9-15-2019

Multispectral tracing in densely labeled mouse brain with nTracer

Douglas H Roossien

John M Webb

Cristina Mazuski

Erik D Herzog

et al.

OXFORD

Bioimage informatics

Multispectral tracing in densely labeled mouse brain with nTracer

Douglas H. Roossien[1], **Benjamin V. Sadis**[1,†], **Yan Yan**[1,†], **John M. Webb**[2,†], **Lia Y. Min**[1,3,4], **Aslan S. Dizaji**[1], **Luke J. Bogart**[5], **Cristina Mazuski**[2], **Robert S. Huth**[1], **Johanna S. Stecher**[1], **Sriakhila Akula**[1], **Fred Shen**[1], **Ye Li**[1], **Tingxin Xiao**[6], **Madeleine Vandenbrink**[1], **Jeff W. Lichtman**[7,8], **Takao K. Hensch**[5,7,8], **Erik D. Herzog**[2] and **Dawen Cai** [1,*]

[1]Department of Cell and Developmental Biology, University of Michigan Medical School, Ann Arbor, MI 48109, USA, [2]Department of Biology, Washington University in St. Louis, St. Louis, MO 63130, USA, [3]Life Sciences Institute and [4]School of Art and Design, University of Michigan, Ann Arbor, MI 48109, USA, [5]Department of Neurology, Boston Children's Hospital, Harvard Medical School, Boston, MA 02115, USA, [6]Department of Biology, University of Toronto, Toronto M5S 1A1, ON, Canada, [7]Department of Molecular and Cellular Biology and [8]Center for Brain Science, Harvard University, Cambridge, MA 02138, USA

*To whom correspondence should be addressed.

†The authors wish it to be known that these authors contributed equally.

Associate Editor: Robert Murphy

Abstract

Summary: This note describes nTracer, an ImageJ plug-in for user-guided, semi-automated tracing of multispectral fluorescent tissue samples. This approach allows for rapid and accurate reconstruction of whole cell morphology of large neuronal populations in densely labeled brains.

Availability and implementation: nTracer was written as a plug-in for the open source image processing software ImageJ. The software, instructional documentation, tutorial videos, sample image and sample tracing results are available at https://www.cai-lab.org/ntracer-tutorial.

Contact: dwcai@umich.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The successful application of multispectral fluorescent labeling techniques [such as Brainbow (Cai *et al.*, 2013; Livet *et al.*, 2007)] to fully dissect the complex network of neural circuits in the brain has been hampered by a lack of quantitative analysis tools. Currently, popular commercial software such as Neurolucida and iMaris allow either manual or user-guided tracing, but none handle multispectral images (Parekh and Ascoli, 2013). We thus wrote nTracer, an ImageJ-based program to facilitate post-acquisition processing and tracing of Brainbow multispectral images (Supplementary Fig. S1). Because the current bottleneck in existing fully automated tracing algorithms is proof reading and error correcting (Chothani *et al.*,

2011; Liu *et al.*, 2011; Peng *et al.*, 2011), combined with the lack of any ground-truth Brainbow tracing results for validation, we developed nTracer as a user-guided semi-automated tracing software which allows for 'on-the-fly' editing of tracing results. We found nTracer to be a successful platform for accurately tracing a variety of different neuronal subtypes in the mouse brain at an average rate of a few hours per neuron. In addition, nTracer's annotation function of synaptic sites may be used to establish connectivity between multiple neurons for network analysis.

The tracing results obtained by current nTracer described here will also serve as a ground-truth reference for future automatic tracing algorithm development. nTracer will also benefit from adapting

the more versatile data structure of the ImageJ2/Fiji platform to handle large dataset (Schindelin *et al.*, 2015). Combining Brainbow labeling with emerging super-resolution light microscopy techniques, such as Expansion Microscopy (Chen *et al.*, 2015) and its variant protein-retention ExM (Chozinski *et al.*, 2016; Tillberg *et al.*, 2016), is an exciting possibility for producing images at the spatial resolutions that are suitable to distinguish the closely positioned synapses and neuronal processes. Nonetheless, the current nTracer presented herein is an important tool that will allow neuroscience researchers to analyze morphology and anatomy of large populations of neurons within single samples using the light microscope.

2 Features

2.1 Post-acquisition processing functions

There are two major sources of color defects when imaging Brainbow samples. The first is due to absorption, scattering and photobleaching, which causes a gradual decrease in fluorescence intensity when imaging deeper into the tissue. We added a histogram matching correction to nTracer to normalize signal intensity while maintaining the intensity ratios between channels. The user is asked to select a reference image slice to which nTracer matches the histogram (see below) of each slice in the image stack. While the reference slice can be chosen from any channel or focal plane, the optimal reference will contain an evenly distributed histogram with minimal pixel values >95% of the maximum bin (Supplementary Fig. S2). This ensures that FP intensity remains constant in any depth of the 3D stack while minimizing amplification of background noise (Supplementary Fig. S2). The reference image's cumulative probability distribution function $CDF_{ref}()$ of its histogram $H_{ref}()$ is calculated. For each target image slice in all channels of the whole 3D stack, a new histogram $H_{tar}()$ is applied, which satisfies the condition that for each gray scale level G_{ref}, a G_{tar} is determined to satisfy

$$CDF_{ref}(G_{ref}) = CDF_{tar}(G_{tar}) \qquad (1)$$

The second source of color defects results from imperfect optical alignments and chromatic aberrations in the microscope system. Eliminating highly correlated background pixels, the masking function increase the sensitivity of correlating the fluorescent neurons.

Both the intensity correction and channel alignment can be done in batch mode on all images (e.g. all stacks if a multi-tile image was taken) in a selected folder. In addition to these corrections, nTracer provides a 3D stitching function that allows rapid merging from overlapping Brainbow image tiles to create a single image stack that covers a large tissue volume.

2.2 Tracing function and data structure

We incorporated two algorithms into nTracer to accurately trace neurites between user defined anchor points from specific neurons. The first prevents user error by not allowing anchor points to be assigned to different neurons, and the second prevents the skeletal trace from jumping to the wrong neuron when joining anchor points. To start a tracing, the user identifies a neuron to be traced and uses a mouse-click to suggest a start point on its process and to measure the neuron's color signature around the start point. In most cases, the mouse clicks hardly land onto the 'right' spot, which results in inaccurate color sampling. nTracer solves this problem by applying a mean-shift algorithm to automatically refine the user input and settle the start point onto the center or membrane wall of
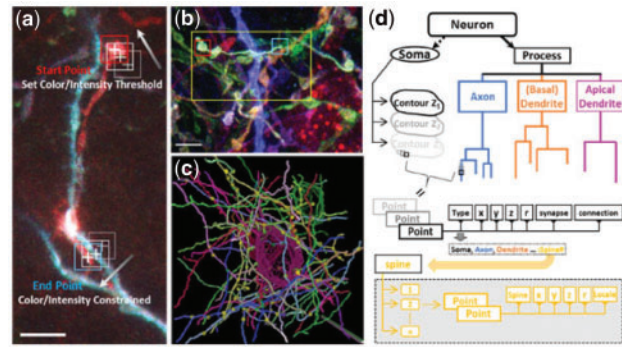

Fig. 1. nTracer. (**a**) nTracer uses a mean-shift algorithm to automatically refine the user mouse clicks near the targeted neurite to precisely define start and end points. To define the start point (red box), nTracer iteratively calculates the intensity 'center-of-mass' (cross) within a defined window (gray boxes) and moves the input point to the calculated 'center-of-mass' point until the distance between the previous and current iterations is smaller than one pixel. The end point (cyan box) is defined in a similar way with additional constraints set by the spectral and intensity values sampled around the start point (red box). The user can therefore avoid setting an end point onto a different neuronal process due to human visual or computer display limitations, in particular with Brainbow images comprised of more than three spectral channels. Scale bar is 10 μm. (**b**) To start tracing, a mouse-click is placed in the vicinity of the neurite to be traced. nTracer utilizes the color profile and a mean-shift algorithm to accurately reposition this mouse input onto the neurite as the start point (red box). The tracing end point (cyan box) is determined in the same way with additional constraints to make sure that the end point has a similar color profile. A keyboard hotkey is then used to trace as a neurite in 3D. Scale bar is 5 μm. (**c**) 3D tracing rendering of a cortical interneuron soma with putative synaptic contacts with neighboring axons. Scale bar is 10 μm. (**d**) Diagram of nTracer results data structure

the targeted neurite with high labeling intensity (Yizong Cheng, 1995) (Fig. 1a). The end point is defined in a similar way with additional constraints set by the color signature sampled around the start point. The user can therefore avoid setting an end point onto a different neuronal process due to human visual or computer display limitations, in particular with Brainbow images composed of more than three spectral channels (Supplementary Fig. S3). To generate a smooth track along the neuronal process (Fig. 1b), nTracer utilizes the A* algorithm (Hart *et al.*, 1968) to connect the two anchor points with a least-cost path, similar to that implemented in 'Simple Neurite Tracer', which was designed for tracing monochromic images (Longair *et al.*, 2011). nTracer defines the A* cost at voxel i as a weighted sum of the normalized spectral and intensity difference between the start point p and voxel point i, which can be formulated as:

$$G_i = \alpha \times \sqrt[2]{\Sigma_n \left(\frac{I_{p,n}}{\bar{I}_p} - \frac{I_{i,n}}{\bar{I}_p}\right)^2} + \beta \times \left|\frac{\bar{I}_p - \bar{I}_l}{\bar{I}_p + \bar{I}_l}\right|, \quad \text{with } \alpha + \beta = 1, \qquad (2)$$

where n denotes the nth spectral channel. $(\bar{I}_p)$ and $(\bar{I}_i)$ are the total intensities in all spectral channels at start point p and voxel point i, respectively. By constraining the pathfinding range to the voxels enclosing the two anchor points and by choosing optimized heuristic values calculated based on windowing-smoothed voxels, nTracer creates an optimal minimal path almost instantaneously, while variance thresholds ensure that any path containing large intensity or color gaps will be rejected. The tracing speed largely depends on the complexity of the images, which arises from labeling density and/or neuronal morphology. Supplementary Figure S4

shows maximum projections from image stacks with different labeling densities that were each traced by five users of varying nTracer experience (from beginner to expert). The rate of tracing from the sparse image stack (Supplementary Fig. S4a) was 7.6 ± 0.78 mm/h (mean ± SE) whereas the tracing rate was 4.4 ± 0.47 mm/h in the dense sample (Supplementary Fig. S4b). Conversely, we found the error rate between users to be dependent on image resolution and less on complexity. The pixel dimensions in Supplementary Figure S4b ($150 \times 150 \times 300$ nm^3) were half that of those in Supplementary Figure S4a and produced inter-user errors at 0.18 and 0.25, respectively, using the DIADEM metric.

3 Analysis

3.1 Data structure

nTracer utilizes the generic JTree structure of JAVA to allow flexible storage and modification of tracing points of multiple neurons in the computer memory. Three JTrees are built for each traced cell to store the tracing points of the somas, processes and spines independently (Fig. 1d). The soma tree contains parallel nodes, each of which stores soma tracing points on a Z plane. The process tree contains parallel nodes, each of which is a bifurcated branching tree that stores connected branches of an axon, or a dendrite. The soma contour or neurite branch is composed of connected tracing points, each of which is a seven-element data array containing the type of the tracing point (Soma, Dendrite, Axon, Spine etc.), x, y, z coordinates, radius at the point (0 for a soma point or for where the process radius is not determined), whether or not a synapse and its connection status. Spines can also be traced off from a dendrite or soma point (has a type of Spine) and stored as parallel non-branching nodes in the third tree-structure database. Each spine tracing point is a six-element data array that stores the type (Spine), x, y, z coordinates, radius at the point and its locale information (soma or dendrite name).

3.2 Visualization

The tracing results (including connectivity information), raw image information and nTracer setting parameters can be saved in files of custom format and exported as line art image stacks for volume rendering (Fig. 1c; Supplementary Fig. S5 and Supplementary Videos). These can be used to perform analyses of putative synaptic connections (Supplementary Fig. S6) and whole populations of neuron subtypes (Supplementary Figs S7 and S8).

3.3 Quantification

Tracing results of each neuron can also be exported as separate files in standard SWC format (Cannon *et al.*, 1998) for morphology analysis and rendering with other software, such as L-measure (Scorcioni *et al.*, 2008). These can be used to perform morphometric analyses of many neurons from single densely labeled samples.

References

Cai,D. *et al.* (2013) Improved tools for the Brainbow toolbox. *Nat. Methods*, **10**, 540–547.

Cannon,R.C. *et al.* (1998) An on-line archive of reconstructed hippocampal neurons. *J. Neurosci. Methods*, **84**, 49–54.

Chen,F. *et al.* (2015) Optical imaging. Expansion microscopy. *Science*, **347**, 543–548.

Chothani,P. *et al.* (2011) Automated tracing of neurites from light microscopy stacks of images. *Neuroinformatics*, **9**, 263–278.

Chozinski,T.J. *et al.* (2016) Expansion microscopy with conventional antibodies and fluorescent proteins. *Nat. Methods*, **13**, 485–488.

Hart,P. *et al.* (1968) A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Syst. Sci. Cybern.*, **4**, 100–107.

Liu,Y. (2011) The DIADEM and beyond. *Neuroinformatics*, **9**, 99–102.

Livet,J. *et al.* (2007) Transgenic strategies for combinatorial expression of fluorescent proteins in the nervous system. *Nature*, **450**, 56–62.

Longair,M.H. *et al.* (2011) Simple Neurite Tracer: open source software for reconstruction, visualization and analysis of neuronal processes. *Bioinformatics*, **27**, 2453–2454.

Parekh,R. and Ascoli,G.A. (2013) Neuronal morphology goes digital: a research hub for cellular and system neuroscience. *Neuron*, **77**, 1017–1038.

Peng,H. *et al.* (2011) Proof-editing is the bottleneck of 3D neuron reconstruction: the problem and solutions. *Neuroinformatics*, **9**, 103–105.

Schindelin,J. *et al.* (2015) The ImageJ Ecosystem: an open platform for biomedical image analysis. *Mol. Reprod. Dev.*, **82**, 518–529.

Scorcioni,R. *et al.* (2008) L-Measure: a web-accessible tool for the analysis, comparison and search of digital reconstructions of neuronal morphologies. *Nat. Protoc.*, **3**, 866–876.

Tillberg,P.W. *et al.* (2016) Protein-retention expansion microscopy of cells and tissues labeled using standard fluorescent proteins and antibodies. *Nat. Biotechnol.*, **34**, 987–992.

Yizong Cheng,Y. (1995) Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, **17**, 790–799.