DeepACSON: Automated Segmentation of White Matter in 3D Electron Microscopy

Ali Abdollahzadeh[1], Ilya Belevich[2], Eija Jokitalo[2], Alejandra Sierra[1*+], and Jussi Tohka[1+]

[1]A. I. Virtanen Institute for Molecular Sciences, University of Eastern Finland, Kuopio, Finland

[2]Electron Microscopy Unit, Institute of Biotechnology, University of Helsinki, Helsinki, Finland

[*]alejandra.sierralopez@uef.fi

[+]These authors share senior authorship

ABSTRACT

To trace the entirety of ultrastructures in large-scale three-dimensional electron microscopy (3D-EM) images of brain tissue, automated segmentation techniques are essential. The current automated techniques use deep convolutional neural networks (DCNNs), approaching human-level segmentation, but designed for tracing small numbers of neurites and for high-resolution datasets. Therefore, we developed DeepACSON to segment 3D-EM datasets of white matter at low-resolutions and for tens of thousands of myelinated axons. DeepACSON performs DCNN-based semantic-segmentation and shape decomposition-based instance-segmentation: a top-down design to account for severe membrane discontinuities inherited with low-resolution imaging. We applied DeepACSON to ten serial block-face scanning electron microscopy datasets of rats after sham-operation or traumatic brain injury, segmenting hundreds of thousands of long-span myelinated axons, thousands of cell nuclei, and millions of mitochondria with excellent evaluation scores. DeepACSON quantified the morphological and spatial aspects of white matter ultrastructures.

Introduction

Recent advances in automated serial-sectioning electron microscopy (EM) techniques allow for acquiring 3-dimensional (3D) image datasets of brain ultrastructures, with high-resolutions and from hundreds of micrometers of tissue[1–3]. The acquired image datasets are of big sizes, ranging from gigabytes to hundreds of terabytes or even petabytes of data. Converting such big datasets into quantitative information, such as mapping neuronal connections known as connectome or morphological data of the ultrastructures, requires annotation of individual components. But manual annotation of even a small section of a 3D-EM dataset is a tedious task, consuming thousands of hours of experts' time. For example, Berning *et al.*[3] reported that manual annotation of 215 neurites in 5×10^8 voxels required 1 500 h of time of trained students, or we estimated that the segmentation of $15 \times 15 \times 15$ µm^3 white matter ultrastructures requires 2 400 h of an expert's time[4]. Semi-automated segmentation methods based on machine learning approaches[3,5] have improved the rate of segmentation. However, these methods still require a

considerable amount of manual interaction as the segmentation is driven on the manually extracted skeletons of neurites, proofreading, and correction of errors.

Deep convolutional neural networks (DCNNs) have emerged as the key to the state-of-the-art automated segmentation techniques[6–11]. DCNN-based segmentation methods generally comprise two steps: generating a membrane probability map from 3D-EM datasets referred to as semantic-segmentation, followed by instance-segmentation that turns probability maps into individual object instances. DCNNs are typically used for semantic-segmentation, and more traditional image analysis techniques favoring a bottom-up design, i.e., over-segmentation and subsequent merge, are used for instance-segmentation. An example is DeepEM3D[7], applying a 3D watershed transform on the membrane probability maps. To avoid instance-segmentation of a two-step design, Funke *et al.*[8] proposed learning voxel affinities. Their method, however, continued with a 3D watershed segmentation and iterative region agglomeration. Recently, Januszewski *et al.*[9] suggested flood-filling networks, a single-object tracking technique, and Meirovitch *et al.*[11] introduced cross-classification clustering, a multi-object tracking technique, merging the semantic- and instance-segmentation into recurrent neural networks. These techniques have a bottom-up design, offering a sophisticated training-based alternative for over-segmentation and subsequent merge.

The bottom-up design of the state-of-the-art EM segmentation techniques[6–11] is subjected to local optimization. Hence, segmentation requires distinctive image features such as high contrast between the cellular membrane and other structures. Therefore, common to these techniques is that they are operating on 3D-EM images with high in-plane resolutions of 4-10 nm[26–9]. Another limitation of the bottom-up design is that it is computationally inefficient for segmenting tens or hundreds of thousands of neuronal processes on over a million cubic micrometers of tissue. In the segmentation of low-resolution 3D-EM datasets of white matter, we encountered both of the above mentioned difficulties: first, the membrane of myelinated axons is not resolved, meaning that there is no distinctive feature to differentiate the intra- and extra-axonal space of a myelinated axon at locations such as nodes of Ranvier. Second, tens of thousands of myelinated axons pass through a volume of about a million cubic micrometers. Therefore, we developed DeepACSON, a **Deep** learning-based **A**utomati**C** **S**egmentation of ax**ON**s. DeepACSON is also a two-step segmentation technique, but it approaches the segmentation problem from a top-down perspective, i.e., under-segmentation and subsequent split. This approach is made possible by the use of *a-priori* knowledge of the topology of myelinated axons and cell nuclei. DeepACSON performs DCNN-based semantic-segmentation and 3D shape analysis based instance-segmentation.

We obtained our 3D-EM datasets using serial block-face scanning electron microscopy (SBEM) from the corpus callosum and cingulum of rats after sham-operation or traumatic brain injury (TBI). The SBEM images were acquired at two resolutions simultaneously: high-resolution images, $15 \times 15 \times 50$ nm^3 voxel size in a small field-of-view $15 \times 15 \times 15$ µm^3, and low-resolution images, $50 \times 50 \times 50$ nm^3 voxel size in a large field-of-view $200 \times 100 \times 65$ µm^3. We segmented the high-resolution datasets using our earlier automated ACSON pipeline[4]. We used the ACSON segmentation of the high-resolution datasets to train DeepACSON, eliminating the need for manually annotated training sets. DeepACSON segmented ten large field-of-view SBEM datasets that sum to 1.09×10^7 µm^3 of white matter tissue into myelin, myelinated axons, mitochondria, and cell

nuclei. From these ten datasets, we segmented about 288 000 myelinated axons span to 9 m, 2 600 cell nuclei, and millions of mitochondria with excellent evaluation scores. Our segmentation of white matter enabled quantifying axonal morphology, such as the axonal diameter, eccentricity and tortuosity, and also spatial organization of ultrastructures, such as the spatial distribution of mitochondria, and the density of cells and myelinated axons.

Results

DeepACSON segmented and analyzed ten low-resolution large field-of-view SBEM datasets. We acquired the images from the ipsi- and contralateral of white matter from two sham-operated rats and three TBI rats. Figure 1 illustrates different steps of DeepACSON, where DCNNs perform semantic-segmentation, and 3D shape-analysis techniques perform instance-segmentation.

Datasets

We acquired SBEM images of the white matter simultaneously at low- and high-resolution (Fig. 2a). The images were from the ipsi- and contralateral corpus callosum of the sham-operated and TBI animals. The low-resolution datasets were $200 \times 100 \times 65$ μm^3 in size, with a voxel size of $50 \times 50 \times 50$ nm^3. Two-thirds of the low-resolution images were from the corpus callosum and one-third from the cingulum (Supplementary Table S1). The high-resolution datasets were $15 \times 15 \times 15$ μm^3 in size and imaged with an anisotropic voxel size of $15 \times 15 \times 50$ nm^3. The high-resolution images were acquired from the corpus callosum. Figure 2a shows the contralateral corpus callosum and cingulum of a sham-operated rat in low- and high-resolution. In Fig. 2b, we show images of the low- and high-resolution datasets from the same location. Ultrastructural components such as myelin, myelinated axons, mitochondria, and cell nuclei were resolved in both settings. However, other components such as unmyelinated axons (2b fuchsia panel, asterisks) and exposed axonal membrane, e.g., at nodes of Ranvier (2b cyan panel, arrowheads), were only resolved in the high-resolution datasets. Figure 2b (purple panel) shows a cell nucleus from the low-resolution dataset, the membrane of which is resolved, but not continuously.

Semantic-segmentation of white matter ultrastructures

We segmented the high-resolution SBEM datasets using our ACSON pipeline[4]. First, we down-sampled the high-resolution datasets and their corresponding segmentation to the resolution of low-resolution datasets to train the DCNNs (Fig. 1, step 1). Figure 2c shows a 3D rendering of myelinated axons of a high-resolution SBEM dataset segmented by the ACSON pipeline. Only four of the total ten datasets were used as training data, while the remaining six datasets were reserved for testing. We trained two DCNNs denoted as DCNN-mAx and DCNN-cN. DCNN-mAx generated the probability maps of myelin, myelinated axons, and mitochondria, and DCNN-cN generated the probability maps of cell nuclei and the membrane of cell nuclei (Fig. 1, step 2). The receptive field of DCNN-mAx was (45, 45, 15) voxels in (x, y, z) directions with 2 881 978 trainable parameters, and the receptive field of DCNN-cN was (45, 45, 15) voxels in (x, y, z) directions with 2,881,777 trainable parameters. We trained the DCNNs on a single NVIDIA Tesla P100-16 GB graphics processing unit (GPU) for one day. From 85 GB of aligned

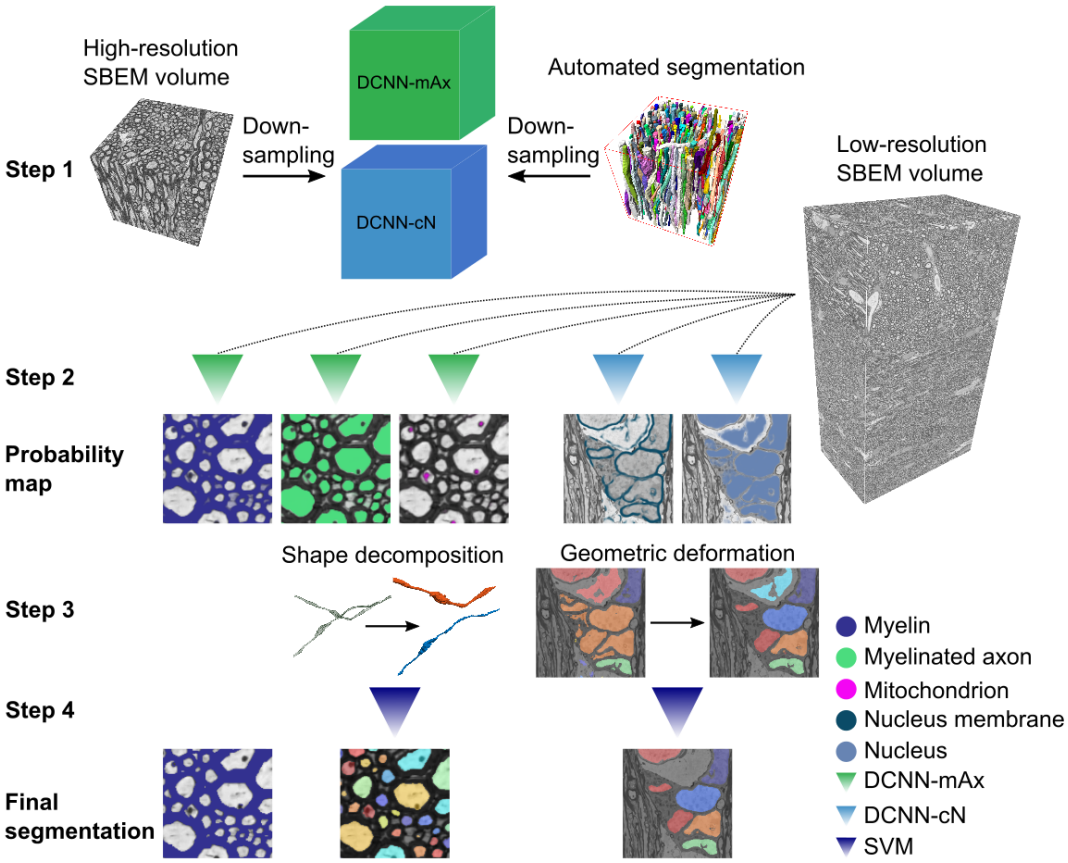

Figure 1. DeepACSON pipeline. **Step 1:** The ACSON segmentation of the high-resolution SBEM datasets, down-sampled to the resolution of the low-resolution datasets, was used for training DeepACSON. We trained two DCNNs denoted as DCNN-mAx and DCNN-cN. **Step 2:** DCNN-mAx returned the probability maps of myelin, myelinated axons, and mitochondria, and DCNN-cN returned the probability maps of cell nuclei and the membrane of cell nuclei. **Step 3:** The segmentation of myelin was finalized by thresholding the myelin probability map. We performed the initial segmentation of myelinated axons by the binarization and connected component analysis. Subsequently, the geometry of the segmented components was rectified using our newly developed cylindrical shape decomposition[12] (CSD) technique. We performed the segmentation of cell nuclei in a geometric deformable model (GDM) framework by applying elastic deformations to the initial segmentation of cell nuclei. **Step 4:** The segmentation of myelinated axons and cell nuclei was finalized by eliminating non-axonal and non-nucleus structures with support vector machines (SVMs).

and denoised low-resolution SBEM datasets, we acquired 1.24 TB (in 82 h: 0.297 MB/s) and 0.97 TB (in 81 h: 0.299 MB/s) of probability maps using DCNN-mAx and DCNN-cN, respectively. Figures 3a and b present the probability maps of myelinated axons, mitochondria, and myelin returned from DCNN-mAx. In Figs. 4a and b, we show the probability maps of cell nuclei and the membrane of cell nuclei returned from DCNN-cN.

Instance-segmentation of myelin, myelinated axons, and mitochondria

In low-resolution SBEM datasets, myelin was a single connected ultrastructural component. The segmentation of myelin was accomplished by binarizing its probability map (see Fig. 1, step 3 and Fig. 3b). To segment the intra-axonal space of myelinated axons, we binarized their probability map and applied connected component analysis. This preliminary segmentation was under-segmented (divided into too few segments) because the exposed axonal membrane was not fully resolved, e.g., at nodes

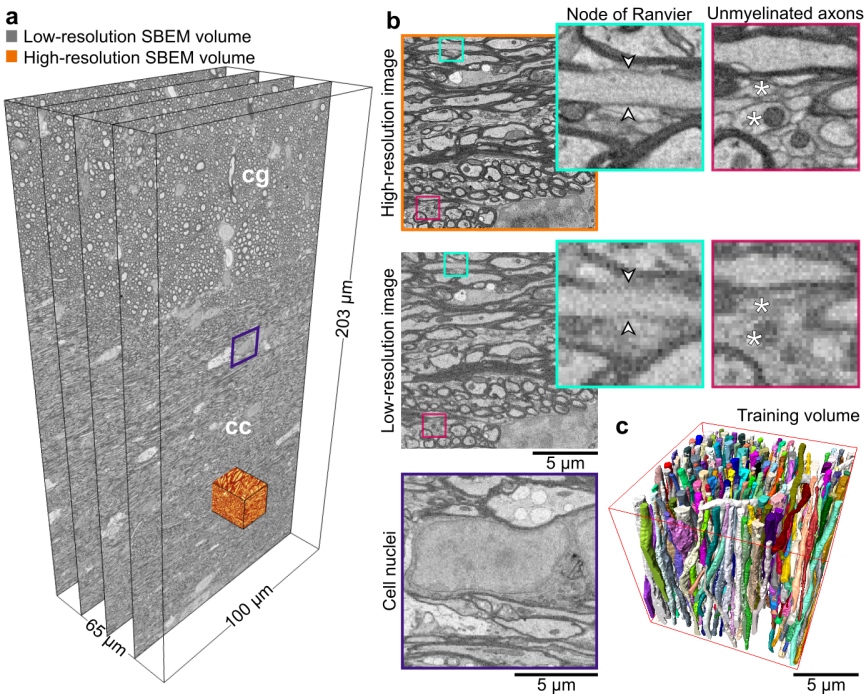

Figure 2. Low- and high-resolution SBEM imaging of the contralateral corpus callosum and cingulum of the sham #25 dataset. (**a**) We acquired SBEM images of the white matter, corpus callosum (cc) and cingulum (cg), simultaneously at high- and low-resolution. The field-of-view of the low-resolution dataset is $204.80 \times 102.20 \times 65.30\ \mu m^3$ equivalent to $4\,096 \times 2\,044 \times 1\,306$ voxels in x, y and z directions respectively, which is about 400 times larger than the field-of-view of the high-resolution datasets. (**b**) Images of the low- and high-resolution datasets acquired from the same location (the orange-rendered volume in (**a**)). The visualization of the high- and low-resolution images shows that myelin, myelinated axons, mitochondria, and cell nuclei were resolved in both settings, while the axonal membrane at nodes of Ranvier (cyan panel, arrowheads) and unmyelinated axons (fuchsia panel, asterisks) were only resolved in the high-resolution images. The purple panel shows a cell nucleus from the low-resolution dataset (**a**), which membrane is resolved, but not continuously. (**c**) A 3D rendering of myelinated axons in the high-resolution SBEM dataset (contralateral sham #25) segmented by the automated ACSON pipeline.

of Ranvier (Fig. 3c). Therefore, the preliminary segmentation of the intra-axonal space of myelinated axons required a further shape analysis step (Fig. 1, step 3). We devised an algorithm for the decomposition of cylindrical objects, called cylindrical shape decomposition[12] (CSD) (Supplementary Fig. S5). The technique decomposes under-segmented myelinated axons and subsequently rectifies their shape using generalized cylinders (Fig. 3c, Fig. $3c_1$, and Supplementary Fig. S6). To finalize the segmentation, we excluded non-axonal instances from the segmentation using a support vector machine (SVM) with a quadratic kernel (as implemented in MATLAB, Statistics and Machine Learning Toolbox, *fitcsvm* function) (Fig. 1, step 4). Figure 3d shows the 3D rendering of the DeepACSON segmentation of myelinated axons in the contralateral corpus callosum and cingulum of a sham-operated rat dataset. Figure 3e shows the 3D rendering of myelinated axons randomly sampled at two locations from the corpus callosum and a location from the cingulum. This figure depicts the orientation dispersion in bundles of myelinated axons.

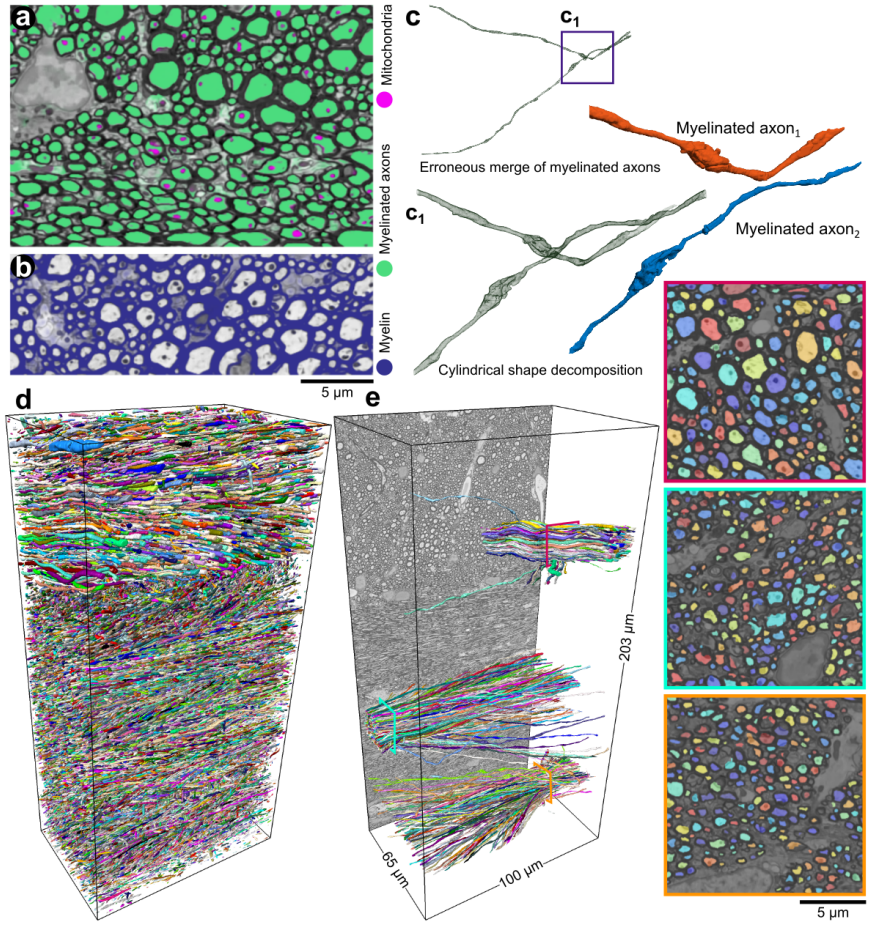

Figure 3. DeepACSON segmentation of myelin, myelinated axons, and mitochondria in white matter. (**a, b**) The probability maps of myelinated axons, mitochondria, and myelin returned from DCNN-mAx, overlaid on their corresponding BM4D filtered images. (**c**) The CSD algorithm decomposed myelinated axons with erroneous merges. (**d**) 3D rendering of DeepACSON final segmentation of myelinated axons (at one-third of the original resolution) in the contralateral corpus callosum and cingulum of sham #25 low-resolution dataset. (**e**) 3D rendering of myelinated axons sampled at the corpus callosum and cingulum illustrates the variance of the axonal diameter among myelinated axons and the orientation dispersion in these bundles.

Instance-segmentation of cell nuclei

The membrane of cell nuclei was not fully resolved at low-resolution ($50 \times 50 \times 50$ nm^3 voxel size), and hence its probability map exhibited discontinuity. We segmented cell nuclei in a geometric deformable model (GDM) framework, where the initial segmentation of a cell nucleus was rectified for its topological errors (see Fig. 1, step 3 and Fig. 4c and d). Non-nucleus instances were excluded from the segmentation using an SVM with a quadratic kernel (Fig. 1, step 4). Figure 4e shows a 3D rendering of cell nuclei in the contralateral corpus callosum and cingulum of a sham-operated rat dataset.

Evaluations

We evaluated DCNN-mAx on a tissue-type level against a test set: six unseen high-resolution SBEM datasets labeled by the automated ACSON pipeline, and down-sampled to the resolution of low-resolution datasets. We calculated the precision (positive predictive value), recall (sensitivity), and F1 score (harmonic mean of precision and recall) for the segmentation

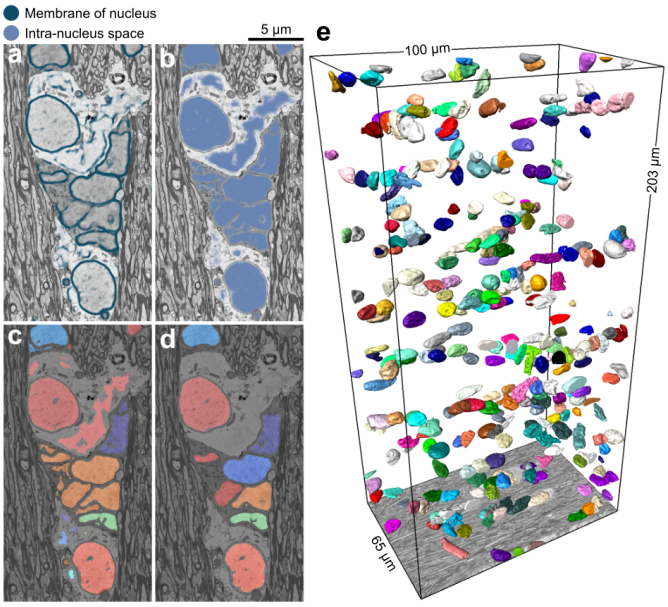

Figure 4. Segmentation of cell nuclei. (**a, b**) The probability maps of cell nuclei and their membrane were returned from DCNN-cN and overlaid on their corresponding BM4D filtered images. (**c**) The initial segmentation of cell nuclei contained topological errors as the membrane of cell nuclei exhibited discontinuity. (**d**) We rectified the segmentation of cell nuclei in a GDM framework and excluded non-nucleus instances by an SVM with a quadratic kernel. (**e**) 3D rendering of cell nuclei in the contralateral corpus callosum and cingulum of sham #25 dataset.

of myelin and myelinated axons (including mitochondria) against the labels provided by the automated ACSON pipeline (Supplementary Table S2). The average F1 score for myelin was 0.886 ± 0.049, and for myelinated axons was 0.861 ± 0.037.

We evaluated the performance of the two SVMs by a leave-one-group-out (LOGO) cross-validation, where the classifier was trained excluding the data from one group of animals (sham-operated or TBI) from training and evaluated against it (Supplementary Table S2). F1 scores were 0.988 and 0.955 for eliminating non-axonal structures while excluding sham datasets and TBI datasets, respectively. For eliminating non-nucleus structures, F1 scores were 0.969 and 0.981, excluding sham datasets and TBI datasets, respectively. The LOGO cross-validation showed that the performance of the SVMs was equally robust in all datasets, regardless of the experimental condition, i.e., sham-operated or TBI.

An expert (A.S.) evaluated DeepACSON segmentation of myelinated axons and mitochondria in an object-level using a GUI-based software tool we developed for this purpose (Supplementary Fig. S1). For this, we randomly sampled every dataset for five non-overlapping windows of size 300×300 voxels (10 datasets, 50 samples). The expert had no access to the dataset ID nor the sampling location. The expert evaluated the sampled images of the final segmentation by counting the number of true-positives, false-positives, and false-negatives. This resulted in following evaluation scores: myelinated axons (precision: 0.965 ± 0.027, recall: 0.877 ± 0.061, and F1 score: 0.918 ± 0.038) and mitochondria (precision: 0.856 ± 0.100, recall: 0.804 ± 0.091, and F1 score: 0.823 ± 0.067).

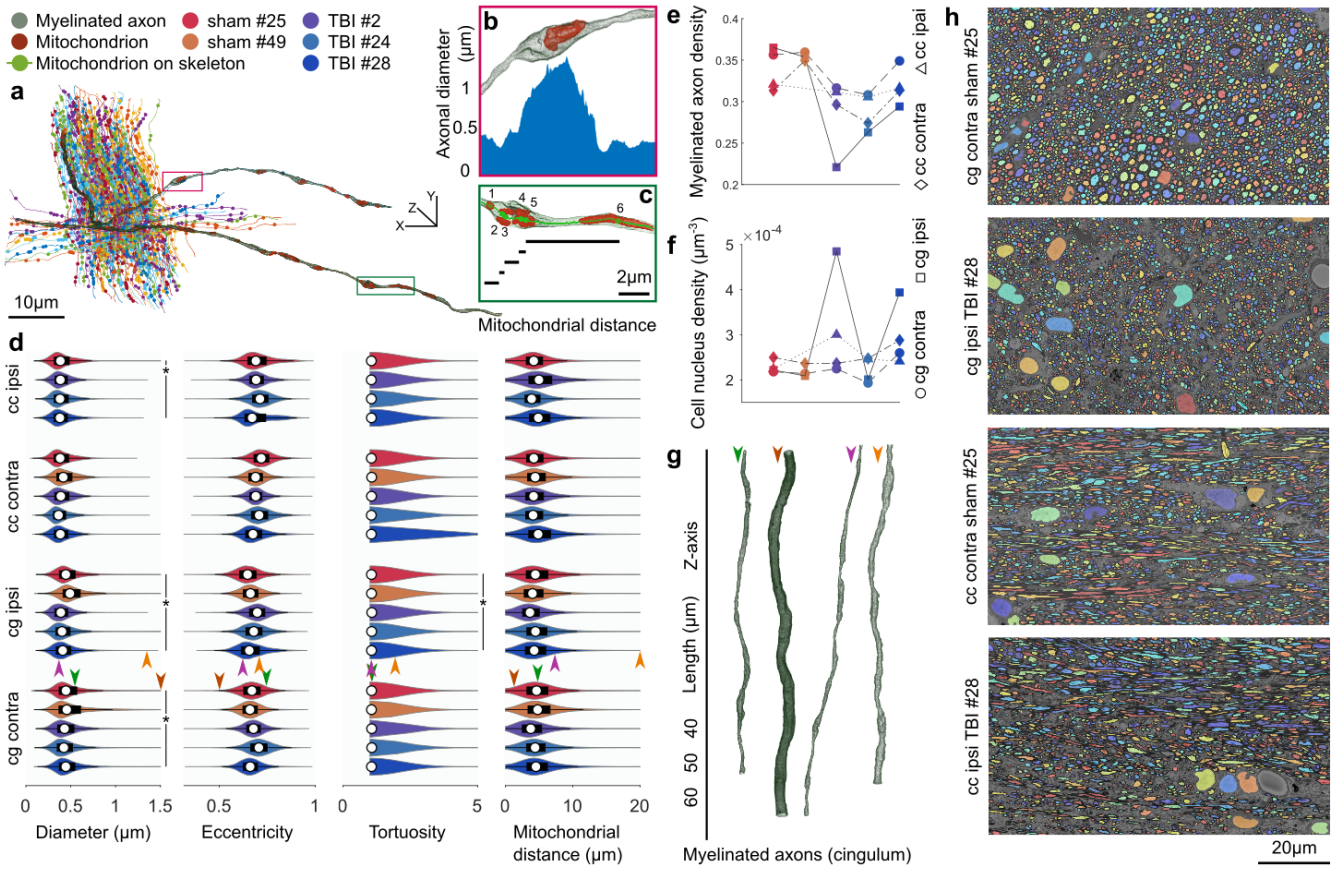

Figure 5. White matter morphology analysis. (**a**) A bundle of myelinated axons was sampled from the cingulum of the sham #25 dataset. Myelinated axons are represented by their curve skeletons. The centroids of mitochondria were projected on the axonal skeletons, shown as filled circles. (**b**) A small section of a myelinated axon from (**a**) represents how the axonal diameter can vary substantially along its length. The increased axonal diameter can be related to the accumulation of mitochondria. The plot shows the axonal diameter along the magnified section of the myelinated axon. (**c**) A small section of a myelinated axon from (**a**) shows the measure of inter-mitochondrial distance. Five mitochondria are accumulated with distances less than 1 µm, and one mitochondrion is distant from others with over 5 µm distance. (**d**) DeepACSON quantified the axonal diameter, eccentricity, and tortuosity of about 288 000 myelinated axons and the inter-mitochondrial distance of about 1 800 000 mitochondria. On each bean plot, the central mark indicates the median, and the left and right edges of the box indicate the 25th and 75th percentiles, respectively. The whiskers extend to the most extreme data points not considered outliers. The colors correspond with the animal ID. (**e**) Comparing the density of myelinated axons as the ratio of the volume of myelinated axons to the myelin volume plus the volume of myelinated axons. The color of the indicators corresponds with the animal ID. (**f**) Comparing the density of cells defined as the number of cell nuclei over the dataset volume. The color of the indicators corresponds with the animal ID. DeepACSON segmented about 2 600 cell nuclei in the ten large field-of-view datasets. (**g**) 3D rendering of myelinated axons from the cingulum visualizes the normative and outliers of the axonal diameter distribution. Each myelinated axon is given an arrowhead to mark its measurements in panel (**d**). (**h**) Representative images of the cingulum and corpus callosum in sham-operated and TBI rats visualize the smaller density of myelinated axons caused by the injury.

White matter 3D morphology analysis

We quantified several morphological and volumetric aspects of the segmented ultrastructures to demonstrate the applications of the white matter segmentation (Fig. 5). For every myelinated axon, we extracted cross-sections along its axonal skeleton, i.e., central axis, with a plane perpendicular to the skeleton. The cross-sections were quantified by the equivalent diameter, i.e., the diameter of a circle with the same area as the cross-section, and the eccentricity, i.e., how much a conic section deviates from

being circular, of the fitted ellipse. (Fig. 5a and b). We measured the tortuosity of myelinated axons as $\tau = \frac{l_{Gd}}{l_{Ed}}$, where l_{Gd} is the geodesic distance and l_{Ed} is the Euclidean distance between the two endpoints of the axonal skeleton. To quantify the spatial distribution of mitochondria, we measured the inter-mitochondrial distances along each myelinated axon. For this, we projected the centroid of mitochondria on the axonal skeleton and measured the geodesic distance between the consecutive projected centroids (Fig. 5a and c).

The analysis indicated that the diameter of myelinated axons varies substantially along the axonal skeleton. Moreover, the distribution of diameters was bimodal, which can partially be related to the location of mitochondria[13] (Fig. 5a, b, and d). The cross-sections of myelinated axons were elliptic rather than circular (Fig. 5d). Myelinated axons were almost straight as the mean of τ in the corpus callosum was 1.068 and in the cingulum was 1.072, but there was a big variation in the tortuosity as the standard deviation of τ was 0.731 and 0.396 in the corpus callosum and cingulum, respectively (see Fig. 5a and d). The distribution of the inter-mitochondrial distance along a myelinated axon was bimodal because mitochondria were either accumulated or appeared distant from each other (Fig. 5a, c, and d).

For the statistical hypothesis testing between the sham-operated and TBI animals, myelinated axons were represented by the median of equivalent diameters, the median of the eccentricities, tortuosity, and the mean of inter-mitochondrial distances (Fig. 5d). We subjected the measurements to the nested (hierarchical) 1-way analysis of variance (ANOVA) separately for each hemisphere[14]. The nested ANOVA tests whether there exists a significant variation in means among groups, or among subgroups within groups. The myelinated axons were nested under the rats' ID, and the rats' ID was nested under the groups (sham-operated and TBI). We set the alpha-threshold defining the statistical significance as 0.05 for all analyses. The rats' IDs were treated as random effects, and the group was treated as a fixed effect. The equivalent diameter was significantly smaller in the ipsi- ($F = 16.27$, $p = 0.027$) and contralateral cingulum ($F = 29.28$, $p = 0.011$) and in the ipsilateral corpus callosum ($F = 15.75$, $p = 0.047$) of TBI rats as compared to sham-operated rats. Also, the tortuosity in the ipsilateral cingulum of TBI rats was significantly greater than sham-operated rats ($F = 25.23$, $p = 0.018$). The equivalent diameter of the contralateral corpus callosum of TBI rats was slightly smaller than sham-operated rats, but not significantly ($F = 5.78$, $p = 0.095$). In the ipsilateral cingulum, the inter-mitochondrial distance was slightly smaller in TBI rats, but not significantly ($F = 6.27$, $p = 0.086$). We did not observe a difference in the tortuosity of the contralateral cingulum ($F = 3.20$, $p = 0.134$), ipsilateral corpus callosum ($F = 1.30$, $p = 0.292$), and ipsilateral corpus callosum ($F = 0.03$, $p = 0.878$) when comparing sham-operated and TBI rats. We did not observe a difference between sham-operated and TBI rats in the eccentricity (contralateral cingulum: $F = 1.57$, $p = 0.299$; ipsilateral cingulum: 3.33, $p = 0.165$; contralateral corpus callosum: $F = 0.67$, $p = 0.471$; ipsilateral corpus callosum: $F = 0.10$, $p = 0.780$) and inter-mitochondrial distances of the contralateral cingulum ($F = 0.33$, $p = 0.603$), contralateral corpus callosum ($F = 0.067$, $p = 0.812$), and ipsilateral corpus callosum ($F = 1.04$, $p = 0.414$) (Fig. 5d). We also quantified the volumetric aspects of the ultrastructures (Supplementary Tables S3). We could not directly compare the volume of the structures between datasets as the volume of cell processes was different in each dataset. Therefore, we calculated the density of myelinated axons as the ratio of the volume of myelinated axons to the myelin volume plus the volume of myelinated axons (Fig. 5e). We observed that

the density of myelinated axons was significantly smaller in the ipsilateral cingulum ($F = 13.03$, $p = 0.037$) of TBI compared to sham-operated rats (Fig. 5e). We did not observe a significant difference in the density of myelinated axons in the contralateral cingulum ($F = 4.29$, $p = 0.130$), ipsi- ($F = 3.42$, $p = 0.162$) and contralateral corpus callosum ($F = 2.13$, $p = 0.282$) of TBI rats (Fig. 5e). We also calculated the density of cells defined as the number of cell nuclei over the volume of the corresponding dataset. We did not observe a significant difference in the density of cells comparing TBI and sham-operated rats (Fig. 5f): (contralateral cingulum: $F = 0.16$, $p = 0.717$; ipsilateral cingulum: $F = 1.79$, $p = 0.273$; contralateral corpus callosum: $F = 0.48$, $p = 0.540$; ipsilateral corpus callosum: $F = 1.02$, $p = 0.419$). Figure 5g shows examples of 3D-rendered myelinated axons in the cingulum, demonstrating the high variability of axonal diameters. Fig. 5h demonstrates the sparsity of myelinated axons due to the injury in the cingulum and corpus callosum of TBI rats.

Computation time

The pipeline required about five days to segment one raw low-resolution SBEM dataset of size $4\,000 \times 2\,000 \times 1\,200$ voxels into its final segmentation. The details are presented in Supplementary Table S4.

Discussion

We developed DeepACSON applying DCNN-based semantic-segmentation and introducing 3D-shape analysis as an instance-segmentation methodology. DeepACSON annotated hundreds of thousands of long-span myelinated axons, thousands of cell nuclei, and millions of mitochondria in large field-of-view 3D-EM datasets. We acquired our EM datasets in a multi-resolution scheme: low-resolution imaging to trace myelinated axons, at about one-tenth of a millimeter of their length, and high-resolution imaging to resolve ultrastructural details. This image acquisition scheme enabled providing human-annotation-free training sets for DeepACSON, as our earlier fully automatic ACSON pipeline[4] segmented the high-resolution datasets. The validation scores of DeepACSON demonstrated excellent performance in the segmentation of white matter ultrastructures. DeepACSON quantified the morphology of myelinated axons, the spatial distribution of mitochondria, and the density of myelinated axons and cells in the white matter of sham-operated and TBI rats. Our findings indicated that changes in the axonal diameter, tortuosity, and the density of myelinated axons were persistent even five months after the injury.

Segmentation of large field-of-view 3D-EM datasets with tens of thousands of myelinated axons requires automated techniques that are not only accurate but also error-aware. The vast majority of automated neurite segmentation techniques in 3D-EM datasets apply a bottom-up design[7–9,11,15–17]. These techniques rely on local optimization, which requires distinctive local image features to differentiate among regions. EM imaging at low-resolutions, about 40-50 nm in-plane resolution, do not continuously resolve the cellular membrane based on which two distinct regions can be differentiated. For example, low-resolution imaging did not resolve the axonal membrane at nodes of Ranvier, not in-plane, nor for several successive images in-depth. Therefore, a bottom-up segmentation is prone to contain under-segmentation errors. In the presence of a-priori knowledge, we applied a top-down design to account for the under-segmentation errors. A related study is MaskExtend[10], proposing a 3D watershed transform and subsequent merge and continuing with detecting X-shape objects to find under-

segmentation errors. Unlike this method, which makes restrictive assumptions about the geometry of under-segmented objects, we address a generic geometry for the under-segmented objects as X-shape objects are a special case in our CSD algorithm. Also, we reconstruct/rectify an under-segmented object on its own by applying generalized cylinders, as opposed to MaskExtend, that requires accessing the watershed segments and merging them at a high merge-threshold[10]. At a more general level, Falk *et al.*[18] developed a plugin in ImageJ software[19] and addressed under-segmentation errors in cells by inserting an artificial one-pixel wide background ridge between touching instances in the training set. This approach is similar to our technique for segmenting cell nuclei, where we inserted an artificial membrane around cell nuclei using morphological operations and assigned high-weights while training. Another approach is CDeep3M[6], which uses a cloud-based residual inception-network architecture for the segmentation of image datasets from diverse modalities. This implementation assumes to obtain highly accurate semantic-segmentation from the network, hence continues to instance-segmentation with thresholding. This assumption ignores the topological errors of the segmentation.

Acquiring large field-of-view image datasets at high-resolutions and for several animals and experimental conditions is time-consuming and resource-wise costly. Despite the improvement in the rate of image acquisition[1,20], microscopes should run continuously for several months to acquire a cubic millimeter at an in-plane resolution of 4 nm^2[1]. Low-resolution imaging at an in-plane resolution of 40-50 nm^2 reduces the imaging time by approximately a factor of 100 as compared to the in-plane resolution of 4 nm^2. Low-resolution imaging allows for axonal tracking[21], morphology analysis of myelinated axons, spatial and orientation distribution analysis of myelinated axons, and spatial distribution of cell nuclei and mitochondria. As well, low-resolution imaging and the corresponding segmentation provide a base for neuronal tissue modeling.

The simultaneous high- and low-resolution imaging enables to use the high-resolution images as the training data to segment low-resolution datasets. The high-resolution images can be segmented using automated methods; hence, we provided a human-annotation-free training set. In our case, we used the ACSON pipeline to segment the high-resolution SBEM datasets[4]. ACSON is fully automated, requires no training data, and segments one of our high-resolution datasets, containing about 400 myelinated axons, in approximately 24 hours. Furthermore, high-resolution imaging resolves the membrane of tissue ultrastructures in several voxels wide, as opposed to low-resolution imaging that does not fully resolve the membrane, resulting in a more accurate segmentation of ultrastructures. Accurate segmentation of ultrastructures yields generating an accurate training set, which technically means that we have smaller label-noise, i.e., correlating confusing labels with underlying tissue ultrastructures, in the training set.

The validation of DeepACSON demonstrated an equally robust segmentation performance in datasets of both sham-operated and TBI rats, remarking that abnormalities in the shape of myelinated axons, such as myelin delamination, are more frequent in TBI datasets[4]. As well, the expert evaluation showed that the sensitivity of DeepACSON was smaller than its precision; we had a bigger number of false-negatives, i.e., myelinated axons that were not segmented, than false-positives. We speculate that very thin myelinated axons have been over-segmented into small-components. And because we filtered out small-components before the shape decomposition step, it increased the number of false-negatives. For example, an axon with a diameter

smaller than 0.2 µm was resolved with less than 13 cross-sectional voxels at $50 \times 50 \times 50$ nm^3 resolution, which was prone to over-segmentation. Segmentation of myelinated axons at low-resolutions can be difficult for an expert as well. In Mikula *et al.* study[21], an expert made an error in every 80 nodes of Ranvier in white matter, where the resolution was $40 \times 40 \times 40$ nm^3 and for the axonal diameter greater than 0.5 µm.

Large field-of-view imaging makes possible quantifying parameters whose measurement in a small field-of-view is not reliable, because the measurement may reflect a very local characteristic of the underlying ultrastructure. Therefore, we quantified the tortuosity of myelinated axons, inter-mitochondrial distance, and cell density only in the large field-of-view datasets. We also evaluated DeepACSON quantification. For that, we compared DeepACSON measurements of the axonal diameter and the eccentricity in the low-resolution datasets with measurements from the high-resolution datasets in our earlier work[4]. This comparison indicated that the quantification in the low-resolution datasets was consistent with the high-resolution datasets[4]. For example, we measured an average equivalent diameter for myelinated axons in the contralateral corpus callosum of sham #25 (sham #49) equal to 0.41 µm (0.44 µm) and 0.44 µm (0.50 µm) in the low- and high-resolution imaging, respectively. Additionally, we measured an average eccentricity for myelinated axons in the contralateral corpus callosum of sham #25 (sham #49) equal to 0.71 (0.69) and 0.72 (0.71) in the low- and high-resolution imaging, respectively. This consistency in the morphology analysis of low- and high- resolution datasets suggests that we have been able to correctly capture the morphology of myelinated axons, despite coarser resolution in the large field-of-view datasets.

We quantified morphological changes in myelinated axons in white matter five months after a severe TBI. The white matter pathology after TBI is extremely complex. The initial response of an axon to a brain injury can be either degeneration or regeneration[22,23], however, morphological alterations in axons can persist for years after injury in humans[24,25] and for up to one year in rats[26]. We found that the axonal diameter of myelinated axons in the ipsilateral corpus callosum, ipsilateral cingulum, and contralateral cingulum of TBI rats has significantly reduced. We further measured that the density of myelinated axons has reduced significantly in TBI, which reflects degeneration of axons after injury[25]. We also found that the tortuosity of myelinated axons has increased in the ipsilateral cingulum of TBI animals. We speculate that prolonged damage in microtubules might underlie the increase in axonal tortuosity[27].

Ultrastructural tissue modeling is an active research field aiming to bridge the gap between macroscopic measurements and cellular- and sub-cellular tissue level. Currently, tissue modeling are based on simplistic representations and assumptions of the ultrastructural features. These assumptions typically assume axons to be perfect cylinders, or the variation of the axonal diameter along axons are neglected[28,29]. Segmentation of brain tissue ultrastructures in 3D-EM datasets can substitute simplistic biophysical models and provide realistic models. Such realistic 3D tissue models open the possibility to investigate underlying reasons for the diffusion magnetic resonance imaging contrast and its macroscopic changes in brain diseases[30–33] or to investigate conduction velocity in myelinated and unmyelinated axons in electrophysiology[34,35].

Materials and Methods

Animal model, tissue preparation, and SBEM imaging

We used five adult male Sprague-Dawley rats (10-weeks old, weight 320 and 380 g, Harlan Netherlands B.V., Horst, Netherlands). The animals were singly housed in a room (22 ± 1 °C, 50% $-$ 60% humidity) with 12 h light/dark cycle and free access to food and water. All animal procedures were approved by the Animal Care and Use Committee of the Provincial Government of Southern Finland and performed according to the guidelines set by the European Community Council Directive 86/609/EEC.

TBI was induced by lateral fluid percussion injury[36] in three rats (TBI #2, #24, #28). Rats were anesthetized with a single intraperitoneal injection. A craniectomy (5 mm in diameter) was performed between bregma and lambda on the left convexity (anterior edge 2.0 mm posterior to bregma; lateral edge adjacent to the left lateral ridge). Lateral fluid percussion injury was induced by a transient fluid pulse impact (21-23 ms) against the exposed intact dura using a fluid-percussion device. The impact pressure was adjusted to 3.2-3.4 atm to induce a severe injury. The sham-operation of two rats (sham #25, #49) included all the surgical procedures except the impact. Five months after TBI or sham operation, the rats were transcardially perfused using 0.9% NaCl, followed by 4% paraformaldehyde. The brains were removed from the skull and post-fixed in 4% paraformaldehyde 1% glutaraldehyde overnight.

The brains were sectioned into 1-mm thick coronal sections with a vibrating blade microtome. Sections from -3.80 mm from bregma from each brain were selected and further dissected into smaller samples containing the areas of interest. We collected two samples from each brain: the ipsi- and contralateral of the cingulum and corpus callosum. The samples were osmium stained using an enhanced staining protocol[37]. After selecting the area within the samples, the blocks were further trimmed into a pyramidal shape with a 1×1 mm^2 base and an approximately 600×600 µm^2 top (face), which assured the stability of the block while being cut in the microscope. Silver paint was used to electrically ground the exposed block edges to the aluminum pins, except for the block face or the edges of the embedded tissue. The entire surface of the specimen was then sputtered with a thin layer of platinum coating to improve conductivity and reduce charging during the sectioning process. The details of the animal model and tissue preparation are described in ACSON study[4].

The blocks were imaged in a scanning electron microscope (Quanta 250 Field Emission Gun; FEI Co., Hillsboro, OR, USA), equipped with the 3View system (Gatan Inc., Pleasanton, CA, USA) using a backscattered electron detector (Gatan Inc.). The face of the blocks was in the *x*-*y* plane, and the cutting was in *z* direction. All blocks were imaged with a beam voltage of 2.5 kV and a pressure of 0.15 Torr. We acquired the high- and low-resolution datasets consistently at one specific location in the white matter in both sham-operated and TBI animals and in both hemispheres. The images were collected with an unsigned 16-bits per voxel. Supplementary Table S1 shows the volume size of the low-resolution datasets.

DeepACSON segmentation pipeline

We devised DeepACSON: a DCNN-based segmentation pipeline to automatically annotate white matter ultrastructures in the large field-of-view low-resolution 3D-EM datasets (Fig. 1). DeepACSON pipeline annotated white matter such that myelin and

each myelinated axon, mitochondrion, and cell nucleus carries its label. Figure 1 shows the four steps of the DeepACSON pipeline. **Step 1: Training set generation for semantic-segmentation**. DeepACSON requires labeled EM datasets as the training material to learn to perform the segmentation task. To automatically provide a training set for DeepACSON, we utilized our previously developed ACSON segmentation pipeline[4]. **Step 2: Semantic-segmentation of the white matter ultrastructure**. The volume of white matter ultrastructures was highly imbalanced, i.e., numbers of voxels representing different ultrastructures were not equal. Thus, we trained two separate DCNNs: DCNN-mAx to generate the probability maps of myelin, myelinated axons, and mitochondria and DCNN-cN to generate the probability maps of cell nuclei and the membrane of cell nuclei. **Step 3: Instance-segmentation of myelin, myelinated axons, mitochondria, and cell nuclei.** Myelin was segmented by thresholding the myelin probability map at $\theta_{myelin} = 0.5$. The segmentation of myelinated axons and cell nuclei required further geometrical and topological corrections after the binarization and connected component analysis. The CSD algorithm was applied to decompose under-segmented myelinated axons into their axonal compartments. In a GDM framework, elastic deformations of the initial segmentation of cell nuclei segmented this ultrastructure. **Step 4: Automatic elimination of false-positive instances.** We eliminated non-axonal and non-nucleus instances, i.e., false-positives, from the segmentation of myelinated axons and cell nuclei by training two separate SVMs.

Pre-processing: data alignment and denoising. We used Microscopy Image Browser[38] (MIB; http://mib.helsinki.fi) to pre-process the collected images from the electron microscope. We found the bounding box to the collected images and run voxel-size calibration. We aligned the images by measuring the translation between the consecutive SBEM images using the cross-correlation cost function (MIB, Drift Correction). We acquired a series of shift values in x direction and a series of shift values in y direction. From each series, the running average of the shift values (window size was 25 values) was subtracted to preserve the orientation of myelinated axons in the translation. We applied contrast normalization such that the mean and the standard deviation of the histogram of each image matches the mean and the standard deviation of the whole image stack. The images were converted to the unsigned 8-bits per voxel format. The size of low-resolution SBEM datasets ranged from 10-15 GB, summing up to 85 GB (8-bits per voxel). To denoise SBEM datasets, we used a non-local filtering algorithm named block-matching and 4D filtering (BM4D)[39]. The algorithm was run in its low-complexity mode to reduce the computation time. We processed the low-resolution SBEM datasets in non-overlapping patches with an approximate size of $1000 \times 1000 \times 300$ to account for RAM requirements and enabling parallel BM4D denoising on CPU clusters (see Supplementary Fig. S2).

Step 1: training sets for DCNNs. We used high-resolution SBEM datasets of the corpus callosum segmented by the automated ACSON pipeline to generate a training set for DeepACSON. The automated ACSON pipeline was designed for segmenting high-resolution small field-of-view SBEM datasets into myelin, myelinated and unmyelinated axons, mitochondria, and cells. The ACSON pipeline segments white matter based on a bounded volume growing algorithm in which seeds are defined automatically using regional maxima of the distance transform of myelin maps[4]. We generated two training sets: one for training DCNN-mAx and one for training DCNN-cN. The segmented high-resolution datasets were modified before using in the training sets. The labels of unmyelinated axons and cells were set equal to the background label to generate the DCNN-mAx

training set. The labels of all ultrastructures except for the cell nuclei were set equal to the background label to generate the DCNN-cN training set. We created an additional label for the membrane of cell nuclei by a morphological gradient with a flat 11×11 square structuring element applied to the segmented cell nuclei. The membrane of cell nuclei was relatively thick: 22 voxels or 0.33 µm wide. We over-emphasized the membrane of cell nuclei because the number of voxels representing it was very small as compared to the nucleus and background labels. After modifying labels, we uniformly down-sampled the high-resolution datasets in both training sets by a window size of 3×3 in $x - y$ plane to conform to the resolution of the low-resolution datasets. We generated nine down-sampled datasets out of each high-resolution dataset by shifting the sampling window for one voxel at a time in x and y directions. The DCNN-mAx training set included datasets of the ipsi- and contralateral of sham-operated and TBI rats. It is important to include both intact and injured axons to enrich the training set. The DCNN-cN training set included only ten cell nuclei.

Step 2: semantic-segmentation of the white matter ultrastructure, architecture and implementation of DCNNs. We implemented the DCNNs and data augmentation using ElektroNN2 library[5] based on the Python Theano framework[40]. ElektroNN was optimized for short model training times and fast inference on large datasets by eliminating the redundant computations of sliding window approaches[41]. The training set was augmented using randomized histogram distortions to become invariant against varying contrast and brightness gradients. In addition, half of the training set underwent image warping consisting of affine and random perspective transformations. We determined the optimal architectural parameters and hyperparameters of the DCNNs experimentally in the training set. We used the same architectural parameters for DCNN-mAx and DCNN-cN, as shown in Supplementary Fig. S3; receptive field was (45, 45, 15) voxels in (x, y, z) directions with 25 205 prediction neurons and 2 881 777 trainable parameters. The DCNNs were trained using the softmax function in tandem with the multinomial negative log-likelihood as a loss function. We set the batch size equal to 1 because there was a sufficient number of prediction neurons for updating the gradient. For the optimization, we used Adam optimizer[42] and set its initial learning rate $\alpha = 5 \times 10^{-5}$, exponential decay rate for the first moment $\beta_1 = 0.9$, exponential decay rate for the second-moment $\beta_2 = 0.999$, and the weight decay = 5×10^{-6} (we use the same notation as in[42]). Also, after every 1 000 training steps, the learning rate α was decayed by a factor of 0.995. For DCNN-mAx, we set the class weights to 0.2, 0.5, 1, and 1 for the background, myelin, myelinated axons, and mitochondria, respectively. The class weights for DCNN-cN were set to 0.1, 1 and, 4 for the background, cell nuclei, and the membrane of cell nuclei, respectively. The weights were proportional to the inverse of the volume of ultrastructure in the training datasets.

Step 3.1: myelin, myelinated axons, and mitochondria segmentation. DCNN-mAx generated probability maps of myelin, myelinated axons, and mitochondria. We binarized the probability maps of myelin, myelinated axons, and mitochondria with a threshold of $\theta_{myelin} = 0.5$, $\theta_{mAxon} = 0.8$, and $\theta_{mitochondria} = 0.5$. The binarized volume of myelin required no further processing, and it was considered as the final myelin segmentation. The binarized volume of mitochondria was dilated with a flat cubic structuring element of size three voxels, and unified with the binarized volume of myelinated axons to form the intra-axonal space of myelinated axons, I_{IAS}. Initial segmentation of individual myelinated axons was obtained using connected component

analysis on I_{IAS}. The segmented volumes underwent a morphological volume closing with a flat cubic structuring element of size three voxels to close the holes. We excluded volumes whose size was smaller than 4712 voxels (the volume of a circular cylinder with a base radius of 0.25 µm and height of 3 µm, resolved with an isotropic $50 \times 50 \times 50$ nm^3 voxel size).

In the low-resolution datasets, the axonal membrane was not resolved, e.g., at nodes of Ranvier. Thus, the initial segmentation of myelinated axons might exhibit under-segmentation error (Supplementary Fig. S6). We developed a specific shape analysis algorithm, CSD, to decompose an object into its semantic-components. As shown in Supplementary Fig. S5a, b, we defined the object as the union of several tubular components; similar to an axon with under-segmentation errors. The decomposition of such an object requires, first, to identify its semantic components, and second, to correct the ambiguity at intersections of branches, i.e., to decompose the shape at intersecting branches. The mathematical details of the algorithm and its validation are described in a different manuscript[12], but we summarize the algorithm here:

Step 3.1.1: skeleton partitioning and critical points. We propose to decompose an object into its semantic-components using its curve skeleton and cross-sectional analysis. We determined the curve skeleton of the object Ω by applying the method from Hassouna & Farag[43]. This method also returns the collection of skeleton branches $\Gamma = \{\gamma_1, \ldots, \gamma_n\}$ (see supplementary Fig. S5c). Based on Γ, CSD partitioned the curve skeleton of the object into $m \leq n$ maximal-length sub-skeletons denoted as $\psi_i, i = 1, \ldots, m$ (Supplementary Fig. S5d) via minimizing an orientation cost-function. Each sub-skeleton corresponds to a semantic-component. In other words, in this step, skeleton branches were merged if they were thought to belong to the same semantic component. To identify intersections of the semantic-components, we analyzed the object cross-sections along intervals of ψ, called decomposition intervals. A decomposition interval occurs in the proximity of junction-points where skeleton branches connect, denoted as $j \in \psi$. The boundaries of a decomposition interval at j were defined using the radius of the maximal inscribed ball at j, and two factors $\alpha_s = 10$ and $\alpha_e = 1.5$. At each decomposition interval, we search for a critical point. It is such a point that the object cross-section changes substantially; the normalized Hausdorff distance H_ρ[12] between a cross-sectional contour and the mean of visited cross-sectional contours exceeds $\theta_H = 0.85$ (Supplementary Fig. S5e). We cut the object at critical points to obtain object-parts. We assigned the same label to the object-parts along the same sub-skeleton to acquire a semantic-component (Supplementary Fig. S5f).

Step 3.1.2: object reconstruction. To reconstruct a semantic component at an intersection, we used generalized cylinders. We constructed a generalized cylinder $\Phi : \mathbb{R}^2 \to \mathbb{R}^3$ as a linear homotopy between two simple closed curves C_{c_1} and C_{c_2} elongated on a simple curve $\zeta \subset \mathbb{R}^3$. To define C_{c_1} and C_{c_2}, let $x_{c_1}, x_{c_2} \in \psi$ be two critical points in the proximity of the joint j on a sub-skeleton ψ. We cut the surface of the object $\partial\Omega$ at these two critical points. The cross-sectional curves of $\partial\Omega$ at x_{c_1} and x_{c_2} are C_{c_1} and C_{c_2}. We defined ζ to be the spline interpolation of ψ between x_{c_1} and x_{c_2}. The acquired generalized cylinder was used to reconstruct the object at the intersection which interior includes j (Supplementary Fig. S5f).

Step 3.2: cell nucleus segmentation. DCNN-cN classifier was trained to generate probability maps of cell nuclei and the membrane of cell nuclei from the low-resolution datasets. SBEM imaging with $50 \times 50 \times 50$ nm^3 voxel size, however, might not fully resolve the membrane of cell nuclei. Thus, the binarization and connected component analysis of DCNN-cN

probability maps at sites where several cell nuclei appear adjacently come with an under-segmentation error. We devised an approach in a GDM framework, i.e., an active surface implemented with level sets[44,45], to segment cell nuclei. Let the probability map of cell nuclei be denoted as $PR_n(x) : X \subset \mathbb{Z}^3 \rightarrow [0,1]$, and the probability map of the membrane of cell nuclei as $PR_m(x) : X \subset \mathbb{Z}^3 \rightarrow [0,1]$, where x is a 3D coordinate belonging to the 3D image domain X.

The position of the initial surface ∂U was determined by binarizing the curvilinear enhanced version of PR_m. We used Frangi filter[46] to enhance the curvilinear structures of PR_m, i.e., the membrane of cell nuclei, and suppress the background based on Hessian of PR_m (see Supplementary Fig. S4). The Frangi filtered PR_m is denoted as $P\hat{R}_m$. We subtracted $P\hat{R}_m$ from PR_n and thresholded the result at $\theta_{nucleus} = 0.5$. This resulted in a binary image $I_N(x) : X \rightarrow \{0,1\}$, where $x \in X$ was a foreground voxel and $I_(x) = 1$ if $PR_n(x) - P\hat{R}_m(x) > \theta_{nucleus}$. The initial position of deformable surfaces was determined using connected component analysis: I_N was segmented to n' individual cell nuclei denoted as $\hat{U}_i$, where $i = 1, \ldots, n'$. At locations at which several cell nuclei appear adjacently, the initial segmentation may include under-segmented cell nuclei. For that, the initial segmentation of cell nuclei required a further topological alteration. Therefore, the surface of each initially segmented cell nuclei underwent a primary deformation, where the surface was propagated for $-0.3\,\mu$m using a 3D Euclidean distance transform. The inward propagation of the surfaces splits the merged cell nuclei. Relabeling the split cell nuclei resulted in U_i, where $i = 1, \ldots, n$ and $n \geq n'$. We used the volumes in which the number of voxels was greater than 1 000 as initial surfaces for a secondary deformation. The evolution of a surface started at ∂U and stopped at the desired nucleus membrane because we defined the speed function equal to PR_m. Each initial surface ∂U was deformed for 300 iterations, where the smoothing term of GDMs preserved the surface sphericity from taking irregularities at sites where membrane probability map PR_m was discontinuous.

Step 4: eliminating false positives. The final step of the DeepACSON pipeline was to eliminate non-axonal and non-nucleus instances, i.e., false positives. We eliminated false positives from the segmentation of myelinated axons using an SVM with a quadratic kernel (MATLAB, Statistics and Machine Learning Toolbox). For each component segmented as a myelinated axon, we computed several features (at one third of the original size): volume, surface area, equivalent diameter, extent (the ratio of voxels in the component to voxels in the smallest bounding box of the component), solidity (the volume of the component divided by the volume of the convex hull of the component), and principal axis length (the length of the major axis of the ellipsoid that has the same normalized second central moment as the component). Features were standardized to the mean of zero and the standard deviation of one. To generate a training set, we randomly picked five datasets out of ten segmented large field-of-view datasets. From the selected datasets, we manually selected representative myelinated axons and non-axonal instances from both the cingulum and corpus callosum (1 097 myelinated axons and 189 non-axonal instances; less than 0.2% of the total number of components segmented as myelinated axons in all ten datasets). To determine the optimal SVM hyperparameters, the regularization parameter C and kernel parameter σ, we selected the pair that minimized the error sum of 5-fold cross-validation on the training set. Using Bayesian optimization algorithm[47,48] to search for the optimal C and σ pair constrained in $[10^{-6}, 10^{6}]$, we set $C = 1.12$ and $\sigma = 8.48$. The trained SVM was applied to eliminate non-axonal

structures from all the datasets. We applied the same approach to eliminate false-positives from cell nuclei. For each component segmented as a cell nucleus, we calculated the same features (as mentioned for myelinated axons) before and after the geometric deformation. To generate a training set, out of ten segmented large field-of-view datasets, we picked four random datasets. From the selected datasets, we manually selected representative cell nuclei and non-nucleus instances (263 cell nuclei and 113 non-nucleus instances; 7.3 % of the total number of components segmented as cell nuclei in all ten datasets) to train an SVM with a quadratic kernel. We set the optimal hyperparameters of the SVM as $C = 6.51$ and $\sigma = 4.17$ using aforementioned methods.

Acknowledgements

This work was supported by the Academy of Finland (grant 316258 to J.T. and grant 323385 to A.S.), and Biocenter Finland and University of Helsinki (I.B., E.J., and SBEM imaging). We would like to thank Maarit Pulkkinen for her help with the animal handling, and Mervi Lindman and Antti Salminen for help preparing the samples for SBEM. We would like to thank Andrea Behanova for developing gACSON, the segmentation evaluation software. The authors acknowledge CSC-IT Center for Science, Finland, for computational resources.

Author contributions

A.A., J.T., and A.S. conceived the project and designed the study. A.A. implemented the methods and performed the experiments. A.A., J.T., and A.S. analyzed the data. I.B. and E.J. contributed to electron microscopy imaging. A.A., J.T., and A.S. wrote the manuscript. All authors commented on and approved the final manuscript.

Competing interests

The authors declare that they have no conflict of interest.

Availability of data and material

The datasets used in the study will be available upon request to the corresponding author. The source code of DeepACSON will be available at https://github.com/aAbdz/DeepACSON.

References

1. Zheng, Z. *et al.* A Complete Electron Microscopy Volume of the Brain of Adult Drosophila melanogaster. *Cell* **174**, 730–743 (2018). URL https://doi.org/10.1016/j.cell.2018.06.019. DOI 10.1016/j.cell.2018.06.019.
2. Hildebrand, D. G. C. *et al.* Whole-brain serial-section electron microscopy in larval zebrafish. *Nat.* **545**, 345–349 (2017). URL http://dx.doi.org/10.1038/nature22356. DOI 10.1038/nature22356.

3. Berning, M., Boergens, K. M. & Helmstaedter, M. SegEM: Efficient Image Analysis for High-Resolution Connectomics. *Neuron* **87**, 1193–1206 (2015). DOI 10.1016/j.neuron.2015.09.003.

4. Abdollahzadeh, A., Belevich, I., Jokitalo, E., Tohka, J. & Sierra, A. Automated 3D Axonal Morphometry of White Matter. *Sci. Reports* **9**, 6084 (2019). DOI 10.1038/s41598-019-42648-2.

5. Dorkenwald, S. *et al.* Automated synaptic connectivity inference for volume electron microscopy. *Nat. Methods* **14**, 435–442 (2017). URL http://dx.doi.org/10.1038/nmeth.4206. DOI 10.1038/nmeth.4206.

6. Haberl, M. G. *et al.* CDeep3M—Plug-and-Play cloud-based deep learning for image segmentation. *Nat. Methods* **15**, 677–680 (2018). URL http://dx.doi.org/10.1038/s41592-018-0106-z. DOI 10.1038/s41592-018-0106-z.

7. Zeng, T., Wu, B. & Ji, S. DeepEM3D: approaching human-level performance on 3D anisotropic EM image segmentation. *Bioinforma. (Oxford, England)* **33**, 2555–2562 (2017). DOI 10.1093/bioinformatics/btx188.

8. Funke, J. *et al.* Large Scale Image Segmentation with Structured Loss Based Deep Learning for Connectome Reconstruction. *IEEE Transactions on Pattern Analysis Mach. Intell.* **41**, 1669–1680 (2019). DOI 10.1109/TPAMI.2018.2835450.

9. Januszewski, M. *et al.* High-precision automated reconstruction of neurons with flood-filling networks. *Nat. Methods* **15**, 605–610 (2018). URL http://dx.doi.org/10.1038/s41592-018-0049-4. DOI 10.1038/s41592-018-0049-4.

10. Meirovitch, Y. *et al.* A Multi-Pass Approach to Large-Scale Connectomics. *arXiv:1612.02120 [q-bio.QM]* (2016).

11. Meirovitch, Y. *et al.* Cross-Classification Clustering: An Efficient Multi-Object Tracking Technique for 3-D Instance Segmentation in Connectomics. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2019). URL http://arxiv.org/abs/1812.01157.

12. Abdollahzadeh, A., Sierra, A. & Tohka, J. Cylindrical Shape Decomposition Algorithm for 3D Segmentation. *arXiv:1911.00571 [cs.CV]* (2019).

13. Wang, L., Dong, J., Cull, G., Fortune, B. & Cioffi, G. A. Varicosities of Intraretinal Ganglion Cell Axons in Human and Nonhuman Primates. *Investig. Ophthalmol. & Vis. Sci.* **44**, 2–9 (2003). DOI 10.1167/iovs.02-0333.

14. McDonald, J. H. Handbook of Biological Statistics. *Sparky House Publ.* 291 (2009). DOI 10.1017/CBO9781107415324.004.

15. Funke, J., Andres, B., Hamprecht, F. A., Cardona, A. & Cook, M. Efficient automatic 3D-reconstruction of branching neurons from EM data. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 1004–1011 (IEEE, 2012). URL http://ieeexplore.ieee.org/document/6247777/. DOI 10.1109/CVPR.2012.6247777.

16. Nunez-Iglesias, J., Kennedy, R., Plaza, S. M., Chakraborty, A. & Katz, W. T. Graph-based Active Learning of Agglomeration (GALA): A python library to segment 2D and 3D neuroimages. *Front. Neuroinformatics* **8**, 1–6 (2014). DOI 10.3389/fninf.2014.00034.

17. Maitin-Shepard, J., Jain, V., Januszewski, M., Li, P. & Abbeel, P. Combinatorial energy learning for image segmentation. *Adv. Neural Inf. Process. Syst.* 1974–1982 (2016).

18. Falk, T. *et al.* U-Net: deep learning for cell counting, detection, and morphometry. *Nat. Methods* **16**, 67–70 (2019). DOI 10.1038/s41592-018-0261-2.

19. Schneider, C. A., Rasband, W. S. & Eliceiri, K. W. NIH Image to ImageJ: 25 years of image analysis. *Nat. Methods* **9**, 671–675 (2012). DOI 10.1038/nmeth.2089.

20. Wanner, A. A., Kirschmann, M. A. & Genoud, C. Challenges of microtome-based serial block-face scanning electron microscopy in neuroscience. *J. Microsc.* **259**, 137–142 (2015). DOI 10.1111/jmi.12244.

21. Mikula, S., Binding, J. & Denk, W. Staining and embedding the whole mouse brain for electron microscopy. *Nat. Methods* **9**, 1198–1201 (2012). DOI 10.1038/nmeth.2213.

22. Mierzwa, A. J., Marion, C. M., Sullivan, G. M., McDaniel, D. P. & Armstrong, R. C. Components of myelin damage and repair in the progression of white matter pathology after mild traumatic brain injury. *J Neuropathol Exp Neurol* **74**, 218–232 (2015). URL http://www.ncbi.nlm.nih.gov/pubmed/25668562. DOI 10.1097/NEN.0000000000000165.

23. Armstrong, R. C., Mierzwa, A. J., Marion, C. M. & Sullivan, G. M. White matter involvement after TBI: Clues to axon and myelin repair capacity. *Exp. Neurol.* **275**, 328–333 (2016). URL http://dx.doi.org/10.1016/j.expneurol.2015.02.011. DOI 10.1016/j.expneurol.2015.02.011.

24. Chen, X. H., Johnson, V. E., Uryu, K., Trojanowski, J. Q. & Smith, D. H. A lack of amyloid β plaques despite persistent accumulation of amyloid β in axons of long-term survivors of traumatic brain injury. *Brain Pathol.* **19**, 214–223 (2009). DOI 10.1111/j.1750-3639.2008.00176.x.

25. Johnson, V. E., Stewart, W. & Smith, D. H. Axonal pathology in traumatic brain injury. *Exp. Neurol.* **246**, 35–43 (2013). URL http://dx.doi.org/10.1016/j.expneurol.2012.01.013. DOI 10.1016/j.expneurol.2012.01.013.

26. Rodriguez-Paez, A. C., Brunschwig, J. P. & Bramlett, H. M. Light and electron microscopic assessment of progressive atrophy following moderate traumatic brain injury in the rat. *Acta Neuropathol.* **109**, 603–616 (2005). DOI 10.1007/s00401-005-1010-z.

27. Tang-Schomer, M. D., Patel, A. R., Baas, P. W. & Smith, D. H. Mechanical breaking of microtubules in axons during dynamic stretch injury underlies delayed elasticity, microtubule disassembly, and axon degeneration. *The FASEB J.* **24**, 1401–1410 (2010). URL http://www.ncbi.nlm.nih.gov/pubmed/20019243. DOI 10.1096/fj.09-142844.

28. Kamiya, K. *et al.* Diffusion imaging of reversible and irreversible microstructural changes within the corticospinal tract in idiopathic normal pressure hydrocephalus. *NeuroImage: Clin.* **14**, 663–671 (2017). URL http://dx.doi.org/10.1016/j.nicl.2017.03.003. DOI 10.1016/j.nicl.2017.03.003.

29. Alexander, D. C. *et al.* Orientationally invariant indices of axon diameter and density from diffusion MRI. *NeuroImage* **52**, 1374–1389 (2010). URL http://dx.doi.org/10.1016/j.neuroimage.2010.05.043. DOI 10.1016/j.neuroimage.2010.05.043.

30. Salo, R. A. *et al.* Quantification of anisotropy and orientation in 3D electron microscopy and diffusion tensor imaging in injured rat brain. *NeuroImage* **172**, 404–414 (2018). DOI 10.1016/j.neuroimage.2018.01.087.

31. Lee, H. *et al.* Along-axon diameter variation and axonal orientation dispersion revealed with 3D electron microscopy: implications for quantifying brain white matter microstructure with histology and diffusion MRI. *Brain Struct. Funct.* (2019). URL http://link.springer.com/10.1007/s00429-019-01844-6. DOI 10.1007/s00429-019-01844-6.

32. Palombo, M., Alexander, D. C. & Zhang, H. A generative model of realistic brain cells with application to numerical simulation of the diffusion-weighted MR signal. *NeuroImage* **188**, 391–402 (2019). URL https://doi.org/10.1016/j.neuroimage.2018.12.025. DOI 10.1016/j.neuroimage.2018.12.025.

33. Ginsburger, K. *et al.* MEDUSA: A GPU-based tool to create realistic phantoms of the brain microstructure using tiny spheres. *NeuroImage* **193**, 10–24 (2019). DOI 10.1016/j.neuroimage.2019.02.055.

34. Chomiak, T. & Hu, B. What is the optimal value of the g-ratio for myelinated fibers in the rat CNS? A theoretical approach. *PLoS ONE* **4** (2009). DOI 10.1371/journal.pone.0007754.

35. Kwong, M. T. *et al.* 3D finite element formulation for mechanical–electrophysiological coupling in axonopathy. *Comput. Methods Appl. Mech. Eng.* **346**, 1025–1050 (2019). DOI 10.1016/j.cma.2018.09.006.

36. Kharatishvili, I., Nissinen, J. P., McIntosh, T. K. & Pitkänen, A. A model of posttraumatic epilepsy induced by lateral fluid-percussion brain injury in rats. *Neurosci.* **140**, 685–697 (2006). DOI 10.1016/j.neuroscience.2006.03.012.

37. Deerinck, T. *et al.* Enhancing serial block-face scanning electron microscopy to enable high resolution 3-D nanohistology of cells and tissues. *Microsc. Microanal.* **16**, 1138–1139 (2010). DOI 10.1017/S14319276100.

38. Belevich, I., Joensuu, M., Kumar, D., Vihinen, H. & Jokitalo, E. Microscopy Image Browser: A platform for segmentation and analysis of multidimensional datasets. *PLoS Biol.* **14**, 1–13 (2016). DOI 10.1371/journal.pbio.1002340.

39. Maggioni, M., Katkovnik, V., Egiazarian, K. & Foi, A. Nonlocal Transform-Domain Filter for Volumetric Data Denoising and Reconstruction. *IEEE Transactions on Image Process.* **22**, 119–133 (2013). URL http://ieeexplore.ieee.org/document/6253256/. DOI 10.1109/TIP.2012.2210725.

40. Soderholm, J., Bevis, B. J. & Glick, B. S. Vector for pop-in/pop-out gene replacement in Pichia pastoris. *BioTechniques* **31**, 306–312 (2001). DOI 10.2144/000113156.

41. Giusti, A., Cireşan, D. C., Masci, J., Gambardella, L. M. & Schmidhuber, J. Fast image scanning with deep max-pooling convolutional neural networks. *2013 IEEE Int. Conf. on Image Process. ICIP 2013 - Proc.* 4034–4038 (2013). DOI 10.1109/ICIP.2013.6738831.

42. Kingma, R. & Prodanova, K. ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION. *AIP Conf. Proc.* **1631**, 58–62 (2014). DOI 10.1063/1.4902458.

43. Sabry Hassouna, M. & Farag, A. A. Robust centerline extraction framework using level sets. *Proc. - 2005 IEEE Comput. Soc. Conf. on Comput. Vis. Pattern Recognition, CVPR 2005* **I**, 458–467 (2005). DOI 10.1109/CVPR.2005.306.

44. Sethian, J. A. A fast marching level set method for monotonically advancing fronts. *Proc. Natl. Acad. Sci.* **93**, 1591–1595 (1996). DOI 10.1073/pnas.93.4.1591.

45. Osher, S. J. Fronts Propagating with Curvature Dependent Speed. *Comput. Phys.* **79**, 1–5 (1988). DOI 10.1007/s13398-014-0173-7.2.

46. Frangi, A. F., Niessen, W. J., Vincken, K. L. & Viergever, M. A. Multiscale vessel enhancement filtering. In Wells, W., Colchester, A. & Delp, S. (eds.) *Medical Image Computing and Computer-Assisted Intervention — MICCAI'98. MICCAI 1998. Lecture Notes in Computer Science*, 130–137 (Springer, Berlin, Heidelberg, 1998). URL http://link.springer.com/10.1007/BFb0056195. DOI 10.1007/BFb0056195.

47. Mockus, J., Tiesis, V. & Zilinskas, A. The application of Bayesian methods for seeking the extremum. *Towards Glob. Optimisation* **2**, 117–129 (1978).

48. Snoek, J., Larochelle, H. & Adams, R. P. Practical Bayesian Optimization of Machine Learning Algorithms. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'12, 2951–2959 (Curran Associates Inc., USA, 2012). URL http://arxiv.org/abs/1206.2944.

Supplementary Information

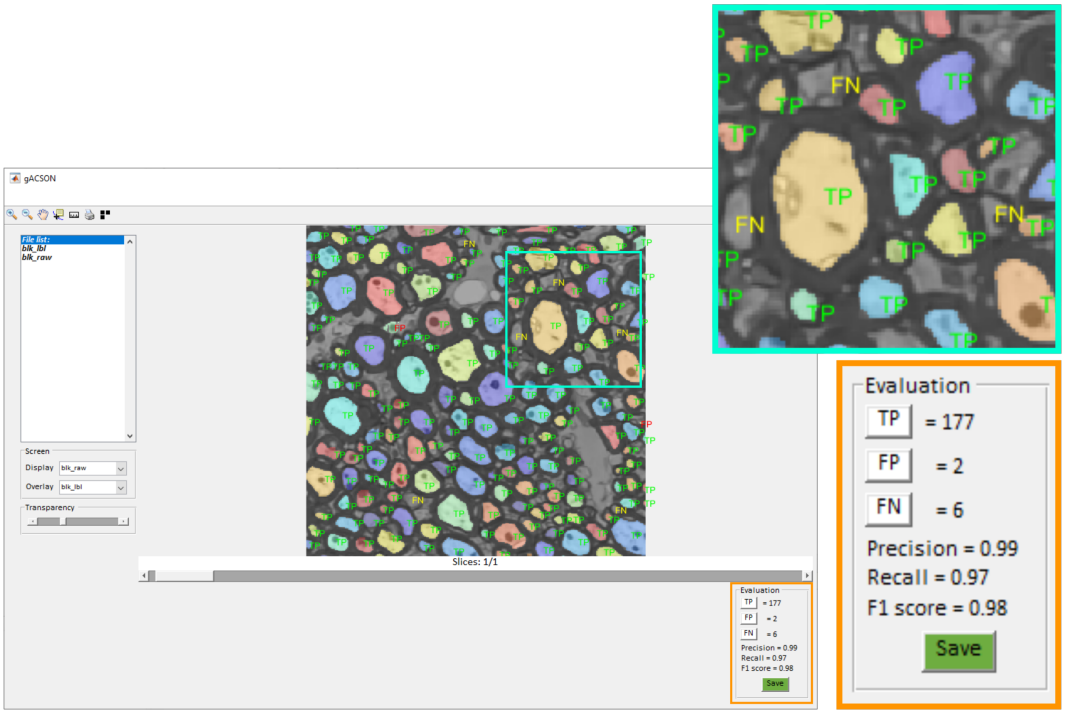

Figure S1. DeepACSON evaluation. We developed a GUI-based software tool, gACSON, in Matlab to load and visualize the segmentation for proofreading. gACSON is designed for the visualization of the large-scale image-datasets/segmentation. Using gACSON, an expert evaluated the DeepACSON segmentation of myelinated axons and mitochondria at the object-level. We randomly sampled each low-resolution dataset and its corresponding segmentation by non-overlapping images of size 300×300 voxels. The sampled images were quantified for the number of true-positives (TP), false-positives (FP), and false-negatives (FN) to calculate the precision, recall, and F1 score. The expert had no access to the dataset ID nor the sampling location.

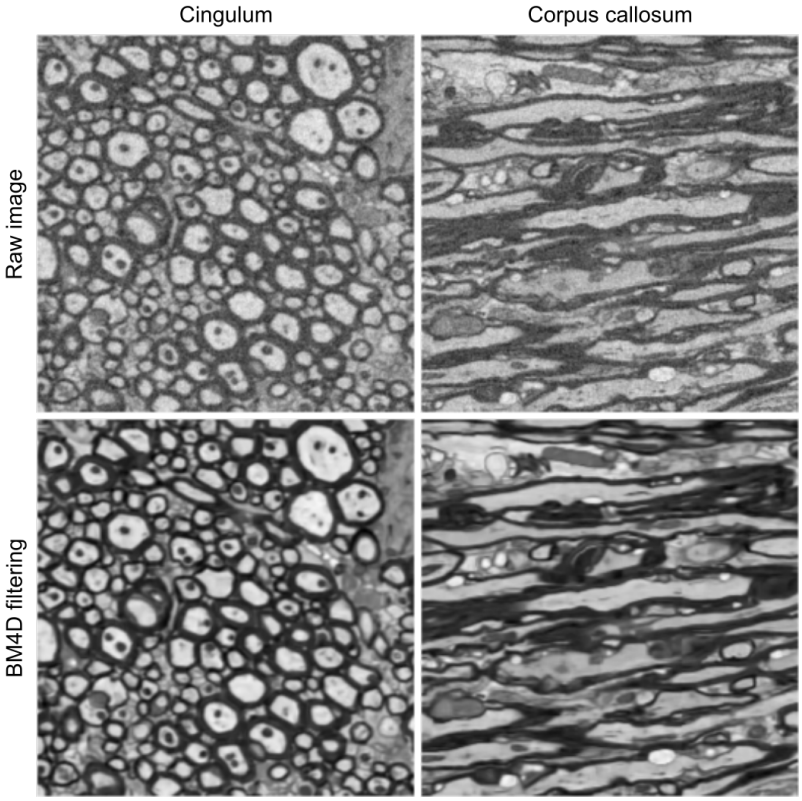

Figure S2. BM4D filtering of the low-resolution SBEM datasets. BM4D recognized the noise as Gaussian-distributed with the standard deviation in the range [17, 22] in our low-resolution SBEM datasets. Shown images were acquired from the cingulum and corpus callosum of sham #25 low-resolution dataset.

Figure S3. The architecture of DCNNs used in DeepACSON. We used the same architecture for DCNN-mAx and DCNN-cN. The size of the convolutional kernels is denoted as $Conv(x,y,z)$. The number of channels/feature maps created from a layer of convolutional kernels is denoted by $@n$. The size of the max pooling operation is denoted as $Max\,pool(x,y,z)$.

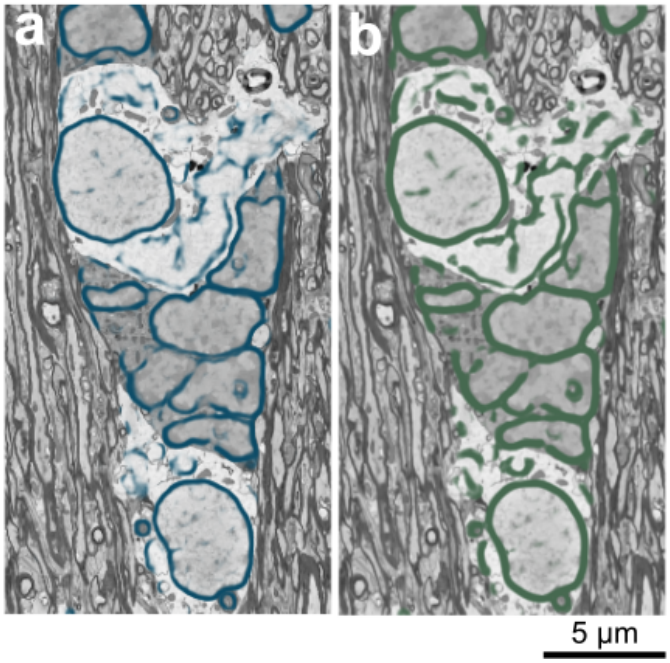

Figure S4. Frangi filtering the probability map of the membrane of cell nuclei. (**a**) The probability map of the membrane of cell nuclei returned by DCNN-cN. (**b**) Application of Frangi filtering on (**a**).

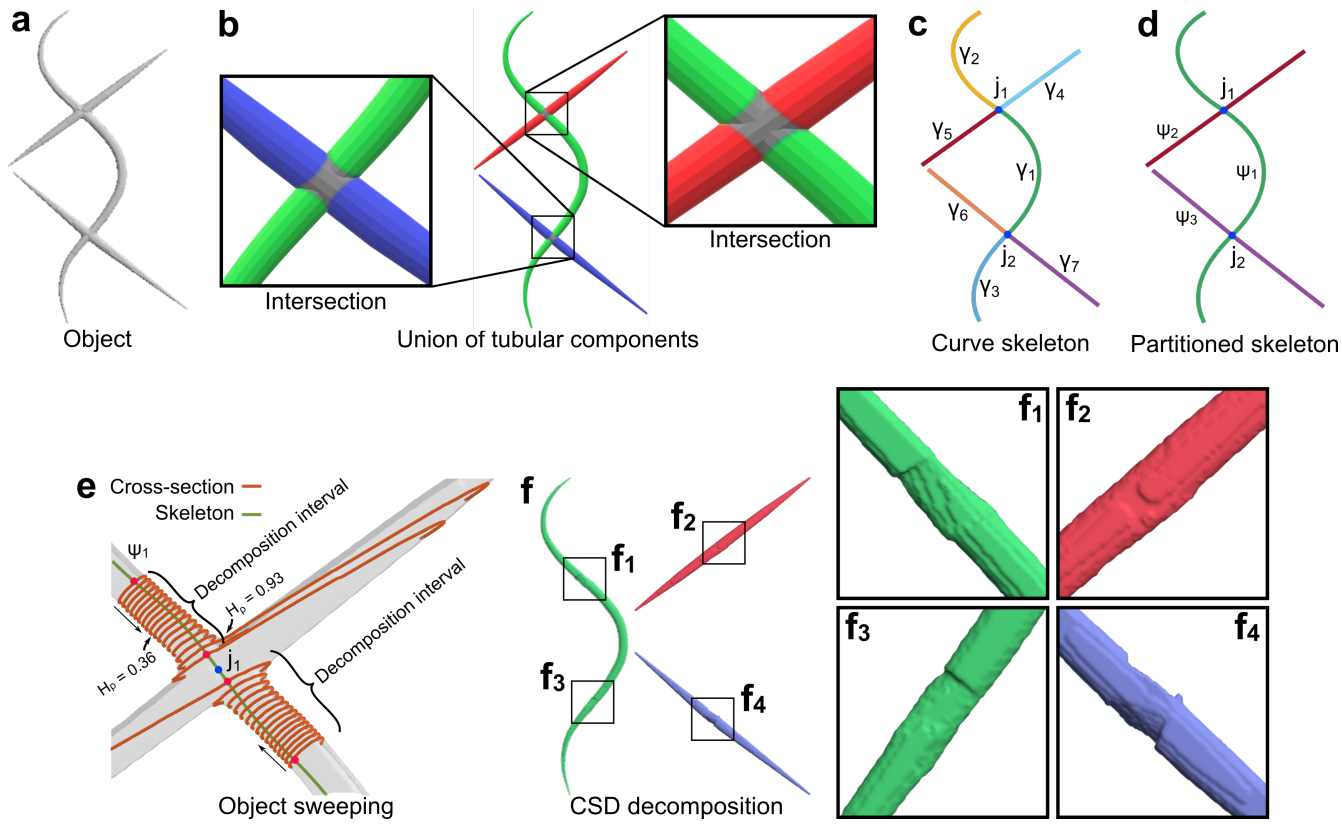

Figure S5. Cylindrical shape decomposition algorithm. (**a**) An object is a union of several semantic-components. (**b**) The semantic-components of the object in (**a**) are color-coded. Intersections of the semantic-components are colored grey. (**c**) The curve skeleton of the synthetic object in (**a**) is the union of all skeleton branches. Skeleton branches are color-coded and denoted as γ. We defined a skeleton junction-point j as such a point that skeleton branches connect. Junction-points are shown as blue filled-circles. (**d**) We found m maximal length sub-skeletons denoted as ψ via minimizing an orientation cost-function. The sub-skeletons are color-coded. (**e**) On a sub-skeleton ψ and in the proximity of a junction-point $j \in \psi$, we defined two decomposition intervals. The boundaries of decomposition intervals are shown with red filled-circles. In each interval, the cross-section of the object was swept along ψ and towards the joint j to find a critical point. At a critical point, the normalized Hausdorff distance H_ρ between a cross-sectional contour and the mean of visited cross-sectional contours exceeds θ_H. Sweeping directions are shown with arrows. (**f**) We cut the object at critical points to obtain object-parts. The object-parts along the same sub-skeleton were assigned the same label to construct a semantic-component. The semantic-components were further reconstructed between their comprising object-parts using generalized cylinders, magnified in (**f$_1$-f$_4$**). The synthetic object in (**a**) comprised of seven object-parts, and our algorithm decomposed it into three semantic-components.

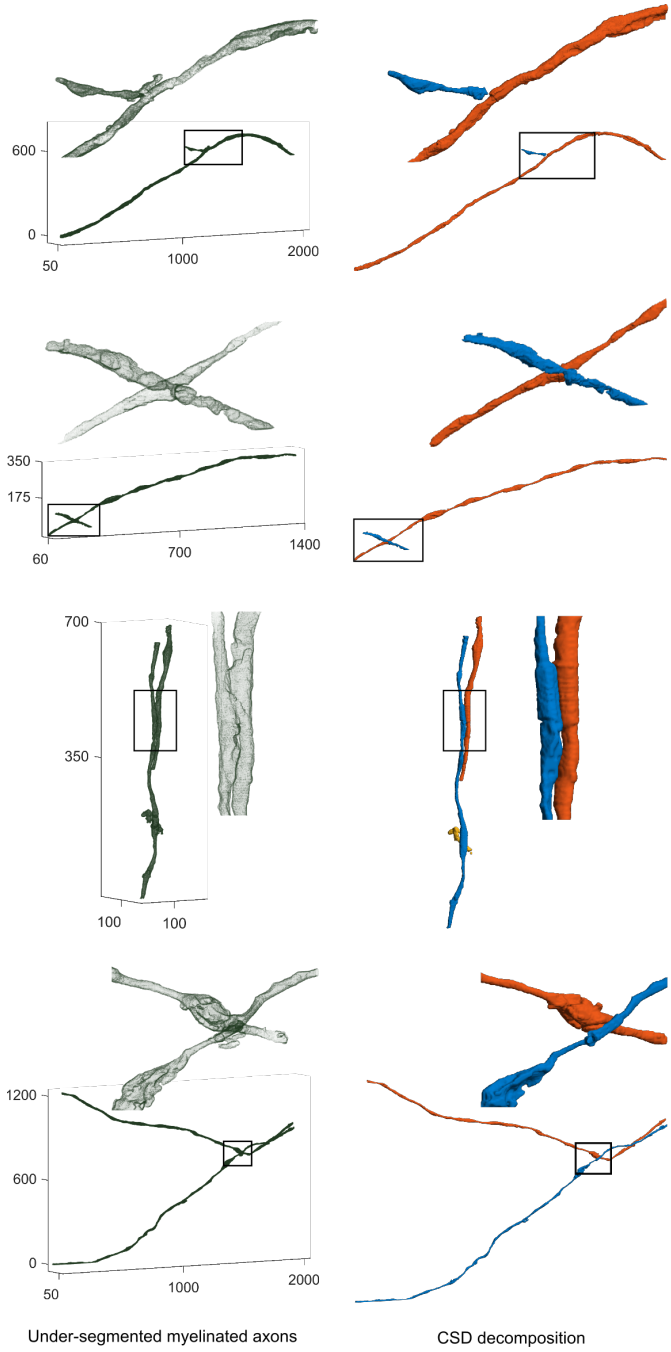

Figure S6. Decomposition of under-segmented myelinated axons into their semantic axonal components using the CSD algorithm.

Table S1. Characteristics of low-resolution SBEM datasets. For each rat, we collected SBEM images with $50 \times 50 \times 50$ nm^3 voxel size from the ipsi- and contralateral corpus callosum and cingulum. The ipsilateral dataset of sham #49 included only the cingulum.

Treatment	Rat ID	Size in voxels (x, y, z)
Sham	#25 contra	$2\,044 \times 4\,096 \times 1\,306$
	#25 ipsi	$4\,096 \times 2\,048 \times 1\,384$
	#49 contra	$4\,096 \times 2\,048 \times 1\,882$
	#49 ipsi	$2\,048 \times 2\,048 \times 1\,210$
TBI	#2 contra	$4\,096 \times 2\,048 \times 1\,086$
	#2 ipsi	$2\,154 \times 4\,134 \times 620$
	#24 contra	$4\,091 \times 2\,028 \times 1\,290$
	#24 ipsi	$2\,946 \times 2\,162 \times 1\,250$
	#28 contra	$4\,096 \times 2\,048 \times 1\,278$
	#28 ipsi	$4\,075 \times 2\,000 \times 1\,300$

Table S2. DeepACSON evaluation scores. The performance of DCNN-mAx was evaluated against the automated ACSON pipeline. The performance of SVMs was evaluated using leave-one-group-out (LOGO) cross-validation (CV). An expert evaluated the final segmentation of myelinated axons (mAxons) and mitochondria. The maximum value of all scores is one.

Evaluation	Structure	dataset	Precision	Recall	F1 score
DCNN-mAx (versus ACSON)	Myelin	sham #25 ipsi	0.931	0.894	0.912
		sham #49 contra	0.746	0.840	0.790
		TBI #2 contra	0.937	0.849	0.891
		TBI #24 contra	0.947	0.882	0.913
		TBI #24 ipsi	0.947	0.898	0.922
		TBI #28 ipsi	0.904	0.877	0.890
DCNN-mAx (versus ACSON)	mAxons	sham #25 ipsi	0.946	0.813	0.875
		sham #49 contra	0.910	0.719	0.803
		TBI #2 contra	0.938	0.805	0.866
		TBI #24 contra	0.893	0.895	0.894
		TBI #24 ipsi	0.893	0.897	0.895
		TBI #28 ipsi	0.931	0.751	0.831
SVM (LOGO CV)	mAxons	Sham	0.988	0.988	0.988
		TBI	0.920	0.991	0.955
SVM (LOGO CV)	Cell nuclei	Sham	0.961	0.977	0.969
		TBI	0.964	1	0.981
Final segmentation (expert)	mAxons	samples of all datasets	0.9647 ± 0.0267	0.8773 ± 0.0607	0.9177 ± 0.0384
	Mitochondria		0.856 ± 0.100	0.804 ± 0.091	0.823 ± 0.067

Table S3. Volumetry of ultrastructures. The volume of myelin and myelinated axons was expressed as a percentage of the corresponding SBEM dataset. The volume fraction in which a cell body/process occupies varies between datasets. Therefore, a direct comparison of the volumes is not reasonable.

Treatment	Tissue	Rat ID	Myelin (%)	mAxons (%)	Cell nuclei	mAxons	Mitochondria
Sham	Cc	#25 contra	48.74	22.24	218	42 318	168 512
		#25 ipsi	49.07	23.17	214	43 209	175 508
		#49 contra	43.53	23.39	161	30 723	116 123
		#49 ipsi	-	-	-	-	-
TBI	Cc	#2 contra	46.42	19.56	133	25 865	91 997
		#2 ipsi	44.13	19.95	124	29 883	58 089
		#24 contra	51.38	19.40	221	49 866	172 241
		#24 ipsi	52.20	22.94	102	23 256	96 032
		#28 contra	42.32	19.31	226	34 804	104 102
		#28 ipsi	40.16	18.61	213	35 553	114 204
Sham	Cg	#25 contra	52.61	29.14	90	16 076	58 583
		#25 ipsi	49.72	28.58	97	15 868	62 838
		#49 contra	50.18	28.19	70	10 537	55 825
		#49 ipsi	48.42	26.49	126	16 932	71 742
TBI	Cg	#2 contra	49.14	22.76	120	23 039	97 777
		#2 ipsi	42.98	12.19	100	14 094	21 722
		#24 contra	49.32	21.98	76	16 495	58 180
		#24 ipsi	41.26	14.72	108	20 374	63 798
		#28 contra	42.68	22.89	131	17 510	77 116
		#28 ipsi	39.79	16.58	167	18 297	48 830

Computation time

Table S4. Computation time. M1: Intel Core i7 6700 CPU 3.4 GHz with 64 GB RAM. M2: 2 $\times$ Intel Xeon E5 2630 CPU 2.4 GHz machine with 512 GB RAM. M3: NVIDIA Tesla P100-16 GB GPU. Because the size of datasets and the number of instances segmented in each dataset are different, we measured the computation time based on the sham #25 dataset in the last column.

Process	Machine	Time	Time/dataset (h)
BM4D filtering	M1 - MATLAB R2017b	0.056 MB/s	54.23
Training DCNN-mAx	M3 - Python 2.7	24 h	-
Training DCNN-cN	M3 - Python 2.7	24 h	-
DCNN-mAx inference	M3 - Python 2.7	0.297 MB/s	10.23
DCNN-cN inference	M3 - Python 2.7	0.299 MB/s	10.16
CSD	M2 - Python 2.7	$\sim$ 113.8 s/myelinated axon	44.66
Myelinated Axon feature extraction	M2 - MATLAB R2017b	$\sim$ 15 s/myelinated axon	5.89
2D Frangi filtering	M1 - MATLAB R2017b	0.267 MB/s	11.38
Cell nucleus feature extraction	M1 - MATLAB R2017b	$\sim$ 30 s/nucleus	5.76
SVM Bayesian optimization	M1 - MATLAB R2017b	173 s	-
SVM inference	M1 - MATLAB R2017b	$\sim$ 4.4 µs/component	5.14×10^{-5}