AxonDeep: Automated Optic Nerve Axon Segmentation in Mice with Deep Learning

Wenxiang Deng[a,b*], Adam Hedberg-Buenz[b,c*], Dana A. Soukup[b,c], Sima Taghizadeh[a], Michael G. Anderson[b,c, d**], Mona K. Garvin[a,b**]

[*] Contributed equally (in first-author role)

[**]Contributed equally (in senior-author role)

From the [a]Dept. of Electrical and Computer Engineering, University of Iowa, Iowa City, IA; [b]Iowa City VA Center for the Prevention and Treatment of Visual Loss, Iowa City VA Health Care System, Iowa City, IA; [c]Dept. of Molecular Physiology and Biophysics, University of Iowa, Iowa City IA; [d]Dept. of Ophthalmology and Visual Sciences, University of Iowa, Iowa City, IA

This report is supported in part by Grants I50 RX003002 (WD, AHB, MGA, MKG); T32DK112751 (AHB); P30 EY025580, I01 RX001481, and R21 EY029991 (MGA)

Disclosure: **W. Deng**, None; **D.A. Soukup**, None; **A. Hedberg-Buenz**, None; **M.G. Anderson,** None, **M.K. Garvin**, None.

Keywords: retinal ganglion cell, axon segmentation, deep learning, semi-supervised learning, mice

Corresponding author: Mona K. Garvin, Ph.D., 4318 Seamans Center for the Engineering Arts and Sciences, University of Iowa, Iowa City, IA 52242, (319) 335-5950, mona-garvin@uiowa.edu.

Word count (manuscript text): 4916

Abstract

Purpose: Optic nerve damage is the principal feature of glaucoma and contributes to vision loss in many diseases. In animal models, nerve health has traditionally been assessed by human experts that grade damage qualitatively or manually quantify axons from sampling limited areas from histologic cross sections of nerve. Both approaches are prone to variability and time consuming. Automated approaches have begun to emerge, but shortcomings have limited wide-spread application. Here, we seek improvements through use of deep-learning approaches for segmenting and quantifying axons from cross sections of mouse optic nerve.

Methods: Two deep-learning approaches were developed and evaluated: (1) a traditional supervised approach using a fully convolutional network trained with only labeled data and (2) a semi-supervised approach trained with both labeled and unlabeled data using a generative-adversarial-network framework.

Results: From comparisons with an independent test set of images with manually marked axon centers and boundaries, both deep-learning approaches performed above an existing baseline automated approach and similarly to two independent experts. Performance of the semi-supervised approach was superior and implemented into AxonDeep.

Conclusion: AxonDeep performs automated quantification and segmentation of axons similar to that of experts without the time- and labor-constraints associated with manual performance. The quantitative and objective nature of AxonDeep reduces variability arising from differences in model, methodology, and user that often compromise manual performance of these tasks.

Translational Relevance: Use of deep learning for axon quantification provides rapid, objective, and higher throughput analysis of optic nerve that would otherwise not be possible.

Introduction

Retinal ganglion cell (RGC) loss is the primary feature of glaucoma, a leading cause of irreversible vision loss.[1, 2] RGCs are also damaged as a part of several other diseases, including forms of traumatic brain injury (TBI),[3, 4] diabetes,[5] multiple sclerosis,[6, 7] and Alzheimer's disease,[8] among others.[9] To evaluate RGC damage, the two fundamental options are to quantify RGC soma in the retina or RGC axons in the optic nerve, which in health typically have a 1:1 relationship. Because axon damage can occur earlier than somal loss,[10, 11] there are many situations in which it is useful to quantify both. While the identification of RGC-specific markers has helped advance techniques for quantification of RGC soma,[12, 13] techniques for quantification of RGC axons, which are smaller and more difficult to label, have lagged. Manual counting of axons has traditionally been adopted by human experts.[14-18] However, manually counting is not only time-consuming, but prone to vary between different experts. Grading of optic nerve appearance based on pathological features has been useful for making some advances,[19-24] but is even more subjective and prone to variation.

First-generation automated approaches for quantifying axons, such as AxonJ[25] and AxonMaster[26, 27] have some known limitations and remain rarely utilized. For example, AxonJ, which members of our group were involved in developing, was designed to recognize axons only in healthy, not diseased, optic nerves. It can be expected that deep-learning approaches, especially given their success in numerous medical application domains[28-33] ultimately would be more robust. In fact, recently, the deep-learning approach AxoNet[34] has been proposed for providing axon counts; however, this approach still does not provide

direct segmentation of the axons and thus the ability to compute additional potential quantitative measures (such as area distributions) using this approach is limited.

In the present work, we propose a deep-learning approach, named AxonDeep, for the segmentation of optic nerve axons in murine tissue. This work was motivated by the need to overcome some of the known limitations of first-generation approaches[25] and a desire to obtain complete segmentations (allowing quantitative measures of multiple axon measurements, beyond just counts) that are not possible with the current tools.[34] Our deep-learning architecture is based on recent work on asymmetric network structures whereby a deep encoder combined with a light-weight decoder can provide an improved performance over symmetric deep-learning architectures for image segmentation problems with more complex scenes.[23, 27] In developing AxonDeep, two deep-learning approaches were evaluated: (1) a traditional supervised approach using a fully convolutional network trained with only labeled data and (2) a semi-supervised approach trained with both labeled and unlabeled data using a generative-adversarial-network framework. Utilizing the semi-supervised approach to be able to also take advantage of unlabeled data, in addition to labeled data, during training was motivated by a need to help address the challenges associated with the manual effort required to generate training sets. Both deep-learning approaches were compared to the AxonJ approach, with the semi-supervised approach found to have the better overall performance. Thus, we introduce AxonDeep as a deep-learning-based axon segmentation tool trained based upon a semi-supervised approach.

Methods

Procurement and preparation of mouse optic nerve specimens for image analysis

Mice. Optic nerves (*n*=56 nerves, 1 nerve from each of 56 mice) were collected from three different mouse strains modeling various presentations of healthy and diseased optic nerves: DBA/2J with various degrees of an inherited age-related form of glaucoma (*n*=16 nerves);[35, 36] C57BL/6J that had been subjected to either blast injury within an enclosed chamber or sham treatment (*n*=27 nerves);[3, 4, 22] and Diversity Outbred (J:DO) with healthy optic nerves but predicted to exhibit genetic background-dependent natural variability in optic nerve features (*n*=13 nerves).[37, 38] Tissues in the current study were collected from mice contributing to prior publications,[39, 40] however all data reported herein arise from new analyses performed uniquely for this study. All mice used in this study were originally purchased from The Jackson Laboratory (Bar Harbor, ME). All animals were treated in accordance with the ARVO Statement for the Use of Animals in Ophthalmic and Vision Research. All experimental protocols were approved by the Institutional Animal Care and Use Committee of the University of Iowa.

Nerve processing. Nerves were processed for histology as previously described.[17, 25, 41] In brief, mice were euthanized by carbon dioxide inhalation with subsequent decapitation. Heads were collected and the skulls opened before fixation in half-strength Karnovsky's fixative (2% paraformaldehyde, 2.5% glutaraldehyde in 0.1 M sodium cacodylate) at 4°C for 16 hours. Optic nerves were dissected from brains and drop fixed in the same fixative for an additional 16 hours at 4°C. Nerves were stained with 1% osmium tetroxide, dehydrated in graded acetone (30%-100%), infiltrated and embedded in resin (Eponate-12; Ted Pella,

Redding, CA), and polymerized in a 65°C oven. Semithin (1-μm) cross sections were cut, transferred to glass slides, stained with 1% paraphenylenediamine (PPD), and mounted.

Procurement of images for training, validation, and testing of the network

Light micrographs (physical dimensions: 90.2 x 67.5 μm; resolution: 4140 x 3096 px) were collected from stained optic nerve cross sections at a total magnification of 1000X using identical camera settings, as previously described.[17, 25, 41] In brief, light micrographs were acquired from representative and non-overlapping fields from one cross-section of each optic nerve using an upright light microscope (BX52, Olympus) equipped with a CCD camera (DP-72, Olympus).

Separation of images for purposes of training/validation and testing of the deep-learning networks. Optic nerve images were divided into two major sub-divisions used for different purposes, a "training/validation" set and a "testing" set (Figure 1). To ensure an equal distribution of nerves with comparable levels of damage between the two sets, a qualitative damage grade was first assigned to each nerve by consensus among a panel of three independent graders (1 = mild or no damage, 2 = moderate, and 3 = severe), as previously described.[14] Among the 56 nerves, images from 38 nerves (1-2 images per nerve from 38 mice) were assigned to the training/validation set (18 grade-1 nerves; 18 grade-2 nerves; 2 grade-3 nerves) and images from the remaining 18 nerves (1 image per nerve from of 18 mice) were assigned to the test set (9 grade-1 nerves; 9 grade-2 nerves). Because of potential challenges in segmenting (manually or automatically) severely damaged nerves, the two grade-3 nerves were assigned to the training set (and only used as additional unlabeled examples in the semi-supervised approach). The remaining 36 mild and moderate nerves in the combined training/validation set were further randomly divided into 26

nerves to be used for training and 10 nerves to be used for validation. Note that, as is standard practice with deep-learning techniques, the training process was used for automatically determining the trainable network weights, whereas the validation process was used for deciding hyperparameters and tuning the network. The random division of the 36 mild/moderate nerves into training and validation sets resulted in 17 grade-1 nerves and 9 grade-2 nerves in the training set, and 7 grade-1 nerves and 3 grade-2 nerves with moderate damage in the validation set. A reference segmentation was manually obtained on a 1024x1024 sub-image as described in the next section on each of the 36 mild/moderate nerves allocated to the training/validation sets (6762 total axons on the 26 training images and 2668 total axons on the 10 validation images); however, all 50 available 4140x3096 full-sized images (in the training set) were also used as unlabeled images (with more than 150,000 axons total) for helping to train the semi-supervised approach (see semi-supervised approach).

Obtaining reference axon segmentations to be used for training and validation

As our deep-learning networks provide pixel-based marking of axons, our reference standard to train the network and optimize hyperparameters needed to include complete segmentations of the axons. In other words, we needed to obtain binary axon masks (white = axon pixels; black = non-axon pixels) alongside the original images to train the networks. The supervised network required all input images for training to have complete segmentation information and the semi-supervised network still required complete segmentation information for a subset of images. Because obtaining completely traced axons from scratch is labor intensive (even for obtaining the subset of complete segmentations needed for the semi-supervised approach), for training purposes, our strategy to obtain

complete segmentations involved manually marking axon centers in combination with manually correcting the boundaries of an automated segmentation. (Note that, as discussed later, our strategy for obtaining complete tracings for evaluation purposes on the test set did not involve an automated step, as was used in the training stage, to avoid any bias associated with involving an automated segmentation.)

More specifically, obtaining the reference tracings to be used for training involved the following. From the 4140x3096 image(s) available per nerve for training purposes, we first randomly selected one of the images and cropped a sub-image of size 1024x1024 (location selected at random) for purposes of obtaining complete segmentation tracings. On each of these 1024x1024 sub-images, we ran the baseline AxonJ approach followed by additional morphological smoothing processes to obtain a smooth starting segmentation for purposes of manual editing. Figure 2(B) shows an example segmentation of the axon image in Figure 2(A). In addition, we independently obtained manual center-point markings by placing a center point at an approximate center of each axon (dead/dying axons are marked separately, but not used in this work), as shown in Figure 2(C).

Next, the border of each segmented axon was detected,[42, 43] and the contours were converted into representative keypoints. Here we used a simple heuristic approach to generate the points. More specifically, the Ramer Douglas Peucker (RDP) algorithm[44] was used to generate a keypoint representation. This greedy algorithm iteratively found a polyline that is close to the contour we segmented, with the maximum distance from the interpolated line between keypoints and the true edge was less than a predetermined distance ε. Here we used an ε of 5.0 pixels. These keypoints were then visualized with a GUI interface, as shown in Figure 2(D).

We next combined the results from the manually marked centers (Figure 2(C)) and smoothed AxonJ contours as follows: 1. False positives (i.e., axons segmented by AxonJ, but not marked by the human expert) were excluded from the mappings; 2. false negatives (i.e., axons not segmented by AxonJ) were annotated with a small starting circle. An example pruned map can be seen in Figure 2(E). Finally, the contours were manually corrected by changing the locations (marked in red) of the interactive keypoints shown in Figure 2(E) and (F).

Obtaining reference axon counts and segmentations to be used for final evaluation (testing)

For the final one-time evaluation of each trained approach on the test set, we also obtained reference manual axon counts and complete segmentations. However, unlike in training, in obtaining complete segmentations we traced the boundaries of the axons from scratch (i.e. not editing an automated result) to avoid any bias associated with use of an automated approach. More specifically, for each of the test images, after obtaining a random 1024x1024 crop, as shown in Figure 3(A), approximated axon centers within the bounding box were manually marked to evaluate the axon count predictions. The axons close (less than 120 pixels) to image borders were ignored in the evaluation because it is hard to accurately assess the axons when part of them are outside the border. For the pixel-based evaluation of the axon segmentations, a randomly chosen 400×400 sized cropped sub-image was used for complete tracing by the same expert (rather than editing an existing segmentation as was done for training). Figure 3(B) shows an example selection of the traced area in the bounding box.

Deep-learning approach 1: Fully convolutional network (FCN) for segmentation

Figure 4 provides an illustration of the architecture of our fully convolution network (FCN) used to provide a pixel-level segmentation of the axons in the image. Overall, it uses an encoder-decoder framework with use of an established encoder (a deep ResNeXt-50[45] in our case) in combination with a more lightweight, asymmetric (compared to the encoder) decoder. This type of architecture is an example of a feature pyramid network (FPN) and has been shown to be successful in image-segmentation and detection tasks.[23, 46-48] For the encoder of the network, we used a deep ResNeXt-50[45] architecture, which uses a very similar structure as ResNet[49] while offering better performance. Use of a deep encoder was motivated by the need to consider the context of a relatively large surrounding region in determining whether a pixel belongs to an axon. Note the asymmetry of the encoder-decoder is different from the symmetric structure used in the popular U-Net framework popular in many pixel-based medical segmentation tasks.[23]

For the training data, in order to minimize the chance of adjacent axons being segmented as a single object, we separately predicted the axons and the borders between adjacent axons with two channels as output (as shown in Figure 4 and Figure 5(B)(C)). To generate the borders for training, we performed a morphological dilation on each manually corrected binary axon segmentation and found its overlap with the rest of the morphologically dilated axons. This overlap was defined as the border between adjacent axons (shown in Figure 5(C)). Note that at test time, these segmented borders can be used to better separate attached axon segmentation masks. As a result, a single channel of axon image with resolution divisible by 32 was used as input, and the network output consisted of two channels: an axon mask and the borders between nearby axons.

In order to train the network, similar to an approach used previously [50], at each training step, the total loss was a function combining a soft Dice loss (L_{Dice}) and binary cross entropy loss (L_{BCE}):

$L_{FCN} = L_{BCE} - \log L_{Dice}$, where

$L_{BCE} = -\sum_i (y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot \log(1 - \hat{y}_i))$ and

$L_{Dice} = \frac{Intersect}{Union} = \frac{2 \sum_i y_i \cdot \hat{y}_i + \alpha}{\sum_i y_i + \sum_i \hat{y}_i + \alpha}$, where

y_i and $\hat{y}_i$ are corresponding truth and prediction at each pixel location i. Here $\alpha = 1$ provides numerical stability and prevents cases of zeros.

At inference time, to generate the final output of the axon segmentation, a watershed algorithm[51] was applied with axon segmentations minus adjacent borders as basins and axon segmentations as masks.[52]

Deep-learning approach 2: Semi-supervised learning

To be able to take advantage of both labeled (time-consuming to acquire) and unlabeled data (i.e., just the input images without any manual annotations) during training, we also used a semi-supervised learning approach based on incorporating the FCN architecture described above into a generative-adversarial-network (GAN) setup. In a traditional GAN,[53] a generator subnetwork G (designed to generate realistic images from noise) is simultaneously trained with a discriminator network D (designed to differentiate between real images and those generated by the generator). Because the subnetworks are trained together (in an alternating fashion), the networks "compete" in a minimax game so that ultimately the generator subnetwork can generate realistic images on its own (in order to fool the

discriminator to try to win the competition). In other words, the addition of the discriminator subnetwork is used to help in the *training* of the generator subnetwork, but the discriminator subnetwork is not needed once training is complete. (Note that in other applications, one can actually keep the discriminator rather than the generator for purposes of having a starting point for a semi-supervised classifier as in Salimans et al.[54] and take advantage of both labeled and unlabeled data for image-level *classification* tasks.) However, in our case, rather than just generating realistic-looking images, we wish to take input images and produce a resulting segmentation. Thus, we follow a similar approach as previously used[55, 56] and a GAN-like framework to help train a segmentation network with both labeled and unlabeled data. An illustration of our approach can be seen in Figure 6. Intuitively, instead of having a generator G to take noise and generate realistic-looking images, we use a subnetwork G to take an input image and produce a resulting segmentation and a subnetwork D to differentiate (at a pixel level) between segmentations produced by the network G and reference segmentations (i.e., being able to tell if the segmentation, at each pixel, was from the generator subnetwork or used as the reference ground-truth). Here both G and D for our semi-supervised learning approach use the same network structure following the FCN approach in the previous section. Note that after training is complete, as with a traditional GAN, we can just use the trained G subnetwork (and not use the discriminator subnetwork) in practice (e.g., during testing) so that the inputs/outputs are just like they would be with the FCN described in the prior section.

Just as with a traditional GAN, each training iteration alternates between training the G network and training the D network. When training the G network, the weights in the D network are frozen and won't be updated (and vice-versa). Both labeled and unlabeled

images can be used to update the weights (in each epoch, all 26 labeled images in the training set are used and a random subset of 26 of the 50 unlabeled images, to match the number of labeled images, is used). In training G (i.e., the segmentation network), when the input is a labeled image (i.e., the input X has a corresponding reference segmentation Y), the loss function is a combination of the same supervised loss used in the fully supervised approach as well as a binary cross-entropy loss based on passing the result of the generator (i.e., segmentation) network through the current version of the discriminator:

$$L_G = \beta \cdot L_{FCN} + L_{bce}(D(G(X), X), 1),$$

where $\beta = 10$ to balance supervised loss and GAN loss and L_{bce} is L_{BCE} with a plain Sigmoid function:

$$L_{bce} = -\sum_i \left(y_i \cdot \log \frac{1}{1+e^{-\hat{y}_i}} + (1 - y_i) \cdot \log \left(1 - \frac{1}{1+e^{-\hat{y}_i}} \right) \right).$$

Note that in this case, the generator wants the discriminator to output a value of 1 at each pixel to indicate that the discriminator is fooled into thinking the generator's output is a true manual tracing. When the input is an unlabeled image X, a reference labeled image is not available, so the loss function is solely based on the binary cross-entropy loss indicated in the second part of the equation above:

$$L_G = L_{bce}(D(G(X), X), 1).$$

Similarly, in training D (the discriminator that tries to differentiate, on a pixel-by-pixel level, using both the segmentation map and original input image as input, whether the resulting segmentation came from the generator/segmentation network or a manual reference), when the input is an unlabeled image X, the loss function is also based on a binary cross-entropy

loss, but this time to encourage the generator's output to result in a value of 0 at each pixel location from the discriminator (i.e., the discriminator wants to correctly predict that the generator's output is not a reference segmentation):

$$L_D = L_{bce}(D(G(X), X), 0) \quad .$$

In training *D*, when the input is a labeled image (i.e., the input *X* has a corresponding reference segmentation *Y*), the current version of the generator is used to create the generated segmentation map, *G(X)*. A "mixed" image *Mix(G(X), Y)* is then created by randomly selecting each pixel to have a value either from the generated segmentation map *G(X)* or the reference image *Y*. A mask image *M* is created with values of 1 at locations where the pixels came from *Y* and 0 where pixels came from *G(X)*. The loss function used is the binary cross-entropy loss between the discriminator output (using the original image and mixed image as input) and the mask image of 1's and 0's:

$$L_D = L_{bce}(D(Mix(G(X), Y), X), M) \quad .$$

As before, this loss function will encourage the discriminator to have an output of 0 at locations where the input was from the generator network and an output of 1 at locations where the input was from the actual reference segmentation.

Training strategies

To implement our approaches, the PyTorch framework was used for all of the experiments. A single Nvidia GeForce 1080Ti GPU with 12GB memory was used for training and testing. To train the networks, we used mini-batch stochastic gradient descent (Adam) as the optimizer.[57, 58] The input image size for training was set to be 512×512, with a batch size of 4, randomly cropped from training images. Before each training mini-batch, data

augmentation was applied to the input images with random resize, cropping, flip, Gaussian blur, and contrast changes.

For the learning rate, the FCN approach used an initial value of 1e-3 and was divided by a factor of 10 when the training loss plateaued. For the semi-supervised approach, a learning rate of 1e-4 for the *D* network and 1e-3 for the *G* network was used.

In order to determine when to stop training, the networks were evaluated on the validation set every 10 epochs. To balance the quality of axon segmentation and the count of axons generated from post-processing, we used an equal combination of accuracy and the absolute axon number difference between prediction and reference as the metric for the segmentation. More specifically, the receiver operating characteristic (ROC) curves with respect to prediction thresholds were measured and area under curve (AUC) of ROC was calculated as a measure of pixel accuracy. The network parameters providing the highest performance on the validation set was used as our final approach for evaluations.

Evaluation

The AxonJ approach and our trained FCN and semi-supervised approach were used to provide both axon counts and complete segmentations for each of the 18 test images (corresponding to 3317 marked axon centers and 1103 fully traced axons from the first expert). The axon counts and complete tracings (obtained as discussed previously) from a single expert were used as the reference for comparisons with the automated approaches. The absolute axon count difference (as a percentage of the reference count) and the Dice similarly coefficient (a pixel-based measure of similarity) between each automated result and the reference were computed for each image and averaged across all images. In addition,

the Pearson correlation coefficient (R) between the counts provided by each automated approach and the manual reference counts were computed. The axon counts and complete tracings were also obtained from a second expert and compared to those from the first expert (again, measuring the average absolute count difference as a percentage, the Pearson correlation coefficient of the counts, and the Dice similarly metric of the complete tracings).

In addition, to further evaluate the semi-supervised approach against a similar structured FCN, we compared these two approaches with different sizes of training data size. Both approaches were trained on different sizes of randomly chosen labeled images: 100% (26 images), 50% (13 images), 25% (6 images), 10% (2 images). Meanwhile, for each training data size, the semi-supervised learning approach was also trained on all the unlabeled/unmarked data (50 images). Metrics of absolute count difference, correlations, and Dice coefficients were then compared between the two models trained on each data size.

Results

Overall quantitative results are summarized in Table 1, with the semi-supervised approach having the best overall performance. Comparing the counts resulting from the semi-supervised learning approach with the reference counts resulted in a mean absolute percent difference in counts of 4.4%, with a Pearson's correlation coefficient of R=0.97. The Dice coefficient (a measure of the relative overlap of the pixel-based segmentations) was 0.81 for the semi-supervised approach. In contrast, the mean absolute percent difference in counts for the AxonJ approach was significantly ($p < 0.05$ using paired t-test) higher (9.0%), the Pearson's correlation coefficient was lower (R=0.86), and the Dice coefficient was

significantly ($p < 0.05$ using paired t-test) lower (0.70). The FCN approach and second expert had similar results that were just slightly worse than the semi-supervised approach. Visually, the semi-supervised approach also performed best, yielding more confident and clean predictions than the FCN approach (Figure 7). Compared with the AxonJ results, fewer false positive regions were segmented.

When visually comparing the performance of the semi-supervised approach vs FCN with different sizes of training data (Figure 8), at each training size, the semi-supervised approach showed cleaner segmentations and less noise, with borders between axons more clearly seen compared to the FCN approach. Quantitatively, with the exception of the absolute percent difference in counts when using only 10% (i.e., 2 images) of the labeled images for training, the semi-supervised approach had a better performance (Table 2). The discrepancy between Dice metric and the absolute percent error of the counts in the 10% case may have been in part due to the counts being more affected by the post-processing stage in its removal of smaller objects.

Discussion & Conclusion

Overall, in this work, we proposed a deep fully convolution network named AxonDeep for segmenting axons in PPD-stained optic nerve cross sections from mice and presented results of training this network using a supervised approach (using only labeled data) and a semi-supervised approach (extending the network in a generative-adversarial-network framework for purposes of training with both labeled and unlabeled data). Our results show a significant improvement over AxonJ using the semi-supervised learning framework both

visually and based on quantitative metrics comparing axon counts and axon segmentations with an expert reference. In the final test set, we did not include cases of severely damaged nerves as it was already known that AxonJ would perform poorly in such cases. While severe nerves were not included as part of supervised training and evaluation, here we include a few qualitative examples to indicate the increased ability of the semi-supervised approach (AxonDeep) in segmenting axons of damaged nerves versus AxonJ. For example, Figure 9 (B)(E) shows example results of AxonJ in cases of severely damaged nerves. In these cases, there are around 500 axons counted with AxonJ, comparing with less than 100 axons annotated by an expert. The output from the semi-supervised learning approach can be seen in Figure 9(C)(F).

Different from existing deep-learning approaches for providing axon counts, such as AxoNet,[34] we directly segment the axons. Use of the semi-supervised training approach to be able to still utilize untraced images was one of our strategies for better being able to deal with the challenge and time-consuming nature of obtaining complete reference segmentations, as is needed for training. Another strategy that we used (for training purposes only) to help address the difficulty in obtaining complete reference segmentations, as previously described in more detail in the methods, was to develop an approach for editing existing segmentations rather than tracing completely from scratch. Before editing, to help avoid any bias in the number of axons marked by the existing segmentation, we only provided the starting automated result for axons whose center points were independently marked by an expert. Thus, while the boundaries of axons themselves for purposes of training may have had a bias towards the starting automated approach, the number of axons in the complete reference segmentations matched the counts provided manually and thus

were not biased by the automated approach. (In a prior iteration on a different dataset, we considered editing the automated result directly without this step, but decided it had the potential for biasing the results towards the automated result too much.) As previously mentioned, while we felt this was an acceptable compromise for training, we did not want to have reference segmentations biased towards an automated approach for purposes of the final evaluation and thus obtained complete tracings on smaller sub-images. In future work, the current version of AxonDeep could potentially be used for editing additional segmentations for training purposes.

Being able to consistently define the actual boundaries of axons is an area that could potentially be further explored in future work. For example, we noted that in both FCN and semi-supervised approach, the Dice coefficient between the reference and semi-automatically generated axon segmentations was around 0.8-0.9 in the validation set, which was higher than the Dice coefficient of 0.79 between two experts in the test set. An example of inconsistency between experts, as well as within an expert (axon mask vs approximate center), is shown in Figure 10. Note the inconsistency between experts are not fully shown in axon count evaluations, since only the consistency of counts is evaluated. In the future, approaches like probabilistic U-Net could be used to help model such uncertainties.

Implementation of AxonDeep could assist in the execution of a broad range of experiments. The mouse optic nerve contains ~50,000 axons, though that number can vary widely according to genetic background.[59] Quantifications of axon number is a gold-standard for measuring disease severity,[17, 60-63] but the labor-intensive nature of manual axon counting often results in studies instead using qualitative grading scales.[19, 64, 65] As with all of the tools that the field has put forward to count RGC somas[26, 27, 39, 66-70] or axons[26, 27, 34, 71-73] in various

animal models, the automated counts performed by AxonDeep greatly reduce the labor of manual counts and eliminate the possibility of user-to-user, lab-to-lab, or model-to-model variability inherent to subjective grading scales. An advantage of AxonDeep is that it performs axon segmentations as well as counts. Thus, it will also be possible to study whether axon size and shape vary during disease state. Given the high interest that has existed for many years in studying differential susceptibility of RGCs with comparatively large versus small soma,[74] and the interesting energetic differences in large versus small axons,[75] AxonDeep could be utilized to study many quantitative aspects of axon morphology which were previously not practical.

Acknowledgements

We acknowledge use of the University of Iowa Central Microscopy Research Facility (a core resource supported by the Vice President for Research & Economic Development, the Holden Comprehensive Cancer Center, and the Roy J. and Lucille A. Carver College of Medicine) and the University of Iowa Multidisciplinary Investigations in Visual Science service cores (a resource supported by a P30 grant from the NIH/NEI to the University of Iowa). The contents do not represent the views of the U.S. Department of Veterans Affairs or the U.S. Government.

References

1. Blonna D, Bonasia DE, Mattei L, Bellato E, Greco V, Rossi R. Efficacy and Safety of Subacromial Corticosteroid Injection in Type 2 Diabetic Patients. *Pain Res Treat* 2018;2018:9279343.

2. Quigley HA. Neuronal death in glaucoma. *Prog Retin Eye Res* 1999;18:39-57.

3. Dutca LM, Stasheff SF, Hedberg-Buenz A, et al. Early detection of subclinical visual damage after blast-mediated TBI enables prevention of chronic visual deficit by treatment with P7C3-S243. *Invest Ophthalmol Vis Sci* 2014;55:8330-8341.

4. Mohan K, Kecova H, Hernandez-Merino E, Kardon RH, Harper MM. Retinal ganglion cell damage in an experimental rodent model of blast-mediated traumatic brain injury. *Invest Ophthalmol Vis Sci* 2013;54:3440-3450.

5. Sohn EH, van Dijk HW, Jiao C, et al. Retinal neurodegeneration may precede microvascular changes characteristic of diabetic retinopathy in diabetes mellitus. *Proc Natl Acad Sci U S A* 2016;113:E2655-2664.

6. Walter SD, Ishikawa H, Galetta KM, et al. Ganglion cell loss in relation to visual disability in multiple sclerosis. *Ophthalmology* 2012;119:1250-1257.

7. Nishioka C, Liang HF, Barsamian B, Sun SW. Sequential phases of RGC axonal and somatic injury in EAE mice examined using DTI and OCT. *Mult Scler Relat Disord* 2019;27:315-323.

8. Dehabadi MH, Davis BM, Wong TK, Cordeiro MF. Retinal manifestations of Alzheimer's disease. *Neurodegener Dis Manag* 2014;4:241-252.

9. Levin LA, Gordon LK. Retinal ganglion cell disorders: types and treatments. *Prog Retin Eye Res* 2002;21:465-484.

10. Buckingham BP, Inman DM, Lambert W, et al. Progressive ganglion cell degeneration precedes neuronal loss in a mouse model of glaucoma. *J Neurosci* 2008;28:2735-2744.

11. Syc-Mazurek SB, Libby RT. Axon injury signaling and compartmentalized injury response in glaucoma. *Prog Retin Eye Res* 2019;73:100769.

12. Nadal-Nicolas FM, Jimenez-Lopez M, Sobrado-Calvo P, et al. Brn3a as a marker of retinal ganglion cells: qualitative and quantitative time course studies in naive and optic nerve-injured retinas. *Invest Ophthalmol Vis Sci* 2009;50:3860-3868.

13. Rodriguez AR, de Sevilla Muller LP, Brecha NC. The RNA binding protein RBPMS is a selective marker of ganglion cells in the mammalian retina. *J Comp Neurol* 2014;522:1411-1443.

14. Anderson MG, Libby RT, Gould DB, Smith RS, John SW. High-dose radiation with bone marrow transfer prevents neurodegeneration in an inherited glaucoma. *Proc Natl Acad Sci U S A* 2005;102:4566-4571.

15. Anderson MG, Libby RT, Mao M, et al. Genetic context determines susceptibility to intraocular pressure elevation in a mouse pigmentary glaucoma. *BMC Biol* 2006;4:20.

16. Libby RT, Li Y, Savinova OV, et al. Susceptibility to neurodegeneration in a glaucoma is modified by Bax gene dosage. *PLoS Genet* 2005;1:17-26.

17. Mao M, Hedberg-Buenz A, Koehn D, John SW, Anderson MG. Anterior segment dysgenesis and early-onset glaucoma in nee mice with mutation of Sh3pxd2b. *Invest Ophthalmol Vis Sci* 2011;52:2679-2688.

18. Templeton JP, Struebing FL, Lemmon A, Geisert EE. ImagePAD, a novel counting application for the Apple iPad, used to quantify axons in the mouse optic nerve. *Exp Eye Res* 2014;128:102-108.

19. Chauhan BC, Levatte TL, Garnier KL, et al. Semiquantitative optic nerve grading scheme for determining axonal loss in experimental optic neuropathy. *Invest Ophthalmol Vis Sci* 2006;47:634-640.

20. Ebneter A, Casson RJ, Wood JP, Chidlow G. Estimation of axon counts in a rat model of glaucoma: comparison of fixed-pattern sampling with targeted sampling. *Clin Exp Ophthalmol* 2012;40:626-633.

21. Evans LP, Woll AW, Wu S, et al. Modulation of Post-Traumatic Immune Response Using the IL-1 Receptor Antagonist Anakinra for Improved Visual Outcomes. *J Neurotrauma* 2020;37:1463-1480.

22. Harper MM, Woll AW, Evans LP, et al. Blast Preconditioning Protects Retinal Ganglion Cells and Reveals Targets for Prevention of Neurodegeneration Following Blast-Mediated Traumatic Brian Injury. *Invest Ophthalmol Vis Sci* 2019;60:4159-4170.

23. Kirillov A, Girshick R, He K, Dollar P. Panoptic feature pyramid networks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*; 2019:6392-6401.

24. Marina N, Bull ND, Martin KR. A semiautomated targeted sampling method to assess optic nerve axonal loss in a rat model of glaucoma. *Nat Prot* 2010;5:1642-1651.

25. Zarei K, Scheetz TE, Christopher M, et al. Automated Axon Counting in Rodent Optic Nerve Sections with AxonJ. *Sci Rep* 2016;6:26559.

26. Reynaud J, Cull G, Wang L, et al. Automated quantification of optic nerve axons in primate glaucomatous and normal eyes--method and comparison to semi-automated manual quantification. *Invest Ophthalmol Vis Sci* 2012;53:2951-2959.

27. Teixeira LB, Buhr KA, Bowie O, et al. Quantifying optic nerve axons in a cat glaucoma model by a semi-automated targeted counting method. *Mol Vis* 2014;20:376-385.

28. Ciresan D, Giusti A, Gambardella L, Schmidhuber J. Deep neural networks segment neuronal membranes in electron microscopy images. *Adv Neural Inf Process Syst* 2012;25:2843-2851.

29. di Scandalea ML, Perone CS, Boudreau M, Cohen-Adad J. Deep active learning for axon-myelin segmentation on histology data. *arXiv preprint arXiv:190705143* 2019.

30. Gomez-de-Mariscal E, Maska M, Kotrbova A, Pospichalova V, Matula P, Munoz-Barrutia A. Deep-Learning-Based Segmentation of Small Extracellular Vesicles in Transmission Electron Microscopy Images. *Sci Rep* 2019;9:13211.

31. Lee JG, Jun S, Cho YW, et al. Deep Learning in Medical Imaging: General Overview. *Korean J Radiol* 2017;18:570-584.

32. Malon CD, Cosatto E. Classification of mitotic figures with convolutional neural networks and seeded blob features. *J Pathol Inform* 2013;4:9.

33. Zaimi A, Wabartha M, Herman V, Antonsanti PL, Perone CS, Cohen-Adad J. AxonDeepSeg: automatic axon and myelin segmentation from microscopy data using convolutional neural networks. *Sci Rep* 2018;8:3816.

34. Ritch MD, Hannon BG, Read AT, et al. AxoNet: A deep learning-based tool to count retinal ganglion cell axons. *Sci Rep* 2020;10:8034.

35. Anderson MG, Smith RS, Hawes NL, et al. Mutations in genes encoding melanosomal proteins cause pigmentary glaucoma in DBA/2J mice. *Nat Genet* 2002;30:81-85.

36. Chang B, Smith RS, Hawes NL, et al. Interacting loci cause severe iris atrophy and glaucoma in DBA/2J mice. *Nat Genet* 1999;21:405-409.

37. Churchill GA, Airey DC, Allayee H, et al. The Collaborative Cross, a community resource for the genetic analysis of complex traits. *Nat Genet* 2004;36:1133.

38. Chesler EJ, Miller DR, Branstetter LR, et al. The Collaborative Cross at Oak Ridge National Laboratory: developing a powerful resource for systems genetics. *Mamm Genome* 2008;19:382-389.

39. Hedberg-Buenz A, Christopher MA, Lewis CJ, et al. Quantitative measurement of retinal ganglion cell populations via histology-based random forest classification. *Exp Eye Res* 2016;146:370-385.

40. Hedberg-Buenz A, Meyer KJ, van der Heide CJ, et al. Biological Correlations and Confounding Variables for Quantification of Retinal Ganglion Cells Based on Optical Coherence Tomography using Diversity Outbred Mice. *bioRxiv* 2020;2020.2012.2023.423848.

41. Trantow CM, Mao M, Petersen GE, et al. Lyst mutation in mice recapitulates iris defects of human exfoliation syndrome. *Invest Ophthalmol Vis Sci* 2009;50:1205-1214.

42. Lorensen WE, Cline HE. Marching cubes: A high resolution 3D surface construction algorithm. *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1987* 1987;21:163-169.

43. Medeiros FA, Sample PA, Weinreb RN. Corneal thickness measurements and visual function abnormalities in ocular hypertensive patients. *Am J Ophthalmol* 2003;135:131-137.

44. Rammer U. An interactive procedure for the polygonal approximation. *CGIP* 1972;1:224-256.

45. Xie S, Girshick R, Dollár P, Tu Z, He K. Aggregated residual transformations for deep neural networks. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*; 2017:5987-5995.

46. Chen L-C, Papandreou G, Schroff F, Adam H. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:170605587* 2017.

47. Seferbekov SS, Iglovikov V, Buslaev A, Shvets A. Feature Pyramid Network for Multi-Class Land Segmentation. *CVPR Workshops*; 2018:272-275.

48. Lin G, Milan A, Shen C, Reid I. RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*; 2017:5168-5177.

49. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*; 2016:770-778.

50. Iglovikov V, Shvets A. Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation. *arXiv preprint arXiv:180105746* 2018.

51. Beucher S. Use of watersheds in contour detection. *Proceedings of the International Workshop on Image Processing*: CCETT; 1979.

52. Nunez-Iglesias J. vdWS, Warner J., Boulogne, F., Gouillart E., Harouche M., Gruter L., Panfilov E. Module: morphology skimage.: scikit-image.

53. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial networks. *Commun ACM* 2020;63:139-144.

54. Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X. Improved techniques for training gans. *arXiv preprint arXiv:160603498* 2016.

55. Hung W-C, Tsai Y-H, Liou Y-T, Lin Y-Y, Yang M-H. Adversarial learning for semi-supervised semantic segmentation. *arXiv preprint arXiv:180207934* 2018.

56. Nie D, Gao Y, Wang L, Shen D. ASDNet: Attention based semi-supervised deep networks for medical image segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*: Springer; 2018:370-378.

57. Kingma DP, Ba J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:14126980* 2014.

58. Reddi SJ, Kale S, Kumar S. On the convergence of adam and beyond. *arXiv preprint arXiv:190409237* 2019.

59. Williams RW, Strom RC, Rice DS, Goldowitz D. Genetic and environmental control of variation in retinal ganglion cell number in mice. *J Neurosci* 1996;16:7193-7205.

60. Howell GR, Libby RT, Jakobs TC, et al. Axons of retinal ganglion cells are insulted in the optic nerve early in DBA/2J glaucoma. *J Cell Biol* 2007;179:1523-1537.

61. Fu CT, Sretavan D. Involvement of EphB/Ephrin-B signaling in axonal survival in mouse experimental glaucoma. *Invest Ophthalmol Vis Sci* 2012;53:76-84.

62. Nuschke AC, Farrell SR, Levesque JM, Chauhan BC. Assessment of retinal ganglion cell damage in glaucomatous optic neuropathy: Axon transport, injury and soma loss. *Exp Eye Res* 2015;141:111-124.

63. Soto I, Howell GR, John CW, Kief JL, Libby RT, John SW. DBA/2J mice are susceptible to diabetic nephropathy and diabetic exacerbation of IOP elevation. *PLoS One* 2014;9:e107291.

64. Libby RT, Anderson MG, Pang IH, et al. Inherited glaucoma in DBA/2J mice: pertinent disease features for studying the neurodegeneration. *Vis Neurosci* 2005;22:637-648.

65. Pang IH, Clark AF. Rodent models for glaucoma retinopathy and optic neuropathy. *J Glaucoma* 2007;16:483-505.

66. Dordea AC, Bray MA, Allen K, et al. An open-source computational tool to automatically quantify immunolabeled retinal ganglion cells. *Exp Eye Res* 2016;147:50-56.

67. Guymer C, Damp L, Chidlow G, Wood J, Tang YF, Casson R. Software for Quantifying and Batch Processing Images of Brn3a and RBPMS Immunolabelled Retinal Ganglion Cells in Retinal Wholemounts. *Transl Vis Sci Technol* 2020;9:28.

68. Hedberg-Buenz A, Christopher MA, Lewis CJ, et al. RetFM-J, an ImageJ-based module for automated counting and quantifying features of nuclei in retinal whole-mounts. *Exp Eye Res* 2016;146:386-392.

69. Masin L, Claes M, Bergmans S, et al. A novel retinal ganglion cell quantification tool based on deep learning. *Sci Rep* 2021;11:702.

70. Salinas-Navarro M, Mayor-Torroglosa S, Jimenez-Lopez M, et al. A computerized analysis of the entire retinal ganglion cell population and its spatial distribution in adult rats. *Vision Res* 2009;49:115-126.

71. Mysona BA, Segar S, Hernandez C, et al. QuPath Automated Analysis of Optic Nerve Degeneration in Brown Norway Rats. *Transl Vis Sci Technol* 2020;9:22.

72. Zarei K, Scheetz TE, Christopher M, et al. Corrigendum: Automated Axon Counting in Rodent Optic Nerve Sections with AxonJ. *Sci Rep* 2016;6:34124.

73. Bosco A, Romero CO, Ambati BK, Vetter ML. In vivo dynamics of retinal microglial activation during neurodegeneration: confocal ophthalmoscopic imaging and cell morphometry in mouse glaucoma. *J Vis Exp* 2015;e52731.

74. Della Santina L, Ou Y. Who's lost first? Susceptibility of retinal ganglion cell types in experimental glaucoma. *Exp Eye Res* 2017;158:43-50.

75. Harris JJ, Attwell D. The energetics of CNS white matter. *J Neurosci* 2012;32:356-371.

Tables

Table 1. Comparison of FCN and semi-supervised approaches training on the full training set with baseline approach

	AxonJ	**FCN**	**Semi**	**Expert 2**
Abs Diff.	9.0%	6.2%	**4.4%**	6.6%
R	0.86	0.93	**0.97**	0.93
Dice	0.70	0.80	**0.81**	0.79

Table 2. Comparison of FCN and semi-supervised approach when training on different training size.

	FCN 100%	**FCN 50%**	**FCN 25%**	**FCN 10%**	**Semi 100%**	**Semi 50%**	**Semi 25%**	**Semi 10%**
Abs Diff.	6.2%	5.7%	8.1%	9.0%	4.4**%**	**4.3%**	6.2%	9.4%
R	0.93	0.93	0.87	0.82	**0.97**	0.96	0.90	0.83
Dice	0.80	0.79	0.78	0.57	**0.81**	0.80	0.78	0.71

Figures

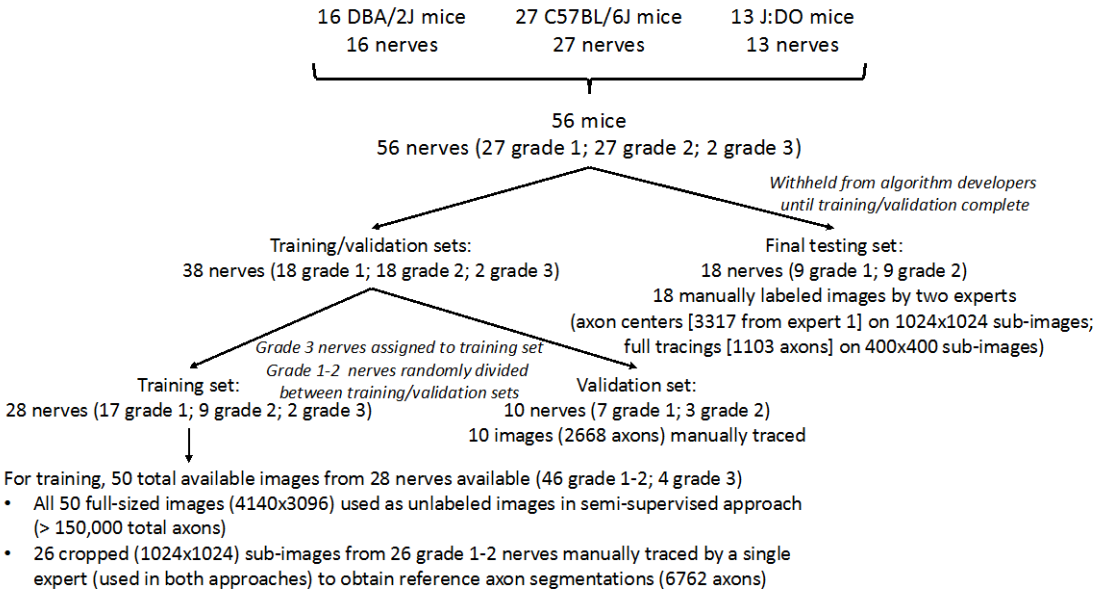

Figure 1. The 56 nerves from 56 mice from different strains and severity levels (grades 1-3) were divided into 28 nerves (50 total images; 26 labeled) for training, 10 nerves (10 labeled images) for validation, and 18 nerves (18 labeled images) for final testing. Based on the labels from expert 1, there were 6762 total axons labeled in the training set, 2668 total axons labeled in the validation set, 3317 axon centers labeled in the final testing set, and 1103 fully traced axons in the final testing set. There were at least 150,000 total axons in all 50 of the unlabeled images in the training set.

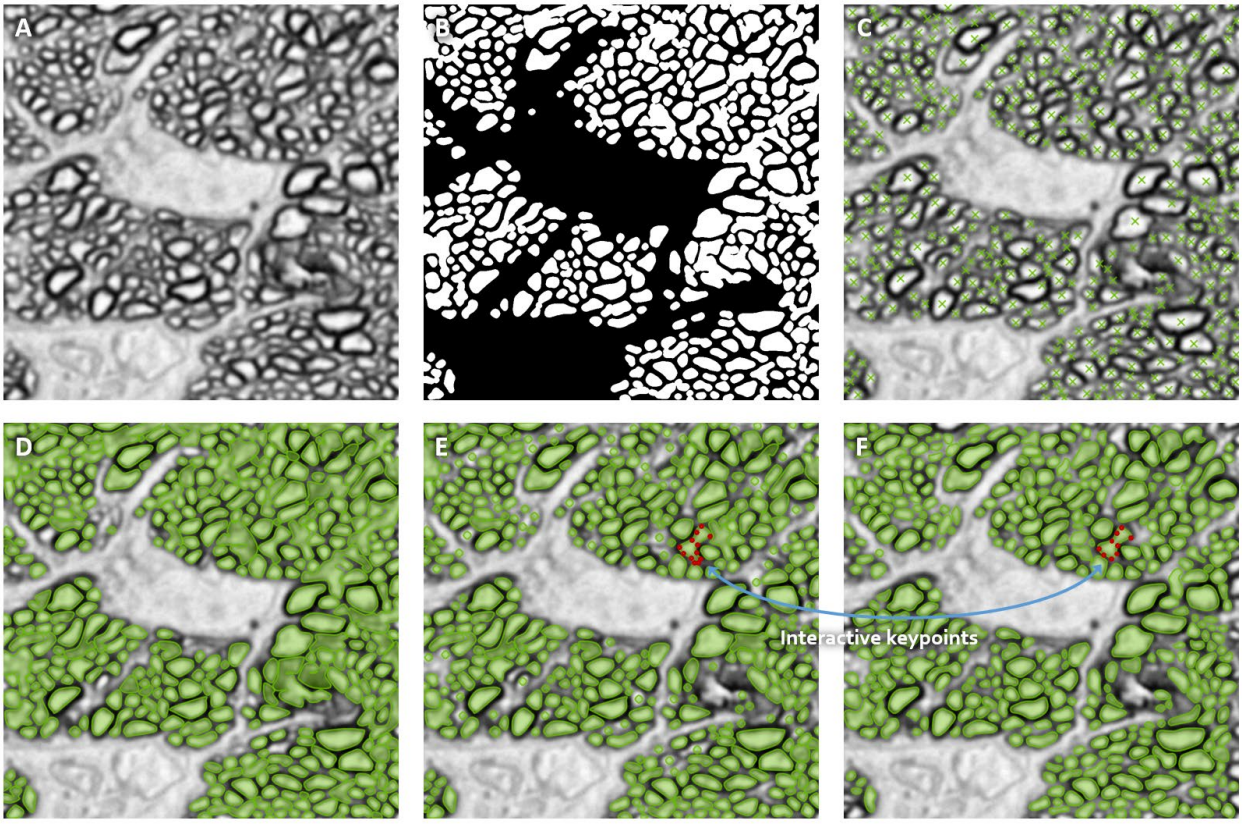

Figure 2. The coupling of manually corrected automated segmentations with manual tracing of center marks of axons was used to construct the set the reference segmentations for training. (A) Light micrograph of a PPD-stained optic nerve cross sections in enhanced format (histogram equalized for better visualization) used as an input image for training. (B) Binary AxonJ result. (C) Manually marked axon centers. (D) AxonJ result in (B) displayed using interactive GUI (clicking on an axon would cause interactive keypoints to appear). (E) Pruned AxonJ result with red interactive keypoints for a sample axon contour indicated. (F) Result of editing sample axon contour. By combining automated segmentation (B), manually traced center marks (C), and manual corrections (E)(F), references for training data can be obtained.

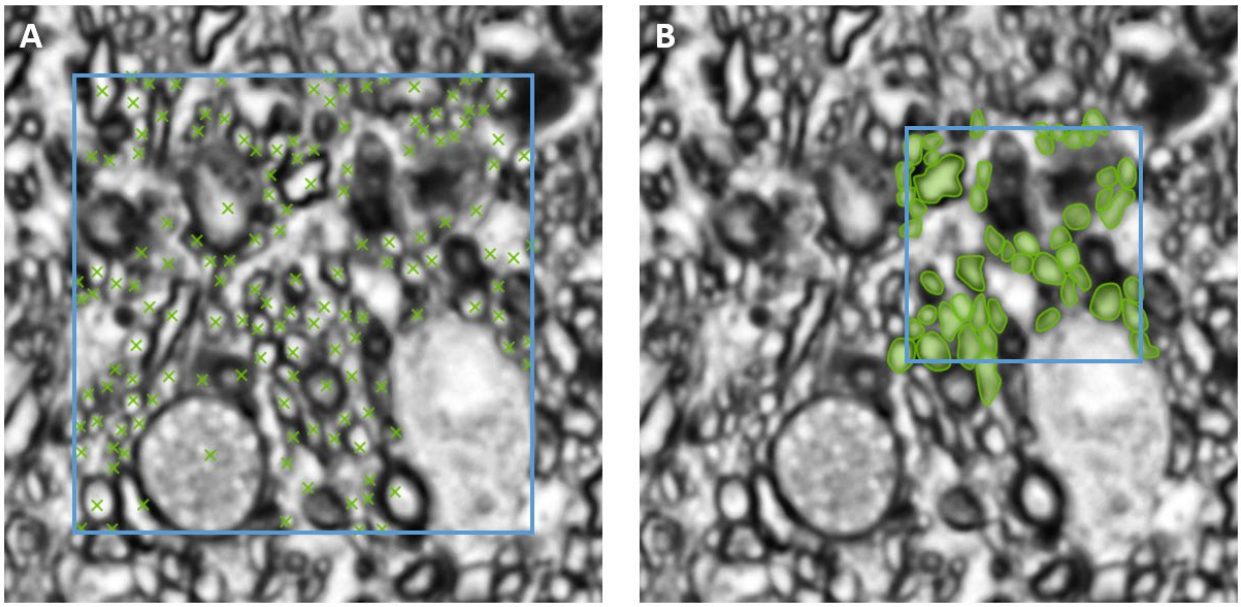

Figure 3. An illustrative example showing the generation of reference axon counts and manual axon tracings used for final evaluation on the test set. (A) Approximated centers within the blue box are used for evaluating axon counts. (B) Axon tracings within the smaller box are used for evaluating axon segmentation masks.

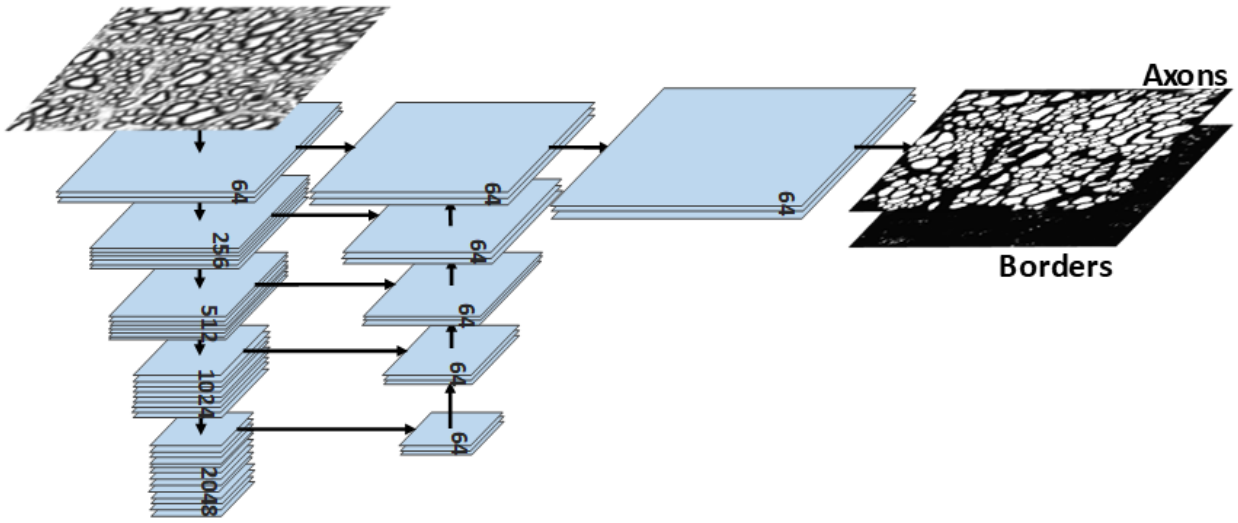

Figure 4. An illustration of the network for our FCN approach. The backbone network is paired with a light-weight feature pyramid (FPN) like decoder. The network takes a single channel axon image as input and outputs an axon probability map and borders between adjacent axons.

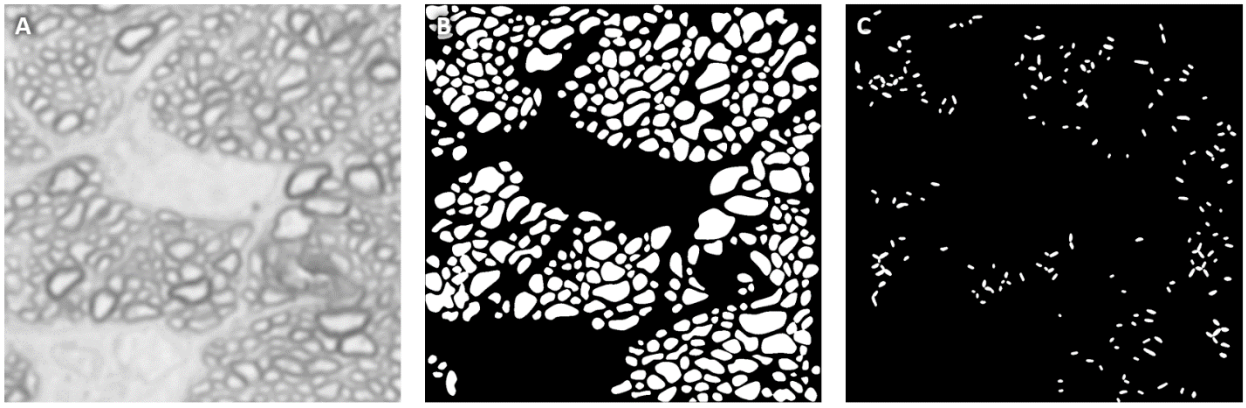

Figure 5. Example training data from analyses of optic nerve micrographs. (A) input axon image, (B) axon masks, and (C) borders between adjacent axons.

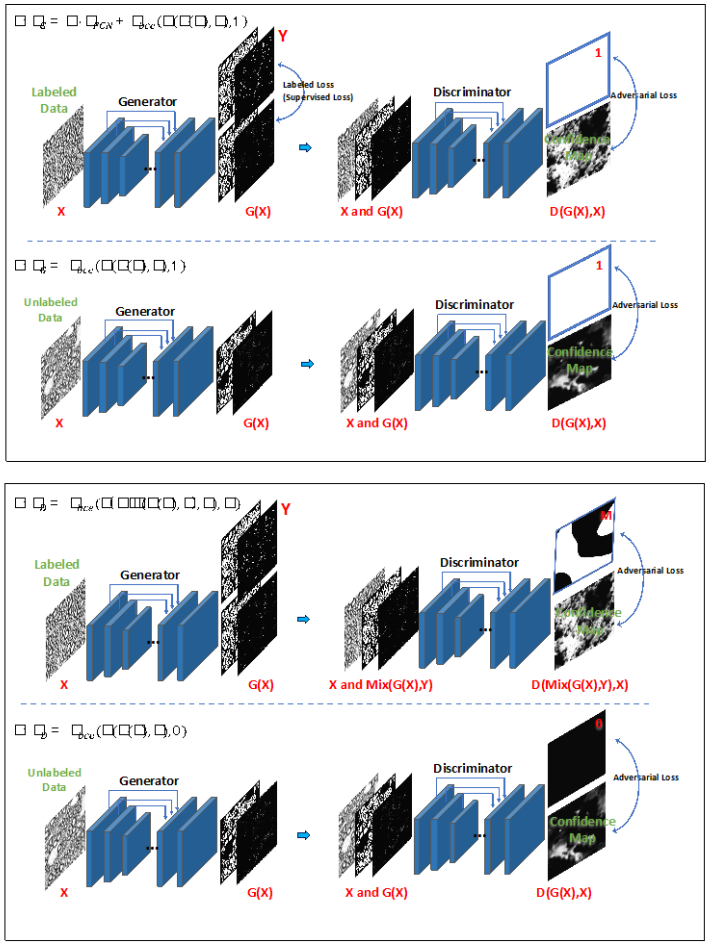

Figure 6. Training the semi-supervised approach alternates between updating the generator network weights and updating the discriminator network weights. Both the generator and discriminator have the same basic underlying architecture as the FCN approach. (After training only the generator network is retained as the final segmentation network.) The input to the generator is a raw axon image and the input to the discriminator is the raw axon image plus a segmentation output. The network uses both labeled data (with both axon images (X) and reference segmentations (Y) available) and unlabeled data (X) in updating the weights. (A) In updating the generator weights with labeled data, the loss function encourages the generator output to match the reference segmentation and to "fool" the discriminator in thinking that the generated output is a reference output. (B) In updating the generator weights with unlabeled data, the loss function encourages the generator to "fool" the discriminator in thinking that the generated output is a reference output. (C) In updating the discriminator weights with labeled data, the loss function encourages the discriminator to correctly output a 1 for pixels coming from a reference segmentation and a 0 for pixels coming from a generated segmentation. Note that a randomly mixed segmentation image (between the generated segmentation and reference segmentation) is provided as part of the input to the discriminator. (D) In updating the discriminator weights with unlabeled data, the loss function encourages the discriminator to correctly output a 0 for pixels not coming from a reference segmentation.

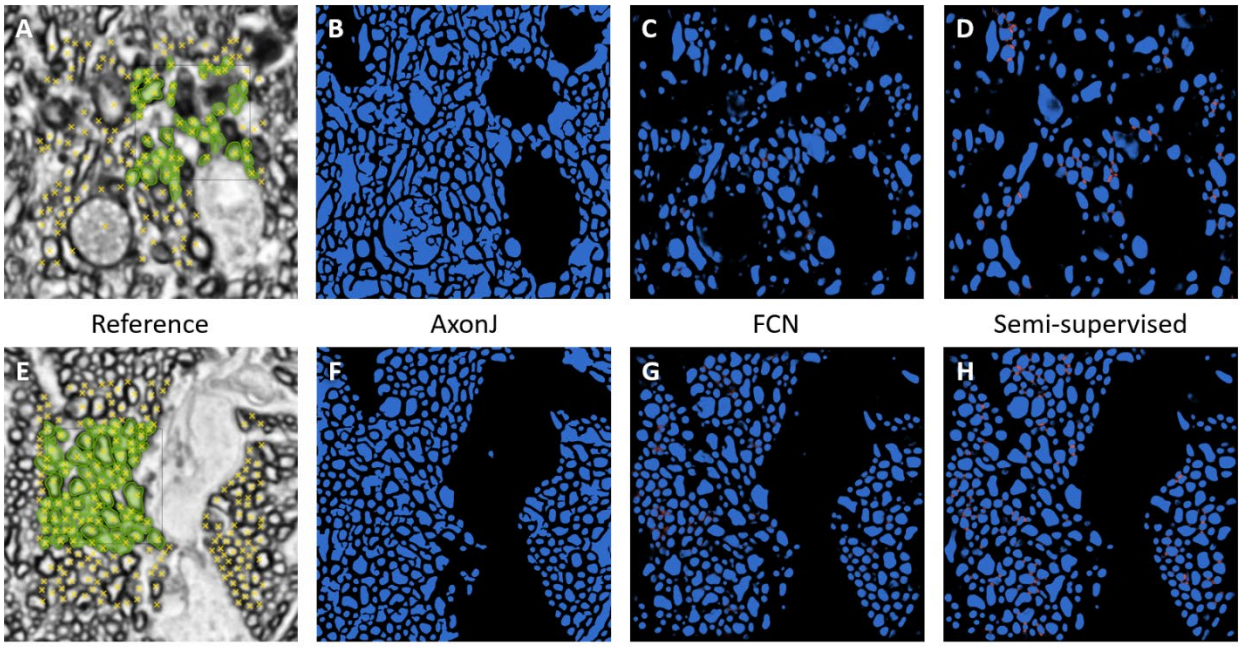

Figure 7. Comparison of AxonJ, FCN, and semi-supervised approach in a damaged axon image (upper row) and healthy axon image (lower row). The semi-supervised learning results (D) and (H) shows the overall best results compare to references (A) and (E). The borders are clearly segmented and axon probabilities are generally cleaner.

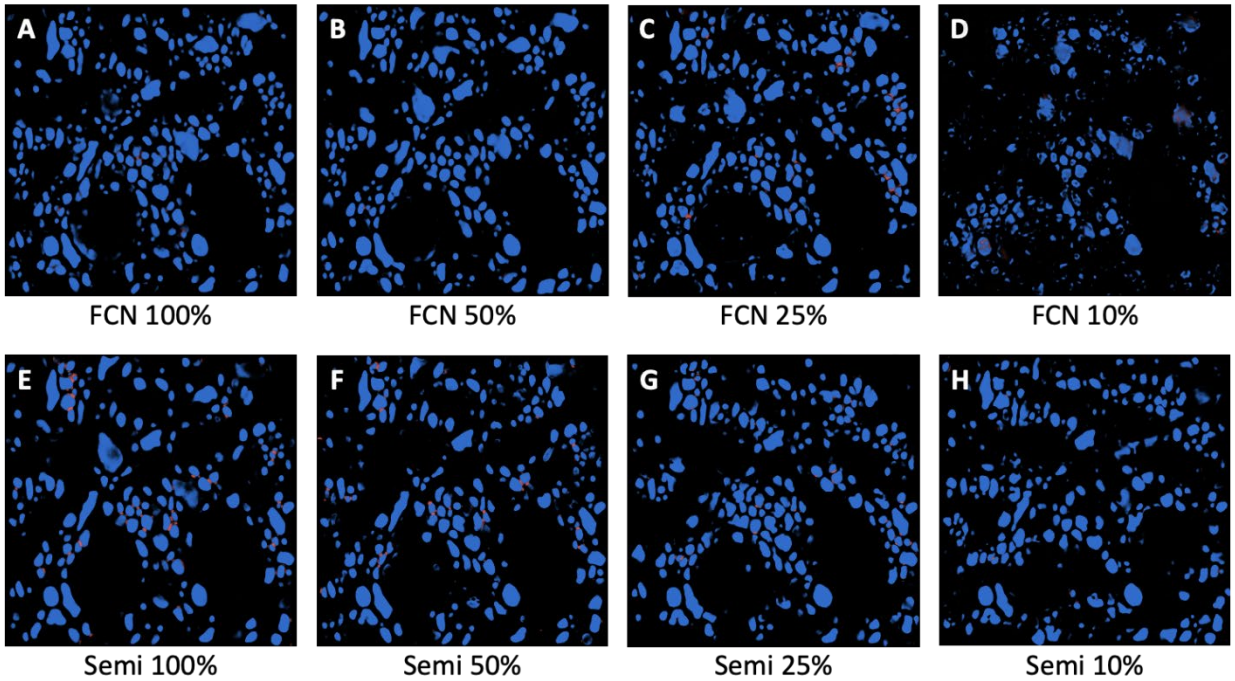

Figure 8. Comparison of FCN approach (upper row) and semi-supervised learning approach (lower row) with different training sizes. (A-D) show example results trained with different sizes under FCN direct training and (E-H) show results of the same example using semi-supervised learning. For each data size, the semi-supervised learning shows a higher quality and cleaner result.

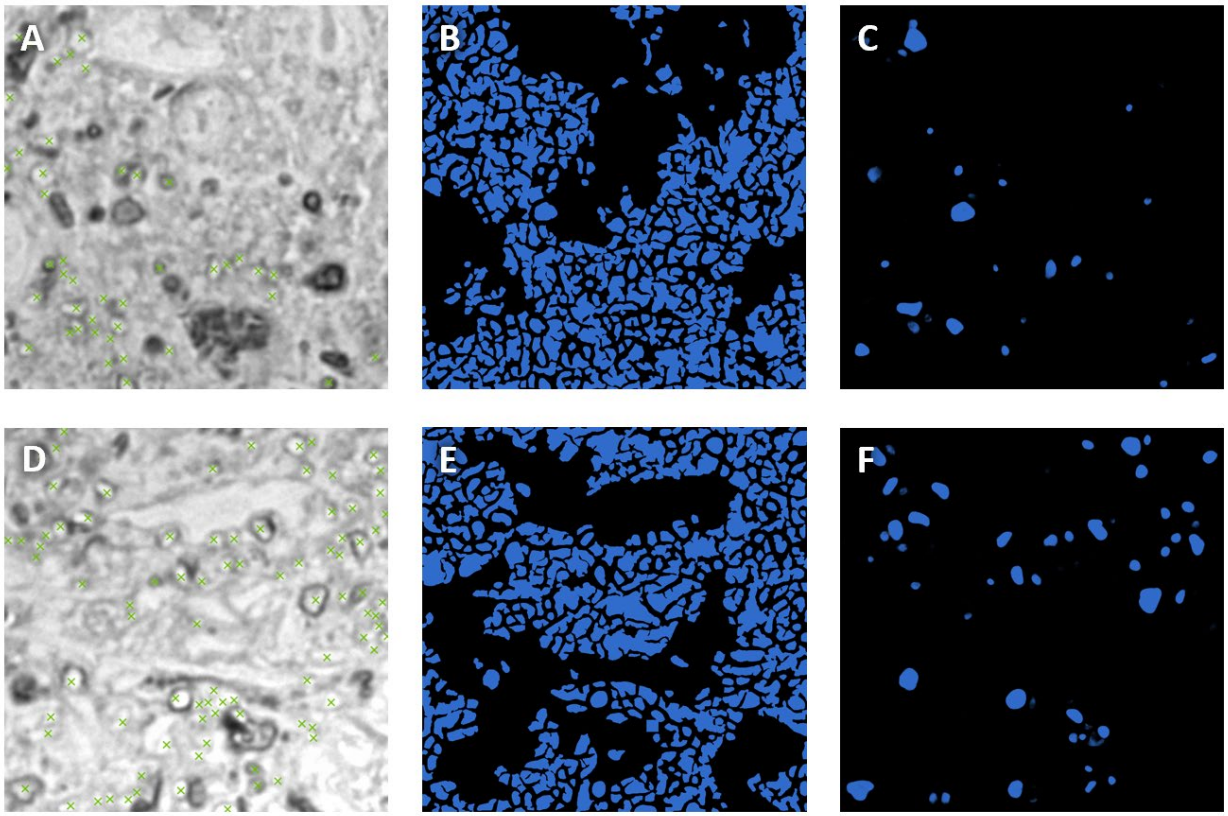

Figure 9. Segmentation comparison between AxonJ and our semi-supervised learning approach in case of severely damaged nerves

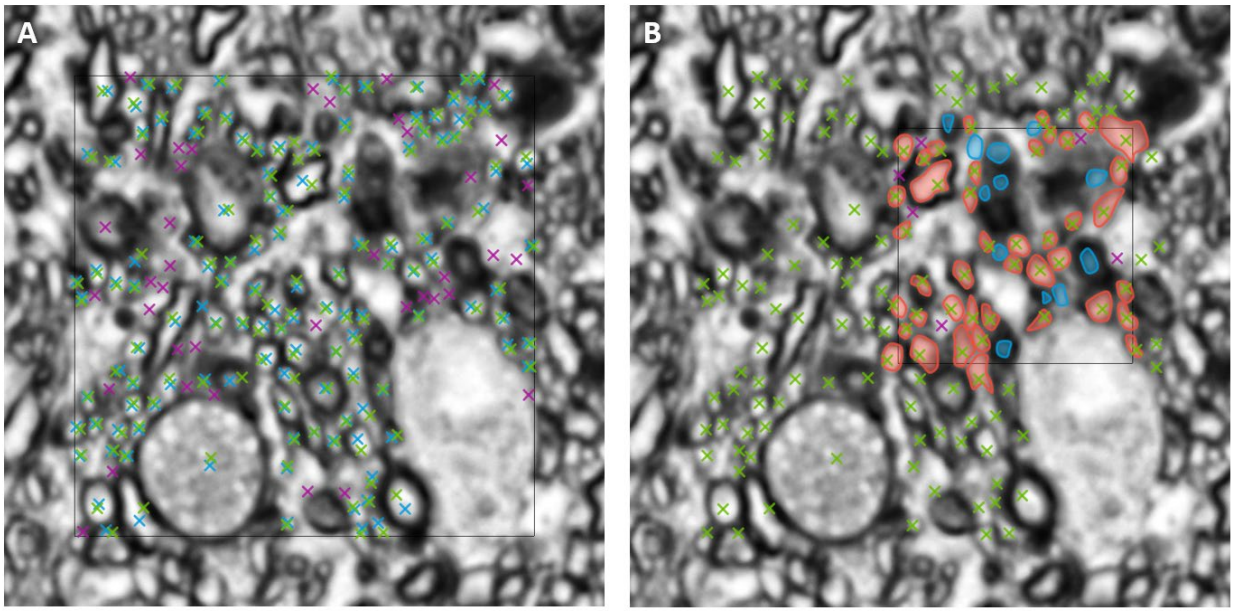

Figure 10. Inter-expert and intra-expert comparison. (A) A comparison between expert 1 (green) and expert 2 (blue). The purple points are the centers only marked by one expert. (B) Comparison between expert 1 center marks and axon tracings. The blue tracings are only traced as axons, and the purple points are only marked as axon centers.