Automated Axon Tracking of 3D Confocal Laser Scanning Microscopy Images Using Guided Probabilistic Region Merging

Ranga Srinivasan[1,2], Xiaobo Zhou[1,4], Eric Miller[3], Ju Lu[5], Jeff Litchman[5], Stephen TC Wong[1,4,*]

[1]Harvard Center for Neurodegeneration and Repair-Center for Bioinformatics, Harvard Medical School, Boston, MA

[2]Department of Electrical and Computer Engineering, Northeastern University, Boston MA

[3]Department of Electrical and Computer Engineering, Tufts University, Medford MA

[4]Functional and Molecular Imaging Center, Department of Radiology, Brigham and Women's Hospital and Harvard Medical School, Boston, MA

[5]Department of Molecular and Cellular Biology, Harvard University, Cambridge, MA, USA

[*] Corresponding to: wong@crystal.harvard.edu

Abstract

This paper presents a new algorithm for extracting the centerlines of the axons from a 3D data stack collected by a confocal laser scanning microscope. Recovery of neuronal structures from such datasets is critical for quantitatively addressing a range of neurobiological questions such as the manner in which the branching pattern of motor neurons change during synapse elimination. Unfortunately, the data acquired using fluorescence microscopy contains many imaging artifacts, such as blurry boundaries and non-uniform intensities of fluorescent radiation. This makes the centerline extraction difficult. We propose a robust segmentation method based on probabilistic region merging to extract the centerlines of individual axons with minimal user interaction. The 3D model of the extracted axon centerlines in three datasets is presented in this paper. The results are validated with the manual tracking results while the robustness of the algorithm is compared with the published repulsive snake algorithm.

Keywords – Maximum Intensity Projection, Segmentation, Guided Region Growing, Watershed

1. Introduction

The orientation of motor axons is critical in answering questions regarding synapse elimination in a developing muscle (Keller-Peck et al., 2001). Biologists have tried to address this issue by identifying the post-synaptic targets using transgenic mice that express fluorescent proteins in small subsets of motor axons. The post-synaptic targets are the cells innervated by the axons. More specifically, in the neuromuscular system, these are the muscle fibers. At neuromuscular junctions of developing mammals, the developing axonal branches of several motor neurons compete with each other resulting in withdrawal of all branches but one (Kasthuri et al., 2003). The biological application of the developed algorithm is to reconstruct the entire innervation field within a skeletal muscle based on images acquired from confocal microscopy. Given a 3D image stack with non-uniform resolution in the x-, y- and z-direction, it is desirable to segment multiple axons contained in the neuron image and reduce them to one-pixel wide medial axis. Our aim here is to better understand the spatial relations of groups of axons traveling in a nerve. The features revealed by the reconstruction help neuroscientists to understand how neurons interact with and control muscular retracting. The size of the individual datasets range from 20-30 Megabytes, but when they are joined together to form a collage, they are typically a few Terabytes in size. Since the data sets to be analyzed are huge, robust and automated algorithms are needed to track the centerlines of the axons accurately.

The tracking approaches introduced in the past can be broadly classified into three groups. The first class of algorithms tracks the centerlines of tubular objects in the *Maximum Intensity Projection* (MIP) image. As the name suggests, the MIP image along a certain direction is the projection of maximum intensities in the dataset along that direction onto a plane perpendicular to it. This is essentially a data reduction step which provides a 2D image of the objects present in

the 3D stack. Several automatic algorithms were introduced in the past for the centerline tracking of elongated tubular objects, such as retinal vasculature (Can et al., 1999) and neurons (Zhang et al., 2006). Though being computationally efficient in tracking the centerlines, they fail to work when the objects seem to cross each other in the MIP image.

The second category of algorithms tracks the centers by segmenting the objects in a sequence of cross-sectional images in the dataset. A popular algorithm in practice today is the snake model (Kass et al., 1987). It improves on an initial boundary by minimizing an energy functional consisting of internal and external energies. The authors of (Cai et al., 2006) proposed an improvement over the original snake model by incorporating a repulsive force to segment close lying cross-sections of axons. Similar to the snake model, the *repulsive snake model* is unable to accurately segment the axons of blurred boundaries. *Level-sets* represent another approach to segment and track using cross-sectional information, but they are computationally expensive and can be slow to converge (Xiong et al., 2006).

The final set of algorithms deploys 3D methods to track the objects in volumes of microscopic data. The main advantage of these algorithms is that they can track the objects regardless of their image orientation. The authors of (Al-Kofahi et al., 2002) built on the template based approach in (Can et al., 1999). The axons over short distances were modeled as cylinders with elliptical cross-sections having a certain curvature. Four sets of directional templates were used to track the centers of the axons in each iteration. Another approach was introduced in (Tschirren et al., 2005), where the authors used fuzzy connectivity for segmentation and analysis of intra-thoracic airway trees. The dataset was analyzed iteratively inside a moving cylindrical region of interest. Inside this cylinder, two regions, foreground and background, were grown simultaneously to compete for voxels. The centerlines could then be

found by skeletonization process. The authors of (Streekstra et al., 2002) used derivatives of a Gaussian kernel to find the direction and the location of the centerlines. The orientation of the centerline is first found by computing the direction along which the second order derivative is the least in magnitude. In a plane perpendicular to this direction, a point where the gradient vanishes is declared to be the center. Though the algorithms that fall in this category are theoretically attractive, their computational costs increase significantly with the size and complexity of the datasets.

Commercially or publicly available software packages, such as *NeuronJ* (Meijering et al., 2004) and *Reconstruct* (Fiala, 2005), on the other hand, track the centerlines in such fluorescent microscopic stacks interactively. *NeuronJ*, a plug-in for the freely available software, *ImageJ* (Abramoff et al., 2004), uses *live-wire* (Barrett et al., 1997) method to track the centerlines in the MIP image. The user has to select at least two points for each axon and has to guide the centerline when there is an ambiguity. *Reconstruct* also needs user intervention when the axons are close to each other. Manually segmenting huge stacks of data is a painstaking task and can lead to fatigue related bias.

A new and computationally efficient tracking algorithm for extracting the centerlines of the axons in fluorescent microscopy datasets is presented in this paper. We introduce a robust hybrid algorithm to track the centerlines of the axons in three-dimensional space with minimal user intervention. The basic approach is to track well-separated axons in the MIP plane using a low-complexity two-dimensional approach developed in (Can et al., 1999) and adopt more computationally sophisticated algorithms when the centerlines appear to intersect each other in the MIP image. In addition, we make use of the cross-sectional information as the cross-over of centerlines can be easily resolved using this approach. Our approach can be summarized as: (1)

Track the centerlines in the MIP image as far as possible, (2) switch to cross-sectional tracking if an axon cross-over is detected and (3) switch back to MIP tracking once the cross-over is resolved. We propose a guided region merging algorithm which is efficient in segmenting the axons when their boundaries are blurred due to imaging artifacts. Both shape and intensity information is used to track the axons which help us segment the axons in the cross-sectional plane accurately.

The organization of the paper is as follows. The method for extracting axon centerlines in two dimensions is mentioned in Section 2.1. In Section 2.2, we describe the approach adopted to solve the cross-over problem by switching to the cross-section based approach. Section 2.3 introduces our guided region growing algorithm which uses a probabilistic method based on the features extracted from the previous cross-sections.

We conclude the paper by demonstrating the results of this algorithm in Section 3. Our algorithm is applied on three datasets, and the results are compared with the manual results. To demonstrate the efficiency of our algorithm, its performance is compared with the centerline extraction method using repulsive snake model (Cai et al., 2006).

2. Materials and Methods

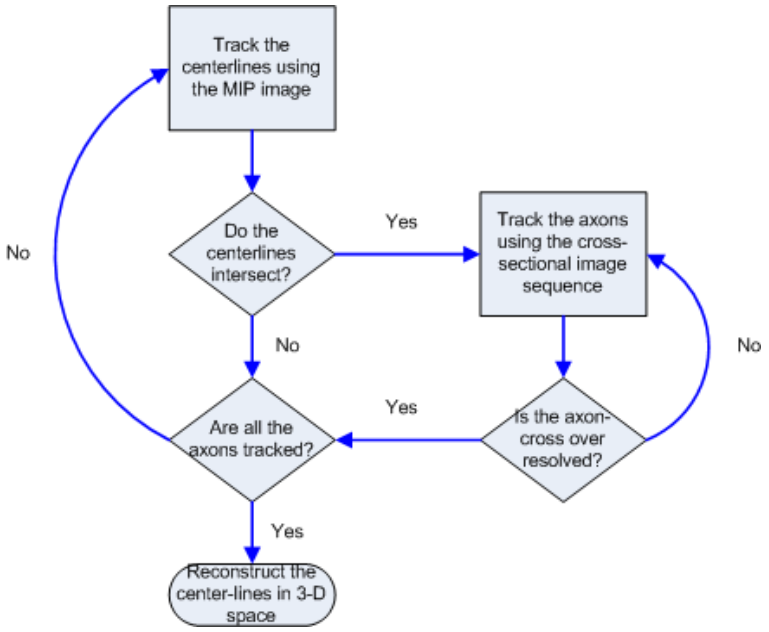
Materials

The raw data available for analysis consists of stacks of cross-sectional images of the axons obtained from neonatal mice using laser scanning confocal microscopes (Olympus Flouview FV500 and Bio-Rad 1024) equipped with motorized stage. For our study, the neurons of interest all expressed cytoplasmic YFP relatively uniformly inside each cell. YFP was excited with a 448nm line of argon laser using an X63 (1.4NA) oil objective with a 520-550nm band-pass

emission filter (Kasthuri et al., 2003). The images were sampled at the Nyquist limit in the imaging plane (pixel size = 0.1 micron) and over-sampled by a factor of 1.5 in the direction normal to the imaging plane (optical section thickness = 0.2 micron), and with a 12 bit dynamic range. Since the axons are not stained, the background is minimal in the acquired images. The goal of the developed algorithm is to build a three-dimensional model of the centerlines of the axons present in the dataset.

Methods

The dataset available for analysis is a sequence of cross-sectional fluorescent microscopic images of axons. These images can be stacked together to get a better picture of the axons present in the dataset. MIP-based tracking algorithms (Can et al., 1999; Zhang et al., 2006) work only when the axons are well separated, which is often not the case. Thus, axon *cross-over* is often encountered when tracking them in two dimensions. Since they never intersect in three dimensions, we resort to a different approach in such special cases. The following flow chart shows the workflow of the algorithm:



2.1 Tracking the Axons in the MIP Image

A simple dataset is shown in Figure 1, containing two axons shown by red and blue structures. Since such a dataset is formed by stacking a sequence of cross-sectional images, each point on the centerlines of the axons corresponds to one cross-sectional slice in the stack of data. As seen in Figure 1, there are several points where the axons seem to cross each other. But when seen from a cross-sectional point of view, it is clear that they do not in fact intersect.

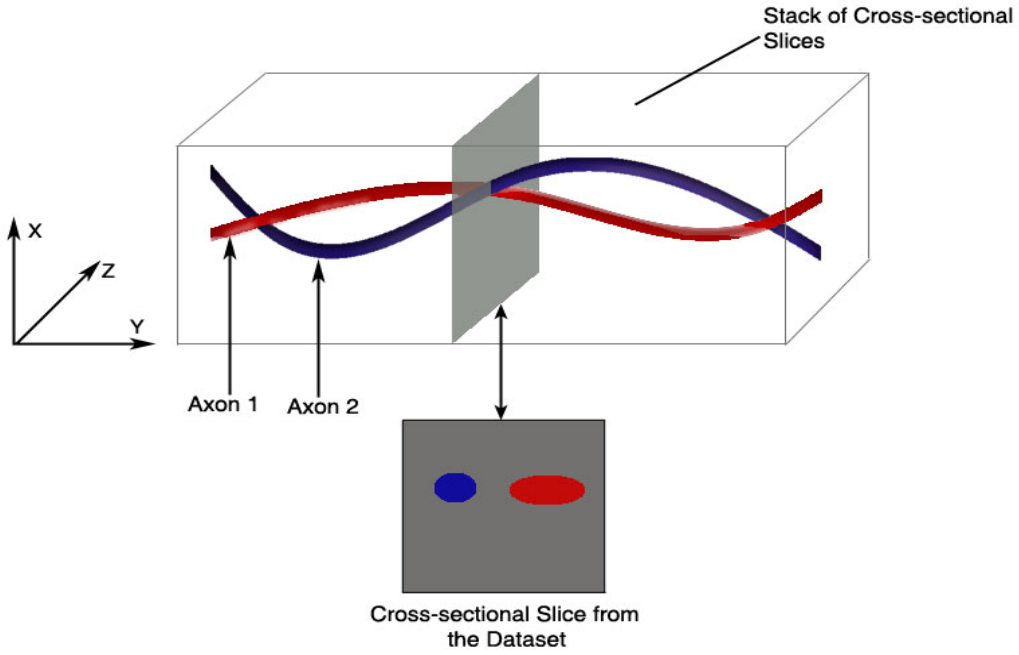


Figure 1: An example of a stack of cross-sectional images of two axons.

The MIP is just a way of visualizing the axons present in the stack of data. As mentioned earlier, the MIP image is the projection of the maximum intensities along a particular direction onto a plane perpendicular to it. Figure 1 shows the three orthogonal directions along which the MIP images could be found. The algorithm begins after the user selects one of the three possible MIP images of the dataset having the clearest details. Figure 2 shows the three MIP images in the three orthogonal directions in one such dataset. Since the MIP image along Z-axis provides the clearest view of the individual axons, and hence is the best for analysis, it has been used in

tracking the centerlines. After an MIP image is chosen, it is preprocessed to enhance the edges of the structures.

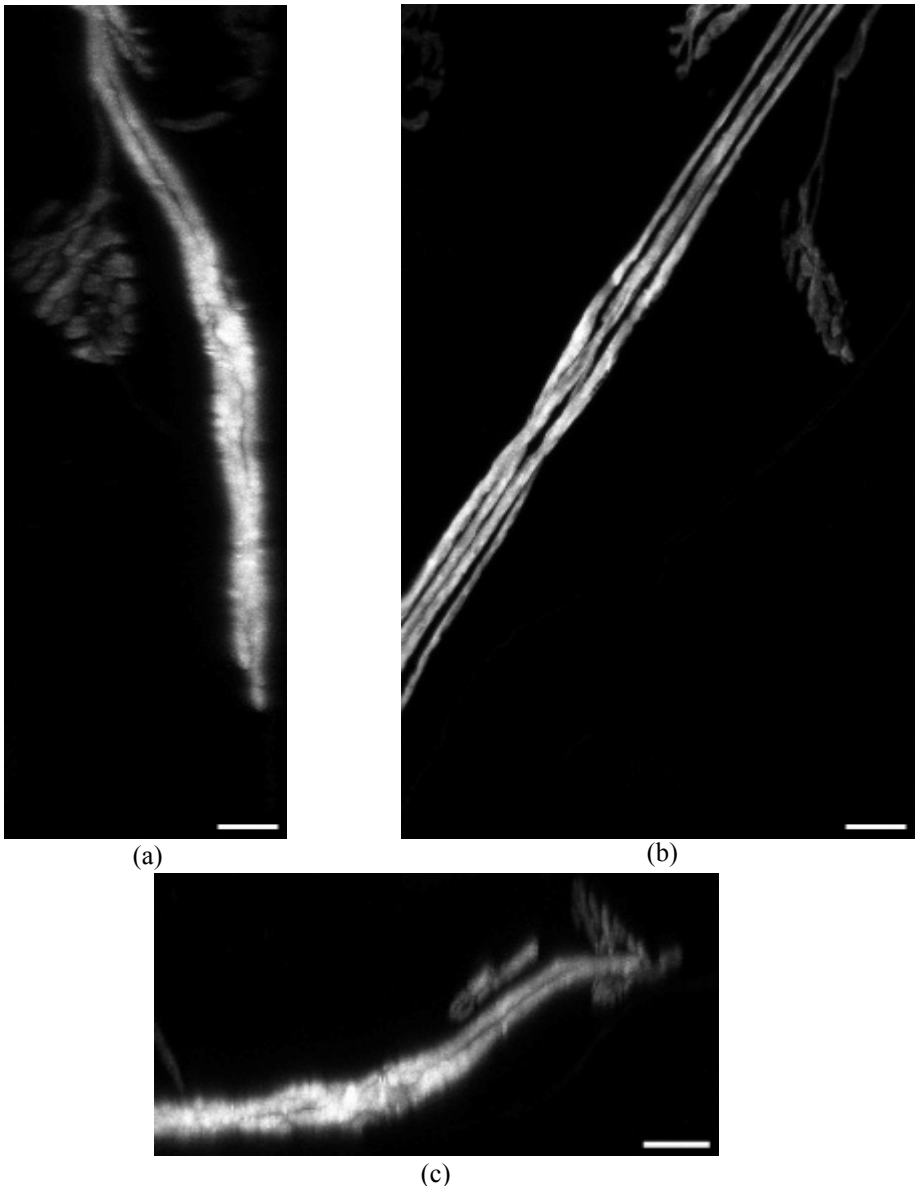


Figure 2 - The Maximum Intensity Projection Images: (a) MIP along Y-axis, (b) MIP along Z-axis, and (c) MIP along X-axis. The scale bar in (a) corresponds to 3μm. The scale bars in (b) and (c) correspond to 6μm.

The method described in (Can et al., 1999) is adopted to track the centerlines of the axons in the edge-enhanced MIP image. In order to automate the initialization of the tracks, a set of *seed points* is defined on the MIP image. They serve as starting points and guide the tracking

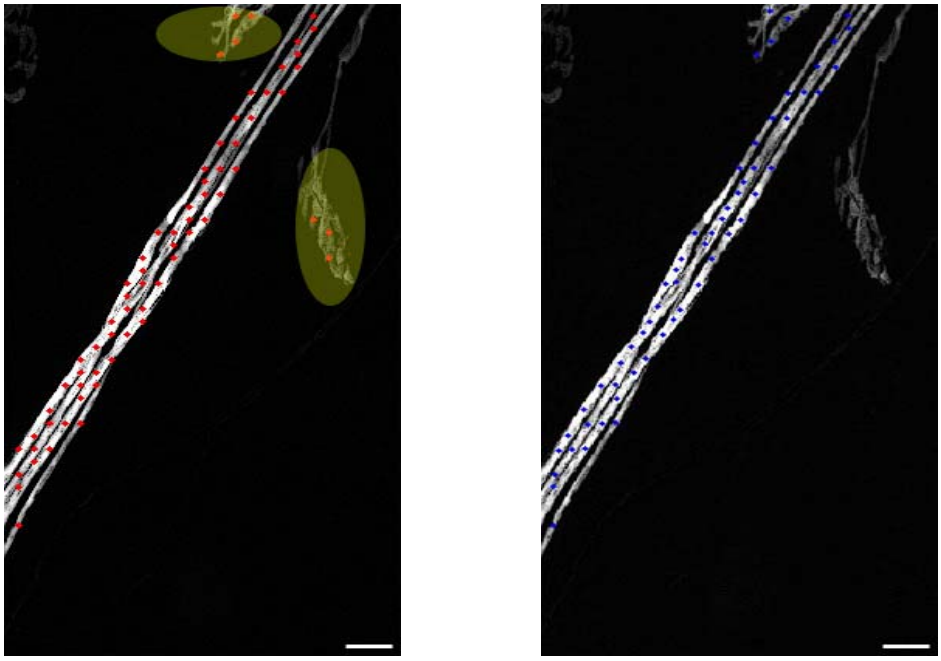
algorithm, and therefore must be inside the axons. Instead of searching the whole image for such points, a grid of predefined resolution, which is determined by the user, is specified on the MIP image. Only the intersection points of the horizontal and vertical grid lines are considered as candidate seed points. Therefore, the user must ensure that the resolution of the grid has to be such that at least a few such intersection points lie on each of the axons. From all the points on the grid, the ones used as seeds are detected based on a threshold applied to the intensity. This threshold is computed as:

$$T = \mu + \sigma \tag{1}$$

where μ and σ are the mean and standard deviation of the intensities of all the points lying on the intersection of the grid lines. In other words, all the points, from the set of the intersection points, whose intensity exceeds the threshold, are considered to be seed points. Due to noise or presence of structures, such as neuromuscular junctions, shown inside the yellow shaded regions in Figure 3(a), there are chances that some seed points are located outside the actual axons. If the algorithm begins at any of these points, a tracking error might result. To minimize such errors, the seed points are aligned towards the center of the axons by using the directional template based tracking method. Directional templates are correlated with the image at varying distances from the seed points to identify the edges of the axons at these locations. The seed points are then aligned towards the center by using the distance of the edges from the seed points.

It can be seen from Figure 3(a) that the neuromuscular junctions are faint and have irregular shapes. Due to this fact, the detected edges, with the seed points on the neuromuscular junctions as the center, will be prone to errors. If we align these seed points based on the left and right edges, the seed points will most likely now lie in the background. Therefore, after the alignment process, we once again use the threshold computed using Equation (1) to determine if the seed

points are lying on the axons or in the background. This minimizes the chances of seed points detected outside the axons, either due to noise or other structures in the image. Figure 3 shows the automatic detected seed points in the MIP image and their alignment towards the center.



(a) (b)

Figure 3 – Automatic detection and alignment of seed points: (a) seed points detected on a grid with a resolution of 20 pixels, and (b) Center-aligned seed points. The scale bars in the figures correspond to 6μm.

The two-dimensional MIP tracking method begins by defining directional templates for each of sixteen quantized directions for detecting the edges, angle of orientation, and the center points of axons present in the MIP image. As mentioned earlier, the seed points detected automatically in the MIP image are aligned towards the center of the axon on which they lie. With the first unaligned seed point as center, the edges of the corresponding axon are detected by correlating the templates with the image at varying distances from this seed point. The angle at which the template response is greatest is considered as the angle of orientation of the axon at that particular location. The distances from either side of the center at which the template response is greatest are considered as the edges. This edge information is then used to align the seed points

towards the center. Similarly, all the seed points in the MIP image are aligned towards the center of the axon on which they lie.

The axons in the MIP image are tracked individually using an iterative process. The algorithm is initiated at the first aligned seed point in the image. Using the angle of orientation of the axon at this particular location, the next center of the axon is predicted by moving by a predefined step size along this direction. The edges of the axons are then detected as mentioned earlier and any error in prediction is corrected by using the distance of the edges from the center. Since the axon thickness is known to be fairly smooth over small distances, the edge information of the axon in the previous iteration is used as a constraint to minimize errors. More specifically, the distance of the edges from the center in the current iteration are bounded to lie within a certain limit near the previously detected edge lengths. This limit is set equal to the step size of the algorithm, as it can be safely assumed that edge lengths corresponding to close lying centers on an axon have more similarity than those that are far spaced. An axon is said to be completely traced in the MIP image if the predicted center lies outside the image boundary or in the background.

During the axon tracking process, seed points are searched in the vicinity of the center, and any seed point found is labeled with the current axon number. Once a particular axon is completely traced, the algorithm begins with the next unlabeled seed point in the MIP image. All the axons are considered to be traced when every seed point in the image is labeled. Though this method is computationally efficient, it cannot track the centerlines when the axons seem to cross-over in the MIP image. Figure 4 shows the cross-over of axons.

2.2 Detection of centerlines using cross-sectional information

A *cross-over* is detected by the algorithm when the centerline of the current axon intersects a

traced axon in the MIP image. To deal with this situation, we have developed an algorithm that works directly on the cross-sectional information. Though this method is accurate, it is computationally expensive as compared the method described in the previous section. Therefore, it is applied to only a few sections in the dataset where there is an ambiguity due to the cross-over.

An outline of our processing scheme is as follows. First, as mentioned earlier, the cross-sectional images of axons in the dataset suffer from intensity non-uniformity and blurriness. Hence, all the cross-sectional images are preprocessed using the Hessian method of adaptive smoothing (Carmona et al., 1998). Besides removing noise from the image, this method also enhances image features. The axons are then segmented using the *seeded watershed algorithm* (Gonzalez et al., 1992) to find their centers in the cross-sections. Hence, another set of *seed-points* are introduced here that are used as the starting point for the segmentation algorithm. Unlike the previous use of seeds, the ones for this stage of the methods are specified within the cross-sectional images being analyzed and are used to roughly determine the centers of the multiple axons located within the cross section. The *mean-shift* algorithm (Debeir et al., 2005) is used to find the seed points. Finally, a *guided region growing* approach is developed to accurately segment the axons when the seeded watershed algorithm fails.

The process begins by identifying a cross-sectional slice where the axons in question are well-separated. Starting from the point of cross-over in the MIP image, a search is initiated to find a location where the centerlines of the axons are separated by more than a certain distance, d, defined as:

$$d = d_{current} + d_{traced} \tag{2}$$

where $d_{current}$ and d_{traced} are the diameters of the current and the intersecting axon. They are determined by using the left and right edge information from the template based MIP tracking

algorithm. The blue line in Figure 4 shows this location.

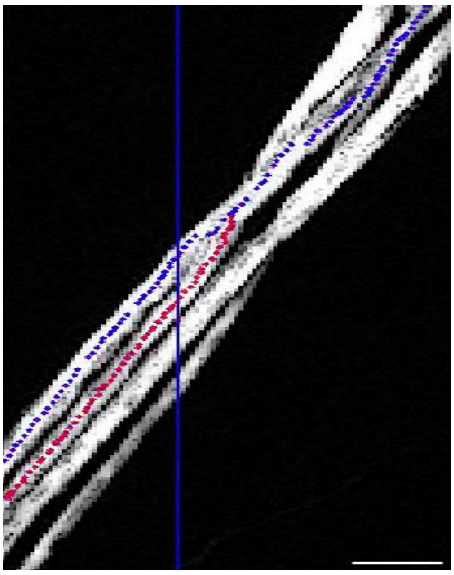


Figure 4 - The phenomenon of axon cross-over. The scale bar in the figure corresponds to 4μm. The figure has been magnified for better visualization.

Since each point on the centerlines of the axons in the MIP image correspond to a cross-sectional slice, as shown in Figure 1, we pull out the corresponding slice from the dataset to start the cross-sectional analysis. Once the axons are found to be well separated again in three-dimensional space, the template based tracking is used again to track the axons in the MIP image.

2.2.1 Tracking

The axons in the cross-sectional slices are segmented in order to find the centerlines of the axons. Apart from providing a good visualization of the axon boundaries in the cross-sections, segmentation helps in the detection of the centers of axons when they have irregular shapes. To avoid errors in finding the centerlines, all the axons present in the current slice, including those that are not yet tracked in MIP image, are tracked together. In order to find the starting points for the segmentation algorithm, the approximate center points have to be estimated. Thus we

introduce another set of seed points that initiate the segmentation.

We begin the analysis by finding the centers of the axons in the first cross-sectional slice. The centers of the axons traced in the MIP image can be easily found by searching for local intensity maxima in that slice. But, as seen in Figure 4, not all the axons are usually traced in the MIP image before encountering a cross-over. Therefore, the seed points for the rest of the axons in the first image are identified manually. These serve as the initial seed points for the next slice in the dataset. The axon cross-sections in the current slice will most likely be shifted from their positions in the previous slice. So, the centers from the previous cross-section cannot be directly used as seeds, and need correction. The mean-shift algorithm is used for this purpose.

Mean-Shift

The mean-shift makes use of the intensity information to find the seed points in the image. It basically tries to find the approximate center points by searching for the center inside a moving kernel, which is a disc shaped region. The kernel is first placed at the initial seed points obtained directly from the previous cross-section. The center inside the kernel is calculated based on the weighted mean with the intensity as the weights. The seed point is then updated with the mean calculated in the current iteration. The kernel is now centered at the new seed point and the computation is repeated again until the algorithm converges or until the number of iterations reaches a preset threshold.

The mean shift algorithm fails when close-lying axons have a considerable difference in their intensities. The algorithm tends to move towards the brighter axon in such cases, which results in seed point being placed near the edges or sometimes entirely outside the actual axon. This is shown in Figure 6(a) inside the yellow ellipse. The corresponding segmentation result using seeded watershed algorithm is shown in Figure 6(b).

To avoid this problem, we introduce constraints in the algorithm that use the information from the segmented regions in the previous slice. First, the radius of the kernel is determined as:

$$R = \frac{\min(r, a)}{2} \tag{3}$$

where, r is the radius of a circle with the same area as the segmented region and a is the minor axis after fitting an ellipse to the segmented region. This helps in minimizing the errors in finding the seed points when the axons have irregular shapes.

The mean shift is then applied inside a window instead of the whole image. This constraint ensures that the seed points do not move out of the actual axon. The size of the window is set to twice the diameter of a circle that has the same area as the segmented axon cross-section in the previous slice. This is a reasonable assumption since the changes in axon position from one slice to the next is known to be smooth. Figure 5 shows the trajectory of the mean-shift for an axon inside a window. The green cross indicates the starting point of the mean-shift. The trajectory of the algorithm is shown by the series of Blue crosses. The Red cross indicates the point to which the mean-shift converges.

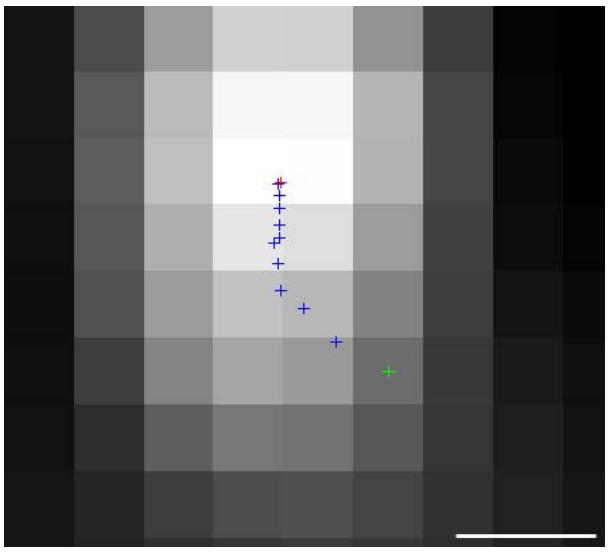


Figure 5 - The trajectory of the mean-shift algorithm. The scale bar in the figure corresponds to 0.2μm. Since this is a magnified version of the actual image, the pixels are seen as block-like structures.

The seeds found by the mean-shift algorithm using the constraints is shown in Figure 6(c) and the corresponding segmentation result using seeded watershed algorithm is shown in Figure 6(d).

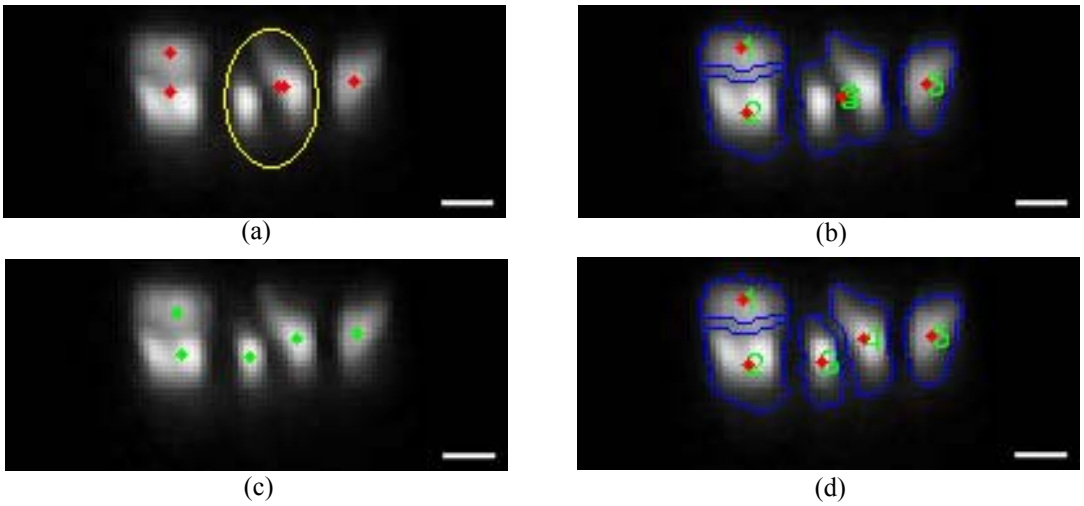
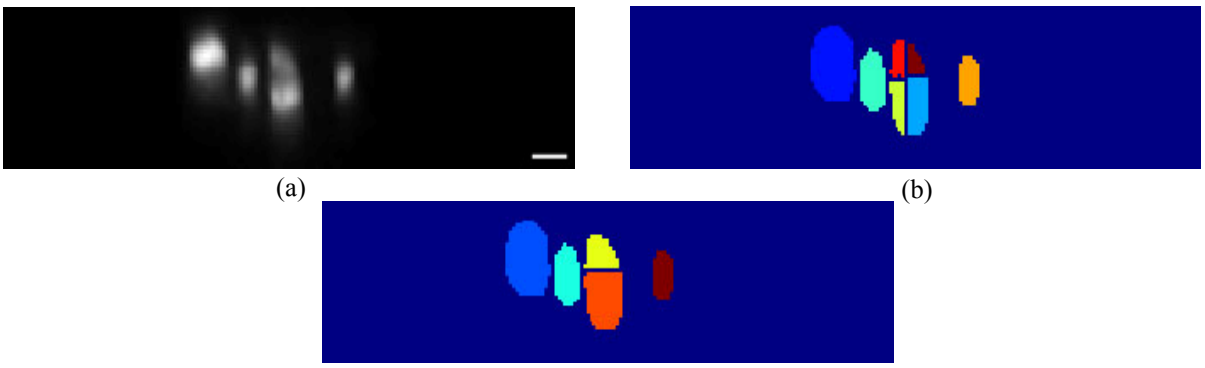


Figure 6 - Seeds estimated by the mean-shift algorithm: (a) Wrongly estimated seeds, shown inside the yellow ellipse, found without constraints, (b) Segmentation results using seeds in (a), (c) Seeds found using constraints, and (d) Segmentation results using seeds in (c). The scale bars in the figures correspond to 1μm.

2.2.2 Segmentation of axon cross-sections

Once the seed points are estimated, the axon cross-sections are segmented using the seeded watershed algorithm. The seeds act as the starting points for the segmentation algorithm and also prevent over-segmentation, a common problem with watershed algorithm. Figure 7 compares the segmentation results of five axons with and without the introduction of seeds.



(a) (b)

(c)

Figure 7 - Segmentation of axons using watershed: (a) Original Image, (b) Segmentation without imposition of seeds, and (c) Segmentation using watershed after imposition of seeds. The scale bar in (a) corresponds to 1μm.

The traditional watershed algorithm does not use a priori information during segmentation. Because it is based only on intensity values in the image, it fails to segment close-lying or close-over axon cross-sections with a large difference in intensities. One approach to address this issue makes use of *seed regions* rather than seed points. Figure 9(b) shows the seed regions initialized to an ellipse whose major and minor axis are set to half their values determined in the previous slice, centered at the seed points detected by the mean-shift algorithm. But, as shown in Figure 9(c) inside the yellow ellipse, the seeded watershed algorithm is unable to resolve the axons.

We have developed a new algorithm to accurately segment the axons in such cases. The developed method uses the watershed algorithm as its basis, but guided by constraints. The guided region growing algorithm is based on both object features and intensity information of the objects in the image. Roughly speaking, we use the watershed method as a preprocessing stage to identify candidate pixels that *could* be added to a segment. A probabilistic approach based on likelihood evaluation using the features to be described below is then used to evaluate the suitability of adding a given pixels to a specific segment. We note that this method is based on an underlying assumption (born out in practice) that the cross sectional structures of axons do not vary drastically from one cross-sectional slice to the next. Hence, a reasonable amount of

similarity can be assumed between consecutive cross-sections. The various parameters involved in the algorithm are discussed here.

Deciding the training set 'N'

Selecting the number of training samples used to estimate statistics of our features is an important decision. A large number of samples will average out the variations in the cross-sections, resulting in improper segmentation while fewer samples mean a larger variance, which causes increased likelihoods to lie in either of the two extremes. In our case, we have chosen training samples from 10 previously segmented slices. We emphasize that this choice was employed quite successfully across the three different data sets analyzed in Section 3.

Deciding the features

The features are extracted from previously segmented slices by fitting an ellipse to the segmented axons. Popular features that are usually used for training a model such as this are *length of major axis, length of minor axis, perimeter, average intensity, difference in centers from one cross-section to the next, area and orientation* (Li et al., 2007; Wang et al., 2007). Since the axon cross-sections have irregular shapes and varying intensity, only a few features can actually be useful in our case. It should be noted that even though the cross-sections have irregular shapes, the change in their shape from one cross-sectional image to the next is not abrupt. The features have to be selected such that the likelihood increases only if the region grows into a particular shape and at a particular angle. We have found that orientation and perimeter are sufficient features to properly guide the algorithm to converge to the actual shape of the axon in the cross-section.

Building the probabilistic feature model

After extracting the features from the previously segmented images, they are fit to Gaussian

distribution and used to train the model. The mean and the covariance matrices for each axon are calculated using these features. The similarity of the shape and size of an axon cross-section between two slices is dependent on the distance between them. In other words, the similarity between the cross-sections of an axon lying close together is more than those that are far-spaced. Thus, the feature vectors used for training are weighted to imply the similarity between the objects in the current slice and the training feature. These weights are set to: $w_n = e^{(N-n+1)}$, where *n* is the distance between the current slice and the slice where the features were extracted, and *N* is the number of slices used to train the model. As it can be seen, the influence of the feature vectors decreases exponentially as we move away from the current slice. The weights are then normalized as: $\hat{w}_n = w_n / W$, where $W = \sum_{n=1}^{N} w_n$.

The weighted mean for the feature vectors are then calculated as:

$$\vec{\mu}_w = \sum_{n=1}^{N} \hat{w}_n . \vec{f}_n \tag{4}$$

where, $\vec{f}_n$ is the feature vector at *n* slices away from the current slice. The weighted covariance matrix is then computed as:

$$\vec{R}_w = \sum_{n=1}^{N} \hat{w}_n . (\vec{f}_n - \vec{\mu}_w) * (\vec{f}_n - \vec{\mu}_w)^T \tag{5}$$

Once the model is built using Equations (4) and (5), it acts as the central force driving the region growing. The likelihood of occurrence of a region with the feature vector, $\vec{f}$, in terms of the probabilistic model can be written as:

$$p_{\vec{f}} = \frac{1}{2\pi \left| \vec{R}_w \right|^{1/2}} e^{-\frac{1}{2}\left[(\vec{f} - \vec{\mu}_w)^T * \vec{R}_w^{-1} * (\vec{f} - \vec{\mu}_w) \right]} \tag{6}$$

Guided region growing

Our algorithm starts with the seed regions in the image as defined in Figure 9(b). As pixels are

analyzed, they are marked with a label that indicates the region to which they belong. The water is filled in the image based on a threshold that is increased in subsequent iterations. In order to make the algorithm less computationally demanding, only the pixels lying inside a *region of interest* are considered for analysis. This is computed by binarizing the image. If Otsu's method (Otsu, 1979), a popular algorithm for determining the threshold for binarization, were to be used, the axons with low intensity are not properly binarized.

In order to correct this, we adopt fuzzy c-means clustering (Dunn, 1973) to compute the threshold. The first step is to cluster the image pixels into three groups based on their intensity. The threshold is then calculated by averaging the highest intensity pixel of the lowest intensity group and the lowest intensity pixel of the medium intensity group, which can be written as:

$$T = \frac{(I_{low} + I_{mid})}{2} \tag{7}$$

where, I^{low} is the highest intensity pixel of the lowest intensity group and I^{mid} is the lowest intensity pixel of the medium intensity pixel group. This makes sure that the threshold is low enough to properly binarize the fainter axon cross-sections. Figure 8 compares the binarized images obtained using the thresholds determined by Otsu's method and the fuzzy c-means clustering method. It can be easily seen, by comparing Figure 8(b) and (c), that the threshold computed using Equation (7) is better in binarizing fainter axons in the image.

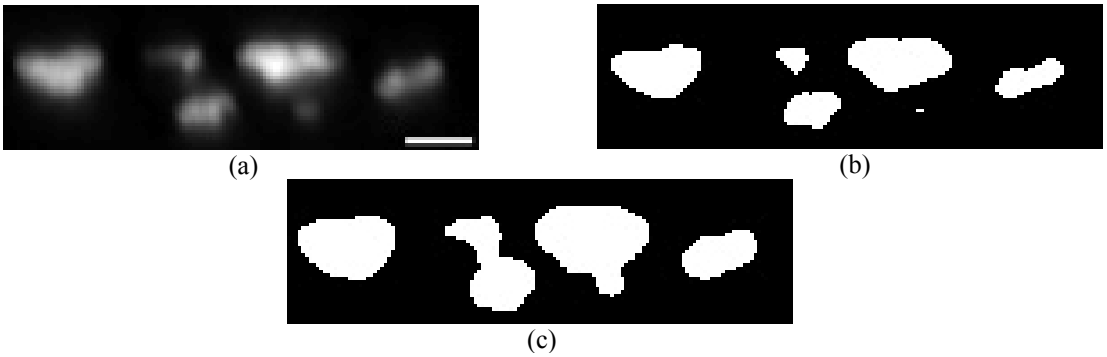


Figure 8 – Determining the threshold for binarization of cross-sectional slices: (a) Original Image, (b) Binarization using Otsu's method, and (c) Binarization using Fuzzy c-means clustering. The scale bar in (a) corresponds to 2μm.

After the region of interest is identified, the threshold for guided region growing algorithm in the first iteration is set to one level higher than the lowest intensity pixel in the region of interest. All the pixels in the image that are lesser than this threshold are considered to be submerged in water in the current iteration. A list of unlabeled pixels is maintained for all the pixels that are submerged so far. For each of the regions in the image, connected components are searched from this list. From the list of pixels in the connected component list, individual pixels are added to the region and the likelihood of occurrence of the new region is computed. The pixels are retained only if there is an increase in the likelihood. The same process is repeated until the threshold reaches to the maximum value of the pixel in the region of interest. Since these pixels are not reordered to find alternate paths for region growing, the algorithm introduced here is a greedy one. In Figure 9(d)-(e), we compare the segmentation results of the seeded watershed algorithm and the guided region growing algorithm.

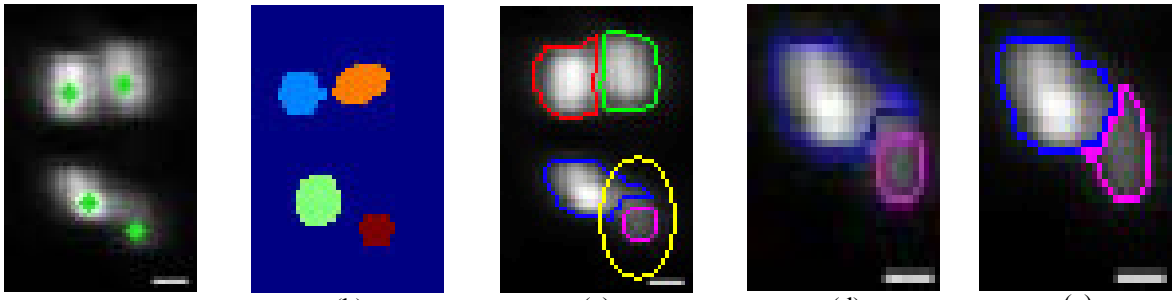


Figure 9 - Segmentation of axons using guided region growing: (a) Original image with seeds found using mean-shift, (b) Seed regions, (c) Boundaries using seeded watershed algorithm with seed regions, (d) Magnified version of boundaries in (c), and (e) Boundaries detected with guided region growing. Scale bars in the figures correspond to 2μm.

It should be noted that the accurate segmentation of the faint and the small axons is crucial in preventing the propagation of errors in tracking to the next slice. Such errors can adversely impact the quantitative analyses of the underlying biological questions.

Once the hybrid algorithm tracks all the axons in the dataset, two kinds of segments of centerlines result: fragments that were tracked using the template-based approach and those that were tracked using a sequence of cross-sectional slices. In order to build the 3D model of the centerlines, we need to locate the third dimension of each of these centers. Since the centerlines tracked using the template-based approach have no cross-over ambiguity, the third dimension can be easily found by searching for the local maximum intensity pixel in the corresponding slices in the dataset. In the latter case, the two dimensions of the centers of the segmented regions, along with the slice number, give us the three dimensions of the centers. The results of the 3D reconstruction of centerlines of axons in three datasets are shown in the next section. The results are validated with the manual tracking results and are compared with the repulsive snake algorithm for robustness.

3. Results

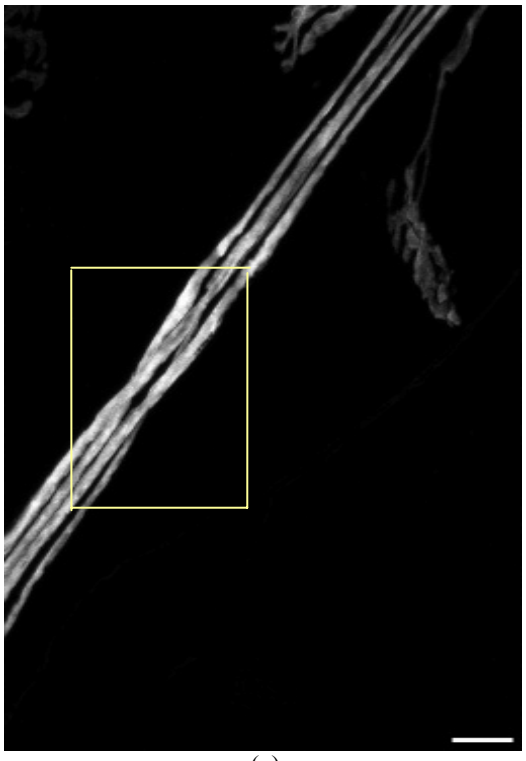
Three datasets were analyzed using the hybrid algorithm proposed in this paper. The centerlines of the axons are presented in both two dimensions in the MIP image and in the three-dimensional domain. The various parameters that were manually initialized for the datasets are:

- The number of cross-sectional slices in the dataset.
- The MIP image of the dataset to be analyzed and the approximate maximum width of the axons in the MIP image.
- The resolution of the grid for the automatic detection of seeds in the MIP image for the 2D directional template based tracking and the step size of the algorithm.
- The initial seed points for the segmentation algorithm in the first axon cross-section each time when an axon cross-over is detected.

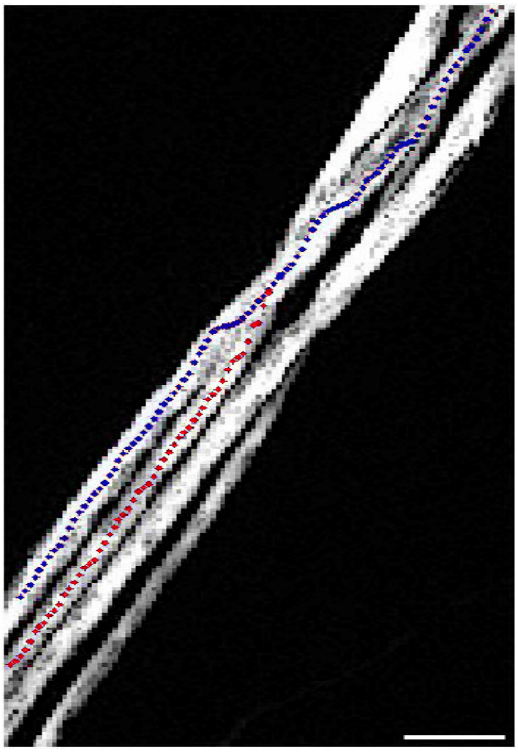
The number of cross-sections to be analyzed by the algorithm is assumed to be known by the user. The user also needs to identify the best MIP image for analysis by looking at the MIP images along the three orthogonal directions. A rough estimate of the width of the biggest axon in this MIP image has to be specified by the user. The resolution of the grid for the detection of seed points in the MIP image needs to be defined such that there are at least a few candidate seed points on each of the axons in the MIP image. Small values of step size of the MIP-based tracking algorithm results in higher accuracy in tracking, but increases the execution time of the algorithm. On the other hand, large values lead to faster tracking, but can result in tracking errors and a coarse centerline. In all the datasets presented in this paper, this value was set to 3 pixels. Once the algorithm switches to the cross-sectional tracking, the user has to specify the seed points for the cross-sections of the axons that have not yet been tracked in the MIP image. This is done only for the first cross-sectional image where the algorithm switches to a different method of analysis.

The results of our algorithm were validated with the manual reconstruction results that were obtained using Reconstruct. Finally, the performance of the guided region growing algorithm is compared with the repulsive snake model on one cross-sectional image from each of the three datasets.

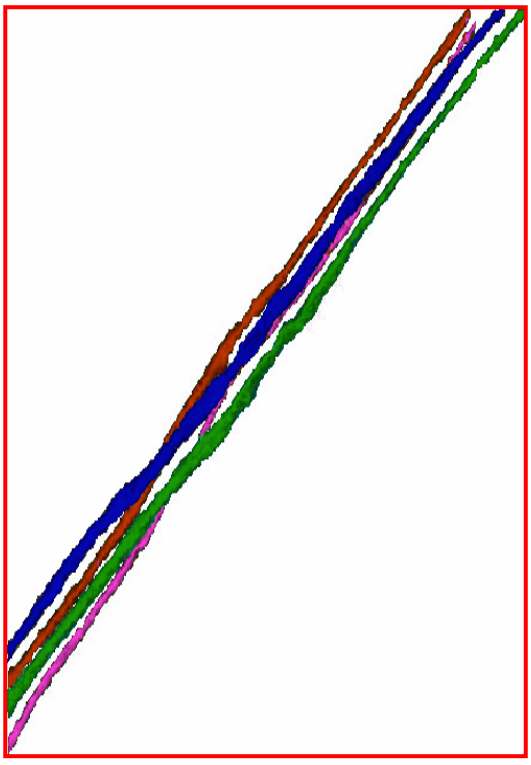
The first dataset contained four axons in total with 256 cross-sectional images, each having a resolution of 141 x 512 pixels. The cross-sectional images were sampled along the Y direction. Figure 10(a)-(b) show the MIP image and the cross-over of axons in the MIP image. The maximum width of the axon was set to twenty pixels and the resolution of the grid for the MIP image was set to ten pixels. Figure 10(c) shows the manual reconstruction of the axons. The 2D MIP and the 3D model of the centerlines are shown in Figure 10(d)-(e).



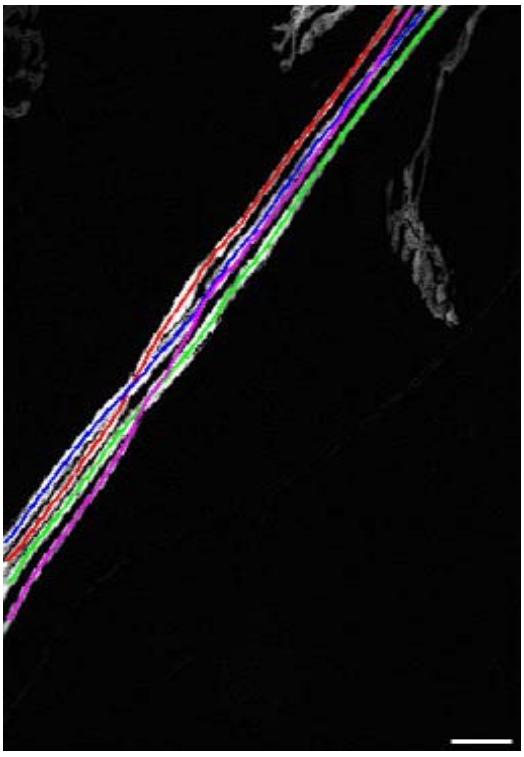
(a)



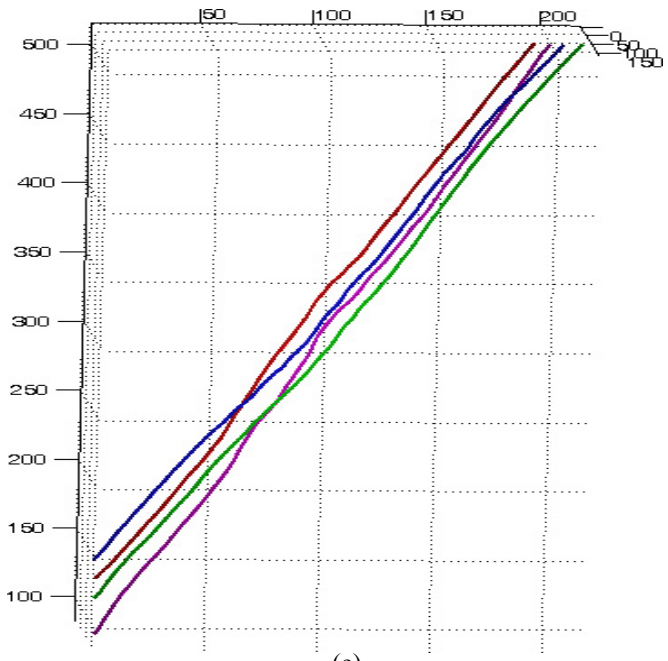
(b)



(c)



(d)

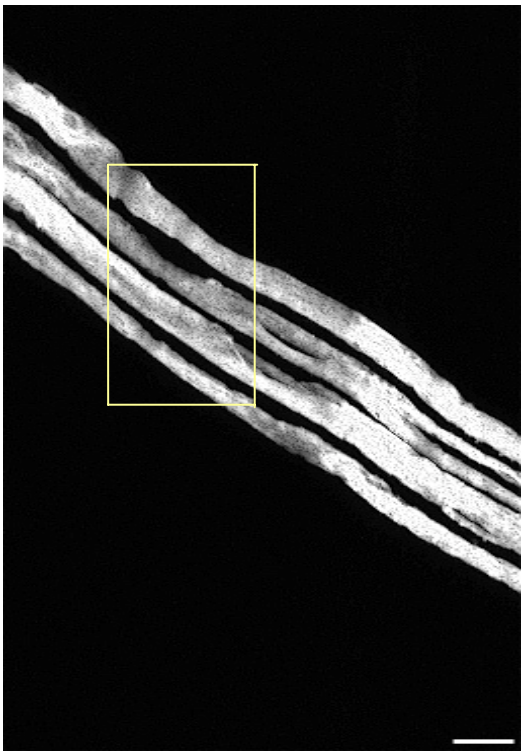


(e)

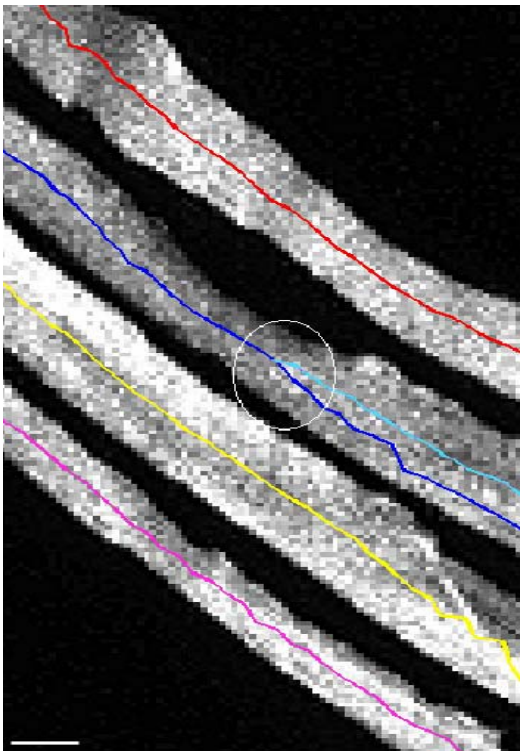
Figure 10 – Tracking results in dataset one: (a) The MIP image of the dataset, (b) Cross-over in the MIP image (The region inside the yellow rectangle in (a) has been magnified for better visualization), (c) Manual tracking results, (d) Centerlines in MIP image, and (e) 3D model of the centerlines. The scale bars in (a) and (d) correspond to 6μm. The scale bar in (b) corresponds to 4μm.

Approximately 50% of the dataset was tracked using the cross-sectional tracking algorithm. The average deviation of the centerlines detected by our algorithm from the manually tracked results was found to be 1.75 pixels.

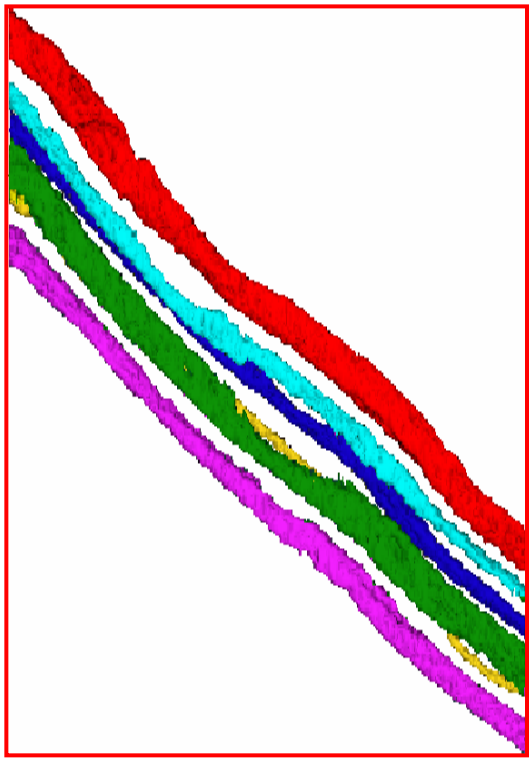
The second dataset contained six axons with 256 cross-sectional images, each having a resolution of 43 x 512 pixels. The cross-sectional images were sampled along the X direction. Figure 11(a) shows the MIP image for this dataset. The cross-over of the axons in the MIP image is shown inside the white ellipse in Figure 11(b). The maximum width of the axon was set to forty pixels while the resolution of the grid for the MIP image was set to fifteen pixels. Figure 11(c) shows the manual tracking results for the dataset. The 2D and the 3D centerlines are shown in Figure 11(d)-(e).



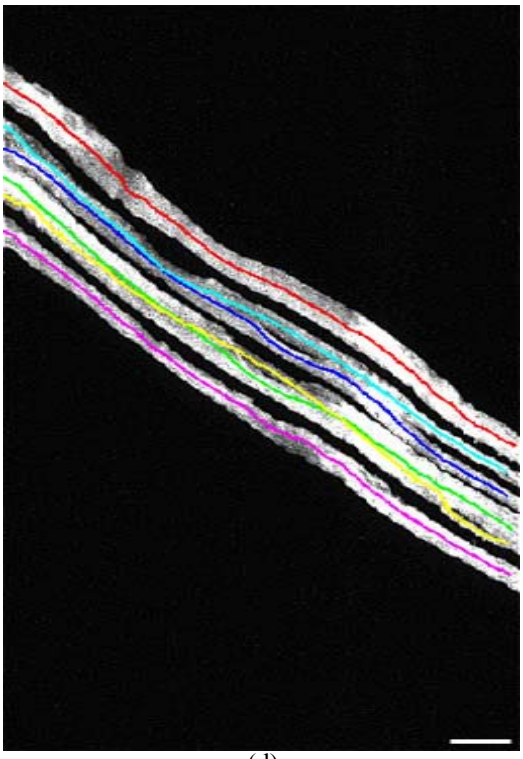
(a)



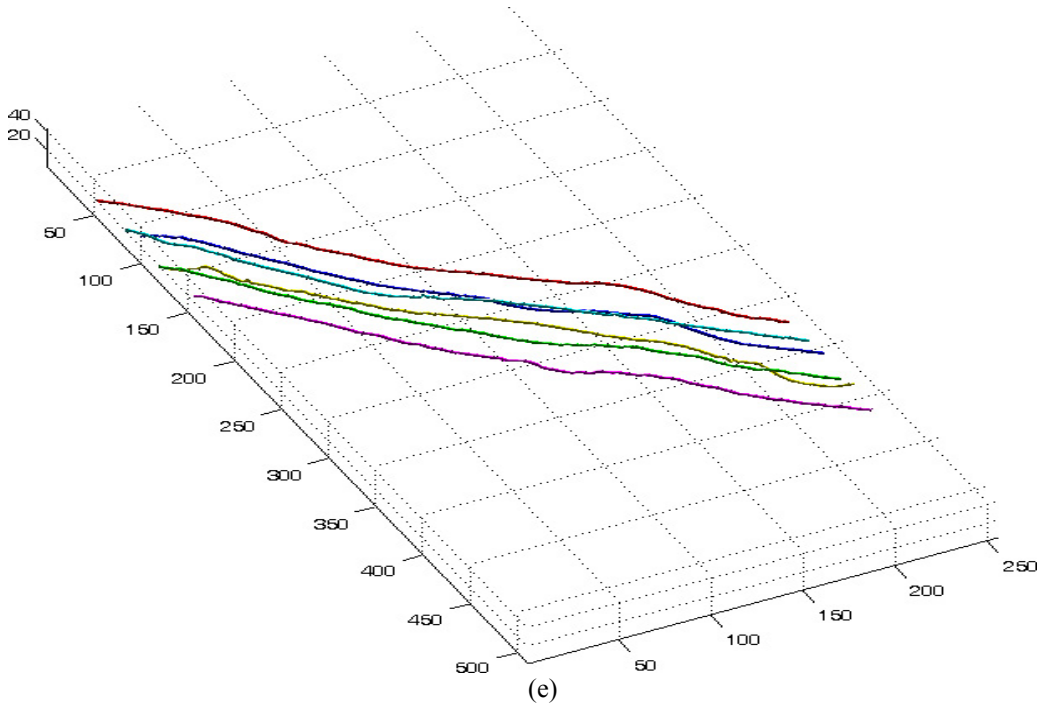
(b)



(c)



(d)



(e)

Figure 11 – Tracking results in dataset two: (a) The MIP image of the dataset, (b) Cross-over in the MIP image (The region inside the yellow rectangle in (a) has been magnified for better visualization), (c) Manual tracking results, (d) Centerlines in MIP image, and (e) 3D model of the centerlines The scale bars .in (a) and (d) correspond to 6µm. The scale bar in (b) corresponds to 2µm.

Approximately 75% of this dataset was tracked using the cross-sectional tracking algorithm. The average deviation of the centerlines detected by our algorithm from the manually tracked results was found to be 1.64 pixels.

As the complexity of the dataset increases, a major part of the centerlines of the axons are tracked using the cross-sectional information. The final dataset contained seven axons with 256 cross-sectional images, each having a resolution of 42 x 512 pixels. The cross-sectional images were sampled along the X direction. Due to the complex nature of the dataset, the centerlines of the axons are tracked entirely using the cross-sectional images. Hence, the centers of the axons in the first cross-sectional image were manually identified. Figure 12(a) shows the MIP image for the dataset. The manual tracking results for this dataset is shown in Figure 12(b). Figure 12(c)-(d) show the 2D MIP and 3D model of the centerlines of the axons.

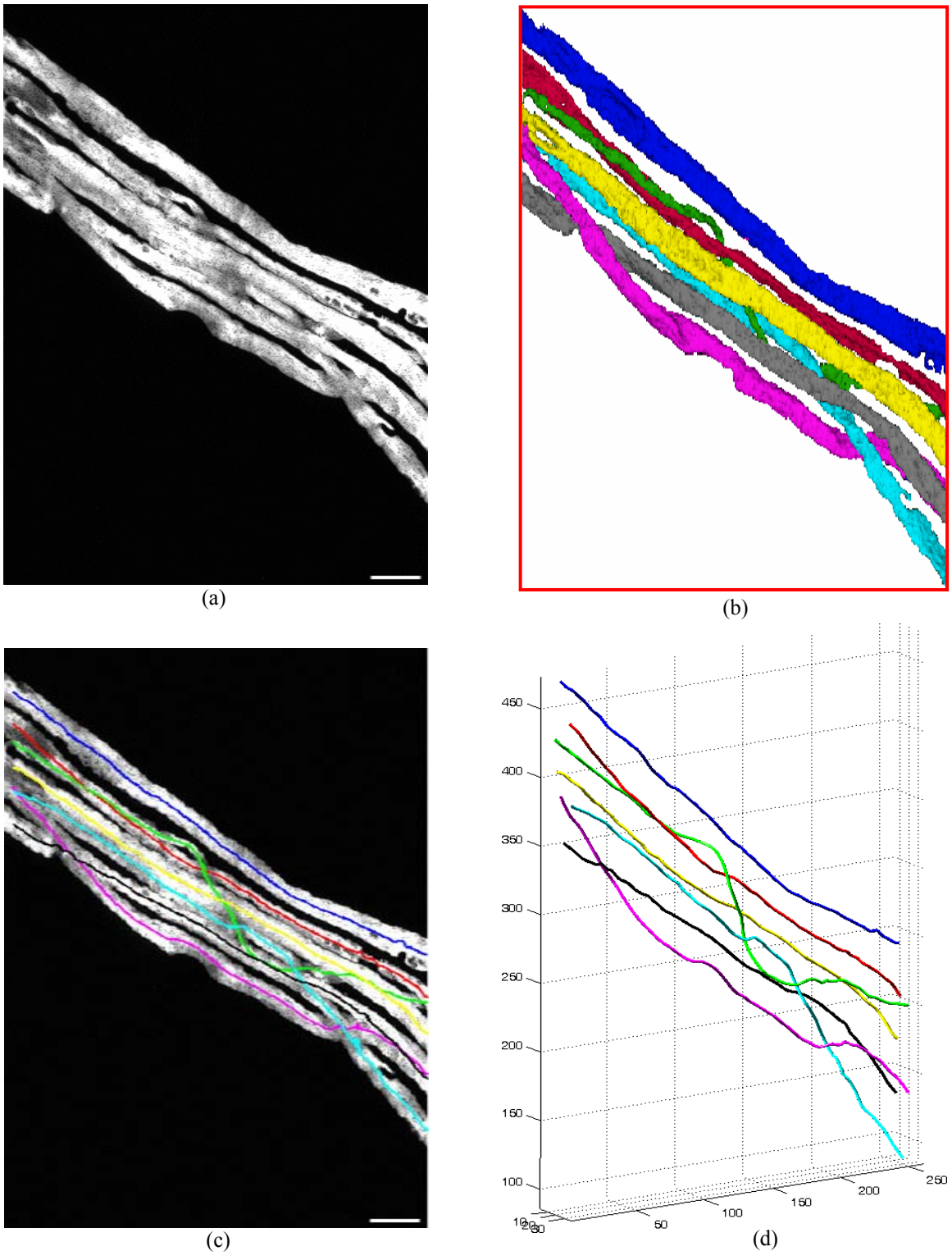
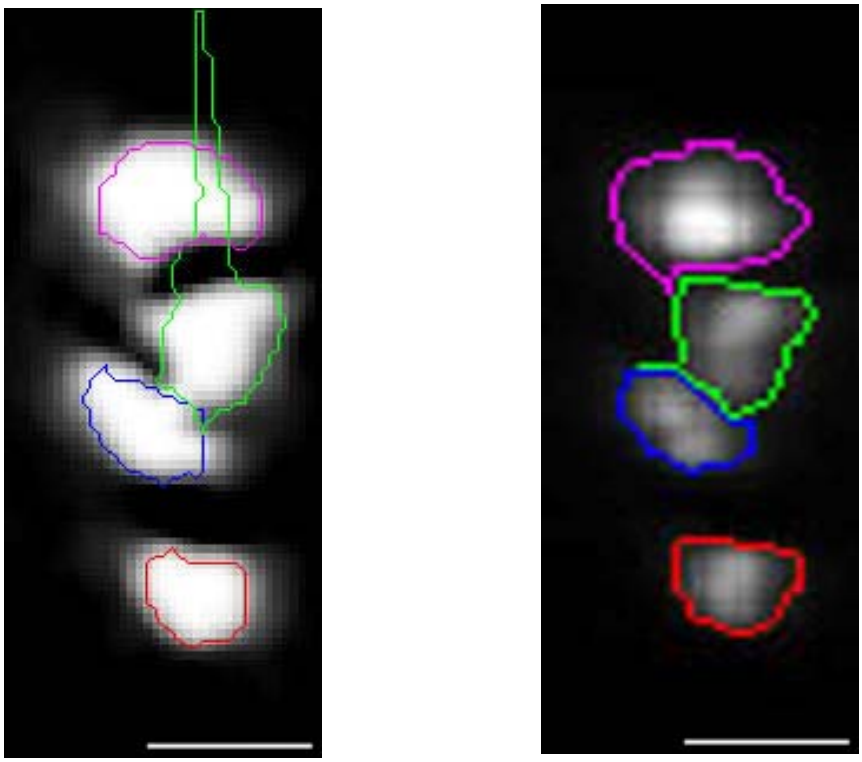


Figure 12 – Tracking results in dataset three: (a) The MIP image of the dataset, (b) Manual tracking results, (c) Centerlines in MIP image, and (d) 3D model of the centerlines. The scale bars in (a) and (c) correspond to 6μm.

In comparison with the manual tracking results, the average deviation of the tracked axon centerlines in the third dataset was found to be 1.81 pixels.

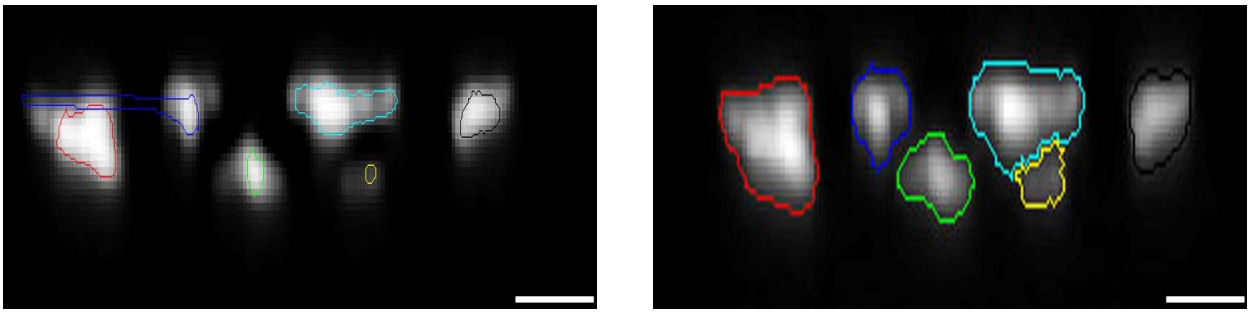
In order to demonstrate the accuracy and robustness of the proposed method in this paper, the segmentation results are compared with that of the repulsive snake model. In this algorithm, the user has to define the initial boundaries and the centers of the axons in the first cross-sectional image of the dataset. The snake then iteratively evolves to the individual boundaries of the axons by minimizing the energy functional. The algorithms were executed on a PC with a 1.6 GHz Intel Pentium processor and 1.2 Gigabytes of memory. Since the datasets mentioned earlier in this section contain hundreds of cross-sectional images, one cross-section from each of the datasets, where a significant difference in segmentation can be noticed, is used to compare the two algorithms. Figure 13 shows the results of the two algorithms in the first dataset.



(a) (b)

Figure 13 - Comparison of segmentation results in dataset one: (a) Repulsive Snake Model, and (b) Guided region growing. The scale bars in the figures correspond to 2μm. The figures have been magnified for better visualization.

In this dataset, the repulsive snake algorithm was able to successfully track the axons in ten consecutive slices. The execution time for ten slices was used to predict the time taken to analyze the entire dataset, and was found to be approximately 60 minutes. On the other hand, the proposed algorithm analyzed this dataset in approximately 20 minutes. It can be seen that the boundaries of axons detected using the repulsive snake model are not accurate when the axonal cross-sections suffer from intensity non-uniformity. Hence, the centers of the axons found in this manner are inaccurate which results in the propagation of error to subsequent slices. The guided region growing, on the other hand, is more robust in handling such situations. Figure 14 compares the two algorithms in another image from the second dataset.



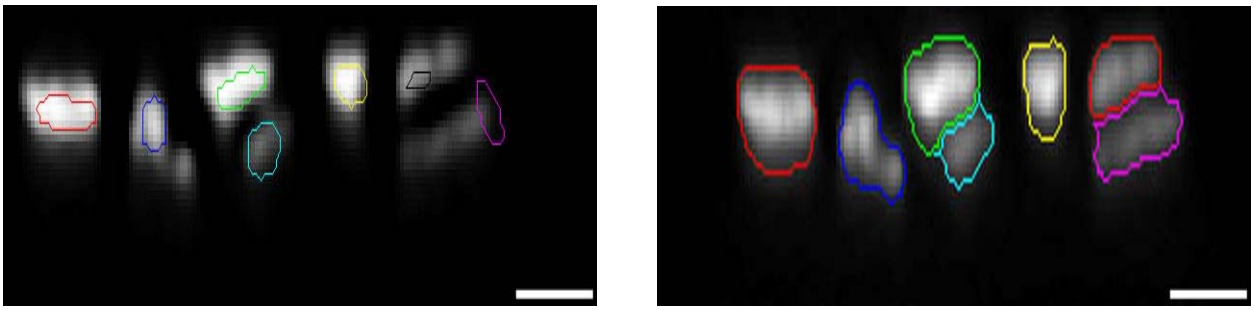
(a) (b)

Figure 14 - Comparison of segmentation results in dataset two: (a) Repulsive Snake Model, and (b) Guided region growing. The scale bars in the figures correspond to 2μm. The figures have been magnified for better visualization.

Due to the imaging artifacts, the repulsive snake model is unable to evolve to the true boundaries of the axons in the image. This is overcome by the guided region growing algorithm by using the shape information from the previous slices. The repulsive snake algorithm failed after two slices in the second dataset. Using the execution time of repulsive snake algorithm in the two slices, the predicted time for analyzing the complete dataset was approximately 78 minutes. Whereas, the proposed hybrid algorithm was able track the axon centerlines in the entire dataset in approximately 16 minutes.

Figure 15 compares the two algorithms in the final dataset. Here, the repulsive snake

algorithm worked for five consecutive slices in the dataset. Based on the execution time of repulsive snake algorithm for five slices, the time taken to analyze the whole dataset was predicted to be approximately 84 minutes. The hybrid algorithm, on the other hand, was able to track the centerlines of the axons in the entire dataset in approximately 24 minutes.



(a) (b)

Figure 15 - Comparison of segmentation results in dataset three: (a) Repulsive Snake Model, and (b) Guided region growing. The scale bars in the figures correspond to 2μm. The figures have been magnified for better visualization.

4. Discussion

Based on the nature of the axons in the MIP image, the datasets can be broadly categorized into three groups:

1. Axons running mostly parallel to each other, with a few intertwined axons.
2. Axons running in bundles, with intertwined axons.
3. Axons with branches.

The first two cases have been examined in the paper. The first two datasets analyzed in the paper fall under the first category, whereas the third dataset belongs to the second group. When the axons run in thick bundles, the boundaries of the axons may not be clearly visible in the MIP image. In such a situation, the entire dataset can be tracked in the cross-sectional domain, as was done in the third dataset. Although the proposed algorithm cannot track axons with branches, we believe that a few changes can help it track such structures.

In the datasets containing branching structures, the hybrid algorithm will switch to the cross-sectional analysis only where an axon cross-over is encountered or at axon branch points. Once the algorithm switches to the cross-sectional analysis, these two cases can be easily distinguished from each other. If a branch point is encountered while tracking in the cross-sectional sequence, there will be a significant change in the size of the axon cross-section, since the axon cross-section will split into two parts. The user can then manually identify if this change is due to a branch point or caused by a sudden change of orientation of the axon with respect to the imaging plane.

The algorithm proposed in the paper requires the structures in the dataset not to be oriented along the imaging plane. This is required since if the axons are aligned parallel to the imaging plane, their cross-sections randomly appear and vanish if the images in the dataset are viewed in a sequence. When the axons are aligned this way, partially or completely, the following changes can improve the algorithm's ability to adapt:

- If all the axons in the dataset are aligned along the imaging plane, the dataset can be re-sampled normal to the imaging plane and the cross-sectional analysis can be performed in this direction. If some parts of the axon bundles are oriented along the imaging plane, the dataset can be subdivided into smaller subsets and each of them can be individually analyzed. By looking at the MIP image, the user can determine which subsets of the dataset need to be re-sampled. The centerlines from each of the subsets can then be merged together to form the complete centerline model.
- Another way to solve this problem is to adopt cross-sectional analysis throughout the dataset and to re-slice the dataset at every step of the process. Once the orientation of an axon is found at a particular point, the next cross-sectional slice is determined after slicing the

dataset in a direction normal to the orientation of the axon. This ensures that the axon cross-sections do not randomly appear and vanish in the cross-sectional image sequence due to their orientation with respect to the imaging plane. Although relatively more expensive than the previously mentioned approach, this method does not require manual intervention except in the first cross-sectional slice where the user has to specify the centers of the axons.

A situation where the proposed hybrid algorithm fails is when the axons completely overlap in the MIP image. In such a case, our algorithm does not detect a cross-over and hence will not switch to the cross-sectional analysis. This way, one or some of the axons in the MIP image might remain untraced altogether. In such rare conditions, the problem can be circumvented by analyzing the whole dataset using cross-sectional analysis. An interesting case one might come across is when the structure size approaches the voxel size. Our algorithm can still be applied in such a case. As mentioned earlier in the paper, to segment the axon cross-section, we have to first detect the seed points. If the structure size approached the voxel size, the seed points can still be detected since they are detected inside a window. The window parameters are determined based on the segmentation result of the previous size. If the window is properly defined, the mean-shift algorithm can accurately determine the seed points. The guided region growing algorithm or the watershed can then easily determine the boundary.

5. Conclusion

From the datasets presented in this paper, it can be seen that the images have high signal to noise ratio, which makes it relatively easy to segment axons as a whole from the background. However, in many instances, axons tend to get close to each other, which make the process of tracking difficult. Significant problems in segmentation and tracking also arise when the intensity of the axon becomes weak. The hybrid algorithm presented in this paper is able to

successfully track the centerlines of such axons in a sequence of cross-sectional images in the presence of both such confounding circumstances. The goal of the algorithm introduced in this paper was to reconstruct 3D medial axes of the axons using the stack of microscopic images. The information available from the 3D reconstruction of the centerlines of the axons helps neurologists understand how neurons interact and control muscular retraction in mammals.

A significant improvement over seeded watershed algorithm was demonstrated using the guided region growing method. The performance of our algorithm was compared with the repulsive snake algorithm and was shown to be more accurate in finding the boundaries of the axonal cross-sections in spite of the presence of imaging artifacts. The future work would involve tracking axons in more complex datasets that contain branching patterns and run in thick bundles. Besides tracking the centerlines of axons, the proposed algorithm potentially can be generalized for extracting centerlines of other tubular objects in a 3D volume consisting of a sequence of cross-sectional images.

Information Sharing Statement

The sample image stacks and the software developed in this paper are available upon request.

ACKNOWLEDGEMENTS

The authors would like to thank Mr. H.M. Cai for the help in testing the data sets using the program of Repulsive Snake Model. They also would like to thank research members of the Life Science Imaging Group of the Center for Bioinformatics (CBI), Harvard Center for Neurodegeneration and Repair (HCNR) and Brigham and Women's Hospital, Harvard Medical School, for their technical comments. The research is funded by the HCNR, Harvard Medical School (Wong).

References

Abramoff M.D., Magelhaes, P.J. and Ram S.J. (2004) Image processing with ImageJ. Biophotonics International. 11(7), 36-42.

Al-Kofahi K.A., Lasek S., Szarowski D.H., Pace C.J., Nagy G., Turner J.N. and Roysam B. (2002) Rapid automated three-dimensional tracing of neurons from confocal image stacks. IEEE Transactions on Information Technology in Biomedicine. 6(2), 171-187.

Barrett W.A. and Mortensen E.N. (1997) Interactive live-wire boundary extraction. Medical Image Analysis. 1(4), 331-341.

Cai H., Xu X., Lu J., Lichtman J.W., Yung S.P. and Wong S.T. (2006) Repulsive force based snake model to segment and track neuronal axons in 3D microscopy image stacks. NeuroImage. 32(4), 1608-1620.

Can A., Shen H., Turner J.N., Tanenbaum H.L. and Roysam B. (1999) Rapid automated tracing and feature extraction from retinal fundus images using direct exploratory algorithm. IEEE Transactions on Information Technology in Biomedicine. 3(2), 125-138.

Carmona R.A. and Zhong S. (1998) Adaptive smoothing respecting feature directions. IEEE Transactions on Image Processing. 7(3), 353-358.

Debeir O., Van Ham P., Kiss R. and Decaestecker C. (2005) Tracking of migrating cells under phase-contrast video microscopy with combined mean-shift processes. IEEE Trans. Med. Imaging. 24(6), 697-711.

Dunn J.C. (1973) A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. Journal of Cybernetics. 3, 32-57.

Fiala J.C. (2005) Reconstruct: a free editor for serial section microscopy. J. Microsc. 218(1), 52-61.

Gonzalez R. and Woods R. (1992) Digital image processing. 2nd ed., Addison Wesley, 617-626.

Kass M., Witkin A. and Terzopoulos D. (1987) Snakes: Active contour models. International Journal of Computer Vision. 1(4), 321-331.

Kasthuri, N. and Lichtman J.W. (2003) The role of neuronal identity in synaptic competition. Letters to Nature. 424(6947), 426-30.

Keller-Peck C.R., Walsh M.K., Gan W.B., Feng G., Sanes J.R., and Lichtman J.W. (2001) Asynchronous synapse elimination in neonatal motor units: studies using GFP transgenic mice. Neuron. 31(3), 381-94.

Li F., Zhou X. and Wong S.T.C. (2007) An automated feedback system with the hybrid model of scoring and classification for solving over-segmentation problems in RNAi high content screening. J. Microsc. *Accepted for publication*.

Meijering E., Jacob M., Sarria J.-C.F., Steiner P., Hirling H. and Unser M. (2004) Design and validation of a tool for neurite tracing and analysis in fluorescence microscopy images. Cytometry Part A. 58A(2), 167-176.

Otsu N. (1979) A threshold selection method from gray-level histograms. IEEE Trans. Sys., Man., Cyber. 9, 62–66.

Streekstra, G.J. and J. van Pelt. (2002) Analysis of tubular structures in three-dimensional confocal images. Network: Comput. Neural Syst. 13: 381-395.

Tschirren J., Hoffman E.A., McLennan G. and Sonka M. (2005) Intrathoracic airway trees: segmentation and airway morphology analysis from low-dose CT scans. IEEE Trans. Med.

Imaging. 24(12), 1529- 1539.

Wang M., Zhou X., King R.W. and Wong S.T.C. (2007) Context based mixture model for cell phase identification in automated fluorescence microscopy. BMC Bioinformatics. 30, 8-32.

Xiong G., Zhou X. and Ji L. (2006) Automated segmentation of drosophila RNAi fluorescence cellular images using deformable models. IEEE Transactions on Circuits and Systems I. 53(11), 2415-2424.

Zhang Y., Zhou X., Degterev A., Lipinski M., Adjeroh D., Yuan J. and Wong S.T. (2006) A novel tracing algorithm for high throughput imaging screening of neuron-based assays. J. Neurosci. Methods. *Article in press.*