SNT: A Unifying Toolbox for Quantification of Neuronal Anatomy

Cameron Arshadi[1], Mark Eddison[1], Ulrik Günther[2,3,4], Kyle I. S. Harrington[1,5], Tiago A. Ferreira[1]

[1]Janelia Research Campus, Howard Hughes Medical Institute, Ashburn, Virginia, USA
[2]CASUS – Center for Advanced Systems Understanding, Görlitz, Germany
[3]Max Planck Institute of Molecular Cell Biology and Genetics, Dresden, Germany
[4]Center for Systems Biology, Dresden, Germany
[5] Virtual Technology & Design, University of Idaho, Moscow, Idaho, USA

Summary

Quantification of neuronal morphology is essential for understanding neuronal connectivity and many software tools have been developed for neuronal reconstruction and morphometry. However, such tools remain domain-specific, tethered to specific imaging modalities, and were not designed to accommodate the rich metadata generated by recent whole-brain cellular connectomics. To address these limitations, we created *SNT*: a unifying framework for neuronal morphometry and analysis of single-cell connectomics for the widely used Fiji and ImageJ platforms.

We demonstrate that SNT —that replaces the popular *Simple Neurite Tracer* software— can be used to tackle important problems in contemporary neuroscience, validate its utility, and illustrate how it establishes an end-to-end platform for tracing, proof-editing, visualization, quantification, and modeling of neuroanatomy.

With an open and scriptable architecture, a large user base, and thorough community-based documentation, SNT is an accessible and scalable resource for the broad neuroscience community that synergizes well with existing software.

Quantification of neuronal anatomy is essential for mapping information flow in the brain and classification of cell types in the central nervous system. Although digital reconstruction ("tracing") of the tree-like structures of neurons —axons and dendrites— remains a laborious task, recent improvements in labeling and imaging techniques allow faster and more efficient reconstructions, with neuroscientists sharing more than 130,000 reconstructed cells across several databases. Powerful toolboxes have been developed for neuronal morphometry (**Sup. Information**). However, such tools remain tethered to specific imaging modalities, computing platforms or workflows, and were not designed to accommodate the rich metadata generated by recent whole-brain connectomic efforts. To address these limitations we established a unifying framework for neuronal morphometry and analysis of single-cell connectomics for the widely used Fiji and ImageJ platforms[1,2].

We re-wrote and extended the popular Simple Neurite Tracer program[3] to create an open-source, end-to-end solution for semi-automated tracing, visualization, quantitative analyses and modeling of neuronal morphology. All aspects of our software, named *SNT* —Simple Neurite Tracer's popular moniker— can be controlled from a user-friendly interface or programmatically, using a vast repertoire of scripting languages (**Fig. 1a**, **Sup. Information**).

For semi-automated tracing we implemented a host of new features, including support for multidimensional images, optimized search algorithms and image processing routines that better detect neuronal processes. To expedite the proof-editing of traced structures, SNT allow users to edit, tag, sort, filter, and rank traced segments, using either ad-hoc labels or morphometric traits. Altogether, these features improve tracing efficiency and reconstruction accuracy (**Sup. Information**).

For visualization, SNT features an interactive 3D viewer dedicated to neuron morphology —*Reconstruction Viewer*— that is hardware accelerated, supports rendering of meshes and detailed annotation of morphometry data. In addition, SNT also integrates with sciview[4], a visualization tool for mesh-based data and arbitrarily large image volumes, supporting virtual, augmented, and mixed reality (**Sup. Information**).

For data retrieval, SNT provides seamless integration with the ImageJ platform, and thus tracing and reconstruction analysis can be intermingled with image processing workflows required to handle multi-dimensional and multi-modal imaging. To exemplify this, we used SNT's scripting abilities to quantify fluorescent in situ hybridization (FISH) imaging of mRNA in

the same volume in which dendrites of pyramidal cells were reconstructed (**Fig. 1b**, **Sup. Information**). Another strength of SNT is that it can connect directly to all the major neuroanatomy databases, including FlyCircuit, NeuroMorpho, MouseLight, and Virtual Fly Brain, supporting several multi-species brain atlases (Drosophila, mouse, zebrafish, **Fig. 1**, **Sup. Information**). While such online databases are highly queryable, they remain constrained by website design limitations. Scripting frameworks that can programmatically parse their data bypass those restrictions, facilitate data sharing, scientific reproducibility, bridge isolated data repositories, and promote the development of new tools and features. Importantly, since SNT adopts the SciJava framework[5,6], it can be scripted using popular computer languages such as Python (and Jupyter notebooks) through pyimagej, Clojure, Groovy, JavaScript, Jython, MATLAB™, R, Ruby, or Scala. To demonstrate the analytical power of SNT, we parsed the MouseLight database, currently containing the most complex reconstructions described in the literature[7]. In particular, we focused on quantifying the repertoire of strategies adopted by individual cells to broadcast information across the brain. In a programmatic, unbiased approach, we retrieved the number of anatomical brain areas innervated by individual axons (**Fig. 1c**). In doing so, we identified two extremes of connectivity: cells that connect exclusively to a single projection brain area, and cells that project broadly over a multitude of brain areas.

SNT provides support for generative models of artificial neurons by utilizing the neurodevelopmental simulation framework, Cx3D[8]. This not only provides capabilities for the algorithmic generation of neuronal morphologies, but enables a new direction of image-based modeling for cellular neuroscience. On the latter, we provide a proof-of-concept example, where artificial neurons are seeded in an image derived from an *in vitro* chemotaxis assay (**Sup. Video 2**). On the former, we challenged SNT's ability to morphometrically distinguish closely-related reconstructions. First, we used gene-regulatory network (GRN) algorithms to define—*in silico*—unique patterns of neuronal growth. Second, we generated thousands of artificial neurons constrained by these patterns. Third, we used built-in metrics[9,10] to statistically differentiate between these computer-generated "neuronal types". We found we could distinguish with high confidence all of the morphological classes, including those closely related (**Sup. Information**). Altogether, these experiments demonstrate how SNT can bridge experimental and modeled data to support model evaluation for both inference and predictive modeling.

In summary, SNT is a powerful tool for tracing, proof-editing, visualization, quantification, and modeling of neuroanatomy. It is based on recent technologies, supports modern microscopy data, integrates well with the ImageJ platform, interacts with major online repositories, and synergizes with post-reconstruction analysis software, and recent data-mining frameworks[11,12]. With a large user base and thorough community-based documentation (https://imagej.net/SNT), SNT is an accessible, scalable and standardized framework for efficient quantification of neuronal morphology.

Acknowledgments

We are extremely thankful to Jayaram Chandrashekar, Albert Cardona and Pavel Tomancak for valuable input. We thank the community of users and contributors of Simple Neurite Tracer, the developers of Scijava, pyimagej, and remaining open source libraries required by SNT, and everyone who helped test the software. W. Rasband, C. Rueden and the ImageJ community for developing and maintaining ImageJ, and Marton Rozsa and Judith Baka for critical reading of the manuscript. Special thanks to all the labs, teams, institutions and initiatives that facilitate public sharing of neuronal data, including 3D InsectBrain, BigNeuron (Allen Institute for Brain Science), Blue Brain, FishAtlas (MPG), FlyCircuit, FlyLight (JRC), MouseLight, NeuroMorpho, OpenWorm, and Virtual Fly Brain.

This work was funded by the Howard Hughes Medical Institute. U.G. was partially funded by the Center of Advanced Systems Understanding (CASUS) which is financed by Germany's Federal Ministry of Education and Research (BMBF) and by the Saxon Ministry for Science, Culture and Tourism (SMWK) with tax funds on the basis of the budget approved by the Saxon State Parliament.

Competing Interests

The authors declare that they have no competing financial interests.

Author contributions

T.A.F. conceived and supervised the project. T.A.F, C.A. wrote core SNT; K.I.S.H, U.G. implemented Cx3D/sciview integration. K.I.S.H, designed and ran simulations. M.E. performed ExFISH experiment. T.A.F. analyzed the data. T.A.F., K.I.S.H., wrote the paper.

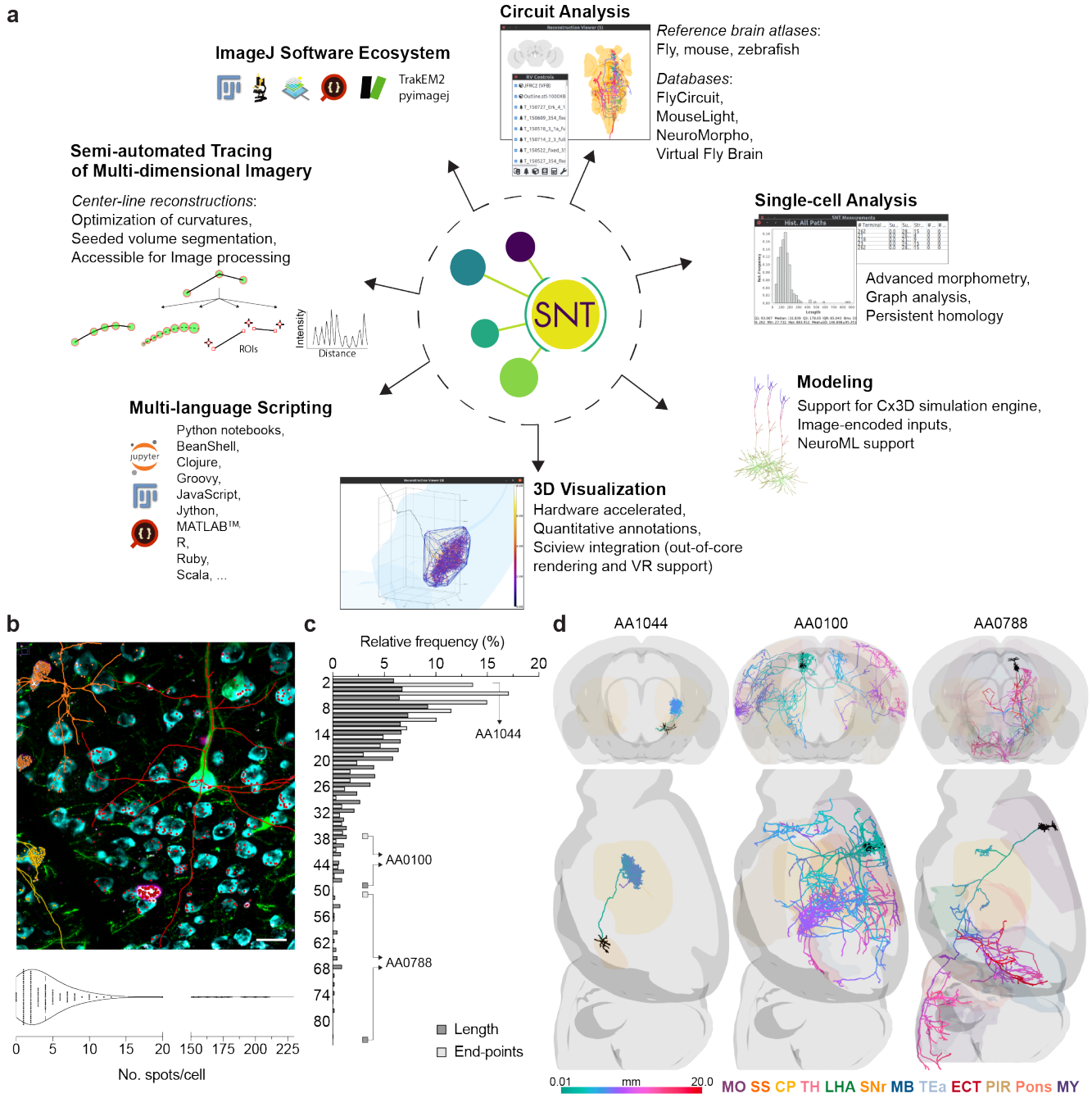


Figure 1 | SNT as an end-to-end platform for data retrieval, visualization, quantification, and modeling of neuroanatomical data.

(**a**) **Overview of the software**. Schematic diagram of major features (counterclockwise): 1) SNT is powered by the rich stack of ImageJ-based software, including: Fiji, ImageJ2, sciview, SciJava, ImgLib2, TrakEM2 and pyimagej. 2) Simple Neurite Tracer original algorithms have been extended to multi-dimensional light-microscopy imagery and paired to several utilities that expedite neuronal reconstruction. 3) All aspects of SNT's functionality are scriptable. 4) Dedicated viewers allow for effective visualization of complex data. 5–7) SNT's bundle of analytical tools allows for morphometry of single-cells, biophysical modeling of neuronal growth, and circuit analysis. Several online databases and reference brain atlases (Drosophila, mouse and zebrafish) are supported.

(**b**) **ImageJ interoperability allows for multi-modal data retrieval.** Top: Maximum intensity projection of a three-channel 3D image of a gelled-brain section processed for ExFISH. Dendrites of GFP-labeled neurons (green) were traced in SNT (center-lines for three cells displayed in orange, red, and yellow). Foci reporting on Somatostatin (SST) mRNA (magenta) were detected on neighboring somata, segmented from a counterstain for total RNA (cyan). Point ROIs reporting foci (circles) were labeled with the same hue of the closest traced cell. All procedures performed within ImageJ. Scale bar: 20µm. Bottom: Violin plot of SST expression for segmented cells in the sub-volume (N=147).

(**c,d**), **Comprehensive analytical tools enable discovery biology.** How many brain areas does a neuron connect to in the mouse brain? The MouseLight (ML) database was programmatically parsed and the number of mid-ontology brain regions innervated by individual axons retrieved under two criteria: (normalized) cable length, and number of end-points at target area. (**c**) Frequency histogram of number of brain areas innervated by single cells (N=1094). (**d**) Ranked examples of axonal connectivity. Left: SNr neuron projecting exclusively to the CP (ML id: AA1044). Its single axonal tuft can cover as much as 4% of the ipsilateral lobe of the target area. Center: SS neuron projecting to many areas in the isocortex (ML id: AA0100). Right: Pyramidal-tract neuron neuron projecting to many areas in the isocortex, midbrain and hindbrain (ML id: AA0788). Dendrites are depicted in red. Neuronal processes are color-coded by "path distance to soma", as per color ramp legend. As a reference, the total axonal length of the cells is: AA1044: 28.598cm, AA0100: 44.649cm, AA0788: 18.454cm. Abbreviation of brain regions: MO: Somatomotor areas; SS: Somatosensory areas; CP: Caudoputamen; TH: Thalamus; LHA: Lateral hypothalamic area; SNr: Substantia nigra, reticular part; MB: Midbrain; TEa: Temporal association areas; ECT: Ectorhinal area; PIR: Piriform area; MY: Medulla.

See **Sup. Information** for details.

Supplementary Information

Overview

SNT is an open-source (GPLv3) program written in Java and distributed with the Fiji[1] distribution of ImageJ[2]. The source code is public and managed by git (https://github.com/morphonets/SNT). SNT stems from the rewrite of Simple Neurite Tracer[3]'s basecode following Scijava[5] design principles and with scripting abilities inspired by several powerful open-source software, namely: L-measure[9], nat[11], TREES[13], btmorph[14], and NeuroM[15]. A special effort was put into backwards compatibility, so that SNT could supersede Simple Neurite Tracer. Wiki-based user documentation is hosted at https://imagej.net/SNT.

SNT can run in headless environments, is fully scriptable and itself extensible. It inter-operates with other core components of the ImageJ ecosystem namely *Sholl Analysis*[10], TrakEM2[16], IJ ops[17], and scenery[4]. SNT functionality can be extended using ImageJ2 commands[6] or executable scripts, allowing both experienced developers and scientists with beginner-level programming experience to customize SNT. Several scripting templates are provided in Fiji's Script Editor and a built-in discovery mechanism automatically registers user scripts in SNT's user interface, as detailed in https://imagej.net/SNT/Scripting. Native Python is supported through pyimagej[18].

Requirements

An up-to-date Fiji installation running Java 8 or newer. Access to "Tubular Geodesics"[19] segmentation requires installation of external binaries, as described in the documentation. Discrete graphics card is recommended for sciview integration. VR support in sciview requires the OpenVR/SteamVR library[4].

Installation

SNT is released through a dedicated "Neuroanatomy" update site, created to streamline and foster contributions from the wider scientific comunity. Sciview has not been officially released and access to its functionality currently requires subscription to a second update site. Detailed instructions are available at https://imagej.net/SNT#Installation.

Supported File Types

Images: SNT accepts any non-RGB image recognized by ImageJ, SCIFIO[20] or Bioformats[21] with up to five axes, including multi-channel and time-lapse sequences. **Neuronal reconstructions**: SNT recognizes all known variants of SWC[22,23], the *de facto* standard for data sharing, MouseLight's JSON[7] and S*imple Neurite Tracer*'s TRACES open formats. **3D graphics**: Wavefront OBJ (reconstruction viewer) and STL, PLY, XYZ (sciview). **Analysis**: Output of SNT analyses can be saved as CSV (tabular data); SVG, PDF, PNG (plots and histograms); MPG (sciview animations); and NeuroML (Cx3D models).

Supported Databases

SNT can download data directly from FlyCircuit[24] (flycircuit.tw), MouseLight[7] (ml-neuronbrowser.janelia.org), NeuroMorpho[25] (neuromorpho.org/), and VirtualFlyBrain[26] (virtualflybrain.org) databases, with ongoing support for the Max Planck Zebrafish Brain Atlas[27] (fishatlas.neuro.mpg.de/). Data can be imported from SNT's user interface or programmatically using its API.

Features

Semi-automated Tracing

The core of SNT's semi-automatic reconstruction remains Simple Neurite Tracer's exploratory approach in which the paths between manually placed points along the centerline of neuronal processes are computed using bidirectional A* search[3]. However, several improvements were made to this procedure, namely: 1) scriptable tracing (**Fig. S1**); 2) refinement of center-lines positioning by post-hoc fitting procedures that take into account the fluorescent signal around each traced node

(**Fig. S2**); 3) Support for multi-channel and timelapse images (**Fig. S3**); 4) Detection of signal within a local 3D neighborhood around the cursor; 5) improved synchronization mechanism of Simple Neurite Tracer's original XY/ZY/XZ tracing views that facilitate accurate node positioning; and 6) computation of curvatures on pre-processed images. The latter allows A* to be computed on mirrored data (called a "secondary image") in which the tube-like structures of neuronal processes have been pre-enhanced using image processing routines. For added convenience, SNT offers filtering pre-sets[28,29] through IJ-ops[17], and allows pre-filtered data to be imported from third-party software, to e.g., allow for processing routines not yet ported into ImageJ. Adoption of other path search algorithms such as *Tubular Geodesics*[19] is also possible through the installation of external binaries. Importantly, these features can be toggled at will during a tracing session.

In SNT, the accuracy and performance of the automated path search can be tuned using Hessian-based analysis of curvatures, optimized to detect tubular structures of a particular size. A key parameter of this filtering operation is σ, the size of the Gaussian kernel used to smooth the image before detection of tube-like structures occurs[3]. In order to provide users with sensible defaults, we ran a series of simulations to assess the impact of σ on reconstruction accuracy. For this purpose, we chose images of Drosophila olfactory axons from the DIADEM challenge[30], because these images share common acquisition parameters and their respective manual reconstructions (the "gold standards") have been well characterized in benchmark studies. The topologies of these axons can be loosely divided into two complexity groups according to their number of branches (**Fig. S1a**). Since A* search can be scripted in SNT, these experiments were fully automated: For each cell, we iterated through the branches in the gold standard reconstruction and performed A* search between the voxels associated with the coordinates of the first and last node of each branch. This procedure was repeated while varying σ, with traced structures compared to the gold standard using the DIADEM metric[31] at each run.

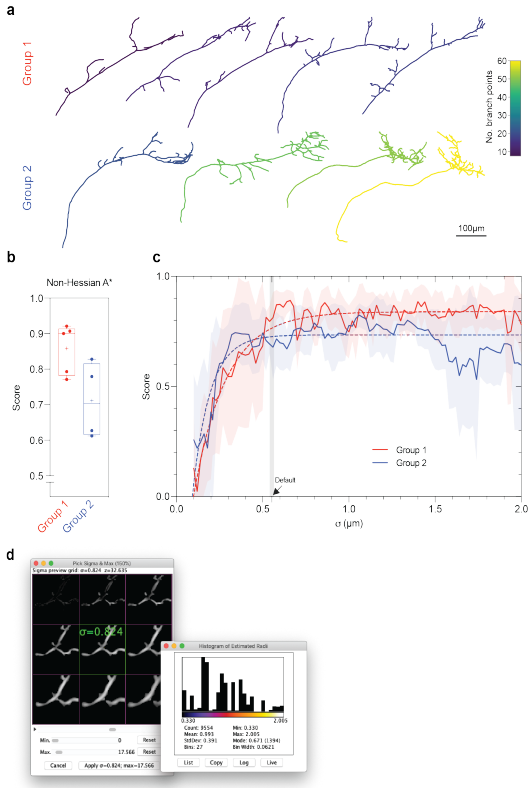


Figure S1 | Characterization of SNT's A* tracing in fully automated tests.

(a) 2D renderings of axonal arbors of Drosophila olfactory projection neurons (DIADEM Challenge "OP" dataset) used in the tests, generated using SNT's *Reconstruction Plotter* command. Cells are color-coded by number of branch points according to hue ramp. The 10 cells can be split into two complexity groups: ≤20 branches (group 1) and >20 branches (group 2).

(b) Accuracy of automated A* without complementary Hessian analysis**.** Each OP cell was traced in the absence of pre-processing routines (the default in SNT) and resulting reconstruction compared to each cell's "gold standard" using the DIADEM similarity score, in which 1 represents perfect similarity.

(c) Effect of σ on accuracy of Hessian-based A* search. For each cell, A* search was performed on Hessian- filtered data under varying σ values. As in b), resulting reconstructions were then compared to each cell's "gold standard". The default value that is proposed to users in SNT's interface is indicated (arrow). Dashed lines indicate 'best-fit' curves (one phase exponential).

(d) SNT features several GUI-based tools that allow users to tune parameters during a tracing session. Here shown two controls for σ adjustment. Left: The "Hessian widget", allowing interactive adjustments of σ at chosen image locations. Right: histogram of predicted radii, computed for the whole image volume being traced using local thickness analysis (depicted data for the OP_1 cell, the second most complex cell of group 2).

First, we found that in the absence of human input, the simpler cells could be traced with high accuracy in the absence of Hessian pre-processing (**Fig. S1b**, median: .90/1.00 similarity score). Second, we found that under the stringent limitations of the test, Hessian pre-processing can enhance the accuracy of the segmentation of the more complex topologies (**Fig. S1c**) but is sensitive to σ choice. It is worth noting that the default σ value —that is proposed to the user at startup upon loading of the image being traced—yielded a median DIADEM score of 0.81 for all cells combined (**Fig. S1c**). Since

reconstructions associated with a score of ≥ 0.8 are considered acceptably similar[31,32], SNT's default settings —that are computed on an image-per-image basis— are reasonably determined. Given that the fine-tuning of Hessian parameters is essential for accurate results, SNT provides users with an interactive widget allowing users to adjust parameters during a tracing session, as well as commands to estimate, *a priori*, the local thickness[33,34] of the structures to be traced (**Fig. S1d**).

Proof-editing of automated segmentation is a major bottleneck in neuron reconstruction[35]. To expedite the proof-editing of traced structures, SNT features an "Edit mode" in which the position of individual center-line nodes can be refined and edited with sub-pixel accuracy. SNT can also execute "path fitting" routines that automatically estimate radii and the optimal position of reconstruction nodes relative to signal (**Fig. S2a**). To expedite its usage, fitting operations are undoable, and can be perused by means of the "Fitting inspector" interface. SNT also allows users to identify and annotate portions of built-up topologies with colors and searchable tags based on custom labels, image data, or computed directly from morphometric traits (**Fig. S2b**).

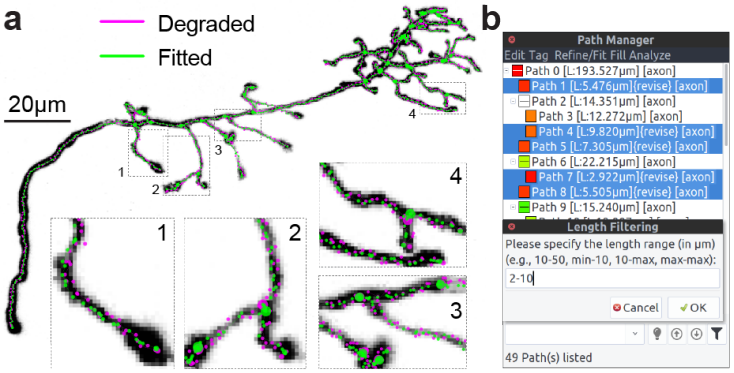


Figure S2 | Post-hoc optimization of curvatures and proof-editing tools.

(a) Refinement of node positioning through automated fitting. The "gold standard" reconstruction of the "OP_1" neuron (depicted as a Z-Projection) was programmatically degraded (magenta). Voxel intensities were used to correct degradation: SNT computed cross-sections along each traced path, 'snapping' nodes to their centroid, as depicted in insets. The same procedure can be used to estimate radii along the traced structure

(b) The Path Manager's search bar provides expedite commands for annotating, filtering, and selecting paths. Such operations can be based on image data, morphometric properties or user-provided tags.

Multidimensional Imaging

A key feature of SNT is seamless integration with ImageJ: Image processing routines can be inter-leaved with tracing tasks, and the entire suite of ImageJ plugins remains accessible during a tracing session. Special emphasis was put into allowing users to access image and reconstructed data in convenient ways. For example, reconstructed paths can be converted into functional ROIs and voxel intensities profiled along their center-lines. With multi-color fluorescence microscopy, this functionality allows users to quantify fluorescent signal along traced structures, and measure the signal from other probes in the imaged tissue (**Fig. 1**). Although such features are of proven utility for many types of data (**Fig. S3**), we anticipate immediate utility in studies focused on neuro-regeration and neuron-glia interactions, in which the cellular environments around neurons is imaged and analyzed.

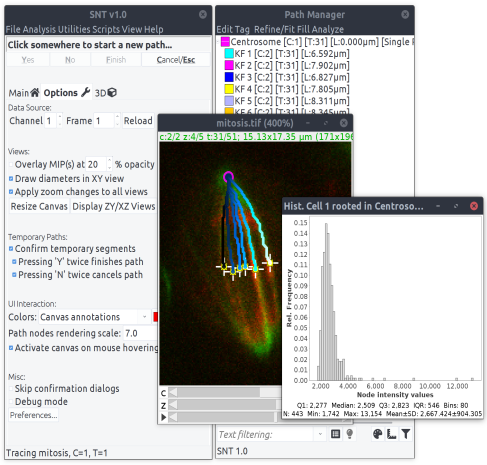


Figure S3 | Analysis of non-neuronal 5D imagery.

K-fibers were traced during anaphase in a multi-channel 3D timelapse of dividing S2 cells[58] (IJ1 "mitosis" 4D sample image). Traces were color-coded uniquely, according to their location along the X- axis. Histogram depicts how the voxel data underlying traced data can be easily accessed from built-in commands. Conversion of tracings into functional ImageJ ROIs is also easily accomplished, as exemplified by point ROIs ('+' markers) highlighting chromosome-attachment sites.

Visualization and Analysis

For simplified visualization of single-cell data SNT implements "Reconstruction Plotter", a 2D canvas for vector-based plotting of reconstructions (**Fig. S1a, S4a**). For rendering more complex 3D data, SNT features two additional viewers: *Reconstruction Viewer* and sciview[4] integration.

Reconstruction Viewer (RV) is hardware accelerated, supporting both surface meshes and reconstruction files (**Fig. S4b**). The viewer can render both local and remote files on the same scene, which allows for direct loading of reconstructions from all of the supported databases and meshes for several template brains, i.e.:Drosophila (larval and adult) via Virtual Fly Brain, FlyLight, and FlyCircuit[24,26,36]; Allen Mouse Common Coordinate Framework[37] (CCF, adult mouse) via MouseLight database[7]; and Zebrafish via the Max Planck Zebrafish Brain Atlas[27]. In the case of the Allen Mouse Brain Atlas[37], the full stack of anatomical ontologies is supported (**Fig. S4b″**). Reconstruction Viewer can also be used as a stand-alone application (i.e., in the absence of SNT's tracing interface), allowing it to be accessed from other environments such as IPython. All Reconstruction Viewer instances can be scripted once displayed, and can be instantiated in "high-performance" mode, suitable for visualization of large amounts of data. As a proof-of-principle we used this feature to visualize the entire MouseLight database on a laptop computer without a discrete graphics card (**Sup. Video 1**). Another key feature of Reconstruction Viewer is its ability to perform geometric analyses on both meshes and reconstructions, which is key for studying the topographic organization of neurons across neuropils (**Fig. S4b**).

Sciview is a powerful SciJava-based visualization tool for volumetric and mesh-based data (**Fig. S4c**). Sciview has been recently improved to support out-of-core volume rendering of images up to 9.261×10^{27} voxels[4] (via BigDataViewer[38]), making it an appealing choice for visualization of imagery from large-scale connectomics projects.

SNT analytical capabilities are three-fold: 1) Analysis of imagery data already discussed; 2) Morphometric analysis of single cells in isolation and 3) Analysis of groups of cells in a common, annotated space (a reference brain/neuropil), which requires handling of neuroanatomical volumes. For morphometric analyses, SNT supports commonly used metrics[9], graph theory based analysis, and popular quantification strategies, such as Sholl[10] and Strahler[39–41]. In addition, SNT's API allows for analyses based on persistent homology[42,43] and other ad-hoc statistical measurements complementary to those available through the user interface (**Fig. S5**).

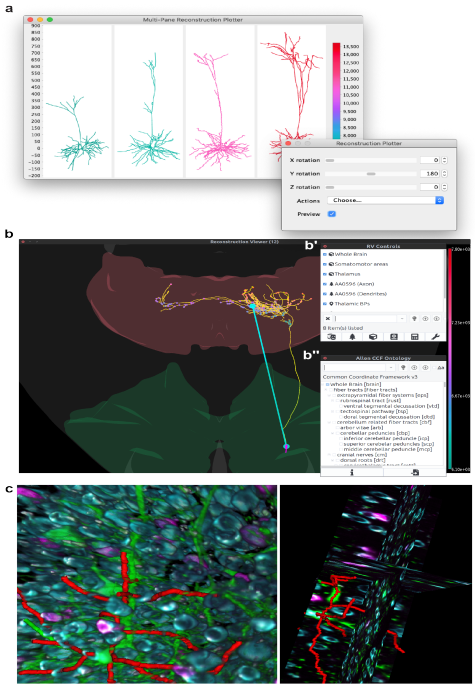


Figure S4 | Neuroanatomy viewers.

(**a**) *Reconstruction Plotter* assembles multi-panel 2D plots of reconstructed cells. Here, four dendritic arbors from the MouseLight database (IDs AA001–4), were automatically aligned, ranked and color-coded by cable length. When called from the GUI (graphical user interface), data can be transformed interactively (inset). Axis and scale bar in µm.

(**b**) *Reconstruction Viewer* (RV) is a 3D visualization tool designed to handle neuroanatomy exclusively. Transverse view of the mouse brain, depicting a MouseLight pyramidal cell (ID AA0596; axon in yellow, dendrites in magenta) in the motor cortex (green). All axonal branch points within the Thalamus (red) have been identified and color-coded by depth, as per hue ramp. The vector connecting the cell soma to the centroid of thalamic branch points is highlighted. RV Controls (b′) are organized in a layout modeled after *Path Manager* (Fig. S2b) in SNT's tracing interface. Extra functionality, such as access to Allen CCF ontologies (b″) is provided through dedicated dialogs.

(**c**) Sciview integration allows SNT data to be rendered with arbitrarily large image volumes. Left: Surface visualization of the multi-channel volume described in **Fig. 1** (reconstructed dendrites in red). Right: Ortho-view of the same data (experimental feature at the time of this writing). For clarity, components of the sciview user interface (scene inspector, interactive scripting shell, Cx3D bridge, etc.) were omitted.

Figure S5 | Overview of analysis API.

Functional scripts (written in Python) showcasing SNT's API (application programming interface) on dendritic arbors from the MouseLight database (cells IDs AA0001–4) that can be access offline from within SNT as "demo trees". For each panel, the script code (as typed in Fiji's script editor) is showed on the left. Script's output on the right.

(**a**,**b**) Statistics of morphometric traits can be extracted from groups of neurons (**a**), a single neuron (**b**), or parts thereof (not shown). Convenience methods allow data to be output in histograms, plots and tables. (**c**) Distribution of cable length across brain areas of Allen CCF. (**d**) Neuronal reconstructions can be converted to graphs succinctly. For convenience, graphs can be displayed in SNT's *Dendrogram Viewer*.

Modeling

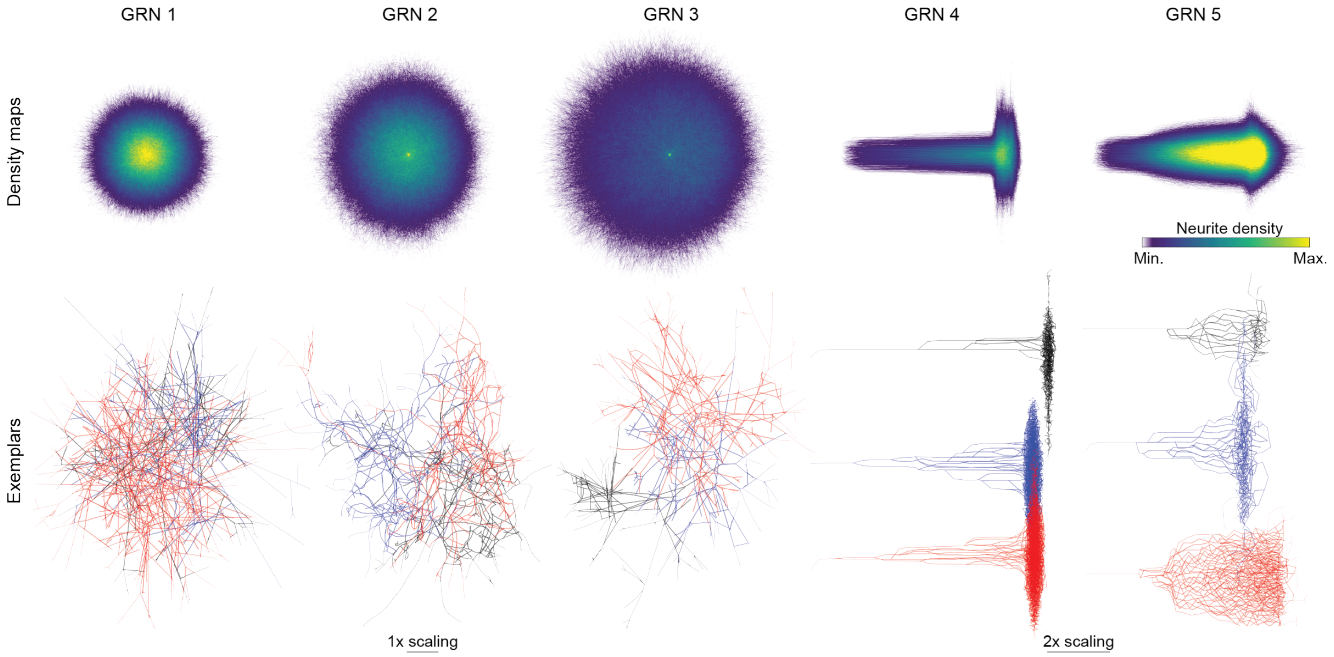
Cortex3D (Cx3D) is a computational modeling tool for simulating neurodevelopmental processes[8] and has been used to define generative models of cortical circuits[44]. We integrate Cx3D with SNT through the sciview visualization package by rewriting Cx3D to grow neuronal processes with sciview's data structures. This facilitates the use of both SNT's and ImageJ's functionality when designing Cx3D models. We present two important examples that benefit from the unification of quantitative neuroanatomy and morphological modeling: statistical discrimination between morphological cell types, and image-based modeling of neural morphologies. However, there are many additional possibilities enabled by this coupling, from generative testing of new quantitative measurements to data-driven model fitting.

Morphological discrimination between neuronal cell types is an important aspect of cell classification. The quantitative measurements made by SNT can be used to statistically test the difference between morphological populations of cells. We demonstrate this approach by randomly generating five unique gene regulatory networks (GRNs)[45] that satisfy minimal viability criteria. A simple test for viability was performed as follows: the GRN is simulated for 100 time-steps with constant input values, with the following criteria assessed at the first and final time-step of the viability test: (1) all behavior regulating protein concentrations should change, (2) the GRN must bifurcate enough to have reasonable growth, (3) the GRN must not over-bifurcate, (4) the GRN must trigger branching, and (5) the GRN must not overbranch. While these criteria will have an impact on the distribution of possible GRNs that are selected, the rationale for this approach is to compensate for the fact that most randomly generated GRNs are not capable of generating physically plausible neurodevelopment.

The neurodevelopment model encodes a minimal artificial environment reminiscent of an *in vitro* assay. A neuron is initialized as a single soma at a randomly selected position within +/- 40 spatial units of the origin (0, 0, 0) along the X-, Y-, and Z- axes. Three extracellular morphogen gradients are established 300 spatial units along the three axes in a Gaussian distribution concentration. The simulation begins by extending an initial neurite segment from the soma. The proteins encoded in the GRN are simulated within each neurite segment where their concentration is proportional to local extracellular morphogen gradients, segment length, segment volume, and branch order within the cell. The GRNs then regulate the concentration of "proteins" controlling the direction of growth, segment bifurcation, and neurite branching. The model proceeds by iteratively simulating physical constraints encoded in the Cx3D simulation engine and the dynamics of the GRN to grow the artificial neuron.

For each GRN, we stochastically generated 1,000 artificial neurons—under the same environmental conditions— and analyzed their skeletons with SNT core metrics. We then performed Principal Component Analysis (PCA) and statistically tested the difference between principal components with two-sample Kolmogorov-Smirnov tests combined with the Fisher combined probability test (**Methods**). We obtained statistical significance on all comparisons. All groups were similarly resolved when extending this approach to other dimensionality reduction techniques (t-SNE[46] and UMAP[47], **Fig. S6**).

By unifying SNT and Cx3D, we can leverage ImageJ within Cx3D to enable support for image-based modeling. We demonstrate a proof-of-concept simulation based upon an in vitro microfluidic assay designed for screening the effect of chemoattractants on neuronal growth[48]. We recreated their microfluidic circuit topology, and define gradients of chemoattractant using an image, analogous to the original experiment's Netrin assay. Neurite outgrowths follow chemical gradients of increasing concentration defined within the microfluidic circuit. We show a simulated neuron and image-based environment in **Sup. Video 2.**



	GRN 1					GRN 2					GRN 3					GRN 4					GRN 5				
PCA	N/A	*	*	*	*	*	N/A	*	*	*	*	*	N/A	*	*	*	*	*	N/A	*	*	*	*	*	N/A
t-SNE	N/A	*	*	*	*	*	N/A	*	*	*	*	*	N/A	*	*	*	*	*	N/A	*	*	*	*	*	N/A
UMAP	N/A	*	*	*	*	*	N/A	*	*	*	*	*	N/A	*	*	*	*	*	N/A	*	*	*	*	*	N/A

Figure S6 | Statistical discrimination between between morphological populations of cells generated *in silico*.

Top: Density maps for the five neurodevelopmental cell types artificially generated using unique GRNs, obtained by overlaying all soma-aligned cells in the group. Color coding reflects the density of neurites at a given location for the 5 groups.

Middle: Representative exemplars of each group. For clarity, cells from GRNs 4 and 5 were separated by a vertical offset.

Bottom: Summary of pairwise statistical measures of morphological differences between the five groups analyzed under different dimensionality reduction techniques. Principal components, t-SNE features and UMAP components were computed from default SNT metrics and compared using a modified two-sample Kolmogorov Smirnov test. *: $p<0.001$; N/A: no test performed between same sample.

Cells from GRNs 4 and 5 were re-scaled in top and middle panel so that all cells could be rendered under equivalent dimensions. N=1000 cells per group.

Supplementary Videos

Video S1 | Example of a programmatic animation: "Cumulative" rendering of the complete MouseLight database in *Reconstruction Viewer*. Animation, was generated on a laptop computer lacking a dedicated GPU, using a single script that downloaded, measured, and rendered each cell. See https://github.com/morphonets/ SNTManuscript for details. Note that the number of cells in the database has meanwhile surpassed those rendered.

Video S2 | Example of image-based modeling using Cx3D and a 3D volumetric image defining a microfluidic circuit designed to assess neurite outgrowth in response to Netrin-1 and and Slit2[48]. The simulation shows a single cell (magenta) with chemotaxis and branching preference for a steady-state chemical gradient (low concentration: purple; high concentration: yellow).

Methods

The figures and analyses from this manuscript can be generated programmatically. See https://github.com/morphonets/ SNTManuscript for details.

Programming

SNT was programmed with Eclipse Java IDE (Eclipse Foundation), IntelliJ IDEA (JetBrains), and Fiji's built-in Script Editor on an Intel i7 laptop running Ubuntu 18.10–20.04.

ExFISH

Histology: Expansion fluorescence *in situ* hybridization (ExFISH) was performed using an optimized protocol not yet published (manuscript in preparation). In short: 150µm-thick cortical slices of an heterozygous Thy-1-GFP-M[49] adult mouse were embedded in a hydrogel using standard embedding procedures for thick tissue[50]. mRNA was detected using HCR 3.0[51]. SST probes and fluorescent hairpins were obtained from Molecular Instruments (molecularinstruments.com). Gelled sections (~2× expanded) were imaged in PBS on a commercial Zeiss Z1 lightsheet microscope.

Spot quantification: Signal from total RNA labeling was segmented using Labkit[52]. Individual cells were then masked using watershed filtering and labeling of connected components using MorphoLibJ[53,54]. Ill-segmented somata were manually eliminated with the aid of BAR tools[55]. Spot density (no. of spots per cell) of SST signal was determined by iteratively running *3DMaximaFinder*[56,57] at locations of each connected component. It should be noted that this approach is rather elementary: it was designed as a proof-of-principle image processing routine that can be performed mid-way through a tracing session using accessible ImageJ tools.

Analysis

MouseLight Single-cell Connectivity

MouseLight (ML) database was programmatically parsed to obtain the number of brain areas (Allen Mouse Common Coordinate Framework (CCF) compartments) associated with individual axonal arbors using two criteria: 1) normalized cable length and 2) number of end-points at target area). Only CCF compartments of ontology depth ≤7 with public meshes available were considered. For 1) axonal length within an anatomical compartment was measured by taking all nodes within the given area and summing the distances to their parent nodes. To exclude enpassant axons, a cell was considered to be associated with the compartment if such length would be at least 5% of the compartment's bounding-box diagonal. For 2) only neurons with at least two end-points were considered. Selected examples in Fig. 1 were chosen by sorting cells by number of associated areas, and selecting those with the largest axonal cable length (AA1044: 28.598cm, AA0100: 44.649cm, AA0788: 18.454cm).

Tracing and Path Fitting Benchmarks

Tracing benchmarks and fitting procedures were performed programmatically. For degradation of traces (Fig. S2), each node in the reconstruction was displaced to a random position within a 1µm neighborhood around each axis.

Synthetic Morphologies

Chemoatraction assay (Video S2): Code accessible from github.com/morphonets/SNTManuscript.

GRNs (Fig. S6): The code for generating GRNs is available at github.com/morphonets/cx3d/, and the five GRNs used in this study are made available at github.com/morphonets/SNTManuscript (together with remaining analysis scripts). Tools

for inspecting GRNs are available at github.com/brevis-us/grneat. Morphometric analysis: Default metrics provided by SNT (41 as of this writing[1]) were retrieved for all artificial neurons. Data was normalized and analyzed using PCA (*Principal Component Analysis*), t-SNE (*t-Stochastic Neighbor Embedding*), and UMAP (*Uniform Manifold Approximation and Projection*). Group comparisons on principal components, t-SNE features, and UMAP components were performed using two-sample Kolmogorov-Smirnov tests adapted for multivariate data. p–values were combined using Fisher's combined probability test. Density maps and examplars: soma-aligned cells were skeletonized, and their skeletons projected into the XY plane using SNT's core functionality. Binary masks of skeletons were then summed up and resulting image normalized to the number of cells. Exemplars were chosen from a random pool of 10 cells.

1 A subset of all available metrics: *Average branch length, Average contraction, Average fractal dimension, Average fragmentation, Average partition asymmetry, Average remote bif. angle, Cable length, Depth, Height, Highest path order, Horton-Strahler bifurcation ratio, Horton-Strahler number, Length of inner branches (sum), Length of primary branches (sum), Length of terminal branches (sum), Mean radius, No. of branch points, No. of branches, No. of inner branches, No. of nodes, No. of primary branches, No. of terminal branches, No. of tips, Width*; [Sholl-based metrics]: *Centroid, Centroid radius, Decay, Degree of polynomial fit, Enclosing radius, Intercept, Kurtosis, Max, Max (fitted), Max (fitted) radius, Mean, Median, No. maxima, No. secondary maxima, Skeweness, Sum, Variance.* Refer to user documentation for details.

References

1. Schindelin, J. *et al.* Fiji: an open-source platform for biological-image analysis. *Nat. Methods* **9**, 676–682 (2012).
2. Schneider, C. A., Rasband, W. S. & Eliceiri, K. W. NIH Image to ImageJ: 25 years of image analysis. *Nat. Methods* **9**, 671–675 (2012).
3. Longair, M. H., Baker, D. A. & Armstrong, J. D. Simple Neurite Tracer: open source software for reconstruction, visualization and analysis of neuronal processes. *Bioinformatics* **27**, 2453–2454 (2011).
4. Günther, U. *et al.* scenery: Flexible Virtual Reality Visualization on the Java VM. *ArXiv190606726 Cs* (2019).
5. Rueden, C., Schindelin, J., Hiner, M. C. & Eliceiri, K. *SciJava Common [Software]. http://scijava.org/*.
6. Rueden, C. T. *et al.* ImageJ2: ImageJ for the next generation of scientific image data. *BMC Bioinformatics* **18**, 529 (2017).
7. Winnubst, J. *et al.* Reconstruction of 1,000 Projection Neurons Reveals New Cell Types and Organization of Long-Range Connectivity in the Mouse Brain. *Cell* **179**, 268-281.e13 (2019).
8. Zubler, F. & Douglas, R. A framework for modeling the growth and development of neurons and networks. *Front. Comput. Neurosci.* **3**, 25 (2009).
9. Scorcioni, R., Polavaram, S. & Ascoli, G. A. L-Measure: a web-accessible tool for the analysis, comparison and search of digital reconstructions of neuronal morphologies. *Nat. Protoc.* **3**, 866–876 (2008).
10. Ferreira, T. A. *et al.* Neuronal morphometry directly from bitmap images. *Nat. Methods* **11**, 982–984 (2014).
11. Bates, A. S. *et al.* The natverse, a versatile toolbox for combining and analysing neuroanatomical data. *eLife* **9**, e53350 (2020).
12. Claudi, F., Tyson, A. L. & Branco, T. Brainrender. A python based software for visualisation of neuroanatomical and morphological data. *bioRxiv* 2020.02.23.961748 (2020) doi:10/ggvd3b.
13. Cuntz, H., Forstner, F., Borst, A. & Häusser, M. One Rule to Grow Them All: A General Theory of Neuronal Branching and Its Practical Application. *PLOS Comput. Biol.* **6**, e1000877 (2010).
14. Torben-Nielsen, B. An Efficient and Extendable Python Library to Analyze Neuronal Morphologies. *Neuroinformatics* **12**, 619–622 (2014).
15. Juan Palacios *et al. BlueBrain/NeuroM*. (Zenodo, 2020). doi:10.5281/zenodo.597333.
16. Cardona, A. *et al.* TrakEM2 Software for Neural Circuit Reconstruction. *PLOS ONE* **7**, e38011 (2012).
17. Rueden, C. *et al. ImageJ Ops [Software]. http://imagej.net/Ops.* (2016).
18. *pyimagej [Software]. https://github.com/imagej/pyimagej*.
19. Türetken, E., Benmansour, F. & Fua, P. Automated reconstruction of tree structures using path classifiers and Mixed Integer Programming. in *2012 IEEE Conference on Computer Vision and Pattern Recognition* 566–573 (2012). doi:10.1109/CVPR.2012.6247722.
20. Hiner, M. C., Rueden, C. T. & Eliceiri, K. W. SCIFIO: an extensible framework to support scientific image formats. *BMC Bioinformatics* **17**, 521 (2016).
21. Linkert, M. *et al.* Metadata matters: access to image data in the real world. *J. Cell Biol.* **189**, 777–782 (2010).
22. Cannon, R. C., Turner, D. A., Pyapali, G. K. & Wheal, H. V. An on-line archive of reconstructed hippocampal neurons. *J. Neurosci. Methods* **84**, 49–54 (1998).
23. Nanda, S. *et al.* Design and implementation of multi-signal and time-varying neural reconstructions. *Sci. Data* **5**, 170207 (2018).
24. Chiang, A.-S. *et al.* Three-dimensional reconstruction of brain-wide wiring networks in Drosophila at single-cell resolution. *Curr. Biol. CB* **21**, 1–11 (2011).
25. Ascoli, G. A., Donohue, D. E. & Halavi, M. NeuroMorpho.Org: A Central Resource for Neuronal Morphologies. *J. Neurosci.* **27**, 9247–9251 (2007).
26. Milyaev, N. *et al.* The Virtual Fly Brain browser and query interface. *Bioinforma. Oxf. Engl.* **28**, 411–415 (2012).
27. Kunst, M. *et al.* A Cellular-Resolution Atlas of the Larval Zebrafish Brain. *Neuron* (2019) doi:10.1016/j.neuron.2019.04.034.
28. Frangi, A. F., Niessen, W. J., Vincken, K. L. & Viergever, M. A. Multiscale vessel enhancement filtering. in *Medical Image Computing and Computer-Assisted Intervention — MICCAI'98* (eds. Wells, W. M., Colchester, A. & Delp, S.) vol. 1496 130–137 (Springer Berlin Heidelberg, 1998).
29. *Tubeness [Software]. https://imagej.net/Tubeness*.
30. Brown, K. M. *et al.* The DIADEM Data Sets: Representative Light Microscopy Images of Neuronal Morphology to Advance Automation of Digital Reconstructions. *Neuroinformatics* **9**, 143–157 (2011).
31. Gillette, T. A., Brown, K. M. & Ascoli, G. A. The DIADEM Metric: Comparing Multiple Reconstructions of the Same Neuron. *Neuroinformatics* **9**, 233–245 (2011).

32. Usher, W. *et al.* A Virtual Reality Visualization Tool for Neuron Tracing. *IEEE Trans. Vis. Comput. Graph.* **24**, 994–1003 (2018).
33. Hildebrand, T. & Rüegsegger, P. A new method for the model-independent assessment of thickness in three-dimensional images. *J. Microsc.* **185**, 67–75 (1997).
34. *Local Thickness [Software]. https://imagej.net/Local_Thickness*.
35. Peng, H., Long, F., Zhao, T. & Myers, E. Proof-editing is the Bottleneck Of 3D Neuron Reconstruction: The Problem and Solutions. *Neuroinformatics* **9**, 103–105 (2011).
36. Jenett, A. *et al.* A GAL4-Driver Line Resource for Drosophila Neurobiology. *Cell Rep.* **2**, 991–1001 (2012).
37. Lein, E. S. *et al.* Genome-wide atlas of gene expression in the adult mouse brain. *Nature* **445**, 168–176 (2007).
38. Pietzsch, T., Saalfeld, S., Preibisch, S. & Tomancak, P. BigDataViewer: visualization and processing for large image data sets. *Nat. Methods* **12**, 481–483 (2015).
39. Horton, R. E. Erosional Development Of Streams And Their Drainage Basins: Hydrophysical Approach To Quantitative Morphology. *GSA Bull.* **56**, 275–370 (1945).
40. Strahler, A. N. Quantitative analysis of watershed geomorphology. *Trans. Am. Geophys. Union* **38**, 913 (1957).
41. Vormberg, A., Effenberger, F., Muellerleile, J. & Cuntz, H. Universal features of dendrites through centripetal branch ordering. *PLOS Comput. Biol.* **13**, e1005615 (2017).
42. Li, Y., Wang, D., Ascoli, G. A., Mitra, P. & Wang, Y. Metrics for comparing neuronal tree shapes based on persistent homology. *PLOS ONE* **12**, e0182184 (2017).
43. Kanari, L. *et al.* A Topological Representation of Branching Neuronal Morphologies. *Neuroinformatics* **16**, 3–13 (2018).
44. Zubler, F. *et al.* Simulating Cortical Development as a Self Constructing Process: A Novel Multi-Scale Approach Combining Molecular and Physical Aspects. *PLOS Comput. Biol.* **9**, 1–15 (2013).
45. Cussat-Blanc, S., Harrington, K. & Pollack, J. Gene Regulatory Network Evolution Through Augmenting Topologies. *IEEE Trans. Evol. Comput.* **19**, 823–837 (2015).
46. Maaten, L. van der & Hinton, G. Visualizing Data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).
47. McInnes, L., Healy, J. & Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *ArXiv180203426 Cs Stat* (2018).
48. Kothapalli, C. R. *et al.* A high-throughput microfluidic assay to study neurite response to growth factor gradients. *Lab. Chip* **11**, 497–507 (2011).
49. Feng, G. *et al.* Imaging neuronal subsets in transgenic mice expressing multiple spectral variants of GFP. *Neuron* **28**, 41–51 (2000).
50. Asano, S. M. *et al.* Expansion Microscopy: Protocols for Imaging Proteins and RNA in Cells and Tissues. *Curr. Protoc. Cell Biol.* **80**, e56 (2018).
51. Choi, H. M. T. *et al.* Third-generation in situ hybridization chain reaction: multiplexed, quantitative, sensitive, versatile, robust. (2018) doi:10.1101/285213.
52. *Labkit [Software]. https://imagej.net/Labkit*.
53. Legland, D., Arganda-Carreras, I. & Andrey, P. MorphoLibJ: integrated library and plugins for mathematical morphology with ImageJ. *Bioinformatics* **32**, 3532–3534 (2016).
54. *MorphoLibJ [Software] (https://imagej.net/MorphoLibJ). https://doi.org/10.5281/zenodo.595022*.
55. *BAR [Software] (https://imagej.net/BAR). https://doi.org/10.5281/zenodo.597784*.
56. *3D ImageJ Suite [Software]. https://imagej.net/3D_ImageJ_Suite*.
57. TANGO: a generic tool for high-throughput 3D image analysis for studying nuclear organization | Bioinformatics | Oxford Academic. https://academic.oup.com/bioinformatics/article/29/14/1840/231770.
58. Vale, R. D., Spudich, J. A. & Griffis, E. R. Dynamics of myosin, microtubules, and Kinesin-6 at the cortex during cytokinesis in Drosophila S2 cells. *J. Cell Biol.* **186**, 727–738 (2009).