Reconstruction of 3D Neuron Morphology Using Rivulet Back-Tracking

Donghao Zhang⋆*, Siqi Liu*⋆*, Sidong Liu*⋆*, Dagan Feng*⋆*, Hanchuan Peng*†*, Weidong Cai*⋆

⋆ School of Information Technologies, University of Sydney, Australia
† Allen Institute for Brain Science, Seattle, WA, USA

ABSTRACT

The 3D reconstruction of neuronal morphology is a powerful technique for investigating nervous systems. Due to the noises in optical microscopic images, the automated reconstruction of neuronal morphology has been a challenging problem. We propose a novel automatic neuron reconstruction algorithm, Rivulet, to target the challenges raised by the poor quality of the optical microscopic images. After the neuron images being de-noised with an anisotropic filter, the Rivulet algorithm combines multi-stencils fast-marching and iterative back-tracking from the geodesic farthest point on the segmented foreground. The neuron segments are dumped or merged according to a set of criteria at the end of each iteration. The proposed Rivulet tracing algorithm is evaluated with data provided from the BigNeuron Project. The experimental results demonstrate that Rivulet outperforms the compared state-of-the-art tracing methods when the images are of poor quality.

***Index Terms*—** Neuron morphology, curvilinear structure tracing, fast marching, vesselness filtering

1. INTRODUCTION

The digital reconstruction of neurons is an important tool in acquiring the morphological information of single neurons, which is critical to understand their functions within neural circuits [1]. Single neurons can be captured from 3D optical microscopic images with various labelling visualization techniques, such as bulk dye loading, intracellular injections, immunolabeling and genetic labelling [2]. The single neuron morphology can be reconstructed from the microscopic images and represented as 3D tubular models. The majority of the neuronal morphology datasets were collected by computer-aided manual tracing [2]. However, manual neuron reconstruction is a labour intensive task that is difficult to scale to large datasets.

Automatic/semi-automatic neuron reconstruction algorithms have been developed in recent years to enable the large-scale acquisition of neuron morphological data. In Neuron Studio [3], the tracing is initialised from a manually chosen source point and shoots rays from a spherical model (Rayburst) at new traced locations. Neuron Tube [4] and MOST algorithm [5] improve the robustness of the Rayburst algorithm by considering more shape specifics. Snake [6] traces from various seeding locations following a preprocessed gradient vector flow (GVF) map. All Path Pruning algorithms (APP1 and APP2) [7, 8] prune the redundant branches from a neuronal tree that results from an initial over-reconstruction based on fast-marching.

Although these powerful algorithms have been proposed, the automatic reconstruction of single neurons remains a bottleneck in the processing pipeline of microscopic neuron images due to the varying image qualities. The broken and fuzzy neurite segments cause major difficulties in neuron tracing. Most existing algorithms are incapable of bridging the gaps between the broken segments without introducing miswiring. The noises from labelling techniques and microscopies cannot be trivially removed by the median filter or Gaussian filter that are widely used in processing general images. These noises cause over-tracing in most seed dependent algorithms [3, 5, 6], and also increase the risk of branch miswiring in post-processing. In addition, the imperfect shapes captured in real microscopic images frequently violate the shape-based assumption relied on by many tracing algorithms.

In this study, we propose a novel tracing algorithm to target the above mentioned challenges. The curvilinear structures are segmented using an anisotropic filter. The multi-stencils fast-marching is applied to a distance field to produce a time-crossing map. Inspired by the all path pruning algorithms [7, 8], the longest remaining neuron segments are discovered iteratively. To achieve this, the Rivulet back-tracking is performed based on the gradient of the time-crossing map to trace from the geodesic farthest point on the segmented foreground to the soma location. After each iteration, the discovered path is erased from the time-crossing map to avoid duplicated tracing. The whole Rivulet back-tracking process only stops when a pre-defined proportion of the foreground area is covered; thus, it is highly robust to under-tracing. The Rivulet back-tracking is also robust to small structural gaps and holes caused by a low signal to noise ratio (SNR). We use the percentile of back-tracking locations traced within

This work was supported in part by ARC grants. Siqi Liu (siqi.liu@sydney.edu.au) and Weidong Cai (tom.cai@sydney.edu.au) are the corresponding authors of this paper.

foreground voxels to measure the confidence of each traced segment. Since the fast-marching is performed on a distance transform map which has brighter intensities near the centreline of the neuron segments, Rivulet is insensitive to tubular structures with non-smooth boundaries.

2. METHODS

2.1. Hessian Based Neuron Segmentation

The image is firstly cropped to remove the empty background regions. Since noise points and irrelevant non-curvilinear structures would cause errors in neuron tracing, we apply anisotropic filter [9] to remove the image noise and preserve the curvilinear neuronal structures. The Hessian matrix $H(x)$ is obtained by calculating the second-order gradient of the Gaussian filtered image. The function $g(x)$ is defined as

$$g(x) = \begin{cases} \sum_{i=1}^{i=3} \alpha_i k_i \lambda_i & (\lambda_1 \approx 0, \lambda_1 \gg \lambda_2, \lambda_1 \gg \lambda_3) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where λ_1, λ_2 and λ_3 are the eigenvalues of $H(x)$; α_i are predefined as α_1 = 0.5, α_2 = 0.5, α_3 = 25; k_i is the normalising term and $k_i = exp(-\lambda_i^2 \,/\, \sum_{i=1}^{i=3} \lambda_i)$. The foreground binary image is segmented by applying a threshold of 0 on the filtered image $f(x) = exp(-|\nabla x|^2)g(x)$. Though the anisotropic filter is capable of eliminating a number of noise points without corrupting the neuron segments of interest, for highly noisy images noise points may remain in the binary image $I_b(x) \in \{0, 1\}$ which are considered in the Rivulet tracing.

2.2. Multi-Stencils Fast-Marching

The first step of Rivulet tracing is to generate time-crossing map T based on multi-stencils fast-marching (MSFM) by solving the Eikonal equation Eq. (2):

$$F = \frac{dx}{dT}, \quad |\nabla T|F = 1, \quad T(\Gamma_0) = 0 \quad (2)$$

where F is the speed evolution function of fast marching, Γ_0 is the starting condition [10]. MSFM solves Eq. (2) along multiple stencils, thus avoiding the numerical errors of conventional fast-marching methods along diagonal directions. The speed image F is computed as a boundary distance transform with sub-voxel precision that has bright intensities near the centre-line of the neuron segments. To compute F, the boundary of the segmented foreground map $B(I_b)$ is firstly obtained by subtracting the I_b by the dilated foreground $I_b \oplus K$, where K is a 3D dilation kernel of size $3 \times 3 \times 3$. Then MSFM is applied at all the points on the boundary [11]. The voxels within the foreground region of F represent the closest distance to any voxels in $B(I_b)$. All voxels in the background region of F are set to zero. Thus, the time crossing map $T(x)$ is defined as:

$$T(x) = \min_{P_{(s,x)}} \int_0^L C(P(t))dt \quad (3)$$

$$C(P(t)) = \left(\frac{F(P(t))}{\max F} \right)^4 \quad (4)$$

where $P_{(s,x)}$ is a set of possible paths L from the source point s to x; $C(.)$ is the cost function for back-tracking procedure to avoid cutting corners. T is then used for the Rivulet back-tracking procedure to determine the centre-lines of neuron segments.

2.3. Rivulet Back-Tracking

The source point s of Rivulet back-tracking is chosen at the location with $\max F$. s is usually the soma position, therefore this step is equivalent to soma detection. Then the back-tracking procedure starts from the point $P(t_0)$ in the segmented foreground with the maximum value in $T(x)$. In other words, $P(t_0)$ is the point with longest accumulative Hamming distance along the centreline from s. $P(t)$ is tracked on $T(x)$ by the gradient descent with fourth-order Runge-Kutta method (RK4).

We define a gap $G > 0$ when back-tracking steps on the background voxels. G increments by 1 for each continuous step on the background and reset to 0 when a foreground voxel is stepped. When G is greater than a predefined threshold, the back-tracking of $P(t)$ stops. G is designed to stop the back-tracking procedure to avoid miswiring. The back-tracking also stops when the Euclidean distance between $P(t_i)$ and $P(t_{i-15})$ is less than one voxel size or $P(t_i)$ is within the voxel of s.

We introduce a simple criterion to evaluate the confidence of each traced branch $P(t)$, as:

$$\overline{P(t)} = \frac{1}{|P(t)|} \int_0^1 I_b(P(t))dt \quad (5)$$

where $|P(t)|$ is the length of $P(t)$. $|P(t)|$ can be illustrated as the proportion of gradient descent steps made on foreground voxels. The obtained branches with low confidences are discarded. $\overline{P(t)}$ makes the Rivulet tracing procedure more robust to the noise points. The first branch is directly added to the neuron tree and considered as the initial trunk and the last point of the branch is set as root. When a new branch is tracked, we apply time sequential wiring to determine whether the branch is connected to the tree. The branch is not connected if the Euclidean distance of termini p and the tree Tr, $D(p, Tr) < R \times (r_p + 3)$ or $D(p, Tr) < R \times (r_{min} + 3)$, where R is the connecting threshold; r_p is the radius of p and r_{min} is the radius of the node with the shortest distance to p. A spherical region ω_i produced by the each new

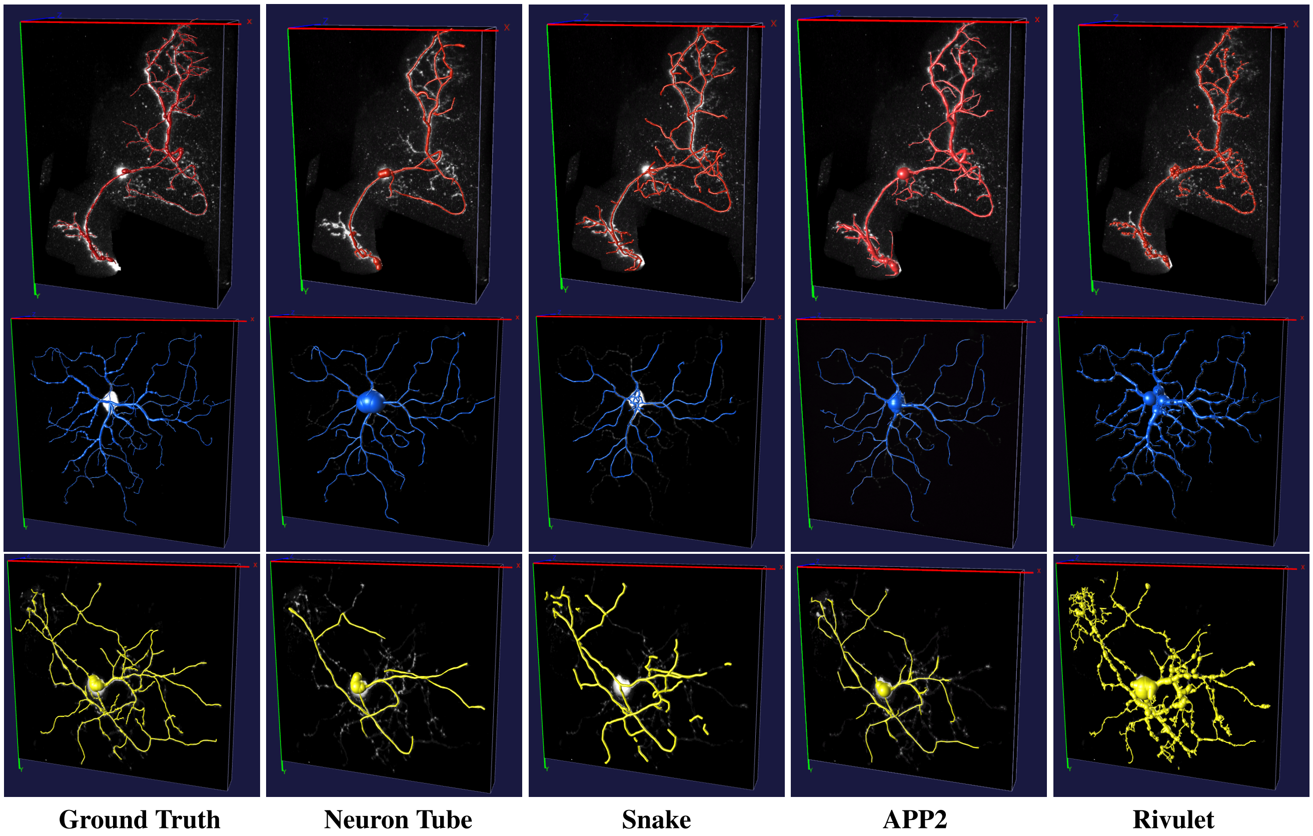


Fig. 1. The first row shows the reconstructions of a fly neuron with non-uniform distributed noises; the second row shows the reconstructions of a fruit-fly neuron with complex arborisation; the third row displays the reconstructions of a noise corrupted frog neuron with serious discontinuity. The 3D neuron reconstructions shown here were all produced automatically without manual correction.

branch point is erased from $T(x)$. The regions on $T(x)$ covered by the tracked branches cannot be revisited in following back-tracking iterations. The foreground point with highest value in the erased $T(x)$ is used as the starting point of next back-tracking iteration. After each iteration, a coverage percentage Ψ is increased by $\sum^{x\subset\Omega} I_b(x)/\sum I_b(x)$, where $\Omega = \{\omega_1 \cup \omega_2 \cup \cdots \cup \omega_n\}$. A higher Ψ reveals more details of small branches. The back-tracking process only terminates when Ψ reaches a predefined threshold (eg. 98%). The time complexity of Rivulet tracing is O($n \log n$), where n is the number of the foreground voxels. Since the time-crossing map erasing avoids duplicated fast-marching and gradient calculation, the $O(n \log n)$ term comes with the fast-marching method alone.

3. EXPERIMENTS AND RESULTS

The data we used to evaluate the proposed Rivulet algorithm were provided by the BigNeuron Project [12][1]. All the tracing results were visualised with Vaa3D [13]. Rivulet has been implemented in an open-sourced GUI Matlab toolbox[2]. All hyper-parameters of the evaluated tracing algorithms were tuned by exhaust searches and validated with visual check.

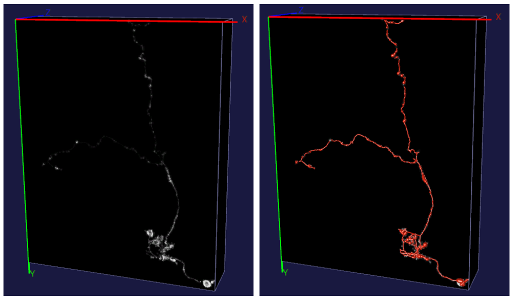
	NeuTube	Snake	APP2	Rivulet
SD	6.63	8.35	4.71	3.59
SSD	12.07	14.25	10.19	5.59
SSD%	0.37	0.45	0.41	0.57

Table 1. The mean SD, SSD and SSD% scores [7] of the compared methods in Fig. 1.

We compared Rivulet with the state-of-the-art automatic neuron tracing algorithms. The results are shown in Fig. 1. In the first row, Rivulet was able to trace the majority of the neuron arbors without being affected by the noises. Rivulet was also shown to trace the image in the second row with any small gaps and complex arborisation. The image in the third row has a low contrast which resulted in blurry boundaries between the neuron and the background. Since the neuron segmentation is not used as the boundary wall in Rivulet tracing,

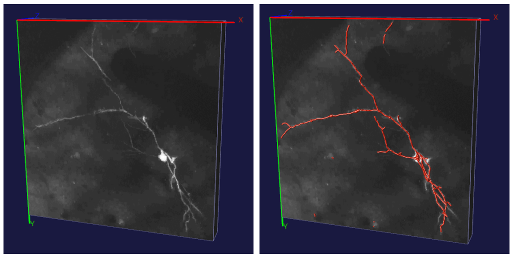
[1] http://alleninstitute.org/bigneuron/about/

[2] https://github.com/lsqshr/Rivulet-Neuron-Tracing-Toolbox



(a) A neuron image with discontinuous structures (b) A complete reconstruction of the corrupted image

Fig. 2. The discontinuous and fuzzy neurons images were constructed by Rivulet.



(a) A noisy confocal microscopic neuron image (b) The reconstruction result by Rivulet

Fig. 3. The Rivulet reconstruction result of an extremely noisy image.

Rivulet was shown to be more robust to dark branches and discontinuous branches. In the quantitative analysis shown in Table 1, Rivulet was also shown to outperform the other compared methods regarding the spatial distance (SD), substantial spatial distance scores (SSD) [7]. The slightly higher mean percentage of substantial spatial distance (SSD%) of Rivulet is due to the larger number of nodes of Rivulet.

The reconstruction results of a fly neuron with discontinuous and fuzzy structures by Rivulet are shown in Fig. 2. The discontinuous structures were caused by imperfect staining and excitation power during image acquisition. The gap threshold and time sequential wiring enable Rivulet to trace discontinuous neuron arbours and resistant to over-tracing at the same time. The unevenly distributed fluorescent markers of fly neuron lead to extremely noisy images. For the noisy image shown in Fig. 3, Rivulet was the only algorithm among the evaluated ones that reconstructed meaningful results, mainly because of the gap threshold and the confidence dumping strategy.

4. CONCLUSIONS

In this study, we proposed a novel automatic tracing algorithm, Rivulet, based on iterative back-tracking. Evaluated with data provided by the BigNeuron project, Rivulet shows great robustness to discontinuous and noisy neuronal structures in confocal microscopic images. Rivulet also outperformed other state-of-the-art neuron tracing algorithms on the challenging images with low SNR, discontinuous arbours and complex topologies.

References

[1] H Peng and M Hawrylycz et al., “BigNeuron: Large-scale 3D neuron reconstruction from optical microscopy images,” *Neuron*, vol. 87, no. 2, pp. 252 – 256, 2015.

[2] R Parekh, G Ascoli, “Neuronal morphology goes digital: a research hub for cellular and system neuroscience,” *Neuron*, vol. 77, no. 6, pp. 1017–1038, 2013.

[3] S.L. Wearne, A Rodriguez, et al., “New techniques for imaging, digitization and analysis of three-dimensional neural morphology on multiple scales,” *Neuroscience*, vol. 136, no. 3, pp. 661–680, 2005.

[4] L Feng, T Zhao, et al., “Neutube 1.0: a new design for efficient neuron reconstruction software based on the swc format,” *eneuro*, vol. 2, no. 1, pp. 49, 2015.

[5] X Ming and A Li et al., “Rapid reconstruction of 3d neuronal morphology from light microscopy images with augmented rayburst sampling,” *PLoS ONE*, vol. 8, no. 12, pp. e84557, 2013.

[6] Y Wang and A Narayanaswamy et al., “A broadly applicable 3D neuron tracing method based on open-curve snake,” *Neuroinformatics*, vol. 9, no. 2-3, pp. 193–217, 2011.

[7] H Peng, F Long, and G Myers, “Automatic 3D neuron tracing using all-path pruning,” *Bioinformatics*, vol. 27, no. 13, pp. i239–i247, 2011.

[8] H Xiao, H Peng, “APP2: automatic tracing of 3D neuron morphology based on hierarchical pruning of a gray-weighted image distance-tree,” *Bioinformatics*, vol. 29, no. 11, pp. 1448–1454, 2013.

[9] J Yang, P.T. Gonzalez-Bellido, et al., “A distance-field based automatic neuron tracing method,” *BMC Bioinformatics*, vol. 14, no. 1, pp. 93, 2013.

[10] D Adalsteinsson and J.A. Sethian, “A fast level set method for propagating interfaces,” *Journal of computational physics*, vol. 118, no. 2, pp. 269–277, 1995.

[11] S.M. Hassouna, A Farag, et al., “Multistencils fast marching methods: A highly accurate solution to the eikonal equation on cartesian domains,” *PAMI, IEEE Transactions on*, vol. 29, no. 9, pp. 1563–1574, 2007.

[12] H Peng, E Meijering, and G Ascoli, “From DIADEM to BigNeuron,” *Neuroinformatics*, vol. 13, no. 3, pp. 259–260, 2015.

[13] H Peng, A Bria, et al., “Extensible visualization and analysis for multidimensional images using Vaa3D,” *Nature Protocols*, vol. 9, no. 1, pp. 193–208, 2014.