segmentations cortex (cyan). Error distances are reported for skeleton-node-to-segmentation object overlap thresholds Θ_n = 1,2 (the larger threshold is less sensitive to node placement noise; see Experimental Procedures). Squares: optimal inter-error distance for Θ_n = 1,2 (3.85 µm, 7.91 µm for retina and 1.58, 4.93 µm for cortex, respectively). Dashed lines: segmentation used in Figures 4 (retina, purple line) and 5 (cortex, blue line); see also (G); solid lines: segmentation used in all other figures (lines connect split-merger point for Θ_n = 1,2.

(G) Cortex axons and overlapping segmentation objects from segmentation in (F) (blue dashed line) to illustrate average object size (2 axons from Figure 5).

(H) Comparison of image data from ISBI 2012 2D-EM challenge and SegEM 3D-EM challenge. Note higher resolution of data in xy (ISBI) versus problematic resolution and alignment in third dimension. Shown data are the entire training dataset for ISBI challenge. SegEM challenge comprises 279 of the volumes shown in (H).

(I) Comparison of all results submitted to ISBI 2012 2D-EM segmentation challenge as reported on http://brainiac2.mit.edu/isbi_challenge/ (black) and SegEM performance on test set of 34 (100 voxel)^3 regions from cortex dataset (blue) evaluated using the ISBI metrics (see Supplemental Experimental Procedures for details of calculation). Scale bars, 1 µm in (D) and (E); 10 µm in (G).

are rarely split longitudinally (example of longitudinal split: Figure S2C). Under this constraint, we find at node threshold θ_n = 2 an inter-merger distance of 151.3 µm and inter-split distance of 7.4 µm (Figure 7F, purple dashed line) to be most useful for whole-cell reconstruction in retina. For cortex, the segmentation used for whole-cell reconstruction had an inter-merger distance of 241.5 µm and inter-split distance of 4.6 µm (Figure 7F, blue dashed line; see Figure 7G for an example skeleton-segmentation overlay of a cortical axon using this segmentation setting).

Thus, in addition to enabling high-throughput circuit reconstruction, SegEM provides all training data and test metrics for large, truly 3D-EM datasets. The supplied training data comprise 987 µm^3 of labeled data, thus about 100 times more than that in the available EM challenges. Importantly, these data are sampled from many locations within a large 3D-EM dataset, providing properly generalizing classifiers. With this, SegEM also becomes a next-generation benchmark for machine learning in connectomics.

DISCUSSION

We have developed SegEM, a semi-automated volume segmentation toolset for circuit reconstruction in fully stained 3D-EM image data (Figure 6). We applied our toolset to neuronal tissue from mouse retina and cerebral cortex. The exemplary synaptic innervation analyses (Figures 4 and 5) illustrate that SegEM resolves the tradeoff between synapse detectability in fully stained EM data and the reconstruction efficiency gain obtained by crowd-sourced skeleton reconstructions, which is required for dense circuit reconstruction in large-scale EM data.

We have proven the applicability to two types of neuronal tissue data even though the voxel-based classifiers do not generalize well between volume datasets from different neuronal tissues (Figures S2D and S2E). This may point to relevant local geometrical differences between peripheral and central nervous tissue.

While throughput of image analysis is still the major bottleneck in high-resolution connectomics, SegEM has made a substantial

difference in our own circuit analyses, reducing the investment per circuit analysis project by greater than 10-fold.

Further improvements in automated segmentation and human-machine interaction will be required for the most ambitious projects, such as whole-brain circuit reconstruction in the farther future, but SegEM has the ambition to make high-resolution connectomics a ready-to-use technique for many laboratories today (Figures 4, 5, and 6) and to furthermore provide a benchmark for fully 3D-EM-based connectomic data analysis (Figure 7).

EXPERIMENTAL PROCEDURES

For the application of SegEM, please follow the detailed instructions in the Supplemental Experimental Procedures.

3D-EM Image Datasets

The "retina" EM dataset (designated ek563, from Briggman et al., 2011 and Helmstaedter et al., 2013) was acquired using serial blockface EM (SBEM, Denk and Horstmann, 2004). Tissue from a P30 mouse retina was stained using a conventional en bloc EM protocol comprising osmium tetroxide, thiocarbohydrazide amplification, and uranyl acetate staining steps (for details, see Briggman et al., 2011). SBEM images were aligned as described in Briggman et al. (2011): briefly, cross-correlation-derived shift vectors were computed in overlapping image regions, shift vectors were globally least-square optimized, and images were shifted using fourier-shift interpolation. The image data (4,096 × 5,304 × 5,760 voxels; voxel size 12 × 12 × 25 nm^3; dataset size 50 × 65 × 145 μm^3; Briggman et al., 2011) were normalized to zero mean and unit SD on a grid with spacing of (128 × 128 × 128) vx^3 and cubes of size (163 × 163 × 143) vx^3 for the whole dataset. Each training volume was accordingly normalized (training volumes varied in size from [256 × 256 × 256] vx^3 to [512 × 512 × 256] vx^3; Table S1).

The "cortex" EM dataset (designated 2012-09-28_ex145_07x2; K.M.B. and M.H., unpublished data) was acquired using SBEM. Tissue from S1 cortex of a P28 mouse was stained using a conventional en bloc EM protocol similar to the retina dataset. Image data (8,274 × 5,338 × 3,321 voxels; voxel size 11.24 × 11.24 × 28 nm^3; dataset size 93 × 60 × 93 μm^3) were normalized to approximately zero mean (unnormalized mean: 122) and unit SD (unnormalized SD: 22) for the whole dataset (see Figure S1H). Each training volume was accordingly normalized (size of training volumes: [100 × 100 × 100) vx^3]). Images were aligned as for ek0563, but shift vectors between images were weighted before global optimization. Note that a re-evaluation of effective cutting thickness indicated a voxel size of 11.24 × 11.24 × 26 nm^3; this correction was not applied to the data in this paper. All procedures were approved by the local animal care committee and were in accordance with the law of animal experimentation issued by the German Federal Government.

Split and Merger Rates

The densely skeletonized segmentation test volume (see above and Figure 7A) was used to calculate the average length between merger (Figure 2C) and split (Figure 2D) errors (Turaga et al., 2010). To allow comparison of split-merger metrics between independently skeletonized datasets of different neurite geometry (training and test skeletonizations from retina and cortex), we first equilibrated skeleton node densities between skeleton sets (resulting average inter-node distance: 460 nm).

Then, we calculated a skeleton node-to-segmentation object overlap matrix $A^{\Omega}_{\alpha k}$ reporting for each combination of skeleton α and segmentation object k (in segmentation Ω) the number of skeleton nodes in α that overlapped with any voxel labeled as segmentation object k (see Figure 7C):

$$A^{\Omega}_{\alpha k} = \sum_{m=1..N_{\alpha}} \delta(\Omega(i(\alpha_m)), k),$$

with N_{α} the total number of nodes in skeleton α, $i(\alpha_m)$ the voxel location of the m^{th} node in skeleton α, and $\Omega(i)$ the label of segmentation Ω at location i. This matrix was then binarized at a node threshold θ_n, which was varied between 1 and 2 to assess the influence of labeling (i.e., skeleton node placement) noise (see Figure 7F): $A^{\Omega\#}_{\alpha k} = \begin{cases} 1 \text{ if } A^{\Omega}_{\alpha k} \geq \theta_n \\ 0 \end{cases}$.

The number of mergers and splits was then calculated as

$$n_{merger} = \sum_{\alpha=1..N_S} \left(\left(\sum_{k=1..N_{\Omega}} A^{\Omega\#}_{\alpha k} - 1 \right) * \Theta \left(-1.5 + \sum_{k=1..N_{\Omega}} A^{\Omega\#}_{\alpha k} \right) \right)$$

and

$$n_{splits} = \sum_{\alpha=1..N_{\Omega}} \left(\left(\sum_{k=1..N_S} A^{\Omega\#}_{\alpha k} - 1 \right) * \Theta \left(-1.5 + \sum_{k=1..N_S} A^{\Omega\#}_{\alpha k} \right) \right),$$

with N_S and N_{Ω} as the number of skeletons and segments, respectively, while Θ is the Heaviside step function. Finally, the average distance between splits d_s and between mergers d_m was calculated as $d_s = L / n_{split}$ and $d_m = L / n_{merger}$, with L the total skeleton path length in the segmentation test volume (L = 0.48 mm and 1.21 mm for cortex and retina test volumes, respectively). In cases where no split or no merger was detected, n_{merger} or n_{split} were set to 1 to limit error distance confidence. Segmentations resulting in only 1 merger or only 1 split were not evaluated. Optimal inter-error distance (the minimum of $1/(1/d_s + 1/d_m)$) was determined by a sliding 3-nearest neighbor average in the ds-dm plane (Figure 7F).

Note that to avoid artifacts by different fractions of wall voxels in different segmentations, the split-merger metric should be computed on fully grown segmentations only (no remaining wall classified voxels; Figure 7F was computed for such fully grown out segmentations only).

The code for computing this metric is contained in evaluateSeg.m (Data S8).

SegEM 3D Image Segmentation Challenge for Connectomics

For participation in the SegEM 3D image segmentation challenge for connectomics, please follow the instructions below. SegEM (segem.io) provides 279 densely volume-annotated volumes in which (100 voxel)^3 are each labeled. These volumes are sampled from throughout the cortex dataset, providing a representative sampling of neurite geometry and image statistics (see Table S2 for a detailed overview). To allow training of classifiers of larger field of view, we supply the raw data for each labeled volume of size 200 × 200 × 150 voxel, i.e., with a border of 100 × 100 × 50 voxel around the labeled volume.

For testing the split-merger metric, we provide a densely skeletonized test set (Data S3), which was used for Figure 7F. Split-merger metrics calculation is contained in cortex/segmentation/evaluateSeg.m of the SegEM package (Data S8 or segem.io). For calculation of pixel, warp, rand error, and variation of information as in Figure 7I, please use the code provided by the ISBI 2012 2D-EM segmentation challenge (see above).

For submission of results to the SegEM 3D image segmentation challenge, an email containing the code to process raw data volumes of the same size as the training data should be sent to segEMchallenge@brain.mpg.de. This will be evaluated on 34 test stacks of the same size as the training stacks given above and on an additional densely skeletonized test set from several regions in the cortex dataset. Random examples of the raw data of these hidden test sets will be provided on segem.io, but not the respective labels. Ranking of segmentation results will be continuously updated on segem.io with respect to the ISBI challenge metrics and skeleton-based split-merger metrics.

SUPPLEMENTAL INFORMATION

Supplemental Information includes Supplemental Experimental Procedures, two figures, two tables, and nine data files and can be found with this article online at http://dx.doi.org/10.1016/j.neuron.2015.09.003.

AUTHOR CONTRIBUTIONS

All work was performed by M.B. and M.H. K.M.B. acquired the cortex dataset.

ACKNOWLEDGMENTS

We thank Viren Jain, Srinivas Turaga, and Sebastian Seung for fruitful discussions in the early phase of the project; Sarah Kaspar for initial classifier training; and Martin Zauser for contributions to code. We thank Yates Buckley, Winfried Denk, Emmanuel Klinger, Helene Schmidt, and Jakob Straehle for comments on earlier versions of the manuscript and Chris Roome and Mario Hilpert for support of the compute infrastructure. We are especially grateful to Heiko Wissler, Fabian Isensee, and Tim Decker for their excellent help with annotation management, synapse annotation, and cell-type classification. We thank Sophie Dittmer, Marco Reiner, Lisa Geiser, Katharina Haase, Bianca Hauber, Julian Sieber, Andre Antunes, Stephanie Best, Simone Pfarr, Claudia Klein, Christina Fianke, Jenny Bauer, Philipp Bastians, Arlie Zegarra, Julia Volz, Patrick Weber, Maximilian Schiedeck, Stephanie Freiss, Sadia Oumohand, Annika Gable, Janika Briegel, Maria Mueller, Emilia Wiegand, Jessica Dietrich, Julia Pesch, Florian Viehweger, Torben Tannig, Niclas Rieger, Patrick Hofmann, Kathrin Schramm, Michael Kirchberger, Julia Hammerich, Stefanie Ehrhardt, Marie Reitz, Erlina Rachmad, Jenny Raetzer, Maximilian Scheller, Matthias Pohrath, Maret Heumannskaemper, Jeannie Schied, Christian Orlik, Lili Hocke, Corinna Eck, Kristina Ernst, Ralf Vogel, Thomas Schackel, Yannick Soehngen, Maike Schramm, Kai Fischer, Edem Atsiatorme, Julia Ricken, Ramtin Lichtenberger, and Martina Jonczyk for neurite skeletonization and Anikó Pálfi, Anna Buntjer, Anna Leinhos, Anna Rommerskirchen, Benedikt Martin, Carina Meyer, Carolin Willburger, Christian Röhrig, Dirk Ollech, Franziska Hentzschel, Hanna Jakobi, Hanna Rogg, Jana Munstermann, Johanna Lott, Josephine Reinhardt, Karin Bretzel, Katharina Kappler, Kathrin Haase, Kathrin Schleich, Kira Garbe, Lisa Keitel, Madeleine Joel, Marcel Smykalla, Martin Moll, Monika Berberich, Nina Wommelsdorf, Patricia Hegemann, Patrick Weber, Philip Gallandi, Raphael Foitzik, Rebekka Wiggers, Sarah Kaspar, Steffen Klein, Stephanie Best, Susanne Dettmer, Susanne Schuster, Victoria Wissdorf, and Wolfram Volkwein for volume tracing. M.H. is a shared stakeholder of the patent "Method and apparatus for image processing," Published Patent Application No. 20100183217.

Received: March 11, 2015
Revised: August 3, 2015
Accepted: August 27, 2015
Published: September 23, 2015

REFERENCES

Andres, B., Koethe, U., Kroeger, T., Helmstaedter, M., Briggman, K.L., Denk, W., and Hamprecht, F.A. (2012a). 3D segmentation of SBFSEM images of neuropil by a graphical model over supervoxel boundaries. Med. Image Anal. *16*, 796–805.

Andres, B., Kroeger, T., Briggman, K., Denk, W., Korogod, N., Knott, G., Koethe, U., and Hamprecht, F. (2012b). Globally Optimal Closed-Surface Segmentation for Connectomics. In Computer Vision – ECCV 2012, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, eds. (Springer Berlin Heidelberg).

Bergstra, J., Yamins, D., and Cox, D.D. Hyperopt: A Python library for optimizing the hyperparameters of machine learning algorithms. Proceedings of the 12th Python in Science Conference, 2013. 13-20.

Briggman, K.L., and Bock, D.D. (2012). Volume electron microscopy for neuronal circuit reconstruction. Curr. Opin. Neurobiol. *22*, 154–161.

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

Cardona, A., Saalfeld, S., Schindelin, J., Arganda-Carreras, I., Preibisch, S., Longair, M., Tomancak, P., Hartenstein, V., and Douglas, R.J. (2012). TrakEM2 software for neural circuit reconstruction. PLoS ONE *7*, e38011.

Ciresan, D., Giusti, A., Gambardella, L.M., and Schmidhuber, J. (2012). Deep neural networks segment neuronal membranes in electron microscopy images. Adv. Neural Inf. Process. Syst. *25*, 2843–2851.

Denk, W., and Horstmann, H. (2004). Serial block-face scanning electron microscopy to reconstruct three-dimensional tissue nanostructure. PLoS Biol. *2*, e329.

Denk, W., Briggman, K.L., and Helmstaedter, M. (2012). Structural neurobiology: missing link to a mechanistic understanding of neural computation. Nat. Rev. Neurosci. *13*, 351–358.

Funke, J., Andres, B., Hamprecht, F.A., Cardona, A., and Cook, M. Efficient automatic 3D-reconstruction of branching neurons from EM data. Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, 16-21 June 2012 2012. 1004-1011.

Hayworth, K., Kasthuri, N., Schalek, R., and Lichtman, J. (2006). Automating the collection of ultrathin serial sections for large volume TEM reconstructions. Microsc. Microanal. *12*, 86–87.

Helmstaedter, M. (2013). Cellular-resolution connectomics: challenges of dense neural circuit reconstruction. Nat. Methods *10*, 501–507.

Helmstaedter, M., Briggman, K.L., and Denk, W. (2011). High-accuracy neurite reconstruction for high-throughput neuroanatomy. Nat. Neurosci. *14*, 1081–1088.

Helmstaedter, M., Briggman, K.L., Turaga, S.C., Jain, V., Seung, H.S., and Denk, W. (2013). Connectomic reconstruction of the inner plexiform layer in the mouse retina. Nature *500*, 168–174.

Jain, V., Murray, J.F., Roth, F., Turaga, S., Zhigulin, V., Briggman, K.L., Helmstaedter, M.N., Denk, W., and Seung, H.S. Supervised learning of image restoration with convolutional networks. Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on, 2007. IEEE, 1-8.

Jain, V., Bollmann, B., Richardson, M., Berger, D.R., Helmstaedter, M.N., Briggman, K.L., Denk, W., Bowden, J.B., Mendenhall, J.M., and Abraham, W.C. Boundary learning by optimization with topological constraints. Computer Vision and Pattern Recognition (CVPR), (2010a) IEEE Conference on, 2010. IEEE, 2488-2495.

Jain, V., Seung, H.S., and Turaga, S.C. (2010b). Machines that learn to segment images: a crucial technology for connectomics. Curr. Opin. Neurobiol. *20*, 653–666.

Jain, V., Turaga, S.C., Briggman, K.L., Helmstaedter, M.N., Denk, W., and Seung, H.S. (2011). Learning to Agglomerate Superpixel Hierarchies. Adv. Neural Inf. Process. Syst. *24*, 2.

Jones, C., Liu, T., Cohan, N.W., Ellisman, M., and Tasdizen, T. (2015). Efficient semi-automatic 3D segmentation for neuron tracing in electron microscopy images. J. Neurosci. Methods *246*, 13–21.

Kasthuri, N., Hayworth, K.J., Berger, D.R., Schalek, R.L., Conchello, J.A., Knowles-Barley, S., Lee, D., Vázquez-Reina, A., Kaynig, V., Jones, T.R., et al. (2015). Saturated Reconstruction of a Volume of Neocortex. Cell *162*, 648–661.

Kaynig, V., Vazquez-Reina, A., Knowles-Barley, S., Roberts, M., Jones, T.R., Kasthuri, N., Miller, E., Lichtman, J., and Pfister, H. (2015). Large-scale automatic reconstruction of neuronal processes from electron microscopy images. Med. Image Anal. *22*, 77–88.

Knott, G., Marchman, H., Wall, D., and Lich, B. (2008). Serial section scanning electron microscopy of adult brain tissue using focused ion beam milling. J. Neurosci. *28*, 2959–2964.

Lichtman, J.W., and Denk, W. (2011). The big and the small: challenges of imaging the brain's circuits. Science *334*, 618–623.

Liu, T., Jones, C., Seyedhosseini, M., and Tasdizen, T. (2014). A modular hierarchical approach to 3D electron microscopy image segmentation. J. Neurosci. Methods *226*, 88–102.

Nunez-Iglesias, J., Kennedy, R., Plaza, S.M., Chakraborty, A., and Katz, W.T. (2014). Graph-based active learning of agglomeration (GALA): a Python library to segment 2D and 3D neuroimages. Front. Neuroinform. *8*, 34.

Saalfeld, S., Cardona, A., Hartenstein, V., and Tomančák, P. (2009). CATMAID: collaborative annotation toolkit for massive amounts of image data. Bioinformatics *25*, 1984–1986.

Seyedhosseini, M., Kumar, R., Jurrus, E., Giuly, R., Ellisman, M., Pfister, H., and Tasdizen, T. (2011). Detection of neuron membranes in electron

microscopy images using multi-scale context and radon-like features. Med Image Comput Comput Assist Interv *14*, 670–677.

Seyedhosseini, M., Sajjadi, M., and Tasdizen, T. Image segmentation with cascaded hierarchical models and logistic disjunctive normal networks. Computer Vision (ICCV), 2013 IEEE International Conference on, 2013. IEEE, 2168-2175.

Snoek, J., Larochelle, H., and Adams, R.P. (2012). Practical Bayesian optimization of machine learning algorithms. Adv. Neural Inf. Process. Syst. *25*, 2951–2959.

Sommer, C., Straehle, C., Kothe, U., and Hamprecht, F.A. Ilastik: Interactive learning and segmentation toolkit. Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on, March 30 2011-April 2 2011 2011. 230-233.

Takemura, S.-Y., Bharioke, A., Lu, Z., Nern, A., Vitaladevuni, S., Rivlin, P.K., Katz, W.T., Olbris, D.J., Plaza, S.M., Winston, P., et al. (2013). A visual motion detection circuit suggested by Drosophila connectomics. Nature *500*, 175–181.

Turaga, S.C., Briggman, K.L., Helmstaedter, M., Denk, W., and Seung, H.S. 2009. Maximin affinity learning of image segmentation. *arXiv preprint arXiv:0911.5372*.

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

Vazquez-Reina, A., Gelbart, M., Huang, D., Lichtman, J., Miller, E., and Pfister, H. Segmentation fusion for connectomics. Computer Vision (ICCV), 2011 IEEE International Conference on, 2011. IEEE, 177-184.