

# Rapid Reconstruction of 3D Neuronal Morphology from Light Microscopy Images with Augmented Rayburst Sampling

Xing Ming<sup>1,2</sup>, Anan Li<sup>1,2</sup>, Jingpeng Wu<sup>1,2</sup>, Cheng Yan<sup>1,2</sup>, Wenxiang Ding<sup>1,2</sup>, Hui Gong<sup>1,2</sup>, Shaoqun Zeng<sup>1,2</sup>, Qian Liu<sup>1,2\*</sup>

**1** Britton Chance Center for Biomedical Photonics, Huazhong University of Science and Technology - Wuhan National Laboratory for Optoelectronics, Wuhan, China,

**2** MoE Key Laboratory of Biomedical Photonics, Department of Biomedical Engineering, Huazhong University of Science and Technology, Wuhan, China

## Abstract

Digital reconstruction of three-dimensional (3D) neuronal morphology from light microscopy images provides a powerful technique for analysis of neural circuits. It is time-consuming to manually perform this process. Thus, efficient computer-assisted approaches are preferable. In this paper, we present an innovative method for the tracing and reconstruction of 3D neuronal morphology from light microscopy images. The method uses a prediction and refinement strategy that is based on exploration of local neuron structural features. We extended the rayburst sampling algorithm to a marching fashion, which starts from a single or a few seed points and marches recursively forward along neurite branches to trace and reconstruct the whole tree-like structure. A local radius-related but size-independent hemispherical sampling was used to predict the neurite centerline and detect branches. Iterative rayburst sampling was performed in the orthogonal plane, to refine the centerline location and to estimate the local radius. We implemented the method in a cooperative 3D interactive visualization-assisted system named fNeuronTool. The source code in C++ and the binaries are freely available at <http://sourceforge.net/projects/flneurontool/>. We validated and evaluated the proposed method using synthetic data and real datasets from the Digital Reconstruction of Axonal and Dendritic Morphology (DIADeM) challenge. Then, fNeuronTool was applied to mouse brain images acquired with the Micro-Optical Sectioning Tomography (MOST) system, to reconstruct single neurons and local neural circuits. The results showed that the system achieves a reasonable balance between fast speed and acceptable accuracy, which is promising for interactive applications in neuronal image analysis.

**Citation:** Ming X, Li A, Wu J, Yan C, Ding W, et al. (2013) Rapid Reconstruction of 3D Neuronal Morphology from Light Microscopy Images with Augmented Rayburst Sampling. PLoS ONE 8(12): e84557. doi:10.1371/journal.pone.0084557

**Editor:** William W. Lytton, SUNY Downstate MC, United States of America

**Received:** July 22, 2013; **Accepted:** November 16, 2013; **Published:** December 31, 2013

**Copyright:** © 2013 Ming et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Funding:** This work is supported by National Natural Science Foundation of China (Grant Nos. 81127002, 81171067, 61102122); Science Fund for Creative Research Group of China (Grant No. 61121004); and Program for New Century Excellent Talents in University of China (Grant No. NCET-10-0386). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

**Competing Interests:** The authors have declared that no competing interests exist.

\* E-mail: [qianliu@mail.hust.edu.cn](mailto:qianliu@mail.hust.edu.cn)

## Introduction

Digital reconstruction of neuronal morphology from light microscopy images provides a powerful technique for the analysis of neural circuits forms and for the investigation of their underlying function [1]. Since the manual reconstruction is very time-consuming, especially for large-scale neuronal analysis, a number of studies have been conducted to develop more efficient computer-assisted approaches to support neurite tracing and neuronal morphology reconstruction [2–4].

Methods of neurite tracing roughly fall in three categories. The first category relies on the sequential presentation of serial 2D images. These methods extract the profiles of neural structures in each 2D cross-sectional plane and then connect the results in the third dimension [5–7]. However, this may be in trouble when local neurite segments lie parallel to the 2D plane. Resampling the original images along different directions might work but is computationally expensive and difficult to cover all neurites.

The second category is based on global image segmentation in 3D. These methods first turn the image into a binary form with certain segmentation algorithms and then extract centerlines of

foreground areas with a skeletonization algorithm [8,9]. Here, efficient segmentation algorithms, such as thresholding, prove successful for uniformly high quality images, although it is not always the case. Various sophisticated filters have been proposed to enhance feature structures and to improve segmentation and skeletonization. Typically, such filters are based on the analysis of eigenvalues of the Hessian [10], Jacobian matrix [11], or steerable filters [12]. However, these filters require multiple scaling or orientations and are computationally expensive because they operate on the entire image. A few other algorithms, such as voxel coding [13,14] and voxel scooping [15], can trace multiple neurite branches directly from the grayscale images but they operate on a voxel-by-voxel basis and are more dependent on the high quality of pre-processed images.

The third category explores images only in local regions around the structures of interest. These methods usually start from a seed, which can be located automatically or manually, and trace recursively where neurite branches go in the 3D image. The direction of tracing is determined according to the distribution of local signal and background. A representative

algorithm is to use a template that comprises four predefined parallel edge detectors, to determine the direction and boundary of a single branch [16–18]. More robust algorithms often rely on certain mathematical or graph models. They usually define a cost or energy measure that is based on local image features and curve regularity; then convert the problem into searching for an optimal path or minimum spanning tree between given crucial points [19–25]. If initial structures or prior knowledge can be obtained, it then becomes optional to fit a snake-based or active contour model to the image data [26–28]. A potential problem is that these local tracing algorithms may fail at branch points or crossover regions because of a lack of global information. Thus, a separate branch merger or segment join step is needed to trace whole tree-like structure [29–31].

To reconstruct 3D neuronal morphology, the neurite tracing can be coupled with radius estimation at each node. The rayburst sampling proposed by Wearne and Rodriguez can be used for this [32,33]. It samples the image data in multiple directions using a pre-computed array of unit vectors from a given node. The process of sampling is defined as the simultaneous casting of rays in multiple directions. Each ray grows from a given node until a specified exit criterion is met, which returns a length representing the forward sampling distance. In the original algorithm, a 2D rayburst is run in the cross-sectional plane for each node, and the diameter is computed from the ray lengths. It is worth noticing that to reconstruct the whole dendritic tree, it is necessary to extract the centerline of branches in advance.

The methods mentioned above are all helpful for the reconstruction of neuronal morphology from light microscopy images. However, these automated reconstructions often contain a substantial number of false segments or short branches, requiring post-correction with significant human effort [34,35]. Thus, to achieve a reasonable balance between fast speed and high accuracy, a rapid reconstruction system incorporating automatic tracing and manual editing is preferable.

In this paper, we present a practical method for the tracing and reconstruction of 3D neuronal morphology from light microscopy images and implement it in an interactive visualization-assisted freeware system named flNeuronTool. We extend the original rayburst sampling algorithm to a marching fashion, which starts from a single or a few initial seed points and marches recursively forward along neurite branches to trace and perform reconstruction on the whole tree-like structure. We validate and evaluate the method using synthetic data and real datasets from the Digital Reconstruction of Axonal and Dendritic Morphology (DIADEM) challenge [36,37]. Then, the system is applied to mouse brain images acquired with the Micro-Optical Sectioning Tomography (MOST) system [38–41], to reconstruct single neurons and local neural circuits.

## Methods

### Neurite Tracing Strategy

At the typical resolution of light microscope images, each segment of a neurite has a smooth tubular shape, which can be approximated as a generic cylinder with a certain radius [42]. Moreover, the diameter and direction of a neurite in a local area never changes abruptly, which means that two adjacent segments have a similar radius and their centerlines often form a modest angle. Thus, it is possible to predict the radius and direction of the next segment according to the current presentation [43].

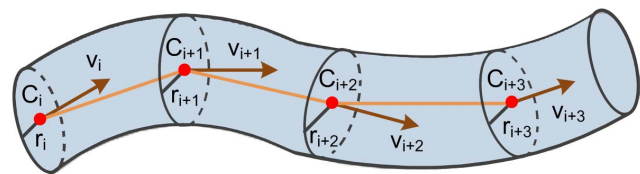
We trace individual neurite using a prediction and refinement strategy. As shown in **Figure 1**, we recursively move a node along the centerline of a neurite. For any given position, the node has

three properties: the location  $C$  is a point on the centerline; the direction  $v$  is a unit vector that represents the tangent of the centerline; and the radius  $r$  is the minimum distance from point  $C$  to the structure boundary, which is estimated at the orthogonal plane of the direction  $v$ . Two consecutive nodes are defined as the parent and child nodes, respectively. If appropriate properties of the parent node have been obtained, we can predict the child node position along direction  $v$ . Then, using local images, the properties of the child node can be refined. Recursively, we can trace the whole tree starting from one single or a few of seed points.

### Centerline Extraction and Branch Detection

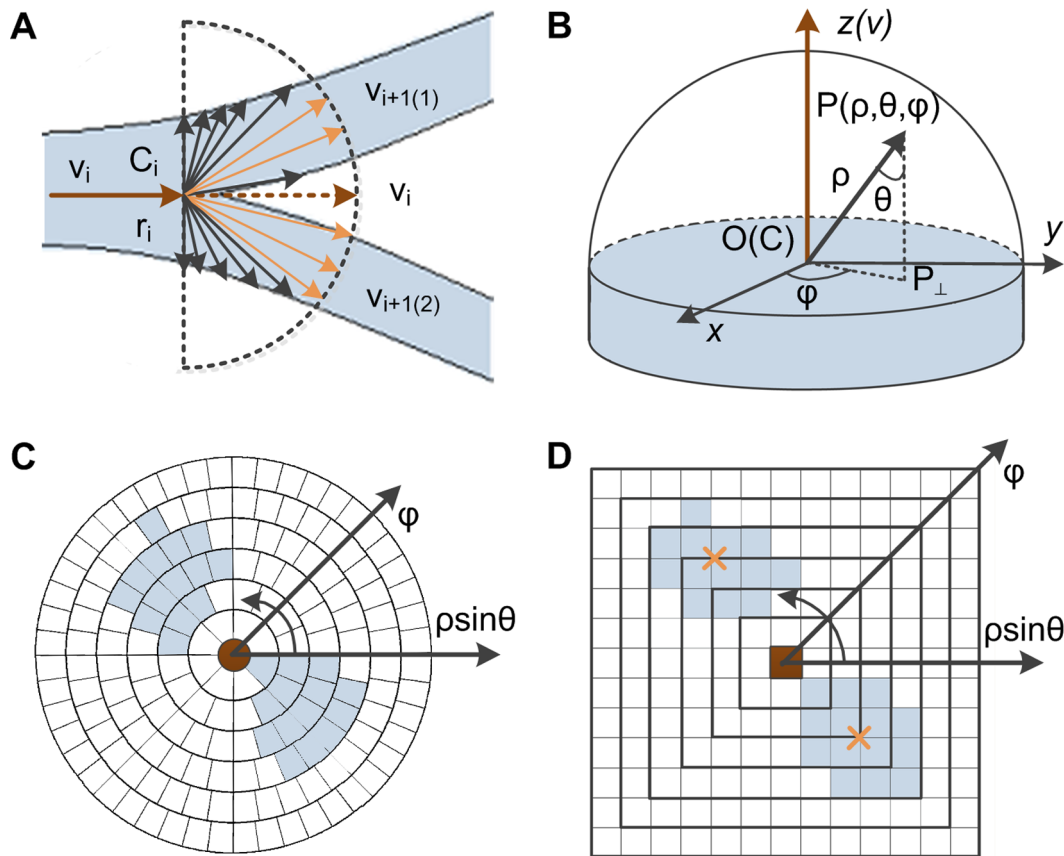
We use augmented rayburst sampling to detect centerlines and branches in the prediction step. Inspecting 2D rayburst sampling at a given node, we find that the longest rays are always close to the direction of the axis that represents the potential centerline of the local neurite segment, while the shortest rays in the orthogonal direction represent the local radius. Consequently, when extending all of the rays from the position of the current node until their length goes over a specified threshold (approximately 2.0–4.0 times the local radius), those rays that are far away from the axis will reach or exceed the boundary of the neurite, and the corresponding samplings will be terminated. The remaining rays that are still inside the structure will be separated into a few clusters according to their directions, which represent potential branches. Specifically, they are together as a single cluster when the branches are absent. The length threshold of each ray is defined as the sampling distance, which is only related to the local radius but independent of the absolute size of the structural structure. Moreover, to avoid backward tracing, we perform sampling only in the forward direction. As shown in **Figure 2A**, actual sampling could be restricted to a half circle, whose origin locates at the current node position  $C$ , and the half angle ray represents the node direction  $v$ .

The improved rayburst sampling is introduced for 3D. Here, to detect branches, sampling is restricted to a hemisphere; thus, it is crucial to generate an approximately uniformly distributed sampling core on the hemisphere surface. The random generation or polygon subdivision used in the original algorithm cannot achieve this goal. As shown in **Figure 2B**, we solve this problem in a spherical coordinate system for which the origin  $O$  is located at the current node position  $C$ , the  $z$  axis coincides with the current node direction  $v$  and the orthogonal plane parallels the cross-section that covers the local radius  $r$ . For any vector  $OP$  with length  $\rho$ , its endpoint coordinate is  $P(\rho, \theta, \varphi)$ , in which  $\theta$  and  $\varphi$  represent the inclination and azimuthal angles, respectively. We divide the whole hemisphere into  $M$  ( $M = 7$ ) stacks according to the inclination angle  $\theta$ , with equal intervals. For each given  $\theta$ , we divide corresponding small circles into  $N$  slices according to the azimuthal angle  $\varphi$ , with equal intervals. The angular distribution is expressed as



**Figure 1. Neurite tracing strategy.** A node is recursively moved along the centerline of a neurite, to reconstruct neurite segments as a sequence of cylinders. At any given position, the node has a location  $C$ , a direction  $v$  and a radius  $r$ .

doi:10.1371/journal.pone.0084557.g001



**Figure 2. Centerline extraction and branch detection using hemisphere sampling.** (A) 2D illustration of the centerline and branch prediction with improved rayburst sampling. (B) 3D illustration of a vector of the hemisphere sampling core in a spherical coordinate system. (C) The sampling state on the hemispherical surface is projected to the orthogonal plane, in which the sampling rays that have the same inclination angle  $\theta$  form a circle, and nonzero value samplings (bright blue) suggest possible branches. (D) In the plane, each circle can be mapped into a circumscribed square, and each local center detected (orange) represents a neurite branch.  
doi:10.1371/journal.pone.0084557.g002

$$\theta(i) = i \frac{\pi}{2M}, \quad 0 \leq i \leq M$$

$$z = \varepsilon \rho \cos \theta$$

$$\varphi(j) = j \frac{2\pi}{N}, \quad N = \max(1, 8i), \quad 0 \leq j \leq N$$

where the function  $\max$  selects the maximum value from two given numbers, and the reason for setting the variable  $N$  values will be explained below. To sample the original data, vector  $OP$  should be converted into the Cartesian form, in which the coordinate of the endpoint  $P(x, y, z)$  is expressed as

$$x = \rho \sin \theta \cos \varphi, \quad y = \rho \sin \theta \sin \varphi, \quad z = \rho \cos \theta$$

It is convenient to set  $\rho$  to one for automated normalization. Nevertheless, in some case, where neurite segments are really thin and comparatively straight, one can speed up the sampling in the  $z$  direction. Thus, an anisotropic sampling core is optional and has no influence on the sampling form, while the conversion to the  $z$  axis is expressed as

where  $\varepsilon$  is defined as the anisotropy coefficient (approximately 1.0~1.1). Then, these vectors could be normalized and used as a pre-computed sampling core.

To perform 3D hemispherical sampling at a given node, we rotate the sampling core to correct the orientation using a transformation matrix, by which we match the current node direction  $v$  with the  $z$  axis. Then, each vector in the sampling core is iteratively extended with a pre-specified step-length from the current position  $C$  until the length goes over the pre-specified sampling distance. Those rays who reach the boundary of the local structure should be terminated early. One could use an adaptive hysteresis threshold to detect the boundary along a sampling ray (supporting information, **Text S1** in **File S1**). In the end of the sampling setup, the sampling state whose ray endpoint is still in the inside of the structure is recorded as a value of one, and the remaining is recorded as a value of zero. A possible set of sampling state on the hemispherical surface is projected to the orthogonal plane as shown in **Figure 2C**, in which the sampling rays that have the same inclination angle  $\theta$  form a circle, and nonzero value samplings suggest possible branches.

In the plane, each circle can be mapped into a circumscribed square with slight deformation, as shown in **Figure 2D**.

Consequently, these sampling squares with different inclination angles  $\theta$  and azimuthal angles  $\varphi$  form a uniform grid. Besides, the number of samplings in the innermost square is only one, and the difference between the adjacent squares is 8, which is the reason for setting the variable  $N$  values for the sampling core generation, as described above. In this uniform grid, nonzero value samplings are separated into a few areas for which the local centers could be detected by multiform ways, such as a simple and effective distance transform [44]. In general, each local center represents a potential neurite branch that can be mapped back to the hemisphere surface and is expressed in both spherical coordinates and the Cartesian form. Then, a new node can be located in the branch as the child whose parent is the current node. The distance between it and the parent node is defined as the location distance, which is approximately equal to a diameter of the parent node.

### Centerline Refinement and Radius Estimation

The location of a node that is predicted by hemispherical sampling may be a bit off the real centerline of local neurite because of holes or other artifacts in light microscopy images, and the radius of predicted child node equals the parent node radius, which should be refined. We run an improved 2D rayburst sampling to perform this step.

As shown in **Figure 3A**, a set of unit vectors that are parallel to the sampling plane and uniformly distributed in a circle are used as a sampling core. The sampling starts at the predicted node  $C$ . Each ray iteratively extends with a pre-specified step-length from the origin point until the structure boundary is reached. We record the intersection of a ray and the boundary as  $P$ . Ideally, the cross-section is a generic ellipse, of which a centroid is simply the average location of all intersections. For real images, to gradually approach to the real position, we iteratively perform multiple sampling in which each calculated centroid is regarded as the origin for the next sampling. This process is terminated when the Euclidean distance  $\delta$  between two calculated positions from successive iterations is less than a pre-specified threshold, which is a radius-related value (the coefficient  $\varepsilon = 0.05$ ). For the last iteration, the lengths of all of the rays are also calculated, of which the mean is regarded to be the estimated radius  $r$  of the current node. This process is expressed as

$$C_{i+1} = \frac{1}{N} \sum_{j=0}^{N-1} P_{i,j}, \quad \text{dst}(C_{i+1}, C_i) \geq \varepsilon r_i$$

$$r_i = \frac{1}{N} \sum_{j=0}^{N-1} \text{dst}(P_{i,j}, C_i)$$

where  $N$  ( $N = 32$ ) is the number of the 2D sampling rays,  $P_{i,j}$  represents the intersection position of the ray numbered  $j$  and the boundary in the iteration numbered  $i$ , and the function  $\text{dst}$  calculates the Euclidean distance between two given points.

The segment from the parent node to the current node represents the local centerline. However, a potential problem is that it could no longer be orthogonal to the reconstructed circle because the sampling origin used to refine the centerline and estimate the radius has shifted during the iterations. There is a space angle  $\delta$  between the initial direction that was predicted by hemispherical sampling and the refined direction, which suggests a calculation error of the centerline, as shown in **Figure 3B**. To improve the precision of the centerline and radius refinements, we iterate the step described above in a new plane that is orthogonal to the current direction until the angle between the last direction and the prior direction falls under a pre-specified threshold ( $\delta = \pi/$

36), which result in an appropriate precision while preventing unnecessary iteration. Then, the cross-section of neurite is reconstructed as a circle with a radius  $r$  that locates at the position  $C$  and is approximately orthogonal to the refined centerline.

### Reconstruction of Neuronal Morphology

The algorithm described above is implemented in a freeware system named `fnNeuronTool`, which is programmed in C++ and potentially supports multiple platforms. This system allows users to reconstruct and proofread neuronal morphologies in a cooperative 3D interactive visualization-assisted environment (supporting information, **Text S2** in **File S1**). The source code and binaries are freely available at <http://sourceforge.net/projects/fneurontool/>.

To run the tracing algorithm, an initial node has to be created as the seed point. We select one single or a few points that are inside of a neuron and are close to the root of the tree-like structure, and pushed them into a queue as seed points. This can be automatically done, but we do prefer a manual way in a 3D interactive environment, which can significantly reduce false positive errors. At each selected location, two hemisphere samplings that have opposite directions are carried out to detection branches. The rest of tracing process is recursive prediction and refinement as described above. In every recursion, generated nodes are all pushed in the queue, which will be orderly popped as new seed points. When all seed points are run out and the sampling end, one could select a new seed point to reconstruct remaining branches of the tree-like structure. This process iterates until the expected reconstruction is achieved.

Due to imperfect images or excessive seed points, the reconstruction will have a number of redundancy paths and short branches. Here, redundancy paths cover the same neurite, in which each pair corresponding nodes have a similar direction and radius, and cover a more or less overlap area. One could find and merge all such node pairs from the root to endpoints based on whether the Cartesian distance between them is less than any one of them radius [23]. The centroid of each pair nodes is calculated to produce a final node location, while radius is averaged. As additional gains, this method uses redundancy paths to generate a more complete and better reconstruction. The length of a branch is defined as the path length that from the last bifurcation to its endpoint. One could remove short branches based on their absolute length and their lengths relative to the local radius [15].

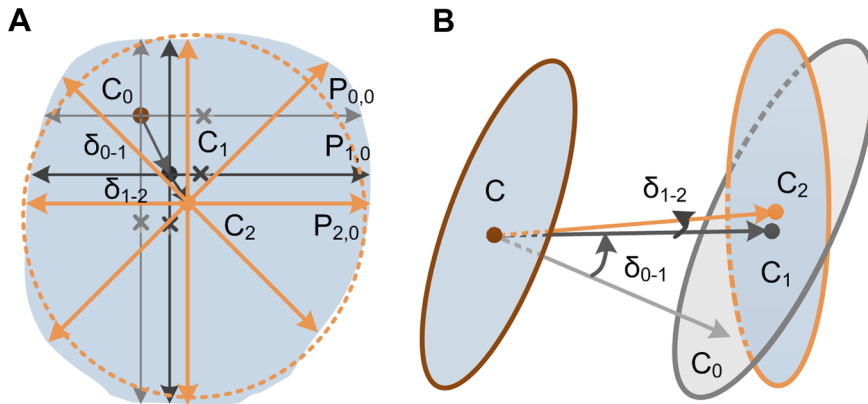
## Experiments and Results

### Parameters Selection

To validate and evaluate the proposed method, we tested it on synthetic data and real datasets from the DIADEM challenge. Then, we applied the system to mouse brain images acquired with the MOST system to reconstruct single neurons and local neural circuits. All of the experiments were performed on an ordinary computer (Intel Core Duo 2.6 GHz CPU, NVIDIA GeForce 9800 GPU, 4GB RAM, Windows 7). **Table 1** summarizes the sampling parameters for these experiments, in which most parameters remain constant across all experiments, while other parameters need to be adapted for each dataset to produce optimal results.

### Validation on Synthetic Data

First, we inspected the performance of the branch detection using synthetic tree-like structure data. The original image and the corresponding truth structure were created by VasuSynth [45] and were downloaded from website (<http://vasusynth.cs.sfu.ca/>), as shown in **Figure 4A**. To find the optimal value of the 3D



**Figure 3. Centerline refinement and local radius estimation using iterative rayburst sampling.** (A) Centroid refinement and local radius estimation using iterative 2D rayburst samplings in which each calculated centroid is regarded as the origin for the next sampling. (B) 3D illustration of centerline refinement. The sampling described above is iterated in a plane that is orthogonal to the current centerline direction.  
doi:10.1371/journal.pone.0084557.g003

sampling distance for the branch detection, we traced the tree in the image with different sampling distance, which ranges from 1.5 to 4.5 with an increment of 0.5 times the local radius (supporting information, **Text S3** in **File S1**). For each tracing, we measured the total length of correct branches and the number of correct bifurcations, and reported the proportions of them to true values. Because the test was run directly on the image without any additional pre-processing or manual editing and used only one seed point that was located at the root of the tree by one mouse click, we could report the computation time for each tracing, as shown in **Figure 4B**. From the trend in these measurements, it is found that a sampling distance too big or too small would lead to the branch detection lost or redundancy, and the optimal value is about 3.0. Considering the difference of images, one may set this value between 2.0 and 4.0. In general, the value of length is always more than the value of the bifurcations and has smaller change, for which lost branches are relatively short.

Then, we validated the robustness of the branch detection. We used ImageJ (<http://rsb.info.nih.gov/ij/>) to generate a set of images based on the original tree-like structure data with additive Gaussian noise, whose standard deviation (SD) ranges from 5 to 30

with an increment of 5. **Figure 4C** shows the highest noised image and the corresponding automated reconstruction. For each image, the value of the sampling distance was set to 3.0. Once again, we reported the length of branches, the number of bifurcations and the computation time, as shown in **Figure 4D**. It is obvious that the tree-like structure becomes more ambiguous when the noise is higher, making the tracing more difficult. However, considering that there is only one seed point for each tracing, the result is acceptable. Overall, low-level noise could lead to more short branches, which must be removed expending additional time. High-level noise tends to cause the early termination of tracing, which is helpful for reducing false positive errors.

#### Evaluation with DIADEM Dataset

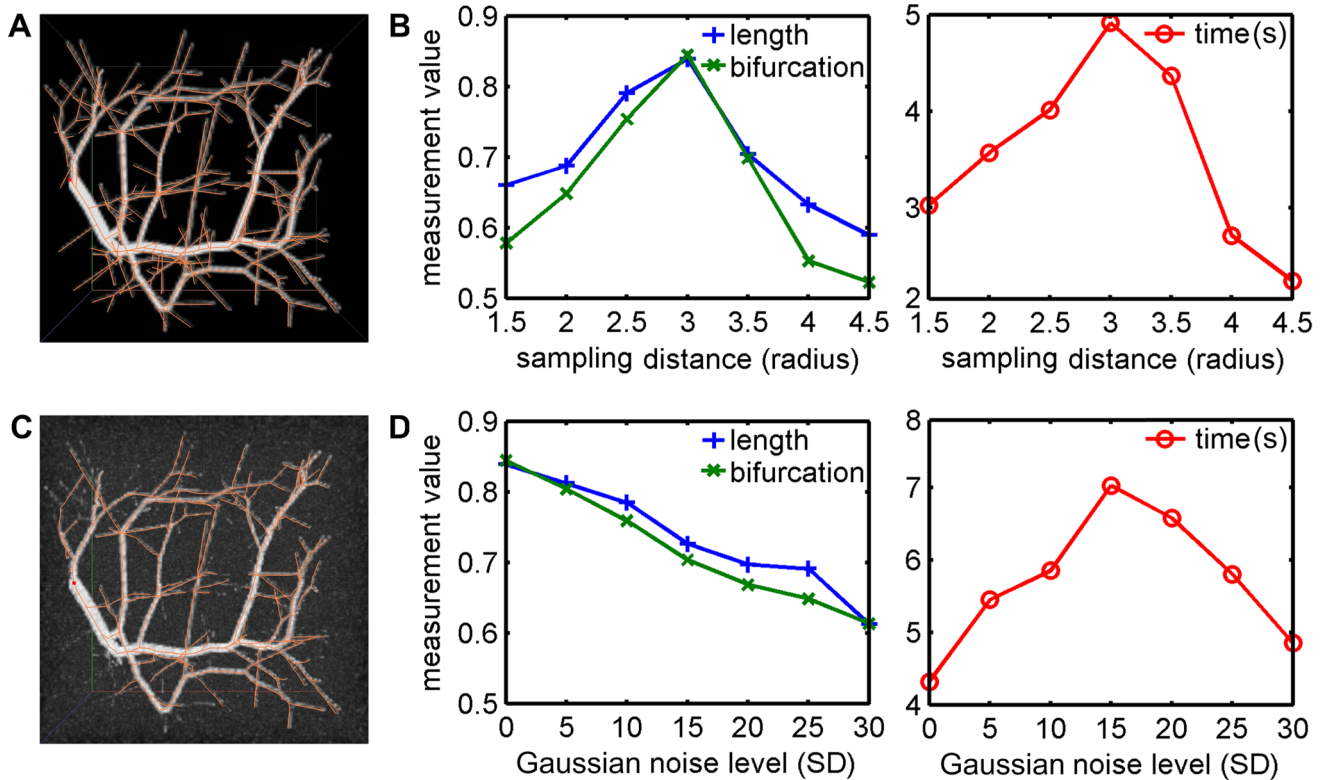
We evaluated the proposed method with two real datasets from the DIADEM challenge. Specifically, the two datasets were drosophila olfactory projection (OP) [46] and mouse neuromuscular projection fibers (NM) [47]. The image stacks and their gold standard reconstructions were downloaded from website (<http://www.diademchallenge.org/>).

**Table 1. Sampling parameters selection.**

Parameters	Value	Notes
3D sampling distance	2.0~4.0 (radius)	Each ray is terminated when its length is over the threshold. A value too big or too small will lead to the branch detection lost or redundancy. Default value is 3.0.
Anisotropy coefficient	1.0~1.1	An anisotropic sampling core can speed up the sampling in the axial direction. Default value is 1.0.
Inclination division number	7	This value determines the number of 3D sampling rays. The larger value will create a more delicate sampling core, but will require more calculated amount.
Sampling step-length	1.0 (voxel)	Each ray is iteratively extending with the step-length from the origin until the exit criterion is satisfied. For anisotropic images, a super sampling should be applied in the low resolution direction.
Location distance	1.5~2.5 (radius)	The distance between the child and the parent node affects the density of nodes in the reconstruction. Default value is 2.0 but no more than the 3D sampling distance.
2D sampling ray number	32	These rays are used to refine centerline and radius.
Iteration exit distance	0.05 (radius)	Centroid refinement is terminated when the distance between two successive locations is less than the threshold.
Iteration exit angle	$\pi/36$	Centerline refinement is terminated when the angle between two successive directions is less than the threshold.

doi:10.1371/journal.pone.0084557.t001





**Figure 4. Validation of the branch detection on tree-like structure data.** (A) The original image (voxel size  $101 \times 101 \times 101$ ) and truth structure (branch length 3325 voxels, bifurcation number 199). (B) The performance analysis for the tracing of the tree structure in the original image with different sampling distance. For each tracing, only one seed point is used. The total length of correct branches and the number of correct bifurcations are measured, and the proportions of them to truth values are reported, as well as the computation time in seconds. (C) The highest Gaussian noised image (SD 30) and the corresponding automated reconstruction (branch length 2036 voxels, bifurcation number 122). (D) The robustness analysis for the tracing of the tree structure in different level Gauss noised images with the same sampling distance (3.0 times local radius).

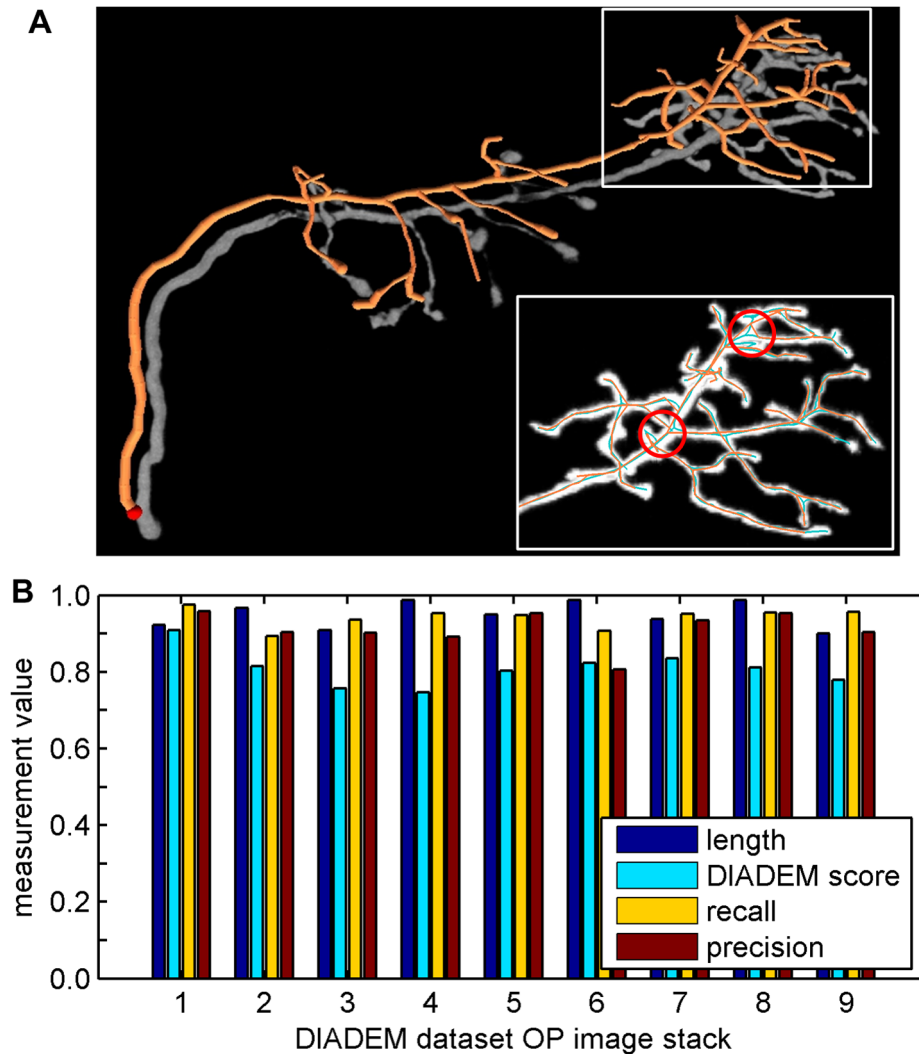
doi:10.1371/journal.pone.0084557.g004

The first test was run on 9 image stacks of dataset OP. For each image stack, we traced three times with different sampling distances that range from 2.5 to 3.5, from which the one that had longest path was selected as final reconstruction. The final reconstruction of the first stack is shown in **Figure 5A**, in which the reconstruction is shifted slightly relative to the voxel data to improve the visibility. Local detail of the reconstruction and the corresponding gold standard are displayed in different colors in the inset, which shows a very small difference between them, especially at several potential topological errors that are indicated with circles. To evaluate the completeness of the reconstructions, we measured their path length, which was reported as the proportion of the length to the gold standard. To quantify the accuracy of automated reconstructions, we compared them to the gold standards by using the DIADEM metric [48]. Let  $G$  be the number of nodes in the gold standard,  $T$  be the number of nodes in our reconstruction,  $M$  be the missing nodes (false negatives) reported by the DIADEM metric, and  $E$  be the extra nodes (false positive) in test reconstruction, we define the recall and precision as

$$recall = 1.0 - \frac{M}{G}, \quad precision = 1.0 - \frac{E}{T}$$

The measurements are shown in **Figure 5B**. Specifically, the path length and DIADEM score are approximately  $0.95 \pm 0.03$  (mean  $\pm$  SD) and  $0.81 \pm 0.05$ , respectively, which indicate that the quality of the reconstructions is generally high. In these automated reconstructions, typical errors such as branch breakings and topological connection mistakes could be easily proofread and corrected in the editing mode of flNeuronTool, which requires approximately 5~10 minutes of user intervention per stack.

The second test was run on dataset NM subset 2. As an example, we reconstructed a piece of volumetric data of 4 image stacks that have been integrated, in which 15 axon fibers start from one of the stacks and continue through the remaining stacks, as shown in **Figure 6A**. It should be noted that we could not reconstruct the entirety of the axon fibers at a high confidence level because of the absence of appropriate pre-processing. Focusing on the local detail, potential errors in the reconstruction include branch breaking from the high unevenness of the voxel intensity, one of which is indicated with a circle. Because of the differences in data dimensionality and tracing root, the DIADEM metric could not be used as expected to evaluate the quality of the reconstruction. Instead, we reconstructed a single image stack, and compared the result with the reconstruction from NeuronStudio. As shown in **Figure 6B**, flNeuronTool can create an acceptable automated reconstruction without additional manual editing. In contrast, as shown in **Figure 6C**, in the reconstruction from NeuronStudio, there are a few errors such as branch breaking or



**Figure 5. Evaluation with the DIADEM dataset OP.** (A) The original data and the automated reconstruction of the first image stack (voxel size  $512 \times 512 \times 60$ ), which is shifted slightly relative to the voxel data to improve the visibility. Local detail of the reconstruction (orange) and gold standard (cyan) are displayed in the inset, in which potential errors are circled (red). (B) The quantitative measurements of nine image stacks. The DIADEM score, recall and precision are report based on DIADEM metric measurements compare with the gold standards. Specifically, the path length and DIADEM score are approximately  $0.95 \pm 0.03$  (mean  $\pm$  SD) and  $0.81 \pm 0.05$ . doi:10.1371/journal.pone.0084557.g005

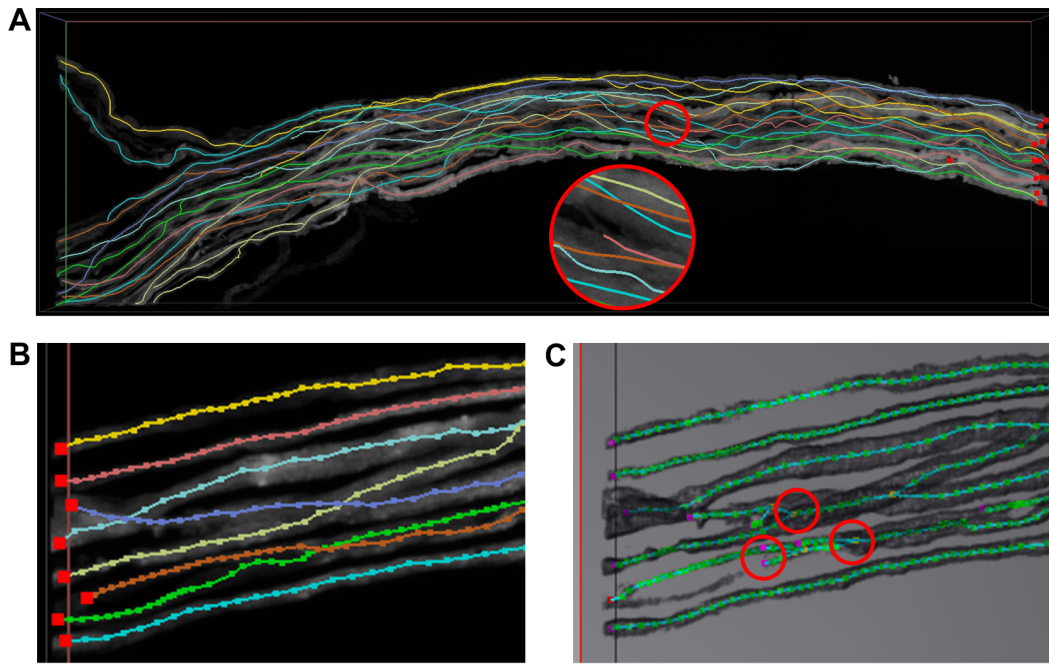
steal that are marked with circles. One possible reason of these differences is that the hemisphere sampling is more or less equivalent to smooth the original image, which could suppress the influence of individual voxels. Indeed, NeuronStudio could create a high-quality reconstruction with appropriate parameters or a small amount of manual editing. However, in our experience, the proofread in 2D projection planes that is adopted by NeuronStudio is not trivial, while the 3D interactive editing adopted by flNeuronTool is more convenient.

#### Application to MOST Image

We applied the proposed system to mouse brain images acquired with the MOST system to reconstruct single neurons and local neural circuits. We used two datasets, which are from a green fluorescence labeled [40,41] and a modified Golgi-Cox stained [38,39] whole mouse brain and provide micron-resolution tomography images at the neurite level. For these original images, pre-processing is necessary, which strongly depends on the details

of the neuron labeling. We applied flNeuronTool to pre-processed datasets to reconstruct a few single neurons and local neural circuits.

The first subset of images was labeled using green fluorescence, which has a 3D resolution of  $0.5 \mu\text{m} \times 0.5 \mu\text{m} \times 2.0 \mu\text{m}$ . An example for the reconstruction of a pyramidal neuron is shown in **Figure 7A**, in which the reconstruction was slightly shifted to improve the visibility. To evaluate the accuracy of the reconstruction, we compared it with the reference reconstruction that was created using commercial software Amira (<http://www.amira.com/>). A part of the details of them is displayed in different colors in **Figure 7B**, and a few of the differences between them are indicated with circles. In general, the result is acceptable. Specifically, potential errors in the reconstruction such as branch contraction could be due to uneven labeling. The proportion of total length and number of bifurcation of the reconstruction to the reference are 0.96 and 0.94, respectively. The time to achieve the final reconstruction using flNeuronTool is less than 10% of the



**Figure 6. Evaluation with the DIADEM dataset NM subset 2.** (A) An example of 4 image stacks (voxel size  $1851 \times 632 \times 125$ ) and the reconstruction, in which an error is circled. (B) An image stack (voxel size  $512 \times 512 \times 57$ ) and the reconstruction from flNeuronTool, which is an acceptable result without additional manual editing. (C) The automated reconstruction of the same stack from NeuronStudio, in which potential errors are circled.

doi:10.1371/journal.pone.0084557.g006

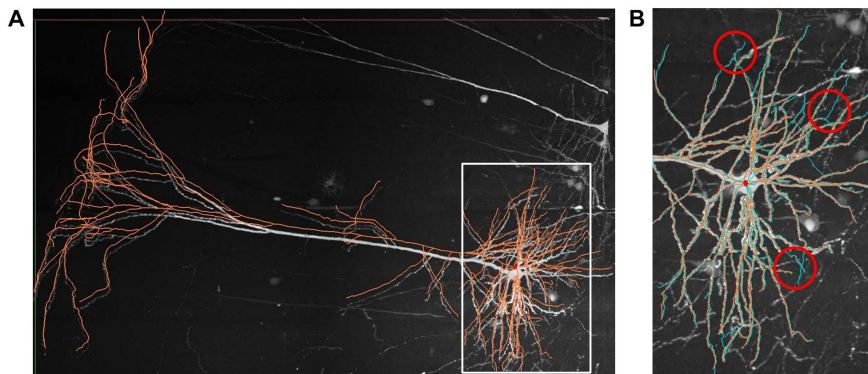
manual method, which is very attractive for large-scale neuronal image analysis.

The second subset of images was stained with the modified Golgi-Cox method, which has a 3D resolution of  $0.3 \mu\text{m} \times 0.3 \mu\text{m} \times 1.0 \mu\text{m}$ . We expected to reconstruct local neural circuits from these images, because the density of neurons in this dataset is suitable. However, with our experience, it is very difficult to achieve this goal even by hand because of the complexity of the neuronal morphologies and local connections and the uneven staining. These images exhibit variable contrast, and there is frequent loss in the continuity of neuron. Hence, the best strategy is to reconstruct neuron only when the confidence level is relatively high. An example of the pre-processed images is shown in

**Figure 8A.** The reconstruction from flNeuronTool is shown in **Figure 8B**, which contains 10 neurons, starting from the somas in the same volumetric subset, as well as 73 separate neurite fragments that have a length of more than 50 voxels. To illustrate the completeness of the neuronal morphologies, we used a sphere to represent a soma location. It is reasonable that the reconstruction contains a few errors, although we have spent approximately an hour on proofreading and correcting them.

## Discussion

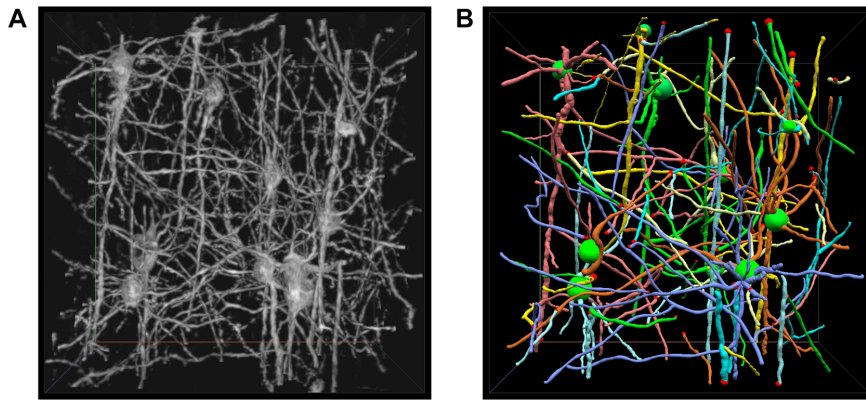
We presented an innovative method for tracing and reconstruction of 3D neuronal morphology from light microscopy images.



**Figure 7. Application the proposed system to MOST fluorescence images.** (A) An example of the original images (voxel size  $1728 \times 1088 \times 176$ ) and the reconstruction of a pyramidal neuron. (B) Detailed comparison of the reconstruction (orange) with the reference reconstruction from Amira (cyan), in which potential errors are circled (red). The proportion of total length and number of bifurcation of the reconstruction to the reference are 0.96 and 0.94, respectively.

doi:10.1371/journal.pone.0084557.g007





**Figure 8. Application the proposed system to MOST Golgi images.** (A) An example of the pre-processed images (voxel size  $450 \times 450 \times 345$ ). (B) The reconstruction that containing 10 neurons with soma (sphere) and a few separate neurite fragments.  
doi:10.1371/journal.pone.0084557.g008

We validated and evaluated the proposed method using synthetic data and the DIADEM datasets, and applied the system to MOST images. The main contribution of this paper is the augmented rayburst sampling algorithm. The original rayburst sampling has been extended to a marching fashion, which is capable of using only local information to perform neurite centerline extraction, branch detection, centroid refinement and radius estimation. The advantages of the method are summarized, as follows. First, all the tracing and reconstruction steps are achieved in a single step, in which the centerline extraction or the extra radius estimation is unnecessary. Second, this method only needs to sample a part of points rather than processing all neuronal voxels one by one. Consequently, the improvement of the reconstruction speed becomes possible, while the influence of random noises is reasonably suppressed so that appropriate computational accuracy could be guaranteed. Third, the sampling is only related to the local radius but not depend on the absolute size of the structural structure, thus, it has the potential to deal with images of different spatial resolutions.

It is worth mentioning that we implemented the method in the freeware system flNeuronTool. The system incorporates automatic tracing and manual editing of neuron reconstruction into a cooperative 3D interactive visualization-assisted environment, which is a powerful tool for analysis of complex neuronal images. Experimental results showed that it provides an effective means of minimizing the human effort that is required to improve the quality of the final reconstruction. In general, the system can be faster than manual reconstruction and be more accurate than fully automatic tracing.

One limitation of the proposed method is its poor performance on beaded neurites. For these images, we think that those graph model-based algorithms may be a better choice [21]. We also found that it would be unreliable or unnecessary to reconstruct the exact morphology for some very fine dendrites which is about one voxel wide in light microscopy images. In this case, the actual application may only need to trace the centerline of neurites, and those snake-based algorithms may be more appropriate [28].

## References

1. Roysam B, Shain W, Ascoli GA (2009) The central role of neuroinformatics in the national academy of engineering's grandest challenge: reverse engineer the brain. *Neuroinformatics* 7: 1–5.
2. Meijering E (2010) Neuron tracing in perspective. *Cytometry Part A*, 77: 693–704.
3. Donohue DE, Ascoli GA (2011) Automated reconstruction of neuronal morphology: an overview. *Brain Research Reviews* 67: 94–102.
4. Halavi M, Hamilton KA, Parekh R, Ascoli GA (2012) Digital reconstructions of neuronal morphology: three decades of research trends. *Frontiers in Neuroscience* 6: 49.
5. Fiala JC (2005) Reconstruct: a free editor for serial section microscopy. *Journal of Microscopy* 218: 52–61.
6. Lu J, Fiala JC, Lichtman JW (2009) Semi-automated reconstruction of neural processes from large numbers of fluorescence images. *PLoS ONE* 4: e5655.

Besides, some computationally expensive but more robust algorithms could be resorted to. For example, it has been shown that the optimal path is useful for automated branch mergers or segment joins [30]. In addition, for a real large-scale dataset, such as the MOST Golgi images that are stained highly unevenly and exhibit variable contrast, a practical pre-processing platform that produces more high quality images for automated tracing and reconstruction is appreciated.

In summary, the proposed system provides an efficient means for the rapid reconstruction of 3D neuronal morphology from light microscopy images, which achieves a reasonable balance between fast speed and acceptable accuracy. This system is very promising for interactive applications of large-scale neuronal image analysis, such as the reconstruction of single neurons and local neural circuits from MOST images.

## Supporting Information

**File S1 Supporting Information for Methods, Experiments and Results.** Text S1, Sampling boundary detection along a sampling ray. Text S2, flNeuronTool: neuron reconstruction system. Text S3, Validation of the reliability of the radius estimation.  
(PDF)

## Acknowledgments

We thank Shangbin Chen and Tingwei Quan for critical reading of the manuscript, and Guozhi Zhang for his help with language editing. The anonymous reviewers are also acknowledged for the constructive comments.

## Author Contributions

Conceived and designed the experiments: HG SZ QL. Performed the experiments: XM. Analyzed the data: XM QL. Contributed reagents/materials/analysis tools: AL JW CY WD. Wrote the paper: XM QL.

7. Myatt DR, Hadlington T, Ascoli GA, Nasuto SJ (2012) Neuromantic – from semi-manual to semi-automatic reconstruction of neuron morphology. *Frontiers in Neuroinformatics* 6: 4.
8. Rodriguez A, Ehlenberger D, Kelliher K, Einstein M, Henderson SC, et al. (2003) Automated reconstruction of three-dimensional neuronal morphology from laser scanning microscopy images. *Methods* 30: 94–105.
9. Weaver CM, Hof PR, Wearne SL, Lindquist WB (2004) Automated algorithms for multiscale morphometry of neuronal dendrites. *Neural Computation* 16: 1353–1383.
10. Streekstra GJ, van Pelt J (2002) Analysis of tubular structures in three-dimensional confocal images. *Network: Computation in Neural Systems* 13: 381–395.
11. Yuan X, Trachtenberg JT, Potter SM, Roysam B (2009) MDL constrained 3-D grayscale skeletonization algorithm for automated extraction of dendrites and spines from fluorescence confocal images. *Neuroinformatics* 7: 213–232.
12. Gonzalez G, Aguet F, Fleuret F, Uner M, Fua P (2009) Steerable features for statistical 3D dendrite detection. *Medical Image Computing and Computer-Assisted Intervention* 12: 625–632.
13. Vasilkoski Z, Stepanyants A (2009) Detection of the optimal neuron traces in confocal microscopy images. *Journal of Neuroscience Methods* 178: 197–204.
14. Chothani P, Mehta V, Stepanyants A (2011) Automated tracing of neurites from light microscopy stacks of images. *Neuroinformatics* 9: 263–278.
15. Rodriguez A, Ehlenberger DB, Hof PR, Wearne SL (2009) Three-dimensional neuron tracing by voxel scooping. *Journal of Neuroscience Methods* 184: 169–175.
16. Al-Kofahi KA, Lasek S, Szarowski DH, Pace CJ, Nagy G, et al. (2002) Rapid automated three-dimensional tracing of neurons from confocal image stacks. *IEEE Transactions on Information Technology in Biomedicine* 6: 171–187.
17. Al-Kofahi KA, Can A, Lasek S, Szarowski DH, Dowell-Mesfin N, et al. (2003) Median-based robust algorithms for tracing neurons from noisy confocal microscope images. *IEEE Transactions on Information Technology in Biomedicine* 7: 302–317.
18. Abdul-Karim M-A, Al-Kofahi K, Brown EB, Jain RK, Roysam B (2003) Automated tracing and change analysis of angiogenic vasculature from in vivo multiphoton confocal image time series. *Microvascular Research* 66: 113–125.
19. Meijering E, Jacob M, Sarria JCF, Steiner P, Hirling H, et al. (2004) Design and validation of a tool for neurite tracing and analysis in fluorescence microscopy images. *Cytometry* 58A: 167–176.
20. Zhang Y, Zhou X, Degterev A, Lipinski M, Adjero D, et al. (2007) Automated neurite extraction using dynamic programming for high-throughput screening of neuron-based assays. *NeuroImage* 35: 1502–1515.
21. Peng H, Ruan Z, Atasoy D, Sternson S (2010) Automatic reconstruction of 3D neuron structures using a graph-augmented deformable model. *Bioinformatics* 26: i38–i46.
22. Peng H, Ruan Z, Long F, Simpson JH, Myers EW (2010) V3D enables real-time 3D visualization and quantitative analysis of large-scale biological image data sets. *Nature Biotechnology* 28: 348–353.
23. Peng H, Long F, Myers G (2011) Automatic 3D neuron tracing using all-path pruning. *Bioinformatics* 27: i239–i247.
24. Türetken E, González G, Blum C, Fua P (2011) Automated reconstruction of dendritic and axonal trees by global optimization with geometric priors. *Neuroinformatics* 9: 279–302.
25. Xie J, Zhao T, Lee T, Myers E, Peng H (2011) Anisotropic path searching for automatic neuron reconstruction. *Medical image analysis* 15: 680–689.
26. Schmitt S, Evers JF, Duch C, Scholz M, Obermayer K (2004) New methods for the computer-assisted 3-D reconstruction of neurons from confocal image stacks. *NeuroImage* 23: 1283–1298.
27. Cai H, Xu X, Lu J, Lichtman JW, Yung SP, et al. (2006) Repulsive force based snake model to segment and track neuronal axons in 3D microscopy image stacks. *NeuroImage* 32: 1608–1620.
28. Wang Y, Narayanaswamy A, Tsai C-L, Roysam B (2011) A broadly applicable 3-D neuron tracing method based on open-curve snake. *Neuroinformatics* 9: 193–217.
29. Al-Kofahi Y, Dowell-Mesfin N, Pace C, Shain W, Turner JN, et al. (2008) Improved detection of branching points in algorithms for automated neuron tracing from 3D confocal images. *Cytometry Part A* 73A: 36–43.
30. Zhao T, Xie J, Amat F, Clack N, Ahammad P, et al. (2011) Automated reconstruction of neuronal morphology based on local geometrical and global structural models. *Neuroinformatics* 9: 247–261.
31. Choromanska A, Chang S-F, Yuste R (2012) Automatic reconstruction of neural morphologies with multi-scale tracking. *Frontiers in Neural Circuits* 6: 1–13.
32. Wearne SL, Rodriguez A, Ehlenberger DB, Rocher AB, Henderson SC, et al. (2005) New techniques for imaging, digitization and analysis of three-dimensional neural morphology on multiple scales. *Neuroscience* 136: 661–680.
33. Rodriguez A, Ehlenberger DB, Hof PR, Wearne SL (2006) Rayburst sampling, an algorithm for automated three-dimensional shape analysis from laser scanning microscopy images. *Nature Protocols* 1: 2152–2161.
34. Luisi J, Narayanaswamy A, Galbreath Z, Roysam B (2011) The FARSIGHT trace editor: an open source tool for 3-D inspection and efficient pattern analysis aided editing of automated neuronal reconstructions. *Neuroinformatics* 9: 305–315.
35. Peng H, Long F, Zhao T, Myers E (2011) Proof-editing is the bottleneck of 3D neuron reconstruction: the problem and solutions. *Neuroinformatics* 9: 103–105.
36. Brown KM, Barrionuevo G, Cauty AJ, Paola V, Hirsch JA, et al. (2011) The DIADEM data sets: representative light microscopy images of neuronal morphology to advance automation of digital reconstructions. *Neuroinformatics* 9: 143–157.
37. Liu Y (2011) The DIADEM and beyond. *Neuroinformatics* 9: 99–102.
38. Li A, Gong H, Zhang B, Wang Q, Yan C, et al. (2010) Micro-optical sectioning tomography to obtain a high-resolution atlas of the mouse brain. *Science* 330: 1404–1408.
39. Zhang B, Li A, Yang Z, Wu J, Luo Q, et al. (2011) Modified Golgi-Cox method for micrometer scale sectioning of the whole mouse brain. *Journal of Neuroscience Methods* 197: 1–5.
40. Gong H, Zeng S, Yan C, Lv X, Yang Z, et al. (2013) Continuously tracing brain-wide long-distance axonal projections in mice at a one-micron voxel resolution. *NeuroImage* 74: 87–98.
41. Yang Z, Hu B, Zhang Y, Luo Q, Gong H (2013) Development of a plastic embedding method for large-volume and fluorescent-protein-expressing tissues. *PLoS ONE* 8: e60877.
42. Ascoli GA, Krichmar JL, Nasuto SJ, Senft SL (2001) Generation, description and storage of dendritic morphology data. *Philosophical transactions of the Royal Society of London Series B, Biological sciences* 356: 1131–1145.
43. Mayerich D, Keyser J (2009) Hardware accelerated segmentation of complex volumetric filament networks. *IEEE Transactions on Visualization and Computer Graphics* 15: 670–681.
44. Sonka M, Hlavac V, Boyle R, editors (2007) *Image Processing Analysis and Machine Vision*: Thomson-Engineering.
45. Hamarneh G, Jassi P (2010) VasuSynth: simulating vascular trees for generating volumetric image data with ground-truth segmentation and tree analysis. *Computerized Medical Imaging and Graphics* 34: 605–616.
46. Jefferis GSXE, Potter CJ, Chan AM, Marin EC, Rohlfling T, et al. (2007) Comprehensive maps of *Drosophila* higher olfactory centers: spatially segregated fruit and pheromone representation. *Cell* 128: 1187–1203.
47. Lu J, Tapia JC, White OL, Lichtman JW (2009) The interscutularis muscle connectome. *PLoS Biology* 7: e1000032.
48. Gillette TA, Brown KM, Ascoli GA (2011) The DIADEM metric: comparing multiple reconstructions of the same neuron. *Neuroinformatics* 9: 233–245.