Chapter 2

New Methods for the Computer-Assisted 3D Reconstruction of Neurons from Confocal Image Stacks

2.1 Abstract

Exact geometrical reconstructions of neuronal architecture are indispensable for the investigation of neuronal function. Neuronal shape is important for the wiring of networks, and dendritic architecture strongly affects neuronal integration and firing properties as demonstrated by modelling approaches. Confocal microscopy allows to scan neurons with submicron resolution. However, it is still a tedious task to reconstruct complex dendritic trees with fine structures just above voxel resolution. We present a framework assisting the reconstruction. User time investment is strongly reduced by automatic methods which fit a skeleton and a surface to the data, while the user can interact, and thus, keeps full control to ensure a high quality reconstruction. The reconstruction process comprises a successive gain of metric parameters. First a structural description of the neuron is built, including the topology and the exact dendritic lengths and diameters. We use generalized cylinders with circular cross-sections. The user provides a rough initialization by marking the branching points. The axes and radii are fitted to the data by minimizing an energy-functional which is regularized by a smoothness constraint. The investigation of proximity to other structures throughout dendritic trees requires a precise surface reconstruction. In order to achieve accuracy of 0.1 micron and below, we additionally implemented a segmentation algorithm based on geodesic active contours which allows for arbitrary cross-sections and uses locally adapted thresholds. In summary, this new

reconstruction tool saves time and increases quality as compared to other methods which have previously been applied to real neurons.

2.2 Introduction

Metric analysis of neurons is indispensible to address (i) the complicated relation between physiology (function) and morphology (form) of neurons by computational approaches (Borst and Haag, 1996; De Schutter and Bower, 1994; Hausser et al., 2000; Koch et al., 1982; Segev and Rall, 1998), (ii) the characterization of cell types and the investigation of neuronal development by statistical analysis of the morphology (da Fontoura Costa and Velte, 1999; Libersat and Duch, 2002; Mizrahi et al., 2000; Uylings et al., 1986; van Pelt et al., 1989), and (iii) the relation of neuron surfaces to other structures in three-dimensional space (Belichenko and Dahlström, 1995; Gray and Weeks, 2003; Hiesinger et al., 2001; Jankowska et al., 1995; Lamotte d'Incamps et al., 1998; Wouterlood et al., 2002) (for an overview see also (da Fontoura Costa et al., 2002)). Depending on the goal a description of the neuron has to comprise the center lines and radii, the topological structure including branching and end points, the order of segments[1], or an exact reconstruction of the surface.

Generally, in order to obtain morphometric measurements, reconstructions of neurons should fulfill the following requirements: (i) sufficient accuracy must be accomplished, (ii) topological constraints based on the assumption of a tree-like structure must be fullfilled, (iii) the relevant measurements must be represented explicitly allowing direct access, and, last but not least, (iv) the amount of the user's time and effort should be reasonably small.

The goal of decreasing the necessary expense of user interaction often acts contrary to that of ensuring the accuracy and the topological correctness of the result. Consequently, available commercial software tools provide methods either for automatic segmentation or manual reconstruction. One of the state-of-the-art software tools of the latter category is *Neurolucida* (by MicroBrightField, Inc., Williston, VT). Unfortunately, the manual tracing of complex dendritic trees is overly tedious. If the user tries to reduce the input actions the result suffers from inaccuracy and abrupt changes of thickness or center line direction.

Automating the reconstruction process is difficult due to noise and the partial volume effect. Noise is generated by different sources in the confocal microscope and the partial

[1] The segment's order is the number of branching points between the segment and the soma.

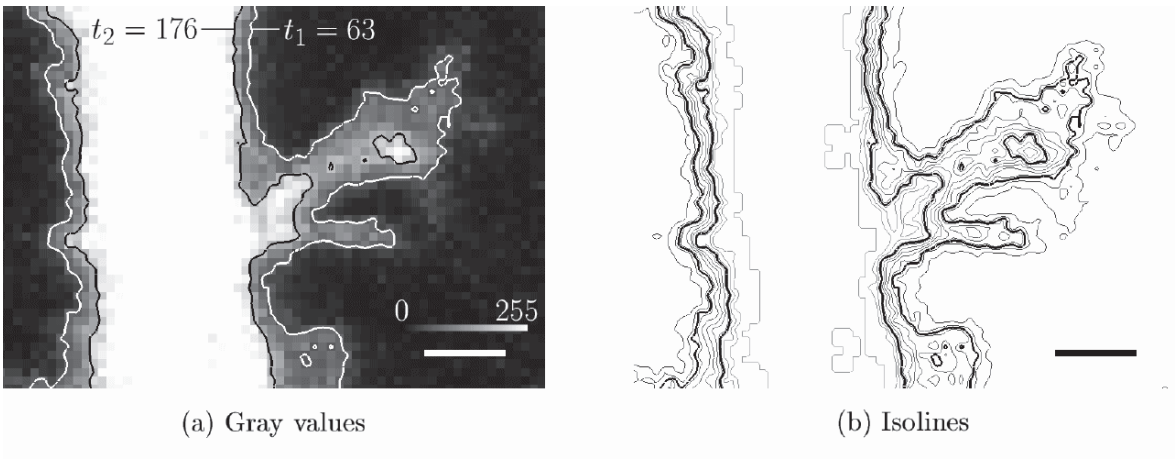


Figure 2.1: Finer dendritic structures show less contrast w.r.t. the background than thicker ones. The figure shows a slice from a confocal image of a neuron. Scalebar: 1μm. (a) Image intensity coded by gray values. The iso-intensity lines for 63 and 176 are shown in white and black. (b) Contour plot of the image. The iso-intensity lines for 63 and 176 are shown in solid black lines. If the threshold for segmentation is close to be 176, it is optimal for the thick branch, but fails to account for the thin branch.

volume effect occurs, when the volume, which is assessed for a measurement, contains labeled and unlabeled tissue. The strength of the latter depends on the point spread function (PSF). Both phenomena cause overlapping intensity histograms of the interior and exterior. In particular fine structures have low contrast to the background which may be decreased additionally by inhomogeneous staining of the cell (see fig. 2.1). Several software tools featuring automated reconstruction procedures are available. To the best of our knowledge all of them work with threshold-based segmentation methods. Here we confine our discussion to the *FilamentTracer* (formerly called *NeuronTracer*) included into the program *Imaris* (Bitplane, Zürich). It provides three modes for the reconstruction of neurons: one works fully automatic, one semi-automatic and the other fully manual. The semi-automatic tracing allows the user to draw along the branches in a two-dimensional projection and automatically computes the position in the third dimension. We discussed the principle problems of manual tracing with respect to *Neurolucida* above. The automatic reconstruction is based on a hysteresis method using two thresholds. Voxels with intensity values above the high threshold are marked as interior with high confidence (say class *inHi*), those between the high and the low threshold are marked as interior with low confidence (say class *inLo*) if there is a connection via other inLo voxels to an inHi voxel. Voxels falling below the low threshold are marked as exterior. The two thresholds are chosen by the user. Afterwards the segmented volume is reconstructed with generalized

cylinders. The result can be manually corrected.
The hysteresis method is an improvement over single threshold segmentations, but still suffers from the same limitations. The appropriate thresholds depend on the staining intensity of a branch, which in turn, results from its true diameter (see fig. 2.1). If the user optimizes the thresholds to a specific diameter, structures with other diameters are miscalculated. In case of neurons with a limited variation of segment diameters, and therefore staining intensities, the hysteresis method improves the completeness of image segmentation. However, extensive variation in the staining intensities of neuronal structures makes the choice of appropriate hysteresis values which include the complete topology of neuronal arborization difficult or even impossible. Therefore, this automatic reconstruction procedure is useful for fast visualization and extraction of the rough topology, whereas morphometric measurements are imprecise, in particular for neurons with a wide range of diameters. A second source of inaccuracy in the FilamentTracer is smoothing which can be applied to the result of the reconstruction. It neglects the image data, hence abrupt changes of the axis direction or the radii are leveled out regardless of actual data values. Thus the demand for the reconstruction of fine structures near the spatial resolution limits of the microscope is not answered by existing automated methods. In order to increase the efficiency of the reconstruction process without losing any quality, we developed a system which highly reduces the time investment by automatic computational methods. It assists the manual reconstruction process by (i) fitting automatically the center lines, radii, and the surface of a neural branch between points given by the user, and (ii) providing an intuitive and convenient user interface which allows immediate access to the result. The combination of the two features yields the highest possible accuracy in practice. On the one hand, the automated fitting relieves the user from most of the necessary user input, thus accelerating the process and eliminating subjective estimations and imprecision due to the user's haste or exhaustion. The reconstruction process is made mostly independent from the user's knowledge. On the other hand the user has full control as an option allowing correction at regions, where computation appears to fail.
This article is organized as follows. In the next section we review related approaches for the detection of tubular structures in three-dimensional images. In the third section we introduce the model for neural structures and describe the methods which fit the model to the data. After showing example results from real neurons in the fourth section, conclusions are drawn in the fifth one.

2.3 Image Analysis Background

The task of reconstructing a neuron from raw three-dimensional image data can be split into two subtasks: First a suitable representation (a model) of the reconstructed neuron has to be defined and then an efficient algorithm has to be designed which transforms the image into a description based on this representation, i.e. which fits the model to the data. According to the requirements discussed in sec. 2.2 the neuron should be described in a geometric way, that is by a graph connecting the individual segments which are specified by their center lines and their radii. The transformation from voxels to this structural description is mostly done by identifying putative center points and linking them to a graph structure.

2.3.1 Extraction of center points

Two approaches exist for the extraction of center points of tubular objects: (i) segmentation followed by skeletonization and (ii) filtering. The first one is based on a segmentation process which separates the image volume covered by the neuron from surrounding tissue. The segmented "neuron" volume is then skeletonized e.g. by a three-dimensional thinning algorithm (Gerig et al., 1993). This approach, however, is sensitive to noise, since thinning algorithms are very sensitive to jagged surfaces resulting in erroneous branches (Kimmel et al., 1995). Furthermore, simple thresholding fails due to the varying intensity profiles of differently sized neurites.

Lorigo et al. (2001) developed a method following the segmentation approach using a *geodesic active contour model* (Caselles et al., 1997) which is an extension of the classical *active contour models* (Kass, 1988) with level set methods. The model of Caselles et al. (1997) allow for the estimation of hypersurfaces of codimension one[2]. Lorigo et al. modified the geodesic active contours to estimate hypersurfaces of codimension two, e.g. one-dimensional curves in a three-dimensional image space. The implicit representation of the curves is sufficient for visualization, but in order to obtain meaningful measurements center points must be extracted and linked to continuous center lines. This is not part of their method, but should give more reliable results than simple segmentation because of the included smoothness and shape constraints.

The second approach to the extraction of center points applies appropriate filters in order to enhance line elements. Sato et al. (1998) introduced a three-dimensional multi-scale line enhancement filter for medical images. It is based on a combination of the eigenvalues

[2]The codimension is the difference between the dimension of the data space and that of the manifold.

of the Hessian matrix whose elements are the second-order partial derivatives of the image intensities. Curvilinear structures are discriminated from other structures such as point-like or planar ones. The corresponding eigenvectors provide the direction of the axis. Multiple scales allow for different radii, where the discretization of the scales determines the possible radii. Their exact extraction is problematic, since in general the number of scales needs to be small in order to reduce the computational cost.
Another method following the second approach was developed by Pizer et al. (1998) and is based on the concept of *cores* in which detect medial points of the object by correlating opposite boundary points. Due to the spatial extent of the applied filter, the detection of center points is accurate and robust in the presence of interfering noise, but the method is computationally more expensive than the calculation of the Hessian. Generally the filter is applied to the image at several scales, too.

2.3.2 Linking the Center Points

The linking is done either by simply connecting nearby center points or by tracing the center line starting from a seed point. The latter procedure has the advantage that the calculation of filter responses, either the eigenvalues of the Hessian (cf. Sato et al. (1998)) or the medialness core (cf. Pizer et al. (1998)), can be accomplished successively. Under the assumption of smoothness, the calculation can thus be restricted to a region and scale of interest. However, the localized focus has the drawback that contextual information is not available which would be necessary in order to find the correct tracing direction at ambiguous and noisy image locations and to detect branching points.
Krissian et al. (2000) follow the first approach. They use both the abovementioned filtering methods for selecting candidates for central points at multiple scales. Subsequently, ridges[3] are extracted in order to find the central points at the scale with maximal response. The vessels are then reconstructed by successive steps of binarization, thinning, smoothing, and linking center points located at neighboring voxels. The topology of the reconstruction is not robust with respect to noise, since unrecognized center points may interrupt the center lines. This is sufficient for visualizing the data where mistakes in the topology are tolerable.
The other approach which traces the center line is adopted by several methods. Aylward and Bullit (2002) use the eigenanalysis of the Hessian for finding the center points and updating the tracing direction. The optimal scale is estimated dynamically by the evaluation

[3]Ridges are local maxima (here of intensity and scale) with respect to a restricted number of dimensions (Eberly et al., 1994; Haralick, 1983)

of an adaptive core. Finding a seed point for each branch is necessary, since branching points are not detected during segment tracing. Flasque et al. (2001) introduced a tracing method, which works on a segmented image and is able to detect branching points and to follow the branches. The segmentation method does not take into account the prior knowledge about the tubular shape of the vascular branches. Neither of the two methods gives the user the possibility to steer or correct the reconstruction process.

The approaches discussed so far were developed specifically for the segmentation of blood vessels in three-dimensional images from magnetic resonance angiography (MRA) or computer tomography (CT). The following methods, however, focus on the reconstruction of neurons from confocal image stacks. Streekstra and van Pelt (2002) use Gaussian derivative kernels and the eigenvalues of the Hessian. The tracing procedure is not specified, instead they focus on the incorporation of the point spread function of the confocal microscope in order to obtain an unbiased estimate of the radius. (Al-Kohafi et al., 2002), however, developed an elaborate method for the tracing of neurons. The basic idea of the approach is the same as in Aylward and Bullit (2002), but the filter kernel used for extracting the center points and radii is based on simple edge detectors and lacks the theoretical foundation of the abovementioned cores. Hence, the detectable diameter of the branches is limited from below. The method, like the other tracing methods, involve heuristic criteria for stopping the process at the end of a branch and bridging discontinuities due to noise and low contrast.

In summary none of the described methods answer the purpose of an accurate reconstruction of the neuron, since, under the inevitable presence of noise, the correct topology is not ensured. The connectivity particularly of the thin processes is often not inferrible from the filter responses. This weakness is critical, if the result cannot be corrected due to the lack of user interaction (Olabarriaga and Smeulders, 2001) which applies to all of the methods. Furthermore, Burl et al. (1998) showed that the "hard" detection strategy which is used by all the methods yields deficient results. A hard detection strategy keeps only the *positions* of the points where the filter response exceeds a threshold, whereas the *value* of the response is discarded. After finding the best candidates for center points, either their location is kept fixed resulting in a center line which is not optimized with respect to its shape, or the center line is smoothed neglecting the data accuracy. Simultaneously optimizing the filter responses and the shape of the object, however, is shown to be a better strategy. Sometimes a smoothness prior is incorporated in the tracing methods (Al-Kohafi et al., 2002; Aylward and Bullit, 2002; Flasque et al., 2001), but these methods suffer from the local focus of the tracing procedure neglecting context information, and thus, still fail to reconstruct regions with low contrast and consequently the following

part of the dendritic tree. The only method available so far which employs a real "soft" detection strategy is that by Lorigo et al. (2001). However, it provides an implicit representation of the neuron without connectivity information, and thus does not allow for immediate user interaction.

2.3.3 Semi-automatic Extraction of Center Lines

In order to avoid the problems of reconstructing the correct connectivity, we pursue a *model-based* approach. A model of the neurite is initialized to lie near the neurite in the image and subsequently fitted to the data. The initialization can either be done manually by the user or alternatively by an automated method, although the latter suffers from the abovementioned difficulties. Such a top-down approach incorporates prior knowledge on tree structure, and thus, overcomes common constraints in localizing center points.
Semi-automatic methods require the input of the two end points of a center line by the user. Then the axis and width of the neurite between them is reconstructed. Frangi et al. (1999) use this scheme for the reconstruction of human *carotid arteries* from MRA. The model-based method incorporates knowledge of the physics of the aqcuisition technique to accurately segment the vessel geometry. The user provides the two end points, then the center line is reconstructed by evaluating the Hessian matrix at multiple scales, and finally the vessel wall is reconstructed. Since the method is tailored for a specialized application to MRA it lacks a data structure to represent complex vascular or dendritic trees.
We therefore developed a framework for the reconstruction of neurons from confocal image stacks which automatically determines the center line and radii of a neuron's segment using start and end point for initialization, employing a "soft" detection strategy, and thus yielding results which combine voxel-based image data with shape priors. It can be used as stand-alone reconstruction tool or can be combined with fully automatic reconstruction methods. When automated methods fail, it can be used to reconstruct unrecognized parts of the neuronal tree with reasonable time investment. Fully automatic reconstruction of recognized parts of the neuronal tree also can be revised by our semi-automatic method, resulting in a reconstruction adapted to original image data at highest accuracy. To the best of our knowledge it provides the only existing semi-automatic reconstruction tool which can be applied to the most complex dendritic trees and a wide variety of confocal image qualities.

2.4 Methods

The *skeleton* of the neuron is reconstructed as follows. The branching and the end points and, if necessary, some points in between are set manually by the user. This initialization roughly approximate the shape of the neuron and provides its topology. It is then fitted to the data with the automatic methods described in section 2.4.1. Initialization and fitting of segments can be iterated. Due to a low signal-to-noise ratio or an improper initialization and parameter setting, the fitting may produce incorrect or deficient results. Therefore, the skeleton can be corrected manually by moving, adding, or deleting nodes at any stage if necessary.

In order to obtain a more detailed segmentation for exact morphometric measurements a *surface* of the neuron can be fitted afterwards. This representation of the neuron is not restricted to the circular cross-section, but lacks the explicit, structural description in terms of center lines, radii, and segments. Hence it gives a better estimate of the volume and surface area, but cannot be used directly for computational models. However, it can be used indirectly by assigning each part of the surface to the corresponding part of the skeleton. This way it is possible to correct the radii of the latter with respect to either the surface area or the volume improving the accuracy of compartment models for computational analysis. The processing flow of the proposed method is described in Fig. 2.2.

2.4.1 Skeleton and Generalized Cylinders

We assume a tubular shape for the neurites: elongated and with circular cross-section. The circular cross-section is an oversimplification, but it reduces the computational costs of the reconstruction process[4]. A neuron is described by a set of tubular segments which are connected to a tree. Branches are not interrupted by gaps. This is critical, because the staining of a cell is typically not uniform and gaps may occur. Furthermore the shape is regularized by a smoothness constraint for the tangents of the center lines and the radii. Our reconstruction is based on the model of generalized cylinders (Binford, 1987), which are generated by sweeping a two-dimensional cross-section along an axis in three-dimensional space. Shape and size of the cross-section as well as the direction of the axis changes with location. We restrict the cross-section to be circular and always perpendicular to the tangent of the axis. Since generally neurons are branching structures several generalized cylinders attached to each other at branching points (see fig. 2.3) are necessary to describe

[4]It may, however, introduce errors in the estimation of the neuron's thickness which can be corrected using the surface reconstruction (see above).

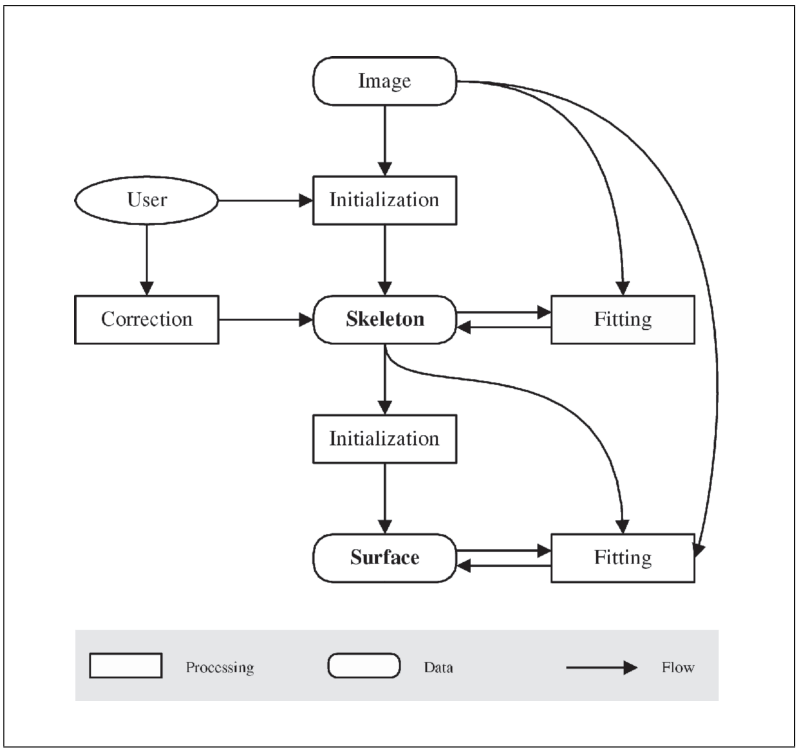


Figure 2.2: The processing flow. The user initializes the skeleton by tracing the neuron roughly (for details see sec. 2.5.1). After fitting the skeleton to the data (see sec. 2.4.1) the user can correct the result if necessary. The fitted skeleton serves as initialization for the detailed surface reconstruction which is then fitted to the data (see sec. 2.4.2). The fitting of the surface is regularized by the skeleton.

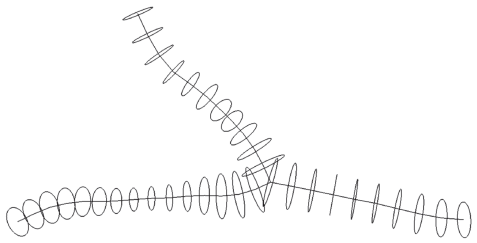


Figure 2.3: A skeleton formed by three generalized cylinders attached to each other. The tubular structure is defined by a circular cross-section which is swept along an axis. It is parametrized by the medial axis and a radius at every point.

the tree. In this work we call such a structure a *skeleton*[5].

We use the *active contour model* (also referred to as *snake*) by Kass (1988) for fitting the skeleton to the data. The model was originally developed to fit just the shape of a curve, that is the axes of the generalized cylinders. We must extend the model in order to also fit the radius to the image data. Here we suggest to estimate the center line and

[5]Note that our description of the neuron does not satisfy the widely used definition of skeletons by Blum (1967).

the radius simultaneously, because fitting of the center line requires a good estimation of the radius and vice versa.
The topology of the skeleton, i.e. the way the branches are connected, is set by the user and remains fixed during the automatic fitting process. The quality of the fitted skeleton is evaluated using an energy functional. The energy is small, if the skeleton meets the assumptions that (i) *bright voxels* indicate the presence of the neuron and (ii) the curvature of the axis and the change of the radii along the axis are *smooth* to a certain extend. Thus the minimization of the energy fits the skeleton to the local image evidences while preserving a certain degree of smoothness.

Medialness

Since the neural structures are modeled as generalized cylinders with circular cross-sections, the medial points – being centers of these circles – as well as the radii at each of these points must be found. The circles lie in the plane which is orthogonal to the medial axis. In order to obtain a robust measure of the *medialness* of an axis point, we define two medialness functions: one for the offset medialness and one for the central medialness (Pizer et al., 1998). Both functions evaluate the medialness with respect to the tangent of the axis. The merits and drawbacks of the two measures are discussed in section 2.5.3.
The *offset medialness* is based on the change of image intensities of points along the circumference of the generalized cylinder. The boundary between object and background at position $\vec{x}$ is characterized by a rapid change of intensities along a direction $\vec{o}$. The Laplace-operator – a common edge detector – locates the boundaries at the local maxima of the gradients, i.e. at the zero-crossing of the second derivative in gradient direction. In this sense we define

$$B(\vec{x}, \vec{o}) = -\vec{o}^T \cdot \nabla I(\vec{x}) \tag{2.1}$$

as the change of the image intensities I measured along the direction $\vec{o}$ (see fig. 2.4). The offset medialness of a position $\vec{x}$ with respect to a radius r and the plane spanned by $\vec{n}_{1,2}$ is then defined by

$$M_O(\vec{x}, r, \vec{n}_1, \vec{n}_2) = \frac{1}{r} \int_0^{2\pi} B(\vec{x} + r\vec{o}(\theta), \vec{o}(\theta)) d\theta, \tag{2.2}$$

where $\vec{o}(\theta) = \cos(\theta)\vec{n}_1 + \sin(\theta)\vec{n}_2$. $\vec{n}_{1,2}$ are two orthogonal vectors of unit length. The offset medialness is high, if strong centripetal gradients lie in a given plane at a given distance to the center. This measure is not affected by gradients lying inside or outside the circle.

The *central medialness* evaluates not just the boundary of a potential generalized cylinder but also the image intensities lying inside and outside this circle. We define a two-dimensional kernel $K(x, y, r)$,

$$\begin{aligned} K(x,y,r) &= -r^2(G_{xx} + G_{yy}) && (2.3)\\ &= (2-R)e^{-\frac{R}{2}}, && (2.4) \end{aligned}$$

where G_{xx} and G_{yy} are the second partial derivatives of the Gaussian function, $R = (x^2+y^2)/r^2$ and r is the radius of the circle. The medialness of a position $\vec{x}$ with respect to a radius r and the plane spanned by $\vec{n}_{1,2}$ is defined by the linear convolution

$$\begin{aligned} M_C(\vec{x}, r, \vec{n}_1, \vec{n}_2) &= K * I && (2.5)\\ &= \int\int K(x,y,r) I(\vec{x} + x\vec{n}_1 + y\vec{n}_2) dxdy. && (2.6) \end{aligned}$$

Due to the Mexican hat type kernel K, strong medialness is given if the interior and exterior of the potential boundary of the generalized cylinder is covered by bright and dark voxels, respectively. Note, that only relative intensity values matter, since maximization of the medialness with respect to the center $\vec{x}$ is independent of its absolute value. This feature is essential, because the scanning of differently sized processes yields different intensities (see sec. 2.2).

In order to compute the medialness of an image location we approximate the integrals in the medialness functions by sums over m equally spaced points. We obtain:

$$M_O(\vec{x}, r, \vec{n}_1, \vec{n}_2) = \frac{1}{m} \sum_{i=1}^{m} B(\vec{x} + r\vec{o}_i, \vec{o}_i) \tag{2.7}$$

and

$$M_C(\vec{x}, r, \vec{n}_1, \vec{n}_2) = \frac{1}{m^2} \sum_{i=-m}^{m} \sum_{j=-m}^{m} K(x_i, y_j, r) I(\vec{x} + x_i\vec{n}_1 + y_j\vec{n}_2), \tag{2.8}$$

where $\vec{o}_i = \vec{o}\big(2\pi\frac{i}{m}\big)$ (see eq. (2.2)), $x_i = 3r\frac{i}{m}, y_j = 3r\frac{i}{m}$ and where $K(x, y, r)$ was set to 0 for $\sqrt{x^2+y^2} > 3r$. The number m of points is chosen such that the distance between adjacent points is approximately one voxel and that $m \geq 8$. Intensity values between the grid points of the image lattice are calculated by trilinear interpolation.

Combining both measures of medialness increases the robustness of the fitting procedure, because they exploit somewhat complementary features of the object. The central medialness evaluates the intensity profile, while the offset medialness makes use of edge information.

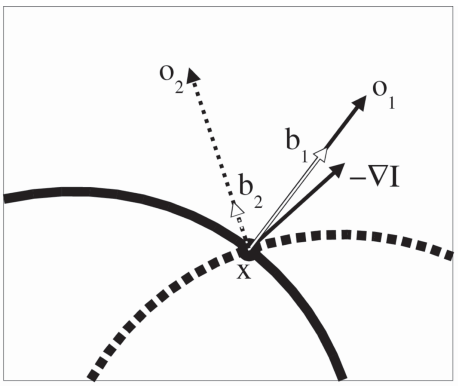


Figure 2.4: The boundariness defined in eq. (2.1). The figure shows two different circular cross-sections (bold lines) defining the radial directions $\vec{o}_{1,2}$ at position $\vec{x}$ and the (negative) image gradient $-\nabla I$. The length of the vectors $\vec{b}_{1,2}$ corresponds to the boundariness measures $B(\vec{x}, \vec{o}_{1,2})$. Depending on the center of the cross-section the same location gives different values for B.

Snakes

The skeleton is composed of a number of segments, each being modeled as a generalized cylinder. The generalized cylinders are parameterized by a set of points $V = \{i : 1 \dots n\}$ (called *snaxels*). Each snaxel is characterized by its location $\vec{x}_i \in \mathbb{R}^3$ in the image, by its radius $r_i \in \mathbb{R}$, and by its neighborhood $N_i \subset V$. The shape of the generalized cylinder is obtained by linear interpolation between adjacent snaxels. Most of the snaxels have two neighbors, except those at the branching points which have three or more neighbors[6], and those at the endings of the branches which have one. We first consider the case that all snaxels have one or two neighbors and that the neighborhood of snaxel i is given by $N_i = \{i-1, i+1\}$ or $N_i = \{i \pm 1\}$. The treatment of the branching points will be explained afterwards.

The snake algorithm fits the skeleton to the data by minimizing an energy functional. It is defined for the whole skeleton as the sum of an external and an internal energy over all snaxels,

$$\widehat{E}_{\text{snake}} = \sum_{i=1}^{n} E_{\text{snake}}(i) = \sum_{i=1}^{n} \left(\alpha E_{\text{ext}}(i) + (1-\alpha) E_{\text{int}}(i)\right). \tag{2.9}$$

n denotes the number of snaxels and $\alpha \in [0, 1]$ is an adjustable weight for controlling the relative importance of the terms.

The external energy is data driven. Its minimization attracts the snaxels (the medial axis of the generalized cylinder) to the medial points of the neural structure and adjusts the corresponding radii. It is defined as the weighted sum of the central and the offset medialness,

$$E_{\text{ext}}(i) = -(\lambda M_C(i) + (1-\lambda) M_O(i)), \tag{2.10}$$

where $M_{C,O}(i) = M_{C,O}(\vec{x}_i, r_i, \vec{n}_{i1}, \vec{n}_{i2})$ is the medialness of the snaxel $i \in \{1 \dots n\}$ at

[6] Bi-, tri-, and multifurcating points exist, and can be reconstructed by our method.

location $\vec{x}_i$ with radius r_i according to equations (2.7) and (2.8). $\vec{n}_{i1,2}$ span the plane which is orthogonal to the tangent of the generalized cylinder at this location. The parameter $\lambda \in [0,1]$ is the weighting factor.
The internal energy is defined as the sum of the first derivatives (i.e. the length of the tangent vectors) of the skeleton. Minimizing this energy gives the snake an elastic behaviour which reduces its curvature and thus smoothes the curve. The radii are aligned as well. For snaxels with one or two neighbors the internal energy is given by

$$E_{\text{int}}(i) = \begin{cases} \frac{1}{4}\left(\beta\left(\|\vec{x}_i - \vec{x}_{i-1}\|^2 + \|\vec{x}_i - \vec{x}_{i+1}\|^2\right)\right. \\ \quad \left. + (1-\beta)\left((r_i - r_{i-1})^2 + (r_i - r_{i+1})^2\right)\right) & : \ |N_i| = 2, \\ \frac{1}{2}\left(\beta\|\vec{x}_i - \vec{x}_{i-1}\|^2 + (1-\beta)(r_i - r_{i-1})^2\right) & : \ |N_i| = 1, \end{cases} \tag{2.11}$$

where N_i is the neighborhood of snaxel i with snaxels at location $\vec{x}_{i\pm1}$ and with radii $r_{i\pm1}$. $\beta \in [0,1]$ is a weighting parameter. Since the internal energy is minimal for minimal distances between the snaxels, the skeleton tends to collapse to one point. This is prevented by constraining the movements of the snaxels to the plane perpendicular to the skeleton. If a snaxel has more than two neighbors, namely at the branching points, the energies are evaluated with respect to each pair of neighboring snaxels and averaged. In order to give more influence to the neighbors belonging to thicker branches, the energy values are weighted by the radii of the respective neighbors,

$$E(i) = \sum_{\{k,l\} \in N_i} \frac{r_k + r_l}{Z} E_{\text{snake}}(i)|_{\{k,l\}}, \tag{2.12}$$

where $Z = 2\sum_{k \in N_i} r_k$ is a normalization factor and $E_{\text{snake}}(\cdot)|_{\{\cdot,\cdot\}}$ denotes the energy with respect to the two given neighbors.
The energy functional given by equation 2.9 is minimized using an iterative gradient descent method which minimizes $\widehat{E}_{\text{snake}}$ with respect to both the locations and the radii (see appendix 2.7.1). Local minima are avoided by a sufficiently accurate initialization which has to be provided by the user who sets the tree nodes.

2.4.2 Surface Reconstruction: Correction for Non-Circular Cross-Sections

If a precise reconstruction of the neuron surface is required, the approximation by generalized cylinders is no longer sufficient. Here we suggest to improve the surface description by using a flexible model based on *geodesic active contours* by Caselles et al. (1997) which,

however, is regularized by the generalized cylinder approximation.
The surface is implicitly defined by a level set of a function $u : \mathbb{R}^3 \to \mathbb{R}$: all points in image space which are mapped to a certain value (e.g. $u(\vec{x}) = 0$) belong to the surface. The fitting process then adjusts the mapping u according to the data and an appropriate smoothness constraint.

Geodesic Active Contours

The geodesic active contours method detects boundaries in three-dimensional space by computing surfaces of minimal area (also called *geodesics*). The area of a two-dimensional surface $\mathcal{S} : \mathbb{R}^2 \to \mathbb{R}^3$ in the three-dimensional Euclidean space is given by

$$A := \int\int d\vec{a}, \tag{2.13}$$

where the surface is parametrized by $\vec{a} \in \mathbb{R}^2$ and $d\vec{a}$ is the Euclidean area element. Surfaces minimizing eq. (2.13) are called *minimal surfaces*.
The fastest way to minimize the area of a surface is the *mean curvature flow* or *Euclidean shortening flow* which is the Euler-Lagrange of A (Chopp, 1993). The curve evolution equation is given by

$$\mathcal{S}_t(\vec{a}) = H(\vec{a})\vec{n}(\vec{a}), \tag{2.14}$$

where the subscript t denotes the derivative with respect to t. H is the mean curvature and $\vec{n}$ is the inward unit normal of the surface. This way the surface moves outward where it is concave and its mean curvature is negative, and it moves inward where it is convex and its mean curvature is positive.
Now let the area be "weighted" at location $\vec{x} = \mathcal{S}(\vec{a})$ according to the image evidences $I(\vec{x})$,

$$\widehat{A} := \int\int g(\mathcal{S}(\vec{a}))d\vec{a}. \tag{2.15}$$

The function g usually depends on the edges in the image:

$$g(\vec{x}) = f(|\nabla I(\vec{x})|), \tag{2.16}$$

where f is a decreasing function, such that $f(r) \to 0$ as $r \to \infty$. Here the function

$$f(x) = \frac{1}{1+x} \tag{2.17}$$

is used. Thus the larger the magnitude of the gradient at the location of a surface element,

the less it contributes to the measured area. The surface which minimizes eq. (2.15) is also called a minimal surface. In order to increase the robustness of the surface reconstruction in particular that of the thin structures an intensity-based term is added:

$$g(\vec{x}) = \lambda f(|\nabla I(\vec{x})|) + (1-\lambda)|I(\vec{x}) - t|. \tag{2.18}$$

λ is a weighting parameter and t is a local threshold which provides intensity information independent from the gradients. The threshold is calculated by averaging the intensities along the circle with center and radius of the snaxel which is closest to $\vec{x}$ and lying in the plane which is perpendicular to the axis of the generalized cylinder at that snaxel. The threshold changes with location, because the central as well as the offset medialness used to fit the snaxels depend just on differences of intensity values.
The Euler-Lagrange of $\widehat{A}$ is then given by (Caselles et al., 1997):

$$\mathcal{S}_t = (gH - \nabla g \cdot \vec{n})\,\vec{n}, \tag{2.19}$$

where the arguments $\vec{a}$ and $\vec{x}$ have been omitted for clarity. Still the Euclidean mean curvature will be reduced by the first term on the right hand side of eq. (2.19) but the speed is controlled by the magnitude of the image gradient. Thus the shortening flow is reduced in the presence of edges. The second term pushes the surface towards the edges except for those lying exactly orthogonal to the surface. $-\nabla g$ points in the direction of image gradients with larger magnitude and $\vec{n}$ is the inward normal of the surface (see fig. 2.5).
In order to formulate the weighted curvature flow in level set notation the evolving surface $\mathcal{S}$ is defined as the zero level set of the function $u(\vec{x}) \in \mathbb{R}$ which is the set of all points for which $u = 0$. The weighted shorting flow for the level set minimizing $\widehat{A}$ is then

$$u_t = |\nabla u| \operatorname{div}\left(g\frac{\nabla u}{|\nabla u|}\right) \tag{2.20}$$

$$= g\,|\nabla u| \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) + \nabla g \cdot \nabla u. \tag{2.21}$$

Closed contours, however, tend to collapse under the influence of the shortening flow. Caselles et al. (1997) therefore proposed an additional constant velocity acting like a "balloon force" in the classical active contour model (Cohen, 1991) in order to avoid this

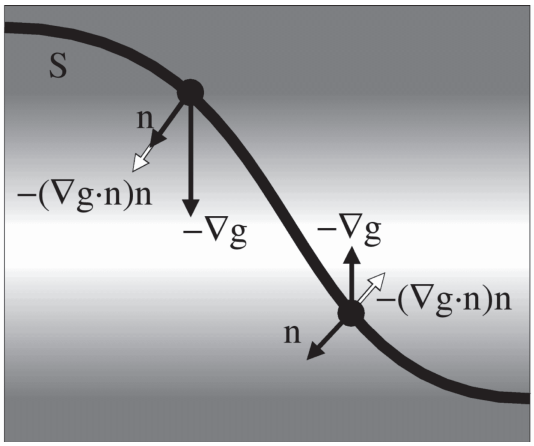


Figure 2.5: The effect of the term $-(\nabla g \cdot \vec{n})\vec{n}$ in eq. (2.19) is illustrated in this figure for a two-dimensional image. The extension to three dimensions is straightforward. The current estimate of the boundary $\mathcal{S}$ is shown as a black line and the edge map $|\nabla I|$ is shown as gray values. White denotes high values of the gradient and small values of g according to eq. (2.18). Black arrows show the normal $\vec{n}$ of the boundary and the direction of highest increase of the edge map: $-\nabla g$. We illustrate the result of $-(\nabla g \cdot \vec{n})\vec{n}$ using white arrows at two different locations of $\mathcal{S}$ marked by solid circles. Both points are to be moved along the normal direction towards the edges.

behaviour. Thus the evolution equation (2.20) is supplemented by an "bias term" $g|\nabla u|$:

$$u_t = |\nabla u| \operatorname{div}\left(g \frac{\nabla u}{|\nabla u|}\right) + cg|\nabla u| \tag{2.22}$$

$$= (c + H)g|\nabla u| + \nabla g \cdot \nabla u, \tag{2.23}$$

where c is a constant which determines the strength of the balloon force. The term $cg|\nabla u|$ prevents the contour from collapsing due to the shortening flow.

The surface of tubular structures is characterized by a high curvature in the direction lying perpendicular to the axis of the tube. This curvature should not contribute to the shortening flow. Hence we incorporate the model of tubular shape by chosing the factor c to be non-constant and depending on the curvature of the boundary of the generalized cylinders:

$$c = -\operatorname{div}\left(\frac{\nabla d}{|\nabla d|}\right), \tag{2.24}$$

where $d(\vec{x})$ is the signed distance to the nearest point lying on the boundary of the generalized cylinders, with positive values outside and negative values inside the cylinders. d is called the signed distance map (note that $|\nabla d(\vec{x})| = 1$ holds in general).

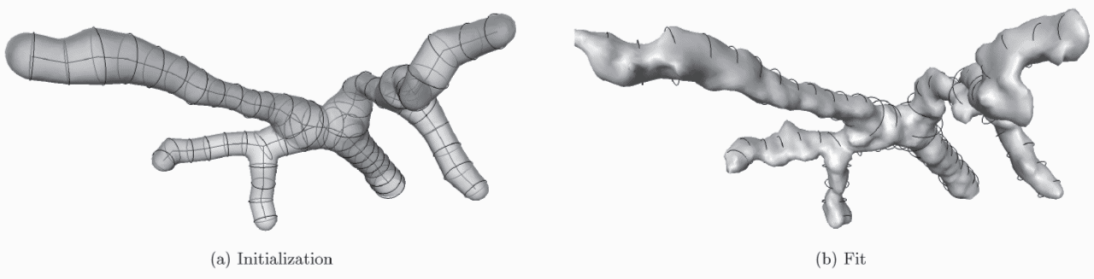


Figure 2.6: Application of eq. (2.27) to the surface reconstruction of a small part of the data set described in sec. 2.5. (a) u is initialized by the signed distance map d of the generalized cylinders which were fitted by the snake algorithm. The figures show the generalized cylinders (solid lines and circles) and the initial zero level set of d as transparent surface. (b) The surface given by the zero level set of u after fitting u to the data using eq. (2.27). Shown are the generalized cylinders and the final (opaque) surface.

The factor c acts as a shape regularizer. It compensates the contributions to the shortening flow which minimize the radius of the cross-section of the neurites leading to a collapse of the surface to a one-dimensional curve. Thus equation (2.21) becomes

$$u_t = g\,|\nabla u|\left(\operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) - \operatorname{div}\left(\frac{\nabla d}{|\nabla d|}\right)\right) + \nabla g \cdot \nabla u \tag{2.25}$$

$$= g\,|\nabla u| \operatorname{div}\left(\frac{\nabla u}{|\nabla u|} - \frac{\nabla d}{|\nabla d|}\right) + \nabla g \cdot \nabla u. \tag{2.26}$$

In order to control the influence of both terms in eq. (2.26) separately, a weighting parameter α is introduced:

$$u_t = (1-\alpha)\, g\,|\nabla u| \operatorname{div}\left(\frac{\nabla u}{|\nabla u|} - \frac{\nabla d}{|\nabla d|}\right) + \alpha\, \nabla g \cdot \nabla u. \tag{2.27}$$

The mapping u is initialized with the signed distance map induced by the skeleton (see fig. 2.6 for an example).

2.5 Results

We applied the neuron reconstruction technique described in this paper to stacks of confocal images of cultured astrocytes, sensory neurons, inter-, and motorneurons. Fig. 2.7(a) shows a maximum intensity projection, and Fig. 2.7(b) and (c) the reconstruction of a

dendritic tree of a motorneuron (MN5) of a *Manduca sexta*[7].

The image in fig. 2.7(a) consists of seven tiles, viz. maximum intensity projections of seven aligned confocal image stacks. The voxel size is $0.1 \times 0.1 \times 0.3\,\mu m^3$. The skeleton was reconstructed by loading successively image stacks covering different subvolumes into the software. This way, large dendritic trees can be reconstructed at highest resolution without a shortage of working memory. The memory usage for the fitting procedure applied on the whole skeleton is very small compared to the memory demands of the image stacks, and therefore, causes no problems. For the surface reconstruction, however, memory usage amounts to the tenfold of the original dataset due to floating point precision and implicit representation of the surface. To circumvent memory limitations the calculations can be run on separate tiles successively.

The number of snaxels amounts to 16578 with an average spacing of about 0.5 μm which allows smooth changes of axis direction and radius. The total length of the reconstructed skeleton is 8323.9 μm, its radii range between 0.1 and 3.5 μm. It has 1022 branching points, 1078 end points, and 2999 segments. We had to set approximately 1500 of 16578 points, i.e. 9%, manually. The tracing procedure is described in detail below. The numbers alone suggest a rather moderate saving of effort, but most of the reduction results from the automated fitting which delivers the user from specifying the radii and corrects inaccurate input. Hence the reconstruction process is accelerated by reducing both the necessary quantity and quality of the input while producing results of highest accuracy. Time investment for defining the branching points of the skeleton strongly depends on the user's experience and demand on completeness. However, to define the 1500 points necessary for the reconstruction of the cell shown in fig. 2.7 took a very experienced user, who sought the highest level of completeness including reconstruction of fine filopodia, one working day. In comparision, the same user needed three hours for a reconstruction restricted to dendrites thicker than $0.3\mu m$. The time needed for the calculation of surface reconstructions depends on the size of the original image stacks. In case of the surface reconstruction shown in Fig. 2.7(c), the computing time was 80 minutes on a standard high-end personal computer, whereas the underlying image data has a size of 780 MB. However, during calculation of surface reconstructions no user interaction is required.

A part of a dendritic tree was reconstructed to evaluate the user invariance (see sec. 2.5.2). This exemplifies the time investment for the processing of simpler structures. It has 123 segments and 520 snaxels. Averaged over three users, the reconstruction time was 20 minutes.

[7] The manduca (*Tobacco hornworm moth*) belongs to the family of the sphings moths which are well known to the layman from the movie "The Silence of the Lambs" (USA, 1991).

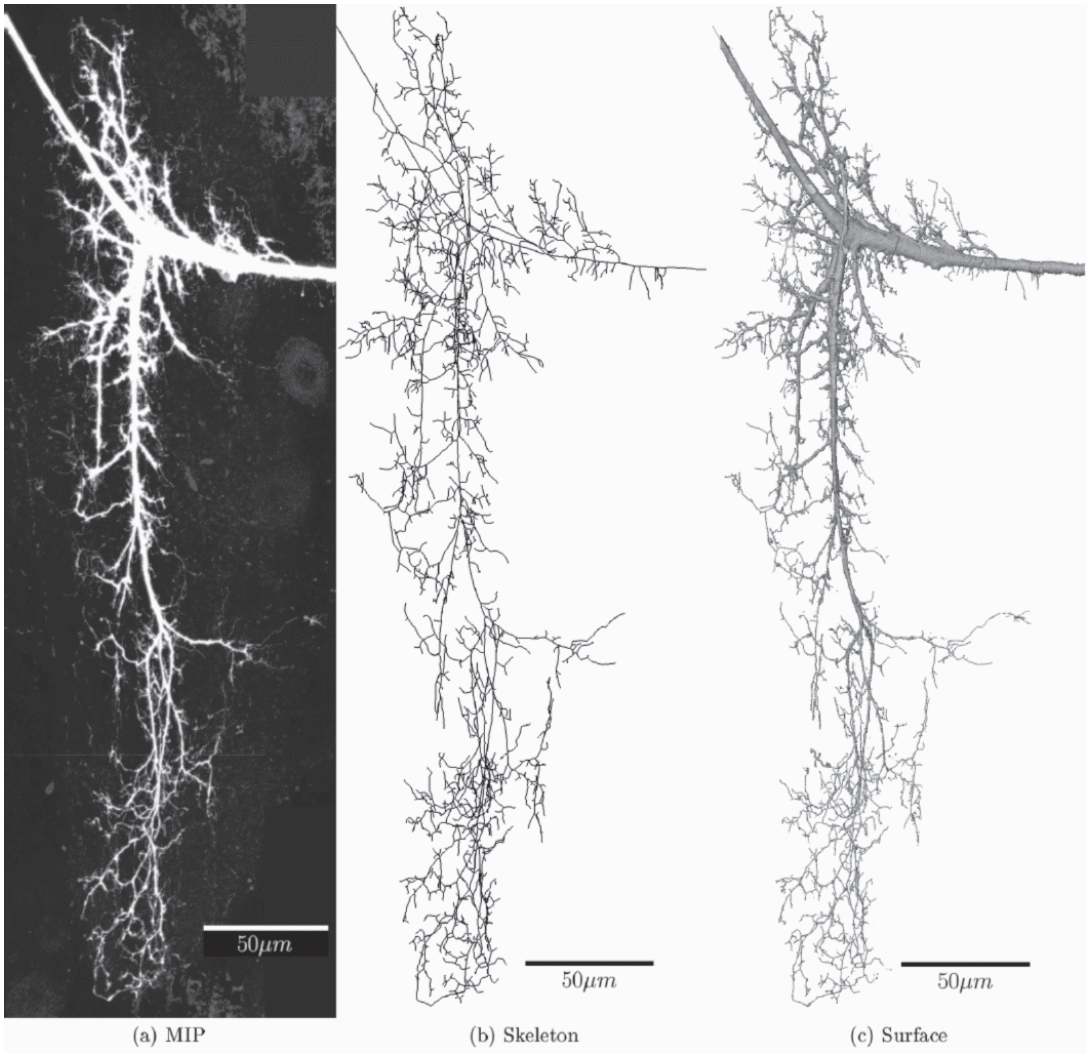


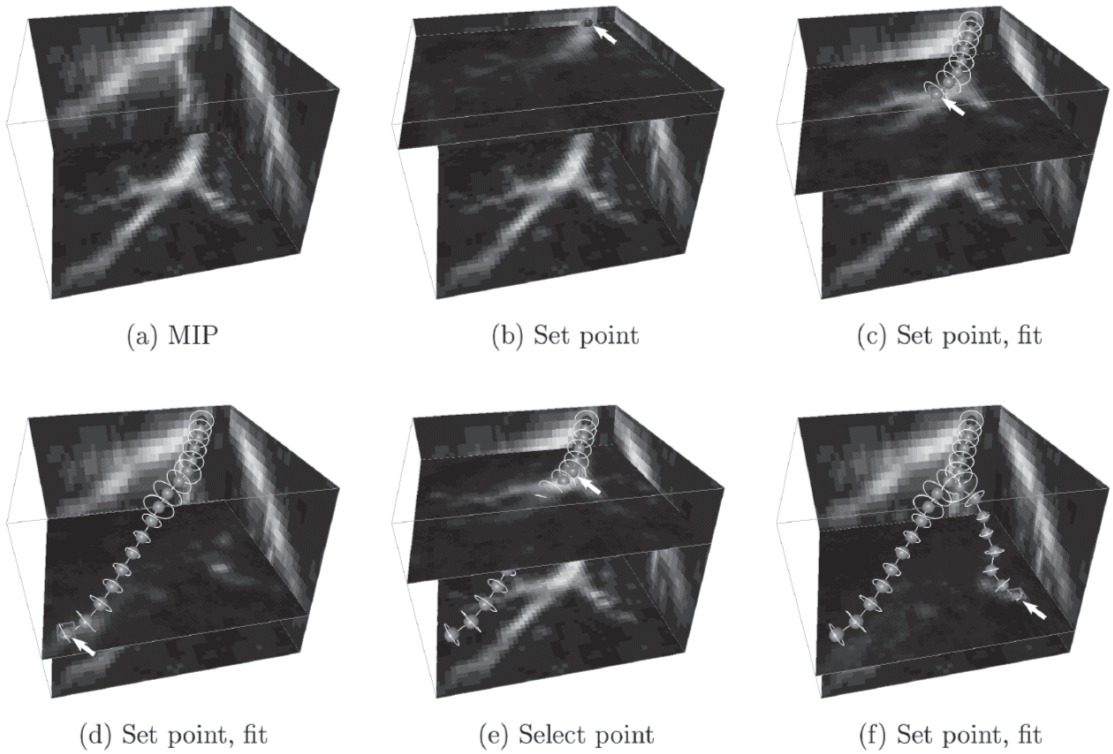
Figure 2.7: The dendritic tree of the Motorneuron 5 of a *Manduca sexta*. The images were captured on a Leica SP2 laser scanning confocal microscope using a 40× oil immersion lens (NA 1.25). The excitation maximum of the laser was at 543 nm and the detection range was 565-600 nm. The voxel size is $0.1\times0.1\times0.3\,\mu m^3$. Scalebar: $50\mu m$. (a) Data shown in a maximum intensity projection. (b) Reconstruction of the skeleton fitted using the active contour model which minimizes eq. (2.9) (parameters: $\alpha = \lambda = 0.5$). The generalized cylinders are indicated by their center lines. The radii are not shown. (c) Surface fitted using the geodesic active contour model of eq. (2.27) (parameters: $\alpha = 0.86$, $\lambda = 0.17$).

2.5.1 Tracing Procedure

The tracing procedure is illustrated in Fig. 2.8. The user traces the dendritic tree iteratively selecting existing snaxels as the starting points of the piece of the skeleton to be added. If the selected snaxel is an end point of a reconstructed segment, the segment will be extended (see fig. 2.8(d)), otherwise the snaxel becomes a branching point and a new segment will be attached to it (see fig. 2.8(f)). To provide the end point of the new piece of the skeleton, the user clicks at the appropriate location in the image. Three-dimensional input coordinates are specified by clicking on a rendered isosurface of the neuron or an image slice which can be dragged through the volume. A new snaxel will be created at the given location and will be connected to the selected one by automatically inserting intermediate snaxels on a straight line with a predefined stepwidth. The computing time for the automatic connection procedure (*snaxelization*) shown in fig. 2.8 is negligible. Afterwards the newly created piece of skeleton can be fitted to the data using the snake algorithm described in section 2.4.1 by minimizing equation (2.9). This way the user can check the resulting fit immediately after the creation of the new piece. Alternatively, the user can set several snaxels, before the fitting procedure is applied to reconstruct the skeleton pieces in between.
The number of iterations which are necessary for the convergence of the snake algorithm depends on the noise and the bending of the neural branch, which is to be fitted. A high noise level requires small update steps in order to avoid oscillatory behaviour resulting in a large number of iterations. Strong bending of a neurite between the start and the end point chosen by the user causes the (linear) initialization of the snake to deviate more strongly from the actual shape of the neurite. This increases the number of iterations as well. Convergence does not introduce a noticeable delay for segments with 20 sampling points and an diameter covering 50 voxels. Increasing the voxel number to 400 introduces a delay of about 10 seconds. The computing time increases with voxel number (linearly for offset medialness and quadratically for central medialness), but can be accelerated by subsampling the image before. Generally the optimal solution is found by the algorithm as long as parts of the initialized center line lie inside the neural structure. Snake-fitting time is by far shorter than the time needed for user interaction.

2.5.2 User Invariance

In order to evaluate the reduction of the user variance which is achieved by our method, three users independently reconstructed the same dendritic tree both manually and with our new method.



(a) MIP (b) Set point (c) Set point, fit

(d) Set point, fit (e) Select point (f) Set point, fit

Figure 2.8: The tracing procedure shown for a detail. The image slice in subfigures (b-f) helps the user to locate the neurites in three-dimensional space and to specify input coordinates by clicking on it. (a) A maximum intensity projection of the data. (b) The user sets the first point (marked by an arrow) which is automatically selected (selected points are highlighted with a wireframe cube), by clicking at the image slice. (c) The user adds a point. The new point is automatically connected with the point which was selected before by inserting additional points. The new points are fitted to the data. (d) The user adds a point. All new points are fitted to the data. (e) The user selects a point (marked by an arrow) by clicking on it to continue the tracing procedure there. Since it is not an end point, it will become a branching point. (f) The user adds a point. All new points are fitted to the data.

The upper row in Fig. 2.9 shows the resulting three manual reconstructions. The user builds the skeleton by defining linear pieces and has to estimate their radii, mimicking the Neurolucida reconstruction procedure. Note that the accuracy of a reconstruction does not depend on the number of snaxels used to represent a linear piece, since the snaxels in between are just linearly interpolated. The quality of these reconstructions is comparable with that of Neurolucida reconstructions. The semi-automated reconstructions shown in the lower row of Fig. 2.9 were produced as described above. The figures in the rightmost column show the center lines of all three skeletons together for the manual

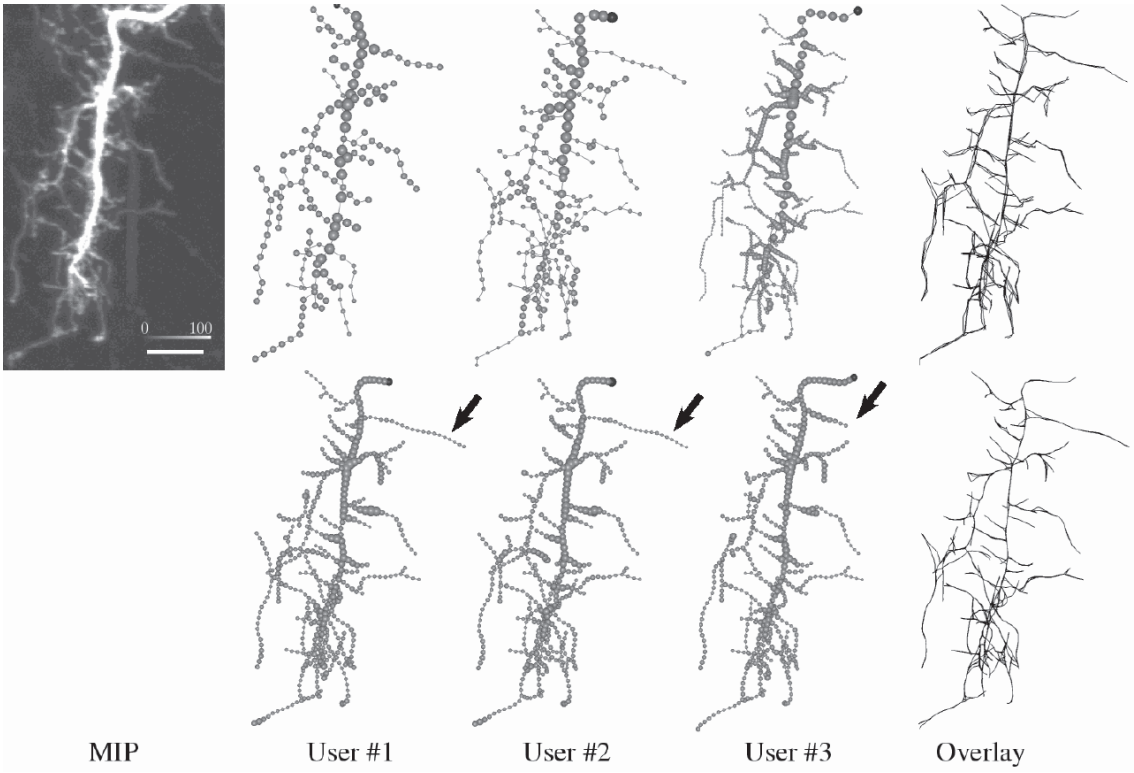


Figure 2.9: Three different users reconstructed a detail of a dendritic tree. It is from the MN5 of a *Manduca sexta* and shown at the top left as a maximum intensity projection. Scalebar: $5\mu m$. The top row shows manual and the bottom row shows semi-automated reconstructions. The snaxels are represented by balls of the same radius as the skeleton at this location. The rightmost column shows the three skeletons together without snaxels to compare the congruence between the results. The arrows indicate a segment which was reconstructed differently by the three users.

(upper row) as compared to the semi-automatic reconstruction (lower row). Obviously the semi-automated reconstructions exhibit a much stronger congruence than the manual ones. Differences between the former occur mainly with respect to the topology, i.e. the lengths and number of segments (see arrows in fig. 2.9). These differences are not caused by the fitting but depend on the user's data interpretation. Therefore, the fitting ensures a high user invariance with regard to center line and branching point locations and segment radii. In contrast, in manual reconstructions higher user variance occurs with regard to these quantities.

For better visualization of user dependent variance the mean radii of the segments are plotted in ascending order for the three manual (fig. 2.10(a)) and the three semi-automated reconstructions (fig. 2.10(b)).

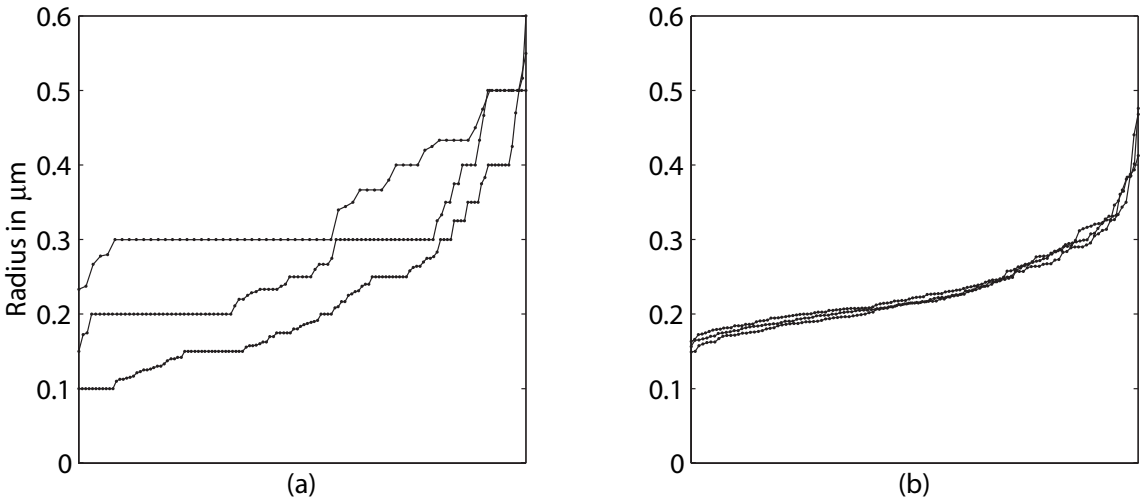


Figure 2.10: The mean radii of the segments plotted in ascending order. Since the reconstructions have different numbers of segments, the x-axis was normalized. (a) The radii of three manual segmentations by different users. (b) The radii of three semi-automated reconstructions by the same users.

Since the number of segments varies, the spacing between the data points is chosen such that the curves are normalized with respect to the x-axis. The estimates of the users differ to a high degree, while those based on medialness measures introduced in section 2.4.1 are very similar.

2.5.3 Skeleton

The maximization of the two medialness measures M_O and M_C (see eqs. (2.7, 2.8)) yields slightly different results for each. In particular the estimates of the radii based on the central medialness M_C are larger than the values obtained for the offset medialness M_O. M_O is dominated by the high values of the gradient in x- and y-direction, while M_C is more influenced by the blurred intensity values in z-direction (see fig. 2.11), while M_O neglects the data in z-direction due to the lack of distinctive maxima of the gradient magnitudes in that direction. It depends on the confocal images which of the two measures is the more appropriate one. If the image is not deconvolved, the offset medialness is preferable. Because the blurring in axial direction is an artefact of the point spread function (PSF) of the confocal microscope, it is reasonable to rely on the estimation of the radius in the x-y-plane and then to adapt this value in z-direction. Here it is done by the restriction to a circular cross-section of the generalized cylinders.

M_C, however, is more robust for very thin structures covering just one or two voxels in x-

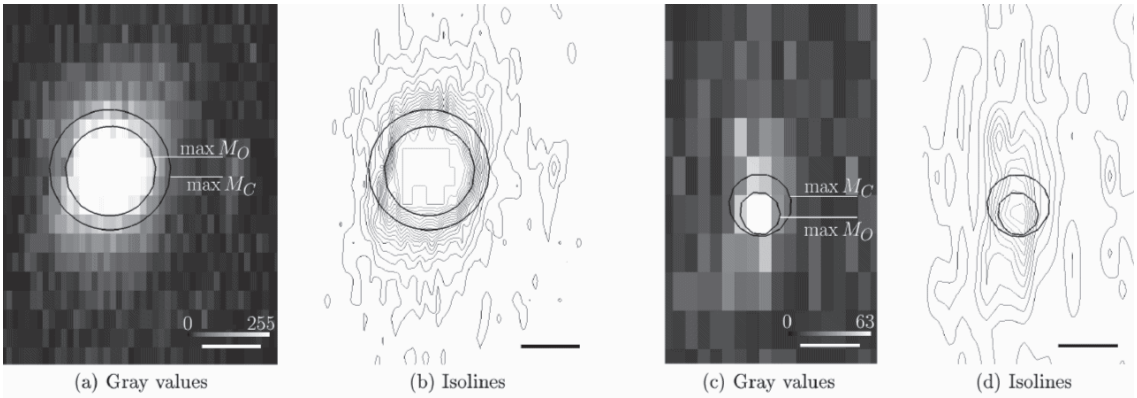
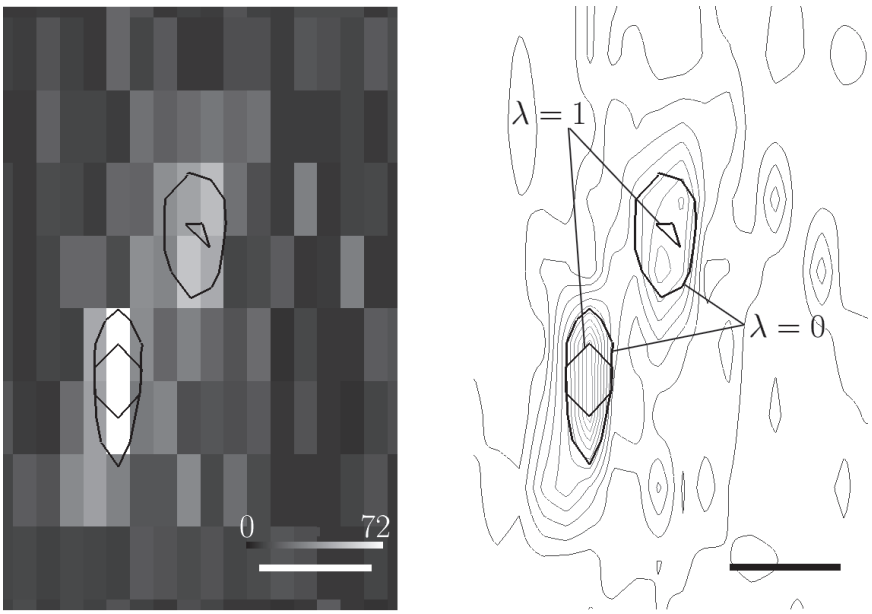


Figure 2.11: Comparison of radius estimates obtained using the two medialness measures M_O and M_C (eqs. (2.7)) and (2.8) for a thick (a,b) and a thin (c,d) neurite. The figures Show the cross-sections of the data and of the generalized cylinders which were fitted to the data. The cross-sections of the generalized cylinders are shown by the thick black contours. The data are visualized using gray values for image intensities (a,c) and as iso-intensity contours (b,d), for the same cross-sections. Note the different ranges of gray values in (a) and (c). Scalebar: (a,b) $1\mu m$, (c,d) $0.5\mu m$. The central medialness M_C is strongly influenced by the blurred gray values in z-direction (vertical in the figures), leading to an estimation of the radius which is larger than that of the offset medialness. The offset medialness M_O, however, considers the local maxima of the image gradients. The gradients are proportional to the density of the isolines in (b) and (d). Because no distinctive maxima are given in z-direction, M_O estimates the radius just with respect to the x- and y-directions (horizontal in the figures).

y-direction, since at the finest scale the image gradients are very sensitive to the inherently low signal-to-noise ratio of the data. After deconvolution the difference between the two measures reduces to a large extend (Hiesinger et al., 2001). Without deconvolution the user may want to choose a proper value for λ (see eq. (2.10)) weighting the two medialness measures such that the method yields the most satisfying results. A reasonable choice is to use M_O in general except for very thin structures, where M_C yields more robust results and should be used in addition.

2.5.4 Surface Reconstruction

The fitting procedure for the surface uses two criteria for the evaluation of the data (see eq. (2.18)), the intensity gradient and a locally adapted threshold which is provided by the skeleton. In order to compare the influence of the different terms we show a cross-section of a detail fo the full data set with two neurites (see fig. 2.12).
The cross-sections which lie perpendicular to the axes of both branches show that the



(a) Gray values (b) Isolines

Figure 2.12: Comparison of the results according to the two data evidence criteria in eq. (2.18) for a thin neurite. The figures show the cross-sections of the data and of the surfaces which were optimized according to the image gradients (parameters: $\alpha = 0.33, \lambda = 1$) and the intensities using a locally adapted threshold (parameters: $\alpha = 0.1, \lambda = 0$). The cross-sections of the surfaces are shown by the thick black contours. The data are visualized using intensity gray values (a) and iso-intensity contours (b), for the same cross-section. Scalebar: $0.5\mu m$. The gradients are proportional to the density of the isolines in (b). Because no distinctive maxima of the gradient magnitudes are given in z-direction (vertical in the figures), the surface resulting from the gradient-based fitting is determined by those in the x- and y-directions (horizontal in the figures). If the latter are not distinctive enough either, the surface tends to collapse (like the right top one). The intensity-based fitting results in a surface, whose cross-sectional shape is strongly influenced by the blurred gray values in z-direction, but does not rely on local maxima of the gradient magnitudes.

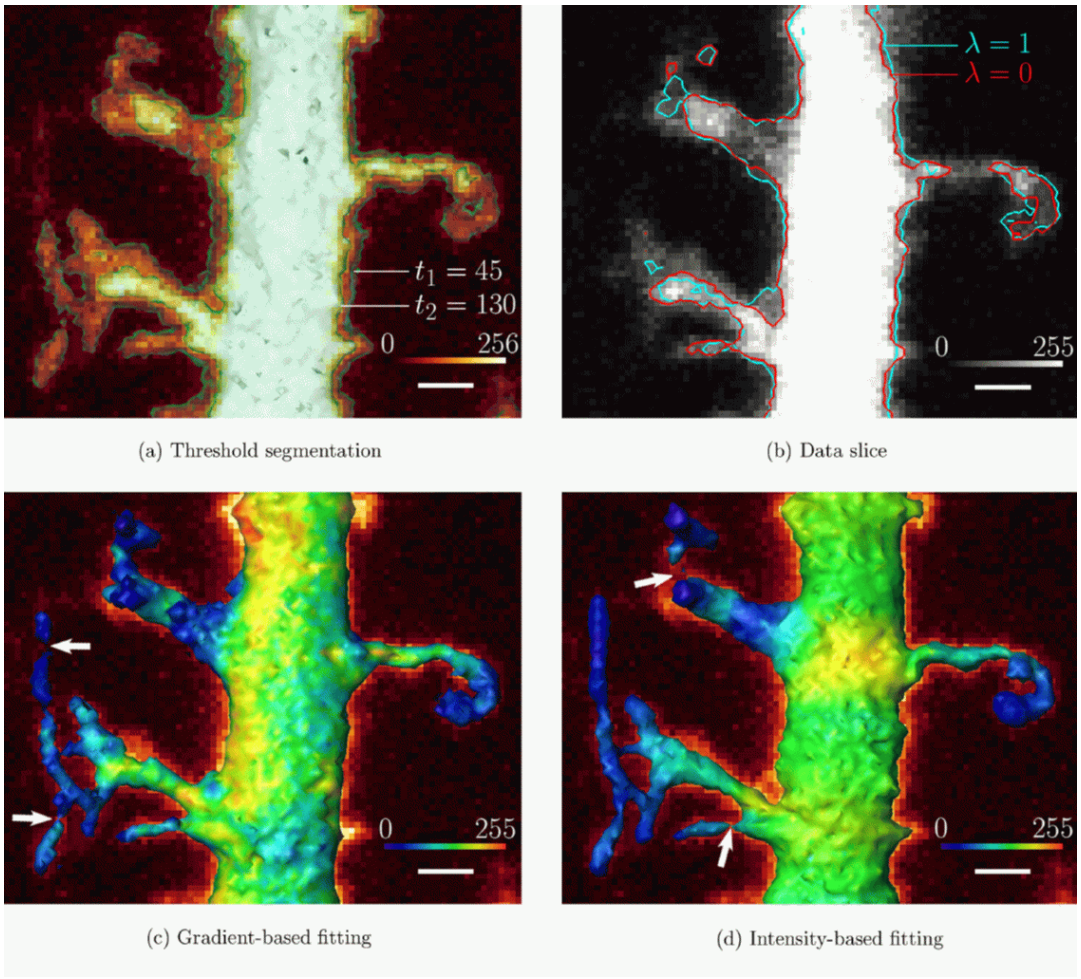
intensity gradients (observable by the density of the isolines in fig. 2.12(b)) do not always provide enough information about the contours of the neural structures. It is inevitable here to reconstruct the surface using a finite threshold value in order to avoid a local collapse of the surface. If, however, the gradients are sufficiently significant, the reconstruction of the tubular surface is dominated by the intensity profile in the x- and y-direction. Due to the lack of distinct maxima in z-direction the first term in eq. (2.27) gives rise to an approximatly circular cross-section. This behaviour can be considered as an invariance with respect to anisotropic blur occurring in confocal images.

The varying contrast which is due to differently sized structures and inhomogeneous staining requires a local adaptation of the threshold (see fig. 2.13). The intensity values at the locations of the reconstucted surface are higher for the thick structures than for the thinner ones. In Fig. 2.13(a) two surface reconstructions are shown according to the global thresholds $t_1 = 45$ and $t_2 = 130$. The threshold t_2, which is optimal for the thick branch, fails to reconstruct the thin branches, while the threshold t_1, which includes the thin branches, is too low for the thick branch.
The intensity-based term in eq. (2.18) depends on the local threshold values determined by the respective snaxel of the skeleton, that is the snaxel, whose boundary is closest to the respective location. These values can be observed at the surface shown in fig. 2.13(d) which was optimized with respect to that term. A strong correlation between the thickness of the branches and the local thresholds is evident. Note, that neither of the medialness measures used for fitting the snaxels (eqs. (2.2) and (2.5)) is based on the absolute intensities.
The local intensity values at the surface which was fitted with respect to the local thresholds approximate those which are found at the surface which was fitted to the gradients (see fig. 2.13(c)). This suggests that both criteria for incorporating the data evidence lead to similar results and thus are reasonable choices (see also fig. 2.13(b)). Nevertheless none of the two is able to avoid the occurrence of little gaps in the reconstruction of thin branches (indicated by the arrows in figs. 2.13(c) and (d)). Since the locations of those gaps differ, it can be expected that the combination of both reduces their number. Indeed, the surface of the same detail (see fig. 2.14) fitted using both criteria ($\lambda = 0.95$) no longer shows these gaps. The difference between the generalized cylinders and the exact surface is shown by the colors. This mismatch may be important, when the proximity of the neural branches to other structures is evaluated which requires an exact surface reconstruction.

2.5.5 2nd Channel Mapping

A detailed surface reconstruction may be used to evaluate the proximity to other structures like other neurons or cell surface molecules. We illustrate the projection of second channel data onto the surface reconstruction in Fig. 2.15 using a reconstruction of an intracellularly filled interneuron and immunocytochemistry of the NO synthetase. To distinguish between both labels in the double staining they were bound to different fluorescent markers and acquired with different laser lines and emission filters. At the moderate laser intensities used for acquisition no cross-talk between both channels was observed.

As an example for examining the spatial relation of two signals in 3-dimensional space we followed the question how NO synthethase immunolabel was distributed within an $0.5\mu m$ vicinity around the dendritic tree of the interneuron. In Fig. 2.15(a) an x-y-slice of the 2nd channel data is shown which is here distributed over the whole volume. This is for instance the case with labelled cell surface molecules, because the labelling cannot be restricted to those molecules belonging to the neuron of interest. In order to project only the relevant data onto the surface the volume can be clipped at a user defined distance from the surface, say 0.5μ (see fig. 2.15(b)). The intensities inside the remaining region is projected onto the surface in Fig. 2.15(c)). If another (pre- or postsynaptic) neuron was stained in the same specimen, it is also possible to mark the locations of the surface where either direct, zero-distance contact or approximation below a user defined threshold between the two appears. In order to obtain reliable results in all these cases it is crucial to have an extremely accurate surface reconstruction which is provided by our method.



(a) Threshold segmentation (b) Data slice

(c) Gradient-based fitting (d) Intensity-based fitting

Figure 2.13: Comparison of the results according to the two data evidence criteria in eq. (2.18) for a detail including neurites of different thickness. All images are views in axial direction, (a), (c), and (d) are shown in front of a maximum intensity projection. Scalebar: $1\mu m$. (a) Two global threshold segmentations (thresholds: $t_1 = 45, t_2 = 130$) shown as transparent surfaces. It is obvious, that branches of different thickness require different thresholds for segmentation. (b) A data slice showing a cut of the volume. The cross-section of the surface, which was optimized w.r.t. the intensity gradients ($\lambda = 1$) is drawn blue, while the one, which was optimized w.t.r. the local thresholds ($\lambda = 0$) is drawn red. Differences between the two occur mainly at the thin branches. (c) The surface reconstruction using the intensity gradients (parameters: $\alpha = 0.95, \lambda = 1$). The intensity values of the image at the locations of the surface are color coded. (d) The surface reconstruction using the local thresholds (parameters: $\alpha = 0.05, \lambda = 0$). The distribution of intensity values at the surface is nearly the same in (c) and (d), indicating that both criteria are reasonable and reliable, although the reconstruction fails at some places of the thin branches (marked by the arrows).

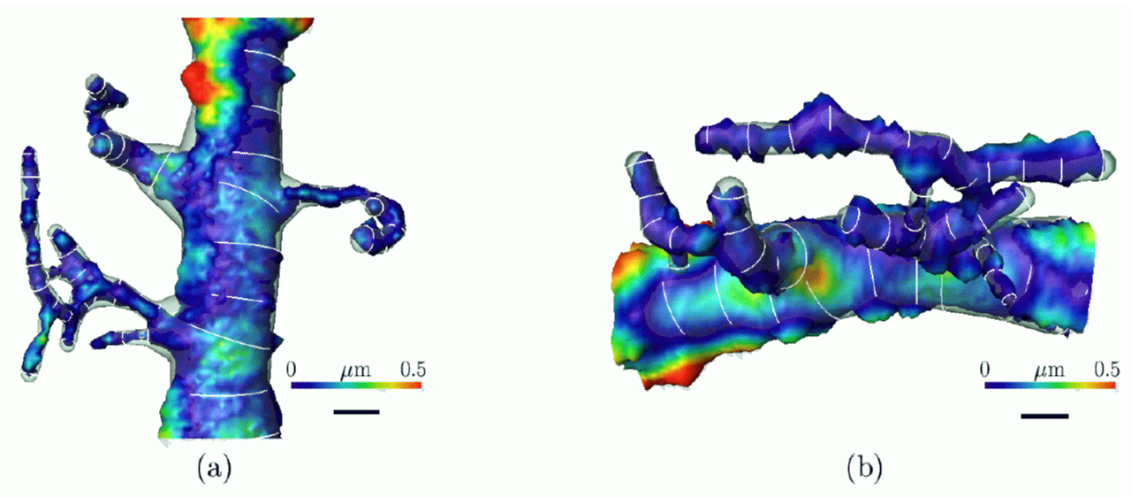


Figure 2.14: The fitted surface (using eq. (2.27)) is compared to the surface of the fitted generalized cylinders (minimum of eq. (2.9)) which is shown by the transparent surface and the white circles. The local distance between the two is color coded (in μm). The left figure is a view in axial direction while the right one is a view from the side. Scalebar: $1\mu m$.

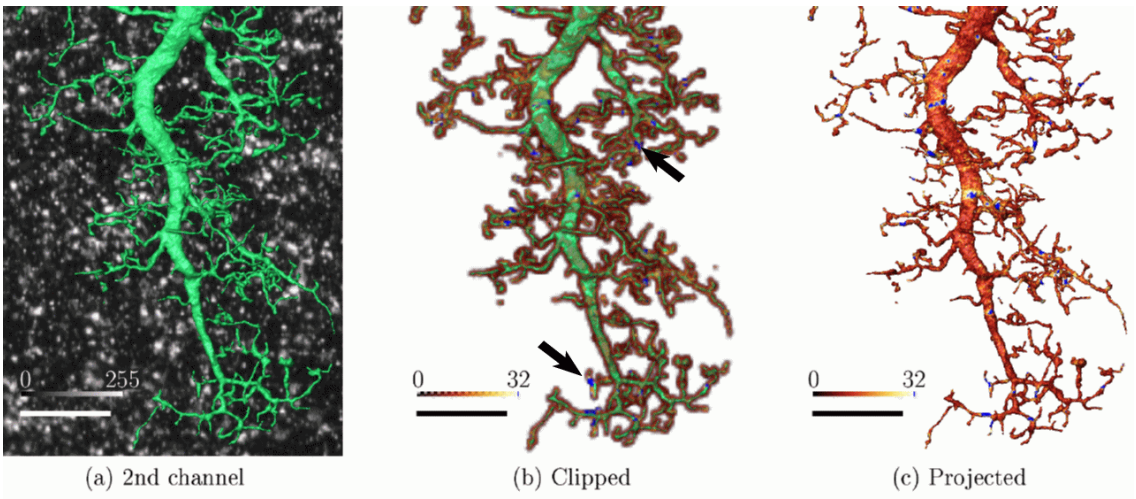


Figure 2.15: Mapping of nearby structures onto a surface reconstruction (Data by courtesy of Daniel Münch, FU Berlin). Scalebar: $20\mu m$. (a) The surface and the maximum intensity projection of the second channel recording. (b) The volume with the second channel staining is clipped at a distance of $0.5\mu m$ of the surface. The remainder is shown as volume rendering. The blue spots (marked by arrows) indicate where the intensity exceeds 32. (c) The remaining structures are mapped onto the surface.

2.6 Conclusions

In this paper, we present a semi-automatic method for high quality three-dimensional reconstructions of most complex neurons from confocal image stacks which can be used for diverse purposes. The skeleton gives the structural description of the neuron with high accuracy and almost arbitrary high sample density of center lines and radii while reducing considerably the user's effort, that is the necessary quantity and quality of input actions. The automatic surface reconstruction achieves highest precision of even the finest neurites with low contrast.

In contrast to other available reconstruction tools our method is characterized by the conjunction of (i) an accelerated process which provides exact results due to automated fitting, (ii) invariance with respect to the reconstructing user and his carefulness, also due to automated fitting, (iii) corrigible results due to extensive possibilities of user interaction, and (iv) independence from computer platforms due to its implementation as a module for *Amira* (Indeed - Visual Concepts GmbH, Berlin, Germany) which is available for the standard operating systems on the PC[8].

Thus we provide an efficient, exact and controllable reconstruction procedure which allows the analysis of a variety of neuronal properties: (i) the physiological properties of neurons can be modeled exactly, since the radii of the segments vary continiously along the axes; (ii) the detailed surface reconstruction allows to evaluate the proximity to other structures reliably; and (iii) the structural description of the neuron allows the statistical analysis of its morphology and structure. Furthermore it is possible to create a metric parameter database for neurons which can be filled by several users, since the user invariance of the tool makes the results comparable.

[8] A binary version of the module will be available free of charge for download at `http://www.neurobiologie.fu-berlin.de/duch.html`. A brief documentation describing the installation and usage of the module will be provided, too.

2.7 Appendix

2.7.1 Calculation of the Gradients for the Snake Energies

The partial derivatives of the external energy with respect to the center point $\vec{x}_i$ and to the radius r_i of snaxel i are

$$\begin{aligned}
\frac{\partial \widehat{E}_{\text{ext}}}{\partial \vec{x}_i} &= -\lambda \frac{\partial M_C(i)}{\partial \vec{x}_i} - (1-\lambda)\frac{\partial M_R(i)}{\partial \vec{x}_i} && (2.28)\\
&= \frac{1}{m}\sum_{j=1}^{m} (\lambda H(\vec{x}_i + r\vec{o}_j)\vec{o}_j \\
&\quad + (1-\lambda)(I(\vec{x}_i + r\vec{o}_j) - t)\nabla I(\vec{x}_i + r\vec{o}_j)) && (2.29)\\
\frac{\partial \widehat{E}_{\text{ext}}}{\partial r_i} &= -\lambda \frac{\partial M_C(i)}{\partial r_i} - (1-\lambda)\frac{\partial M_R(i)}{\partial r_i} && (2.30)\\
&= \frac{1}{m}\sum_{j=1}^{m} (\lambda \vec{o}_j^T H(\vec{x}_i + r\vec{o}_j)\vec{o}_j \\
&\quad + (1-\lambda)(I(\vec{x}_i + r\vec{o}_j) - t)\vec{o}_j^T \nabla I(\vec{x}_i + r\vec{o}_j)), && (2.31)
\end{aligned}$$

where H is the Hessian matrix with

$$H = \nabla^2 I = \begin{pmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{pmatrix}. \qquad (2.32)$$

The partial derivatives of the internal energy with respect to the snaxels having two neighbors are

$$\begin{aligned}
\frac{\partial \widehat{E}_{\text{int}}}{\partial \vec{x}_i} &= \frac{1}{2}(\vec{x}_{i-1} + \vec{x}_{i+1} - 2\vec{x}_i) && (2.33)\\
\frac{\partial \widehat{E}_{\text{int}}}{\partial r_i} &= \frac{1}{2}(r_{i-1} + r_{i+1} - 2r_i). && (2.34)
\end{aligned}$$

Like mentioned in section 2.4.1 the internal energy is neglected for the snaxels with one neighbor. For those having more than two neighbors the internal energy is calculated using a weighted sum of the contributions of all pairs of neigbors.

Acknowledgements

This work was supported by the Federal Government of Germany (S. Schmitt, M. Scholz, K. Obermayer BMBF 0310962) and the Deutsche Forschungsgemeinschaft (J.-F. Evers SFB 515/A7; C. Duch SFB 515/A7, DU331/2-3).

References

Al-Kohafi, K. A., Lasek, S., Szarowski, D. H., Pace, C. J., Nagy, G., Turner, J. N., and Roysam, B. (2002). Rapid automated three-dimensional tracing of neurons from confocal image stacks. *IEEE Transactions on Information Technology in Biomedicine*, 6(2):171–187.

Aylward, S. R. and Bullit, E. (2002). Initialization, noise, singularities, and scale in height ridge traversal for tubular object centerline extraction. *IEEE Transactions on Medical Imaging*, 21(2):61–75.

Belichenko, P. V. and Dahlström, A. (1995). Confocal laser scanning microscopy and 3-D reconstructions of neuronal structures in human brain cortex. *NeuroImage*, 2(3):201–207.

Binford, T. O. (1987). Generalized cylinder representation. In Shapiro, S. C., editor, *Encyclopedia of Artificial Intelligence*, pages 321–323. John Wiley & Sons.

Blum, H. (1967). A transformation for extracting new descriptors of shape. In Whaten-Dunn, W., editor, *Models for Perception of Speech and Visual Form*. MIT Press, Cambridge, Massachusetts.

Borst, A. and Haag, J. (1996). The intrinsic electrophysiological characteristics of fly lobula plate tangential cells: I. passive membrane properties. *J. Comp. Neurosci.*, 3(4):313–336.

Burl, M. C., Weber, M., and Perona, P. (1998). A probabilistic approach to object recognition using local photometry and global geometry. In Burkhardt, H. and Neumann, B., editors, *Computer Vision - ECCV'98, 5th European Conference on Computer Vision, Freiburg, Germany, June 2-6, 1998, Proceedings, Volume II*, volume 1407 of *Lecture Notes in Computer Science*, pages 628–641, Berlin, Heidelberg, New York. Springer.

Caselles, V., Kimmer, R., and Sapiro, G. (1997). Geodesic active contours. *Int. J. Comp. Vision*, 22:61–79.

Chopp, D. L. (1993). Computing minimal surfaces via level set curvature flow. *J. Comp. Phys.*, 106(1):77–91.

Cohen, L. D. (1991). On active contour models and balloons. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 53(2):211–218.

da Fontoura Costa, L., Manoel, E. T. M., Faucereau, F., Chelly, J., van Pelt, J., and Ramakers, G. (2002). A shape analysis framework for neuromorphometry. *Network: Computation in Neural Systems*, 13(3):283–310.

da Fontoura Costa, L. and Velte, T. J. (1999). Automatic characterization and classification of ganglion cells from the salamander retina. *J. Comp Neurol.*, 404(1):33–51.

De Schutter, E. and Bower, J. M. (1994). An active membrane model of the cerebrellar purkinje cell, i. simulation of current clamps in slice. *J. Neurophysiol.*, 71(1):375–400.

Eberly, D., Gardner, R. B., Morse, B. S., Pizer, S. M., and Scharlach, C. (1994). Ridges for image analysis. *Journal of Mathematical Imaging and Vision*, 4(4):353–373.

Flasque, N., Desvignes, M., Constans, J.-M., and Revenu, M. (2001). Acquisition, segmentation and tracking of the cerebral vascular tree on 3D magnetic resonance angiography images. *Medical Image Analysis*, 5(3):173–183.

Frangi, A. F., Niessen, W. J., Hoogeveen, R. M., van Walsum, T., and Viergever, M. A. (1999). Model-based quantitation of 3-D magnetic resonance angiographic images. *IEEE Transactions on Medical Imaging*, 18(10):946–956.

Gerig, G., Koller, T., Székely, G., Brechbühler, C., and Kübler, O. (1993). Symbolic description of 3-D structures applied to cerebral vessel tree obtained from MR angiography volume data. In Barrett, H. H. and Gmitro, A. F., editors, *Information Processing in Medical Imaging IPMI'93*, number 687 in Lecture Notes in Computer Science, pages 94–111. Springer Verlag, Berlin.

Gray, J. R. and Weeks, J. C. (2003). Steroid-induced dendritic regression reduces anatomical contacts between neurons during synaptic weakening and the developmental loss of a behavior. *J. Neurosci.*, 23(4):1406–1415.

Haralick, R. M. (1983). Ridges and valleys in digital images. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 22:28–38.

Hausser, M., Spruston, N., and Stuart, G. J. (2000). Diversity and dynamics of dendritic signaling. *Science*, 290(5492):739–744.

Hiesinger, P. R., Scholz, M., Meinertzhagen, I. A., Fischbach, K.-F., and Obermayer, K. (2001). Visualization of synaptic markers in the optic neuropils of *Drosophila* using a new constrained deconvolution method. *J. Comp Neurol.*, 429:277–288.

Jankowska, E., Maxwell, D., Dolk, S., Krutki, P., Belichenko, P., and Dahlström, A. (1995). Contacts between serotoninergic fibres and dorsal horn spinocerebellar tract neurons in the cat and rat: A confocal microscopic study. *Neurosci.*, 67(2):477–487.

Kass, M. (1988). Snakes: active contour models. *J. Comp. Vision*, 1:321–331.

Kimmel, R., Shaked, D., Kiryati, N., and Bruckstein, A. M. (1995). Skeletonization via distance maps and level sets. *Computer Vision and Image Understanding: CVIU*, 62(3):382–391.

Koch, C., Poggio, T., and Torre, V. (1982). Retinal ganglion cells: Functional significance of dendritic morphology. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 298:227–264.

Krissian, K., Malandain, G., Ayache, N., Vaillant, R., and Trousset, Y. (2000). Model-based detection of tubular structures in 3D images. *Computer Vision and Image Understanding*, 80(2):130–171.

Lamotte d'Incamps, B., Destombes, J., Thiesson, D., Hellio, R., Lasserre, X., Kouchtir-Devanne, N., Jami, L., and Zytnicki, D. (1998). Indications for GABA-immunoreactive axo-axonic contacts on the intraspinal arborization of a Ib fiber in cat: A confocal microscope study. *J. Neurosci.*, 18(23):10030–10036.

Libersat, F. and Duch, C. (2002). Morphometric analysis of dendritic remodeling in an identified motoneuron during postembryonic development. *J. Comp Neurol.*, 450(2):153–166.

Lorigo, L. M., Faugeras, O. D., Grimson, W. E. L., Keriven, R., Kikinis, R., Nabavi, A., and Westin, C.-F. (2001). CURVES: Curve evolution for vessel segmentation. *Medical Image Analysis*, 5(3):195–206.

Mizrahi, A., Ben-Ner, E., Katz, M. J., Kedem, K., Glusman, J. G., and Libersat, F. (2000). Comparative analysis of dendritic architecture of identified neurons using the Hausdorff distance metric. *J. Comp. Neurol.*, 422(3):415–428.

Olabarriaga, S. D. and Smeulders, A. W. M. (2001). Interaction in the segmentation of medical images: A survey. *Medical Image Analysis*, 5(2):127–142.

Pizer, S. M., Eberly, D., Fritsch, D. S., and Morse, B. S. (1998). Zoom-invariant figural shape: The mathematics of cores. *Computer Vision and Image Understanding*, 69(1):55–71.

Sato, Y., Nakajima, S., Shiraga, N., Atsumi, H., Yoshida, S., Koller, T., Gerig, G., and Kikinis, R. (1998). Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images. *Medical Image Analysis*, 2(2):143–168.

Segev, I. and Rall, W. (1998). Excitable dendrites and spines: earlier theoretical insights elucidate recent direct observations. *Trends Neurosci.*, 21(11):453–460.

Streekstra, G. J. and van Pelt, J. (2002). Analysis of tubular structures in three-dimensional confocal images. *Network: Computation in Neural Systems*, 13(3):381–395.

Uylings, H. B., Ruiz-Marcos, A., and van Pelt, J. (1986). The metric analysis of three-dimensional dendritic tree patterns: a methological review. *J. Neurosci. Meth.*, 18(1-2):127–151.

van Pelt, J., Uylings, H. B., and Verwer, R. (1989). Distributional properties of measures of tree topology. *Acta Stereology*, 8:456–470.

Wouterlood, F. G., Vinkenoog, M., and van den Oever, M. (2002). Tracing tools to resolve neural circuits. *Network: Computation in Neural Systems*, 13(3):327–324.