Interactive Visualization–A Key Prerequisite for Reconstruction and Analysis of Anatomically Realistic Neural Networks

Vincent J. Dercksen, Marcel Oberlaender, Bert Sakmann, and Hans-Christian Hege

Abstract Recent progress in large-volume microscopy, tissue-staining, as well as in image processing methods and 3D anatomy reconstruction allow neuroscientists to extract previously inaccessible anatomical data with high precision. For instance, determination of neuron numbers, 3D distributions and 3D axonal and dendritic branching patterns support recently started efforts to reconstruct anatomically realistic network models of many thousand neurons. Such models aid in understanding neural network structure, and, by numerically simulating electro-physiological signaling, also to reveal their function.

We illustrate the impact of visual computing on neurobiology at the example of important steps that are required for the reconstruction of large neural networks. In our case, the network to be reconstructed represents a single cortical column in the rat brain, which processes sensory information from its associated facial whisker hair. We demonstrate how analysis and reconstruction tasks, such as neuron somata counting and tracing of neuronal branches, have been incrementally accelerated – finally leading to efficiency gains of orders of magnitude. We also show how steps that are difficult to automatize can now be solved interactively with visual support. Additionally, we illustrate how visualization techniques have aided computer scientists during algorithm development. Finally, we present visual analysis techniques allowing neuroscientists to explore morphology and function of 3D neural networks.

Altogether, we demonstrate that visual computing techniques make an essential difference in terms of scientific output, both qualitatively, i.e., whether particular

V.J. Dercksen and M. Oberlaender contributed equally to this work.

V.J. Dercksen (✉) · H.-C. Hege
Zuse Institute Berlin, Germany
e-mail: dercksen@zib.de; hege@zib.de

M. Oberlaender · B. Sakmann
Max Planck Florida Institute, Digital Neuroanatomy, Jupiter, USA
e-mail: marcel.oberlaender@mpfi.org; bert.sakmann@mpfi.org

L. Linsen et al. (eds.), *Visualization in Medicine and Life Sciences II*,
Mathematics and Visualization, DOI 10.1007/978-3-642-21608-4_2,

goals can be achieved at all, and quantitatively in terms of higher accuracy, faster work-flow and larger scale processing. Such techniques have therefore become essential in the daily work of neuroscientists.

1 Introduction

One fundamental challenge in neuroscience is to understand how brains process sensory information about their environment and how this can be related to the animal's behavior. A widely used model system to study these relationships, from the molecular up to the behavioral level, is the somatosensory whisker system in rats. Like most nocturnal rodents, rats use their facial whiskers as complex tactile sensory organs to explore their environment. Such sensory input from single whiskers is processed by corresponding segregated neuronal networks in the primary somatosensory cortex (S1). Functional as well as anatomical evidence suggests that these networks resemble cylindric shapes [17, 22], hence being called *cortical columns* (Fig. 1(e)).

To shed light on how single whisker information is processed by a cortical column, recent attempts [12] aim to reconstruct its detailed anatomy and synaptic connectivity. Numerical simulations upon such high-resolution neural networks, with measured electrical input, caused by the deflection of a single whisker in the living animal (*in vivo*), will help to gain mechanistic understanding of sensory information processing in the mammalian brain.

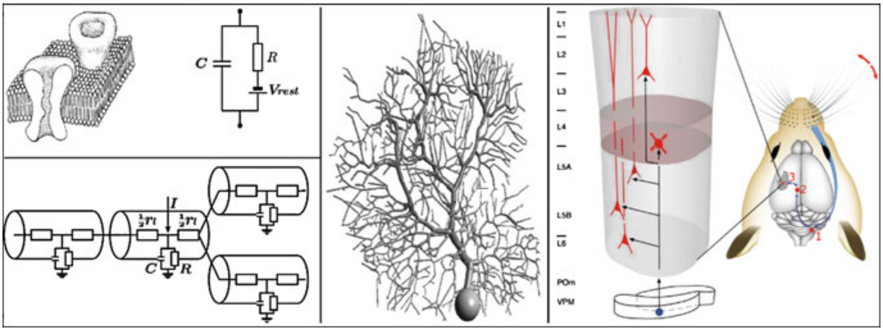

Fig. 1 (**a**) Ion channels provide a conductance through the membrane (Figure adopted from [7], courtesy of Sinauer Associates Inc., Publishers). (**b**) Associated passive electrical RC-circuit. (**c**) Spatial discretization of the cable equation results in multiple coupled compartments, each one locally equivalent to a uniform cable (Figure adapted from [3]). (**d**) Realistic full-compartmental model of a cerebral Purkinje neuron (Figure adopted from [19]). (**e**): Illustration of one-to-one correspondence between a whisker and a cortical column in the somatosensory cortex (3). A stimulus from a whisker is conveyed to its associated cortical column (schematically depicted on the left) via the brain stem (1) and the thalamus (2). Cells in the VPM nucleus of the thalamus project into the column and innervate cells in different layers (Figure adapted from [6])

As derived by [6, 8, 12], anatomical prerequisites to build realistic circuits comprise the number of neurons and neuron-types, their 3D distributions and characteristic dendrite and axon morphologies. Being composed of more than 17.000 neurons [10] of various anatomical as well as functional types, each type displaying complex axonal projection patterns (up to multiple centimeters per neuron), these prerequisites will hardly be satisfied by manual reconstruction approaches. Automated tools that extract the desired anatomical data from microscopic images are hence necessary and were recently described [14, 15].

Interactive visualization proved to be essential during the development of these methods, for validation of the results as well as for integrating the resulting data into neural networks. In this work, we illustrate how visual computing aids neuroscientists to extract and determine previously inaccessible anatomical data, how it facilitates the network reconstruction process and ultimately, how this might result in new scientific insights.

2 Biophysical Concepts of Neural Network Simulations

This section provides background for non-neuroscientists to better understand the anatomy and function of neurons and neuronal networks, as well as the modeling of electro-physiological signaling.

Functionally, neurons (or nerve cells) are the elementary signaling units of the nervous system, including the brain. These basic building blocks can mediate complex behavior, as potentially large numbers of them are interconnected to form neuronal networks. In general, neuronal networks are composed of chemically (synapses) or electrically (gap junctions) coupled local and long-range microcircuits of anatomically and physiologically classified neuron types. Specifically, each neuron is composed of a cell body (soma), multiple dendritic branches and one axonal tree, which receive electrical input from and transfer output towards other neurons, respectively. Hence, as found by the pioneer Ramón y Cajal about a century ago, neurons are usually grouped anatomically with respect to their soma shape, dendrite morphology and/or axonal branching pattern.

Neurons maintain a potential difference across their cell membrane, induced by deviating ion concentrations inside and outside the cell. This resultant resting potential (V_{rest}) is cell-type specific, but typically around -70 mV. Neuronal signaling involves changes in this potential.

The electro-physiological membrane properties can be described by electrical circuits containing a capacitance and resistance (RC-circuit, Fig. 1(b)). The membrane itself acts as a capacitance (C), which contains small pores (ion channels, Fig. 1(a)) with channel-type specific resistances (R_i). These pores mediate ionic currents across the membrane. Some channels are continuously open (passive or leaky) and others open and close actively with respect to voltage or chemical ligand concentration. Such active (e.g., voltage-dependent) conductances were first described in a neuron model by Hodgkin and Huxley (HH) in 1952:

$$C\frac{dV}{dt} = -\left(\sum_i I_i + I_{leak}\right), \tag{1}$$

where

$$I_i = \frac{V - E_i}{R_{i(V,t)}}, \tag{2}$$

represents voltage-dependent channel kinetics. E_i is the reversal potential of each ion channel type, i.e., the membrane potential at which there is no net flow of ions across the membrane. This model successfully explains a fast membrane depolarization, called action potential (AP) or spike, which is believed to be the fundamental electrophysiological unit in information processing.

Further development of electrical neuron models by Wilfred Rall, who believed that the complexity of the dendrites and axonal arborization would affect the neuronal processing, resulted in a cable theory for neurons. The cable equation

$$\frac{r_m}{r_l}\frac{\partial^2 V}{\partial x^2} = c_m r_m \frac{\partial V}{\partial t} + V \tag{3}$$

describes the flow of electric current (and change in potential) along passive one-dimensional neuronal fibers [18]. Here $r_m = \frac{R_m}{\pi d}$ and $c_m = C_m \pi d$ are the membrane resistance and capacitance per unit length, respectively, defined in terms of membrane resistance R_m and membrane capacitance C_m per unit area. The cable diameter is denoted by d; r_l is the (uniform) intracellular resistance per unit length along the cable.

Numerous bifurcations as well as variations in diameter and electrical properties along the neuronal branches, however, diminish the possibility to find analytical solutions for the cable equations. This leads to numerical solutions of spatially discrete, anatomy-based neuron models composed of multiple coupled HH-typed compartments (Fig. 1(d)), each locally resembling a uniform cable (Fig. 1(c)). Proper boundary conditions for the cable endings have to be specified. In particular at branching points, the voltage at the ends of the meeting cables is the same and the sum of all currents is 0. Input obtained from other cells at chemical synapses can be modeled electrically by adding appropriate currents to (3).

Initiated by Roger Traub and colleagues, many of these full-compartmental neuron models with active HH-type properties were synaptically interconnected, resembling realistic microcircuits and carrying out realistic neuronal operations. Following this tradition, attempts to reengineer the 3D anatomy and connectivity of functional networks of many thousand full-compartmental neurons were started [6,8,12]. Such goals became feasible with continuously increasing computing power and recently available large scale imaging, image processing and visualization techniques.

Hence, the reconstruction of a functionally well defined network, the 'cortical (barrel) column' in the rat's primary somatosensory cortex (S1) was started. Sensory information acquired by a single facial whisker is conveyed to dedicated regions in the brain stem, further to the ventral posterior medial nucleus of the thalamus (VPM) and finally to its corresponding cortical column (Fig. 1(e)). Excitatory thalamocortical input from this pathway into the column network, based on single whisker information, is essential to trigger simple behaviors, such as the decision to cross a gap. Hence, simulation of a realistically reengineered cortical column with single whisker input, measured *in vivo*, will potentially yield new insights and understanding of principle mechanisms that explain how the brain translates environmental input into behavioral responses.

Summarizing, the information processing of a neural network is simulated by modeling the propagation of electro-chemical signals through connected compartments representing the neuron geometry. The numerical solution of the resulting large system of ordinary differential equations consists of a potential value at each compartment at each time step. These time-dependent potentials defined on 3D graph structures are the input data for subsequent visual analysis.

A realistic and detailed anatomical model is a very important prerequisite to meaningfully simulate neural network activity. The 3D neuron morphology including neurite lengths, diameters and branching pattern is an important input parameter for the system of coupled cable equations. The number and distribution of nerve cells is important for defining the network and its connections. In the following, we focus on how this anatomical information can be obtained, and the role of image processing and visualization tools therein.

3 3D Reconstruction of Neuron Morphology from Microscopic Image Data

An important step in modeling neural networks is the reconstruction of three-dimensional morphology, particularly 3D axon and dendrite trajectories of an appropriate number of neurons. For modeling the thalamocortical part of the single whisker pathway, morphologies of all cell types within the cortical column as well as from its input region (VPM) are required.

As it is currently impossible to reconstruct all nerve cells from a single animal, a statistically representative number of cells of different anatomical types are reconstructed one by one from different animals and combined into a single network. Thus, for $\sim$10 different cell types and a minimum of $\sim$10 morphologies per type, at least $\sim$100 cells have to be reconstructed. To make this feasible, an efficient reconstruction pipeline is required.

Typically, cells are filled *in vivo* with a tracer. A common choice is biocytin. As it is non-fluorescent, it does not bleach (making a re-scan possible) and causes no outshining of weakly stained axons by strongly stained dendrites. The part of the brain containing the stained cell is then cut into sections, as it is too thick too be imaged in its entirety at high resolution. Typically, there are 20 sections of 100 μm thickness per cell. Large areas have to be scanned to capture all axonal arbor. A transmitted light brightfield (TLB) microscope is used to image the sections at high resolution using a mosaic-scanning technique. This results in a stack of thin, but three-dimensional images of 20–90 gigabytes each.

3.1 Original Workflow: Manual Tracing

The original workflow [14] for tracing the neurons is based on the Camera Lucida technique using, for example, the Neurolucida system [11]. Special software displays a live camera image of a section captured by an attached computer-controlled microscope on the computer screen. The user then manually marks the neuronal structures. By moving the stage in the xy-plane and varying the focus in the z-direction, a progressively larger volume is inspected. The software combines the marked structures into a 3D graph representation.

The most common approach is to start with the section containing the soma and follow the dendritic and axonal branches from there. Whenever a branch reaches the section boundary, the section under the microscope is replaced with the next one. The user has to find structures in the new section that correspond to the end points of the current tracing and transform the current tracing, such that it is aligned with the new section. To find the correct alignment more easily, usually three or more branches are traced concurrently. After alignment, the tracing continues in the next section. This procedure is repeated until all branches have been entirely traced. This approach works well for dendrites, as they usually have localized branching patterns and relatively large diameters ($\sim$ 2–5 μm). Axonal branches, however, can be less than 1μm thick and extend further away. Tracing and finding correspondences between sections is, therefore, extremely tedious and time-consuming (up to 100 h for complex, wide spreading axons). It also requires experienced users to reach a reliable level of reconstruction quality. The number of skilled users is usually the limiting factor when large numbers of cells have to be reconstructed.

3.2 First Improvement: Automatic Tracing

In order to reduce the manual labor an automatic segmentation method for dendrites and axons is required. The segmentation is, however, non-trivial: the structures are very thin and contrast may be poor due to limited dye penetration, especially further away from the soma where the tracer is injected into the cell. Therefore, axons frequently appear fragmented. Also numerous background structures have a gray-value in the same intensity range as the axons and dendrites (see Fig. 2(a)), in particular glia cells that presumably contain biocytin contrast agent intrinsically.

Existing (semi-)automatic methods [9] focus mainly on tracing dendritic branches. Therefore, an algorithm was developed that automatically segments both dendritic and axonal structures from 3D TLB images and computes a graph

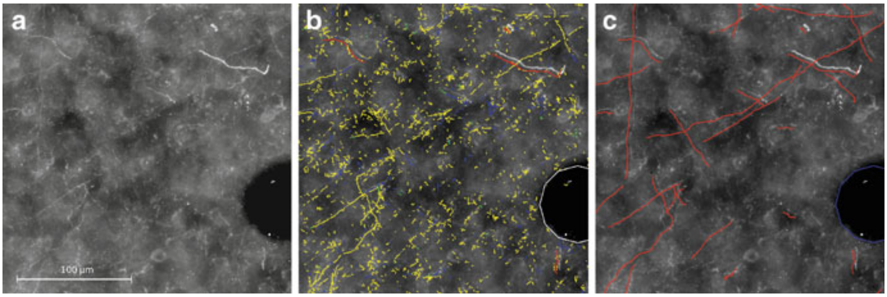

Fig. 2 (**a**) Small region of a maximum intensity projection of an inverted 3D TLB image of a single brain section. The bright filamentous structures are the axon fragments to be traced. However, the image contains a lot of background structures in the same intensity range. (**b**) Result of the automatic tracing algorithm, slightly shifted to show the correspondence with the image. To aid in the interactive post-processing, the fragments are colored according to their position within the section: touching top of the section (blue), bottom (green), both (red) or none (yellow). The round structure is a blood vessel cross-section. (**c**) Final tracing after post-processing

representation of the approximated centerlines [14]. The method is based on deconvolution [13], followed by a sequence of image filters and morphological operations, and finally skeletonization. The method finds virtually all neuron branches. It is, however, very conservative: oversegmentation is accepted in order to assure that no foreground structure is lost (see Fig. 2(b)).

To finish the reconstruction, a number of post-processing steps are performed interactively using the Serial Section Manager in the Neurolucida software. First, non-neuron artifacts are removed and fragmented branches are spliced (connected) within each section separately. The user is presented visual feedback in the form of three orthogonal projection images (xy-, xz- and yz-projections, see Fig. 3(a)). To support the user in deciding which fragments to delete and which to connect, the traced segments are overlaid onto a maximum intensity projection (MIP) image of the section.

Second, the sections are (rigidly) aligned. Alignment is done pair-wise, i.e., two neighboring sections at a time. One section serves as a reference and remains fixed, while the other is interactively rotated and translated. The user has to perform a pattern matching task: he needs to determine the correspondences between neuron fragments in the two slices. Again, having only 2D projection views, this can be extremely difficult, especially for dense, visually symmetric clouds of projected line fragments. In some cases, finding the correspondence is even impossible. To alleviate this problem, blood vessel contours are traced additionally. These contours are usually easier to match, as they are fewer and may have different diameters, giving a hint on which vessels may correspond. After a rough blood vessel match has been found, the neuron fragment correspondence can often be determined more easily.

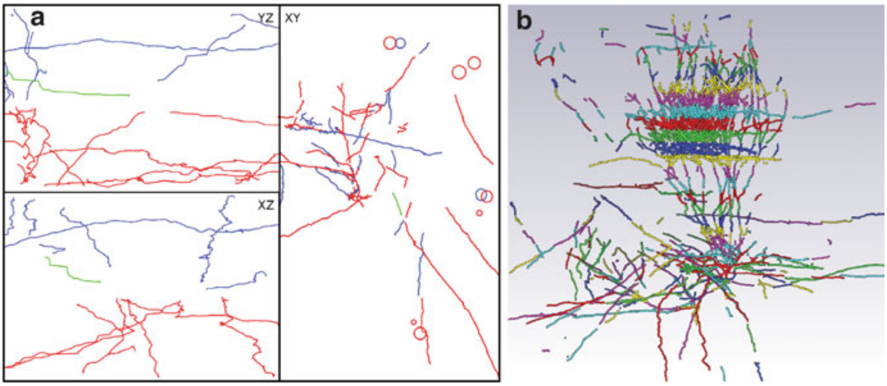

Fig. 3 (**a**) Two sections (red and blue) after alignment. The vessel contours (blue and red circles) are helpful during the manual alignment, as these features are easier to recognize and to match than the segment end points. Based on 2D projection views, it is often difficult to decide which axon fragments are to be spliced, e.g. in case of the green fragment. (**b**) Automatically aligned stack of reconstructed sections, colored by section

Finally, the corresponding fragments are spliced across the section boundaries. Again, the lack of a three-dimensional view on the data makes this step difficult and unnecessarily time-consuming.

3.3 Second Improvement: 3D Interactive Editing

Being based on 2D projection views, the interactive post-processing severely hampered the reconstruction process. Therefore, a 3D interactive environment, the *filament editor* [1], was developed as part of the Amira [20] visualization and data analysis software. Its main components are:

- A graph data structure, consisting of a set of nodes (the branching and end points) and segments connecting the nodes. The trajectory of each segment in 3D space is represented by a sequence of points. Attribute values can be associated with the nodes, segments and/or points. For example, a *radius* attribute can be associated with the points, defining a floating point value at each point along the segments. One could also define anatomical labels to all elements in the combined set of nodes and segments, resulting in the assignment of label values like *dendrite*, *axon* or *vessel* to subgraphs.
- A set of tools to select (parts of) the data structure, including clicking on or drawing a contour around target elements, selection of connected subgraphs, etc.
- A set of operations to manipulate the data structure, like deletion and connecting of nodes and segments. All operations can be undone/redone.

- A graphical user interface that – besides GUI elements for invoking the available selection and editing tools – contains a 3D viewer window for displaying the graph and other objects, like a MIP image. The graph can be colored according to its attribute values.

In the improved work flow, first, each slice is cleaned and spliced individually. The user loads the automatically traced neuron fragments and displays them colored by the attribute indicating whether a particular connected subgraph touches the top of the slice, the bottom, both or neither (see Fig. 2(b)). This information is exported by the tracing program, together with a MIP image of the section. The graph data is superimposed on the MIP image. In the MIP, most neuronal processes are easily distinguishable from falsely traced background structures. Hence, if not already traced perfectly, fragmented branches are spliced according to the MIP interactively. Afterwards, artifacts are removed by drawing a selection polygon around them. Only connected subgraphs that are entirely within the polygon are selected and removed. This feature makes it easy to remove many small artifacts at once, while avoiding the removal of parts of larger, already connected, neuronal branches. However, some processes are too faint to be recognized in the MIP. By rotating the tracing, human pattern recognition can compensate for that. Specifically the coloring of top and bottom structures proved to be helpful. Every branch has to touch either the top or bottom (in most cases both) borders of a slice. Therefore, in most cases, these weak processes resemble dashed 3D lines starting with blue (top), connecting to yellow (intermediate) and ending at green (bottom) fragments.

Second, all manually edited sections are combined into a single graph for interactive alignment in the *filament editor*. The nodes and segments of each section are labeled according to their section number for visualization and fast selection. A section is aligned with its fixed predecessor section by interactive translation and rotation in the 3D viewer using handles. All other sections can be hidden to not obscure the view. By repeated pair-wise alignment the entire stack is processed.

Third, the segments are spliced across the slice boundaries. The alignment and splicing process is regarded as the final quality control for each tracing. Missing or falsely connected branches as well as loops can be easily identified and corrected in the individual sections. As a final and optional step, subgraphs representing different anatomical structures can be labeled as such.

An important aspect of such interactive tools is usability. In the case of the *filament editor*, this not only includes fast access to the selection and editing tools by mouse and shortcut keys, it also requires stability and robustness against operating errors. Important is also responsiveness. All viewing, selection and editing operations have to be immediate, also for the relatively large data sets that may result from the automatic tracing (containing ~1M points).

Altogether, the 3D environment enables efficient interactive post-processing and alignment of the automatically reconstructed sections. The additional depth cues, obtained primarily by interactive rotation of the scene, are the major advantage of the 3D editing, when compared to the 2D Neurolucida workflow. They are essential in order to quickly decide which fragments to discard and which to connect.

Based on the alignment by the interactive tool, all neuron morphologies could be reconstructed with satisfactory quality.

3.4 Third Improvement: Automatic Section Alignment

In order to make the alignment faster and more objective, an automatic alignment algorithm was developed [2]. Given two sections containing multiple segments (polylines), it computes an optimal transform, i.e., a rotation angle around the z-axis, a 2D translation in the xy-plane and an optional uniform scaling, using the following steps:

1. For a pair of sections to be aligned, find the set of fragment end points for each section. This results in two point sets P and Q, one for each section, that have to be matched. A matching M is a set of corresponding point pairs $\{(\mathbf{p}_i, \mathbf{q}_j)\}$.
2. Find a set of candidate matchings by searching for subsets P' and Q' for which the mutual distances between all points in P' are similar to the distances between the corresponding points in Q'. For each candidate matching, compute the optimal transformation. These are the starting transformations.
3. Starting from each starting transformation, further optimize the matching and the transformation T with respect to a scoring function that favors a large match size $|M|$, but penalizes large distances between corresponding points:

$$score(P, Q, M, T) = \frac{|M|}{min(|P|, |Q|)} \cdot e^{-\alpha \cdot \sqrt{\frac{\sum_{(\mathbf{p},\mathbf{q}) \in M} \|\mathbf{p} - T(\mathbf{q})\|^2}{|M|}}} \tag{4}$$

The parameter α balances these contradicting goals. The final result is the matching and transformation leading to the highest score.

This method works very well for this type of data. It is also very fast, it takes a few seconds per slice pair for typical point set sizes of 30–100. As this automatic alignment method is integrated in the *filament editor*, the results are immediately displayed in the 3D viewer and can be visually validated. Whenever a slice pair cannot be successfully aligned, it can be transformed interactively, as before.

3.5 Reconstruction Time

The reconstruction time strongly depends on the neuron type, its spatial extent and axonal complexity. For instance, an axon of a pyramidal neuron in S1 can spread over millimeters, reaching a total length of several centimeters. Reconstructing L5 cells, approximately 20 sections (1.7×1.7×0.1 mm^3) at a resolution of 92×92×500 *nm* need to be imaged, resulting in a data set of $1.7\,TB$. For such neurons the reconstruction time can be divided as follows:

- Scanning time: ∼ 3 days (∼ 4h per section).
- Automatic processing: ∼9 days (∼10 h per section).
- Interactive post-processing and alignment: ∼2–3 days (5–90 minutes per section, 60 minutes on average).

Although this results in a total of two weeks per L5 cell, less than 20 hours of manual labor are required. The pipeline also greatly benefits from task and data parallelization. The time for automatic reconstruction decreases linearly with the number of microscopes, computers and human tracers. Finally, the interactive post-processing does not need to be done by a single neuroscience expert, but by multiple student assistants in parallel, after only a few hours of training. Altogether, this pipeline allows for a much higher throughput of individual cell models, making the modeling of large networks feasible.

4 Automatic Counting of Neurons in Large Brain Volumes

In order to model a realistic neuronal network in 3D, one needs to know how many nerve cells constitute the network and how they are distributed in space. The estimation of absolute numbers of neurons, densities or rates of density change in neuron populations has usually been based on random, sparse sampling methods, such as stereology [21]. These methods determine cell densities by inspecting a representative sub-volume of tissue and extrapolating the obtained density values to a reference volume. However, due to the lack of well-defined reference volumes and usually unjustified assumptions of homogeneous densities across them, it is preferable to detect and count all cells within the volume of interest.

Three-dimensional imaging techniques, e.g., confocal or widefield microscopy, and suitable neuronal stains, like NeuN, which stains all neuron somata, offer possibilities to achieve this. Using these techniques, neuron somata appear as roundish shapes, sometimes prolongated by the onset of the dendritic arbors that have taken up some stain.

4.1 Original Workflow

The simplest way to detect all somata in such 3D images is by manual marking. This is, for example, achieved by moving a 2D image slice through the data set and clicking on the center of each soma to place a landmark at its approximated centroid. The center is set on the slice where the soma has the largest diameter. This procedure, however, is very time-consuming, but can be regarded as rather accurate, with a reported inter-user variability of 2% [10].

An approach to make the workflow more three-dimensional is to display a rough binary segmentation by a semi-transparent isosurface and use this to place the landmarks. Alternatively, a direct volume rendering of the image can be used [16]. These approaches depend, however, on a suitable threshold value or transfer function and are, therefore, not very accurate when the brightness varies strongly across the image, which is typical for these kind of images.

The main problem for both of these approaches is the amount of required manual labor, making a quantification of some cubic millimeter large volumes impossible. Even a volume containing just one single cortical column and its direct surroundings ($500 \times 500 \times 2000\,\mu$m) embodies tens of thousands of cells, making manual marking of each neuron extremely time-consuming.

4.2 Automatic Detection and Counting Algorithm

To make the exhaustive counting of nerve cells in large tissue volumes feasible, we developed a method for automatic detection and counting of neuron somata in 3D images [15]. It consists of three steps (Fig. 5):

1. Segmentation. This step produces a binary image separating foreground from background. The main processing steps are contrast enhancement, thresholding and a number of morphological image filters for speckle removal and hole filling. The exact filters and parameters depend on the imaging and staining procedure. In [15] various combinations for different types of image data are presented. Due to locally high cell densities and limited resolution, clusters of touching somata may, however, remain.
2. Morphological splitting. Touching objects are separated at the connecting neck by marker-based watershed transform.
3. Model-based splitting. Any remaining touching objects are divided on the basis of the volume statistics of the object population. This is an optional step that assumes a dominant population of cells with a Gaussian-distributed soma volume. Any object that could not be split at a thin neck, but is statistically too large to consist of a single object, is divided into the most probable number of components.

The method results in cell counts and densities comparable with manual counting ($\sim$4% deviation). This makes it possible to refrain from manual counting and process large data sets completely automatically. Thus, also the exhaustive counting of entire column populations is now feasible.

4.3 Visualization for Algorithm Development and Validation

The development of a segmentation algorithm that consists of a sequence of image filters, each with its own parameters, is often a trial-and-error process that requires a frequent evaluation of intermediate results. As an objective quantitative measure is not always available, this evaluation is commonly done visually. For an efficient workflow during algorithm development, easy-to-use visualization tools are therefore indispensible.

A common way to judge the result of a binary segmentation algorithm is to display a gray-value image slice overlaid with the segmented foreground in a semi-transparent color, for example, using Amira [20] (see Fig. 4(a)). By moving the 2D slice, the algorithm developer can quickly verify the quality of the segmentation result. If a foreground voxel object needs to be separated into its constituent parts, as, for example, in step 2 and 3 of the automatic cell detection method, the 2D slicing approach is insufficient. The boundary between two touching objects should be located at a thin 'neck', which is very difficult to verify on 2D slices. In this case, the generation of object boundary surfaces using the Generalized Marching Cubes algorithm [5] has proven very useful. As in this surface representation each object has a distinct color, the correctness of the splitting result can be determined almost instantly.

Visualization is also very useful to compare differences between automatically computed soma positions and those of a reference data set, e.g., a 'gold standard', manually created by an expert. In order to discover whether a particular false positive or negative result is caused by the automatic algorithm, by expert error, data artifacts, or differences in treatment of only partially imaged boundary objects, etc., one has to look closely at each differing data point in combination with the image data. In Amira, one can for example display both landmark sets with different colors, and show the image data using 2D textured slices with contrast control, or 3D direct volume rendering. Together with a simple 3D viewer allowing for interactive zooming and rotating, this is a very effective environment for visual method validation.

Although the automatic method performs very well, it may occur that the results for a particular data set are not satisfactory. In such cases, it is useful to have an interactive landmark editor to make corrections.

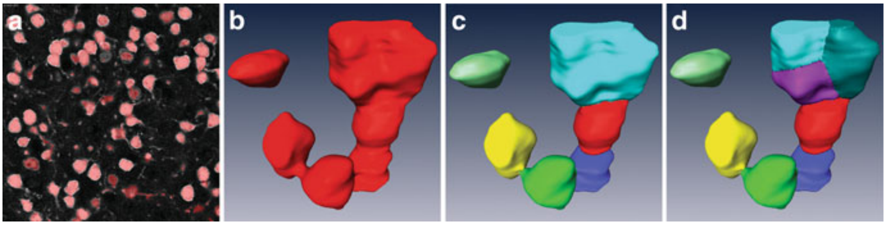

Fig. 4 (**a**) Segmentation validation by semi-transparent overlay. (**b**) Surface reconstruction of a cluster of touching cells after binary segmentation. (**c**) After marker-based watershed splitting. (**d**) After splitting based on soma volume statistics

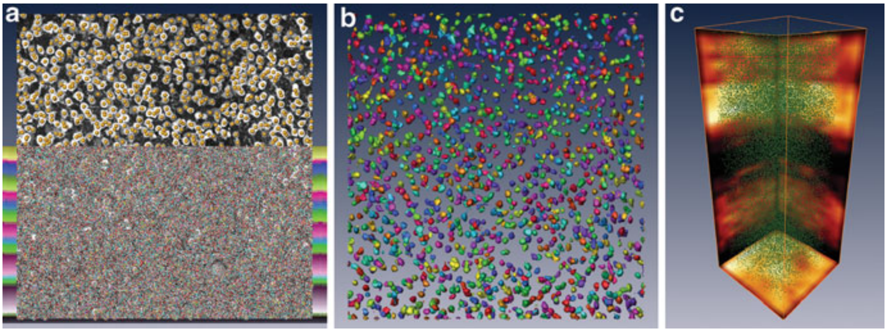

Fig. 5 (**a**) Maximum intensity image of 3D data set containing NeuN-stained somata and automatically detected centers. (**b**) Surface reconstruction of individual cell bodies after automatic segmentation and cluster splitting. (**c**) Automatically detected somata (green) and MIP of the cell density in a large tissue volume (550 μm × 550 μm × 2 mm) containing a cortical column

4.4 Conclusion

Summarizing, we have seen how automatic counting of neuron somata allows the neuroscientist to reliably determine neuron numbers and 3D distributions in large tissue volumes, like a brain region containing cortical columns. Being free of assumptions, like homogeneity across a reference volume, the automated approach potentially leads to more valid results than sparse sampling methods. This example also illustrates that the use of visualization methods is indispensible during algorithm development, as it allows the computer scientist to perform the frequent evaluation of results quickly and effectively.

5 Visual Analysis of Neuron Morphology and Function

In this section we will present two examples of how even simple visualization tools for the analysis of 3D neuron morphology support neuroscientists in their daily work. We also discuss the requirements and challenges for analyzing functional properties of networks of neurons resulting from numerical simulations.

5.1 Evaluation of a Neuron Classification Algorithm

Given a reasonably large set of reconstructed neurons, one can manually and/or automatically group them in multiple morphological cell types. Such a classification is a key prerequisite for many anatomical investigations. For example, correlation of

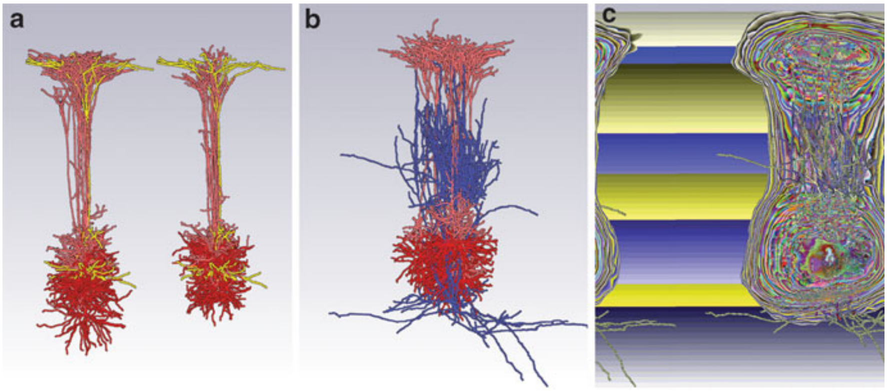

Fig. 6 (**a**) Visual validation of neuron morphology classification. The same highlighted neuron is displayed together with cells of the same type (left) and of a different type (right). In the latter case, the difference in shape is immediately visible. (**b**) Visualization of the spatial relation of two neuron types. The apical (pink) and basal (red) dendrites of a small number of L5B cells are displayed together with one VPM axon (blue). (**c**) Volume-rendered dendritic density field of a large population of L5B cells. Together with an axonal density field (not displayed) this allows for qualitative and quantitative analysis of the innervation domain of these two cell types

anatomical classes with physiological responses to stimuli or determination of the 3D spatial extent (with respect to anatomical position landmarks such as pia surface) of a certain cell type, help neuroscientists to gain insights into network structure and function.

For any given neuron clustering algorithm, for instance the one described in [12], a tool to visually validate and analyze its results is very useful. We developed a simple visualization tool to display sets of morphologies (see Fig. 6(a)). Neurons that have been assigned to the same cluster are overlaid. This gives insight into the typical properties of the particular class. The user can select and highlight one neuron from the list of all neurons. From this, the user can immediately see whether the selected neuron's morphology is similar to the morphology of the other neurons in the group. The user can also compare it to neurons in other clusters. The combined display of morphologies can also reveal errors in the reconstruction or in the registration into a common reference frame.

5.2 Exploring 3D Innervation Volumes of Neuron Populations

To understand the structure of a neural network, a tool that allows for 3D visual exploration of a population of neurons is very helpful. One property of interest is the 'innervation domain' of two neuron types, i.e., the region in space where axons of one type and dendrites of the other type overlap, indicating potential connections between the cell types.

Given a population of neurons that have been registered into a common coordinate frame, e.g., using the method described in [12], the innervation domain can be investigated by simply line-rendering some cells of both types with dendrites and axons colored differently (Fig. 6(b)). This allows one to see the relative positions of the two cell populations and their overlap regions.

For large neuron populations line renderings result in extreme visual clutter. One solution is to sample branch density on a uniform grid, i.e., to compute dendrite/axon length per volume unit. This density map can be interactively explored, using standard visualization techniques like volume rendering or iso-surfaces (Fig. 6(c)). Having such axonal and dendritic density maps for two different cell types, one can easily see where they overlap. In addition, it is possible to quantitatively analyze the individual density fields and their innervation domain.

5.3 Visual Analysis of Simulated Network Activity

Numerical simulation of the activity of single neurons (see Fig. 7) or large networks results in time-dependent potentials, defined on each compartment comprising the neuron morphology. Depending on the size of the network, the spatial resolution and the number of time steps, the simulation of a column-size network can easily lead to data sets that are hundreds of gigabytes in size. Tools for quantitative and visual analysis are required to gain knowledge from such large and complex data. Of interest are, for example, methods to correlate simulation results with measured quantities, like *in vivo* two-photon Ca^{2+} time series imaging [4] for

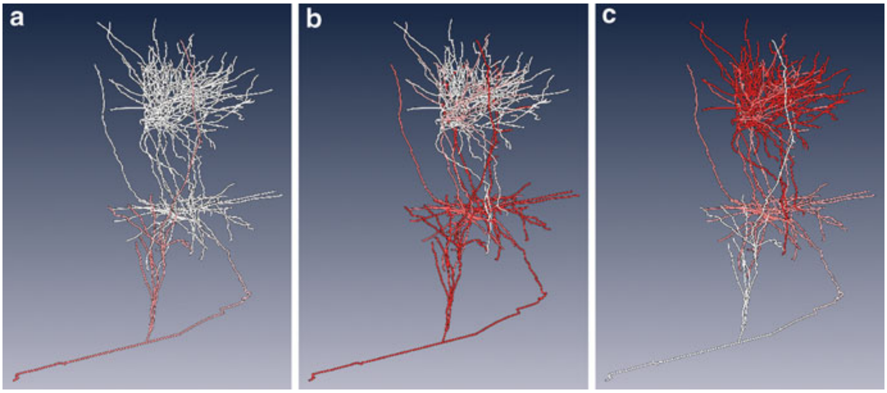

Fig. 7 Numerically simulated action potential propagating through a single VPM axon after 1 (**a**), 3 (**b**) and 5 (**c**) ms after artificial current injection at the axon onset (lower left)

validation purposes. Second, to find and understand yet unknown activity patterns, an interactive exploration tool is required that supports the formulation and verification of hypotheses, for example regarding correlations between anatomical and physiological data. Also the comparison of results of different simulation runs for different input conditions is important for network parameter tuning and sensitivity tests, requiring an even more challenging 5D analysis tool.

6 Discussion and Conclusion

We illustrated, by means of ‘real-world’ problems from the field of neuroscience, that visual computing tools, including interactive visualization, proved to be essential for the extraction and validation of previously inaccessible anatomical data from microscopic images. The resulting increase in throughput is important for obtaining required anatomical data for the reconstruction of large, high-resolution 3D neural networks, e.g., for numerical simulations. Interactive visualization tools also support the integration of anatomical data into such networks, and subsequent exploration and analysis.

The integration of all visualization tools into one software system (the visualization and data analysis software Amira [20]) has proven extremely beneficial as the user has access to many different modules to simultaneously display image, line-like and other data, allowing him to relate different objects in a single 3D interactive environment.

Although important anatomical data could be extracted for the reconstruction of a large neural network representing a cortical column, there are some important open issues that have to be addressed in order to be able to study the functioning of such large networks by numerical simulation of the electro-physiological processes. These problems include the determination of the wiring of the neuronal circuits, i.e., the number and distribution of synaptic connections between cells in the network (“connectomics”). Also other parameters that are important for the simulation need to be determined reliably for all cells in the network, like dendrite/axon radii, spine/bouton distributions, ion channel properties, etc. The current tendency of using higher resolution image modalities as an initial step to address these problems and the according increase in data set size, calls for adequate visual computing techniques to be developed in the near future. Also for the analysis of large-scale numerical simulations of neural networks, adequate visual computing methods will be indispensable.

Acknowledgements We would like to thank R.M. Bruno and C.P.J. de Kock for providing the sections for the neuron morphology reconstruction, H.S. Meyer for providing the stained somata images, P.J. Broser for initiating the collaboration, and finally S. Lang for his work on the NeuroDUNE simulation software.

References

1. Dercksen, V.J., Broser, P.J., Sakmann, B., Hege, H.C., Oberlaender, M.: Efficient 3D reconstruction of single neuron morphology from stacks of transmitted light brightfield microscopy images. In preparation
2. Dercksen, V.J., Weber, B., Günther, D., Oberlaender, M., Prohaska, S., Hege, H.C.: Automatic alignment of stacks of filament data. In: Proc. IEEE International Symposium on Biomedical Imaging, Boston, USA, 971–974 (2009)
3. Gerstner, W., Kistler, W.M.: Spiking Neuron Models: Single Neurons, Populations, Plasticity. Cambridge University Press (2002)
4. Göbel, W., Kampa, B., Helmchen, F.: Imaging cellular network dynamics in three dimensions using fast 3D laser scanning. Nature Methods **4**, 73–79 (2007)
5. Hege, H.C., Seebaß, M., Stalling, D., Zöckler, M.: A generalized marching cubes algorithm based on non-binary classifications. Tech. rep., ZIB Preprint SC-97-05 (1997)
6. Helmstaedter, M., de Kock, C.P.J., Feldmeyer, D., Bruno, R.M., Sakmann, B.: Reconstruction of an average cortical column in silico. Brain Research Reviews **55**(2), 193–203 (2007)
7. Hille, B.: Ionic channels of excitable membranes, 2nd edn. Sinauer Associates, Inc., Publishers (1992)
8. Markram, H.: The blue brain project. Nat Rev Neurosci **7**(2), 153–160 (2006)
9. Meijering, E.: Neuron tracing in perspective. Cytometry. Part A **77**(7), 693–704 (2010)
10. Meyer, H.S., Wimmer, V.C., Oberlaender, M., de Kock, C.P.J., Sakmann, B., Helmstaedter, M.: Number and Laminar Distribution of Neurons in a Thalamocortical Projection Column of Rat Vibrissal Cortex. Cerebral Cortex **20**(10), 2277–2286 (2010)
11. MicroBrightField Inc.: Neurolucida. www.mbfbioscience.com/neurolucida (2011)
12. Oberlaender, M.: Three-dimensional reengineering of neuronal microcircuits. the cortical column in silico. Ph.D. thesis, Univ. of Heidelberg, Germany (2009)
13. Oberlaender, M., Broser, P.J., Sakmann, B., Hippler, S.: Shack-Hartmann wave front measurements in cortical tissue for deconvolution of large three-dimensional mosaic transmitted light brightfield micrographs. J. Microsc. **233**(2), 275–289 (2009)
14. Oberlaender, M., Bruno, R.M., Sakmann, B., Broser, P.J.: Transmitted light brightfield mosaic microscopy for three-dimensional tracing of single neuron morphology. J. Biomed. Opt. **12**(6), 1–19 (2007)
15. Oberlaender, M., Dercksen, V.J., Egger, R., Gensel, M., Sakmann, B., Hege, H.C.: Automated three-dimensional detection and counting of neuron somata. J. Neurosci. Methods **180**(1), 147–160 (2009)
16. Peng, H., Ruan, Z., Long, F., Simpson, J.H., Myers, E.W.: V3D enables real-time 3D visualization and quantitative analysis of large-scale biological image data sets. Nature biotechnology **28**(4), 348–53 (2010)
17. Petersen, C.C.H.: The functional organization of the barrel cortex. Neuron **56**(2), 339–355 (2007)
18. Rall, W., Agmon-Snir, H.: Methods in neuronal modeling: from ions to networks, chap. Cable Theory for Dendritic Neurons, 27–92. C. Koch and I. Segev (eds.) (1998)
19. Schutter, E.D., Bower, J.M.: An active membrane model of the cerebellar Purkinje cell: II. Simulation of synaptic responses. J Neurophysiol **71**(1), 401–419 (1994)
20. Stalling, D., Westerhoff, M., Hege, H.C.: Amira: A highly interactive system for visual data analysis. In: C. Hansen, C. Johnson (eds.) The Visualization Handbook, chap. 38, 749–767. Elsevier (2005)
21. Sterio, D.C.: The unbiased estimation of number and sizes of arbitrary particles using the disector. J Microsc **134**(2), 127–136 (1984)
22. Wimmer, V.C., Bruno, R.M., De Kock, C.P.J., Kuner, T., Sakmann, B.: Dimensions of a Projection Column and Architecture of VPM and POm Axons in Rat Vibrissal Cortex. Cerebral Cortex **20**(10), 2265–2276 (2010)

http://www.springer.com/978-3-642-21607-7

Visualization in Medicine and Life Sciences II
Progress and New Challenges
Linsen, L.; Hagen, H.; Hamann, B.; Hege, H.-C. (Eds.)
2012, VIII, 288 p., Hardcover
ISBN: 978-3-642-21607-7