NEURON RECONSTRUCTION FROM FLUORESCENCE MICROSCOPY IMAGES USING SEQUENTIAL MONTE CARLO ESTIMATION

Miroslav Radojević and Erik Meijering

Biomedical Imaging Group Rotterdam
Erasmus University Medical Center, Rotterdam, the Netherlands
Email: meijering@imagescience.org

ABSTRACT

Microscopic analysis of neuronal cell morphology is required in many studies in neurobiology. The development of computational methods for this purpose is an ongoing challenge and includes solving some of the fundamental computer vision problems such as detecting and grouping sometimes very noisy line-like image structures. Advancements in the field are impeded by the complexity and immense diversity of neuronal cell shapes across species and brain regions, as well as by the high variability in image quality across labs and experimental setups. Here we present a novel method for fully automatic neuron reconstruction based on sequential Monte Carlo estimation. It uses newly designed models for predicting and updating branch node estimates as well as novel initialization and final tree construction strategies. The proposed method was evaluated on 3D fluorescence microscopy images containing single neurons and neuronal networks for which manual annotations were available as gold-standard references. The results indicate that our method performs favorably compared to state-of-the-art alternative methods.

***Index Terms*—** Neuron reconstruction, Bayesian filtering, particle filtering, fluorescence microscopy.

1. INTRODUCTION

The complexity and functionality of the brain depend critically on the morphology and interconnectivity of the neurons [1]. But exactly how neuronal morphology and higher-order brain functions are related remains elusive and is still a subject of intense scientific research [2]. In this endeavor, a prominent tool for visualizing neurons in vitro as well as in vivo is fluorescence microscopy. To allow quantitative measurement and statistical analysis of the physical properties of neurons, a crucial step is to extract the relevant structural information from the images and turn it into a faithful digital representation. State-of-the-art methods for this purpose are still far from perfect, as discussed in various reviews [3–5], and the urgent need for better methods have driven researchers in the field to organize international studies such as the DIADEM challenge [6] and the BigNeuron project [7]. In the context of these and other studies, numerous methods have been reported showing varying performance and usability, and the quest to design better methods continues.

Here we propose a new method for neuron reconstruction from 3D fluorescence microscopy images based on sequential Monte Carlo estimation. An early version of the method is a contender in the ongoing BigNeuron benchmarking study [7] but has not been published before. In this paper we present the concept and the new design aspects of the method for the first time. Employing particle filtering as the underlying tracing algorithm [8, 9], we propose novel state transition and observation models, as well as novel initialization and tree construction strategies. Tracing commences from seed points obtained using a standard multiscale line filter [10] and proceeds by recursively predicting and updating the next node on a branch according to the defined models. This results in a collection of possibly overlapping but probabilistically independent estimates of the neuronal branches in the image, providing more evidence for reconstruction than normally obtained by deterministic methods, and achieving more robustness against uncertainty in the data. The overcomplete estimations are subsequently combined into a refined estimate of the tree node positions and connectivities using mean-shifting. We describe the technical aspects of our method and the results of a preliminary performance evaluation on real fluorescence microscopy images from the DIADEM challenge.

2. METHOD

The proposed method consists of six main steps (Fig. 1), each of which we briefly describe in this section.

Step A: Soma Extraction. The cell body (soma) of a neuron has a larger diameter than its dendritic and axonal arbors and thus can be detected by performing gray-scale erosion using a structuring element with a slightly larger diameter than the largest expected branch diameter. A segmentation of the soma can then be obtained by max-entropy thresholding [11]. In

This work was funded by the Netherlands Organization for Scientific Reseach (NWO grant 612.001.018 awarded to EM).

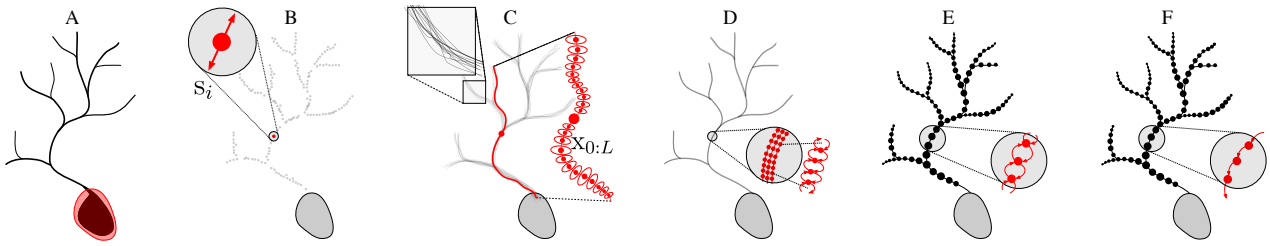


Fig. 1. Schematic overview of the six main steps of the proposed method: (A) soma extraction, (B) seed extraction, (C) branch tracing, (D) trace refinement, (E) node grouping, (F) tree construction.

our applications it is sufficient to model the soma as a single node whose diameter can be estimated by fitting a sphere to the segmented image region.

Step B: Seed Extraction. Seeds $\mathrm{s}_i = [\mathrm{p}_i, \mathrm{v}_i, \sigma_i]$ on the arbors are obtained by applying Hessian-based multiscale tubularity filtering [10] and sampling from the local maxima of the filter response. Here, $\mathrm{p}_i = [x_i, y_i, z_i]$ denotes the local branch position, $\mathrm{v}_i = [v_{x_i}, v_{y_i}, v_{z_i}]$ the local branch direction taken as the Hessian eigenvector corresponding to the smallest eigenvalue, and σ_i the local branch thickness represented by the Gaussian scale at which the tubularity filter yields the highest response. Only those maxima are considered whose max-min difference within a cross-sectional neighborhood orthogonal to v_i is larger than a given tolerance τ.

Step C: Branch Tracing. For each seed point, a sequence of hidden states $\mathrm{x}_{0:L} = (\mathrm{x}_0, \ldots, \mathrm{x}_L)$ is estimated representing a local branch segment, where x_0 and x_L are the state of the start (seed) point and end (last) point of the segment, respectively. The state vector $\mathrm{x}_i = [\mathrm{p}_i, \mathrm{v}_i, \sigma_i]$ consists of an estimate of the position p_i, direction v_i, and scale σ_i of the corresponding branch node. Using a sequence of measurements $\mathrm{z}_{0:L}$ and applying Bayes' rule we can iteratively estimate the posterior distribution of the state vectors by

$$p(\mathrm{x}_i|\mathrm{z}_{0:i}) \propto p(\mathrm{z}_i|\mathrm{x}_i)\int p(\mathrm{x}_i|\mathrm{x}_{i-1})p(\mathrm{x}_{i-1}|\mathrm{z}_{0:i-1})\mathrm{d}\mathrm{x}_{i-1} \quad (1)$$

where $p(\mathrm{x}_i|\mathrm{x}_{i-1})$ denotes the state transition (assumed to be Markovian) and $p(\mathrm{z}_i|\mathrm{x}_i)$ denotes the likelihood (measurements are assumed to be independent). In our algorithm, we solve the estimation problem using sequential Monte Carlo filtering [12], where the posterior at each iteration is approximated by N particles (samples) $\mathrm{x}_i^k = [\mathrm{p}_i^k, \mathrm{v}_i^k, \sigma_i^k]$, $k = 1, \ldots, N$, with corresponding weights w_i^k, $\sum_k w_i^k = 1$, such that $p(\mathrm{x}_i|\mathrm{z}_{0:i}) \approx \sum_k w_i^k \delta(\mathrm{x}_i - \mathrm{x}_i^k)$.

The importance sampling distribution that we use in the prediction step to draw a particle x_i^k given a particle x_{i-1}^k from the previous iteration is given by (Fig. 2)

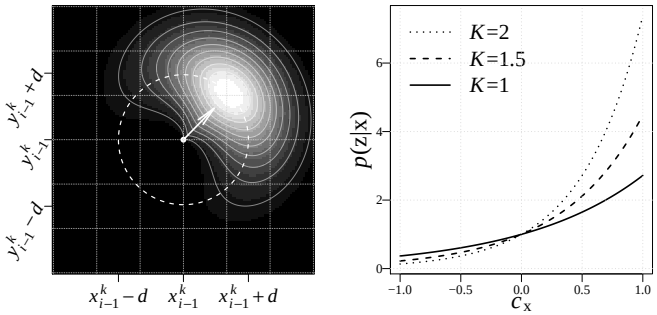
$$p(\mathrm{x}_i|\mathrm{x}_{i-1}^k) = \begin{cases} \frac{\exp\left(\kappa\, \mathrm{v}_i \cdot \mathrm{v}_{i-1}^k - \frac{(d_i - d)^2}{2(d/3)^2}\right)}{2\pi I_0(\kappa)\eta} & d_i \leq 2d \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where η is a normalization factor so that the integral of $p(\mathrm{x}_i|\mathrm{x}_{i-1}^k)$ over $d_i \leq 2d$ is unity, I_0 is the zero-order Bessel function of the first kind, κ is the circular variance parameter, $d_i = ||\mathrm{p}_i - \mathrm{p}_{i-1}^k||$ is the Euclidean distance between the predicted position and the particle position in the previous iteration, and d is the tracing step size. Each predicted state is assigned a unit direction $\mathrm{v}_i = (\mathrm{p}_i - \mathrm{p}_{i-1}^k)/||\mathrm{p}_i - \mathrm{p}_{i-1}^k||$ defined by two consecutive positions.

The likelihood that we use to update a predicted state x is given by $p(\mathrm{z}|\mathrm{x}) = e^{Kc_\mathrm{x}}$ (Fig. 2), with $c_\mathrm{x} = \max_\sigma c_{\mathrm{x},\sigma}$, where $c_{\mathrm{x},\sigma} \in [-1, 1]$ denotes the normalized cross-correlation value of the image at the predicted position with a template having a Gaussian profile with scale σ (see Table 1 for the considered scales) in the plane orthogonal to the predicted direction. Filtering is terminated if the average particle correlation at iteration i, $\sum_k c_{\mathrm{x}_i^k}/N$, drops below the correlation threshold c_min. The particle weights are updated as

$$w_i^k \propto w_{i-1}^k p(\mathrm{x}_i^k|\mathrm{x}_{i-1}^k) e^{Kc_{\mathrm{x}_i^k}} \quad (3)$$

and then renormalized. The final node estimate at each iteration is computed from the weighted particles as the centroid $\hat{\mathrm{x}}_i = \sum_k w_i^k \mathrm{x}_i^k$. To avoid particle weight deterioration, systematic resampling [13] is performed each time the effective sample size N_eff [14] falls below 80% of N.



(a) $\mathrm{x}_i^k \sim p(\mathrm{x}_i|\mathrm{x}_{i-1}^k)$ (b) $p(\mathrm{z}|\mathrm{x}) = e^{Kc_\mathrm{x}}$

Fig. 2. The prediction importance sampling distribution (a) used by our algorithm (for ease of visualization a 2D example is given) and the measurement likelihood (b) for different values of K.

Step D: Trace Refinement. The previous step typically results in multiple traces for each neuron branch segment.

The traces are subsequently refined using mean-shifting [15], which iteratively moves the position of each trace node towards the local mean of the neighbouring nodes. In practice, 3-5 iterations are sufficient to reach satisfactory alignment of the traces. Prior to this, all traces are resampled with a step size of one voxel to achieve better alignment.

Step E: Node Grouping. The refined trace nodes are grouped into a final set of branch node estimates and corresponding interconnections by iteratively taking an as-yet ungrouped node with the highest cross-correlation value (Step C), calculating the centroid position of the nodes within grouping radius r of this node, marking all nodes included in this calculation as grouped to prevent them from being considered again, and repeating this until all nodes have been processed. Any two node groups are connected if at least one pair of nodes from the distinct groups are connected.

Step F: Tree Construction. Finally the branch nodes and connections are traversed using a breadth-first search algorithm to obtain a graph representing the complete neuronal tree which can be exported to the standard SWC file format.

3. RESULTS

The method was implemented in C++ as a plugin for the bioimage visualization and analysis platform Vaa3D [16]. It has several parameters with default values given in Table 1. As a preliminary evaluation we applied our method to two 3D data sets from the DIADEM challenge [17]: OPF (9 image stacks) and NCL1A (16 image stacks). Two sets of measures were used to quantify the match between the obtained reconstructions and the available gold-standard reconstructions: the spatial distance measures SD, SSD, and %SSD [16], and the overlap measures P (precision), R (recall), and F = 2 P R/(P+R) [18, 19]. We compared our probabilistic neuron reconstructor (PNR) to several state-of-the-art methods, including NeuroGPS-Tree (GPS) [18], all-path prunning (APP2) [20], minimum-spanning tree (MST) used in the BigNeuron project [7], and a method based on probability hypothesis density (PHD) filtering [19]. The parameters of each method were optimized for each evaluation measure. To reduce bias in the measurements we resampled each reconstruction (from the methods and from the gold standard) with an inter-node distance of one voxel. Because of space limitations we present in Fig. 3 the results for the SSD (lower is better) and F (higher is better) measures and note that the other measures lead to very similar observations. From the results we conclude that our method performs comparably or better than the alternative methods. The differences are particularly noticeable for the NCL1A data, where, unlike with OPF, there is considerable room for improvement. Example reconstructions with our method are shown in Fig. 4.

Parameter	Value	Description
σ	$\{2, 3\}$ [voxels]	Considered scales
τ	10 [8-bit scale]	Local maxima tolerance
$c_{\min}$	0.5	Correlation threshold
L	200	Iteration limit
N	20	Number of particles
κ	3 [voxels]	Circular variance
d	3 [voxels]	Tracing step size
r	2 [voxels]	Grouping radius

Table 1. Parameters of our method and their default values.

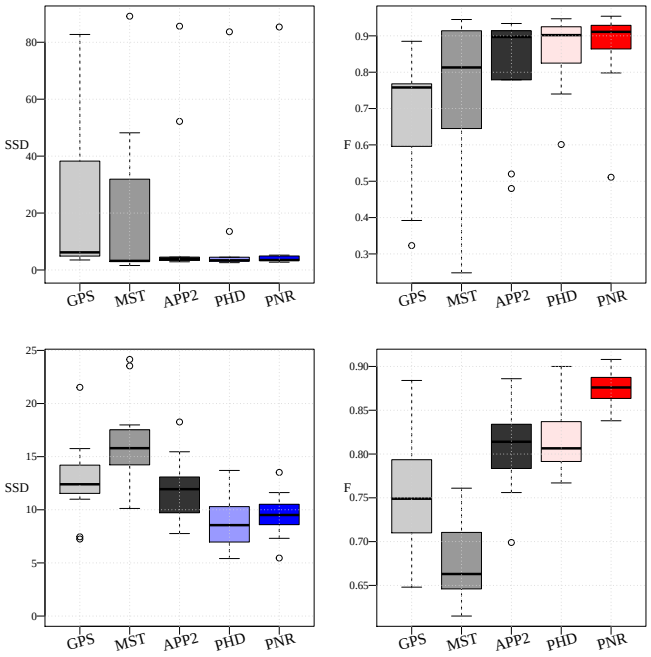


Fig. 3. Performance of the methods on the OPF (top row) and NCL1A (bottom row) data sets according to the SSD (left column) and F (right column) measures. For SSD we used a significant distance threshold of $S = 2$ voxels as is commonly done. Box plots were generated using the statistical software package R.

4. DISCUSSION

Our new method for reconstructing neuronal cell morphology from fluorescence microscopy images employs probabilistic branch tracing and subsequent merging of the traces to obtain a tree representation. The presented sequential Monte Carlo implementation of Bayesian filtering for the branch tracing step features substantial design changes compared to our earlier work [9], including new prediction and update models, extension to 3D, estimation of local branch thickness, and a new sampling procedure. Combined with the new trace merging procedure, our method yields complete reconstructions fully automatically. A main advantage of our method, owing to its probabilistic nature, is that neuron branches are typically traced multiple times with independent estimation results, allowing to better deal with data ambiguities. The

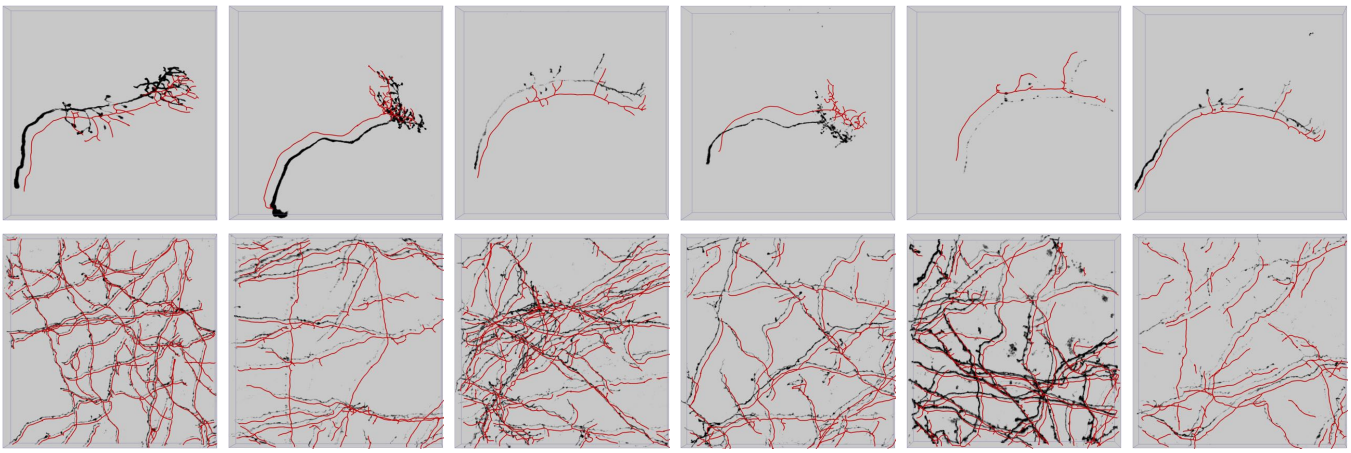


Fig. 4. Examples of neuron reconstructions produced by our method applied to representative cases from the OPF data set (top row) and the NCL1A data set (bottom row). The reconstructions (shown in red color) are overlaid on volume renderings (with inverted gray-scale intensities) of the image stacks and are slightly displaced to facilitate visual inspection.

presented results show favorable performance of our method compared to state-of-the-art alternative methods. Further improvements could be achieved by adding pruning strategies to deal with false-positive traces. In future work we will also test our method extensively on a much wider range of data available from the BigNeuron project.

5. REFERENCES

[1] D. E. Donohue and G. A. Ascoli, "A comparative computer simulation of dendritic morphology," *PLoS Computational Biology*, vol. 4, no. 6, pp. 1–15, May 2008.

[2] G. A. Ascoli, *Trees of the Brain, Roots of the Mind*, MIT Press, Cambridge, MA, 2015.

[3] E. Meijering, "Neuron tracing in perspective," *Cytometry Part A*, vol. 77, no. 7, pp. 693–704, July 2010.

[4] D. E. Donohue and G. A. Ascoli, "Automated reconstruction of neuronal morphology: an overview," *Brain Research Reviews*, vol. 67, no. 1, pp. 94–102, June 2011.

[5] L. Acciai, P. Soda, and G. Iannello, "Automated neuron tracing methods: an updated account," *Neuroinformatics*, vol. 14, no. 4, pp. 353–367, October 2016.

[6] T. A. Gillette, K. M. Brown, K. Svoboda, Y. Liu, and G. A. Ascoli, "DIADEMchallenge.Org: a compendium of resources fostering the continuous development of automated neuronal reconstruction," *Neuroinformatics*, vol. 9, no. 2-3, pp. 303–304, September 2011.

[7] H. Peng, M. Hawrylycz, J. Roskams, S. Hill, N. Spruston, E. Meijering, and G. A. Ascoli, "BigNeuron: large-scale 3D neuron reconstruction from optical microscopy images," *Neuron*, vol. 87, no. 2, pp. 252–256, July 2015.

[8] D. R. Myatt, S. J. Nasuto, and S. J. Maybank, "Towards the automatic reconstruction of dendritic trees using particle filters," in *Nonlinear Statistical Signal Processing Workshop*. IEEE, Cambridge, UK, 2006, pp. 193–196.

[9] M. Radojević, I. Smal, and E. Meijering, "Automated neuron morphology reconstruction using fuzzy-logic detection and Bayesian tracing algorithms," in *Proceedings of the IEEE International Symposium on Biomedical Imaging*. IEEE, New York, NY, 2015, pp. 885–888.

[10] A. Frangi, W. Niessen, K. Vincken, and M. Viergever, "Multiscale vessel enhancement filtering," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, Berlin, 1998, pp. 130–137.

[11] M. Radojević, I. Smal, and E. Meijering, "Fuzzy-logic based detection and characterization of junctions and terminations in fluorescence microscopy images of neurons," *Neuroinformatics*, vol. 14, no. 2, pp. 201–219, April 2016.

[12] B. Ristic, S. Arulampalam, and N. J. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech House, Boston, MA, 2004.

[13] G. Kitagawa, "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1–25, March 1996.

[14] A. Kong, J. S. Liu, and W. H. Wong, "Sequential imputations and Bayesian missing data problems," *Journal of the American Statistical Association*, vol. 89, no. 425, pp. 278–288, March 1994.

[15] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799, August 1995.

[16] H. Peng, Z. Ruan, F. Long, J. H. Simpson, and E. W. Myers, "V3D enables real-time 3D visualization and quantitative analysis of large-scale biological image data sets," *Nature Biotechnology*, vol. 28, no. 4, pp. 348–353, April 2010.

[17] K. M. Brown, G. Barrionuevo, A. J. Canty, V. De Paola, J. A. Hirsch, G. S. Jefferis, J. Lu, M. Snippe, I. Sugihara, and G. A. Ascoli, "The DIADEM data sets: representative light microscopy images of neuronal morphology to advance automation of digital reconstructions," *Neuroinformatics*, vol. 9, no. 2-3, pp. 143–157, September 2011.

[18] T. Quan, H. Zhou, J. Li, S. Li, A. Li, Y. Li, X. Lv, Q. Luo, H. Gong, and S. Zeng, "NeuroGPS-Tree: automatic reconstruction of large-scale neuronal populations with dense neurites," *Nature Methods*, vol. 13, no. 1, pp. 51–54, January 2016.

[19] M. Radojević and E. Meijering, "Automated neuron tracing using probability hypothesis density filtering," *Bioinformatics*, 2017, in press.

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