Three-Dimensional Analysis of Spiny Dendrites Using Straightening and Unrolling Transforms

Juan Morales · Ruth Benavides-Piccione · Angel Rodríguez · Luis Pastor · Rafael Yuste · Javier DeFelipe

Abstract Current understanding of the synaptic organization of the brain depends to a large extent on knowledge about the synaptic inputs to the neurons. Indeed, the dendritic surfaces of pyramidal cells (the most common neuron in the cerebral cortex) are covered by thin protrusions named dendritic spines. These represent the targets of most excitatory synapses in the cerebral cortex and therefore, dendritic spines prove critical in learning, memory and cognition. This paper presents a new method that facilitates the analysis of the 3D structure of spine insertions in dendrites, providing insight on spine distribution patterns. This method is based both on the implementation of straightening and unrolling transformations to move the analysis process to a planar, unfolded arrangement, and on the design of DISPINE, an interactive environment that supports the visual analysis of 3D patterns.

Keywords Visual analysis in neuroscience · Dendritic spines · Irregular and unstructured grids · Point-based data

Introduction

The acquisition of high resolution reconstructions of neurons in 3D is becoming more and more frequent in neuroscience, for purposes such as analyzing detailed aspects of the structure of the neurons' dendritic arbors (DeFelipe 2010; Meijering 2010; Donohue and Ascoli 2011). However, studying the 3D distribution of relevant entities of dendritic processes is not a simple task. It should be noted that inferring 3D structure from 2D projections obtained with the current acquisition techniques (Semwogerere and Weeks 2005) gets

difficult when the geometry is complex and the user has to recognize patterns just visually (Eick and Karr 2002). The video accompanying Fig. 1 illustrates the rotation of a dendrite's segment, showing clearly some of the problems found by domain experts when analyzing such a structure directly in 3D. Typical difficulties come from the irregular distribution of 3D samples, the dendrite's curvature, the presence of occlusions, and the absence of spatial references. The existence of a large number of relevant morphological parameters increases the influence of these problems, hampering further analysis of this kind of data.

So far, direct visual observation of these feature distributions was the only exploration technique available to neuroscientists. In consequence, even drawing preliminary conclusions was a hard task; this situation has resulted in the existence up to now of very few studies dealing with the 3D arrangement of many relevant dendritic structures. But on the other hand, understanding the synaptic organization of the brain depends largely on knowledge about the synaptic inputs to the neurons. Indeed, the dendritic surfaces of pyramidal cells (the most common neuron in the cerebral cortex Defelipe and Fariñas 1992) are covered by thin protrusions named dendritic spines, which represent the targets of most excitatory synapses in the cerebral cortex and therefore, are critical in learning, memory and cognition (Spruston 2008; Kasai et al. 2010; Yuste 2010).

This paper presents a new method that facilitates analyzing 3D structure in dendrites, together with its application for studying the distribution patterns of dendritic spines. It has to be noticed that dendritic spines are dynamic structures which might show motility and changes in length and size, although their insertion points in the dendritic shafts are rather stable (e.g., Portera-Cailliau et al. 2003). In consequence, the spines' insertion points have been chosen as primary references, albeit users can select other references such as the spine heads' centroid or its distal tip.

The insertion of spines in dendrites can be represented as a cloud of points distributed over the dendrite surface, presenting accordingly a tubular arrangement. Nevertheless, the analysis of spine distribution patterns over the dendrite's surface by classical interactive procedures such as zooming, panning or rotating the point of view is difficult. The main reason for this difficulty is that the dendrites' geometry perturbs the 3D locations of the spines' insertion points, destroying the shape from motion cues that facilitate 3D user perception while rotating and interacting with the 3D dendrite representations. Furthermore, sharp bends in the dendrite's axis or even radial displacements in the position of the spines' insertion points, such as those found in Figs. 1c and 6, make things harder.

The method proposed here uses a new approach for simplifying the visual analysis of dendritic spine distributions, by straightening and unrolling the placement of the point-based primitives used to characterize the insertion of spines over the dendritic surface. Essentially, this method is based on applying a transformation that reduces the data dimensionality from 3D to 2D, while

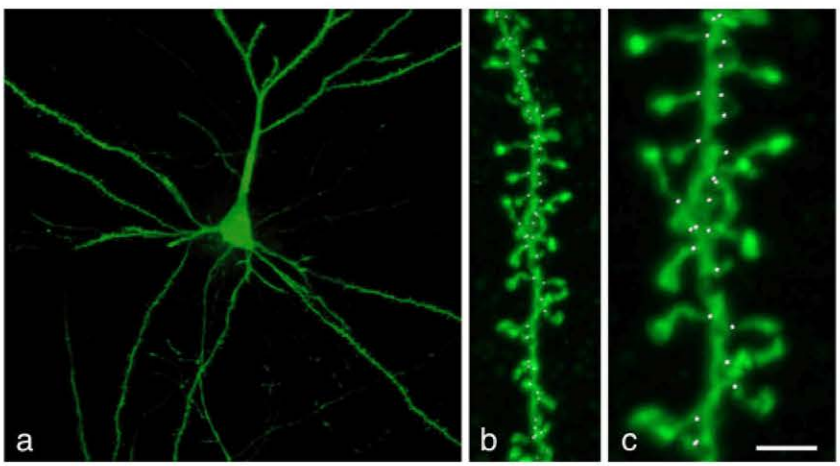


Fig. 1 **a** Confocal microscopy image of an intracellularly injected layer III pyramidal neuron of the human cingulate cortex. **b** and **c** present low and high magnification images showing a basal dendritic segment, including dendritic spines and their points of insertions (*white dots*) in the dendritic shaft. This image represents a single plane from the stack of images used to locate the insertion points in the 3D reconstruction (see text for further details). Thus, only the insertion points present in that plane are shown (see a supplementary video at http://www.datsi.fi.upm.es/~arodri/cbb/ninf11/Figure1_video.avi). Scale bar (in **c**): 26.25 μm in **a**; 4 μm in **b**; and 1.9 μm in **c**

preserving the most relevant spatial properties from the original distribution. Moving from the original 3D domain to a transformed 2D domain might seem odd at first, but the fact is that this transformation eliminates a third, irrelevant dimension, where noise is prevalent. Analyzing dendrite data in this transformed domain is much simpler because this noise is eliminated. At any rate, the user can combine both domains (the original 3D and the transformed 2D), selecting which one to use at every single moment.

The straightening and unrolling transforms have been implemented in DISPINE, a new software tool that supports the interactive visual analysis of 3D patterns. This is especially relevant because, as mentioned by Fuchs and Hauser (Fuchs and Hauser 2009), "Interaction is probably the most important tool for understanding complex data". The interaction process lets users change dynamically the visualization parameters, allowing the representation of data in such a way that the presence of relevant information is maximized. This approach facilitates and accelerates strongly the analysis process, providing neuroscientists with a more powerful method to examine the insertion points, since the joint visualization of data both in the transformed and the original space allows a better visual analysis of surface patterns and trends.

Because of the high magnification applied during the digitization stage, dendrites are divided during the acquisition process into segments of fixed length to be manageable; Fig. 1b shows an example of a dendritic segment captured with confocal microscopy. Once spines are represented as volumetric images, the 3D positions of the spines' insertion points can be manually marked (Fig. 1c) with the help of commercial tools like Imaris (Bitplane 2011). Other possible references that could be marked instead are the spines' center of mass, or the apex of the spine's head. Additionally, other relevant morphological features of spines, like length or volume (Arellano et al. 2007), can also be manually estimated with Imaris.

Figure 2 shows a global view of the results of the marking process applied to a dendritic segment, drawing red dots in the positions of the detected insertion points. Replacing each spine by its insertion point is a first simplification which can facilitate the visual analysis of spine distributions. But as Figs. 1 and 2 show, this process is not enough, in particular for tasks such as detecting 3D hidden surface distribution patterns.

Additionally, the zoomed-in window in Fig. 2 shows clearly some discontinuities in the digitized spines, those arising because the injection technique does not guarantee a complete stain of the neuron structures. This effect makes automatic analysis very hard, which is one of the reasons why interactive visualization procedures are such an important support tool for exploring the anatomy of brain tissues.

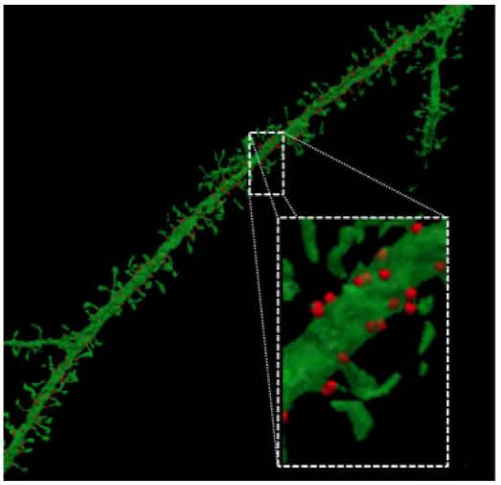


Fig. 2 Example of a dendritic segment with spines represented by *red markers* indicating the spines' insertion points on the dendrite's surface. The volumetric image displayed here has a resolution of $1024 \times 1024 \times 45$ voxels for the following dimensions: 76.88 μm × 76.88 μm × 12.59 μm

The following section describes a spatial transformation that improves the level of interpretability of the cloud of points. For a better explanation of the method, the spatial transform is decomposed in two stages, straightening and unrolling. From now on, the word *"point"* will be a synonymous of the term *"spine marker"*, or simply *"spine"*. This way, each spine will be represented by its insertion point p_i, and the set of spines, by P (the set of spines' insertion points).

Spatial Transformations

Dendrites can be well approximated by tubular geometries, which can be parameterized along three directions (Fig. 3):

1. Axial direction: position along the medial axis[1] of the dendrite measured from the proximal end of the dendritic segment.
2. Angular direction: angular orientation around the dendrite's medial axis.
3. Radial direction: radial distance to the dendrite's medial axis.

[1]The dendrite's sheath medial axis is a simplified representation that is manually created; it is actually a polyline whose vertices are markers which store the tangent unit vector.

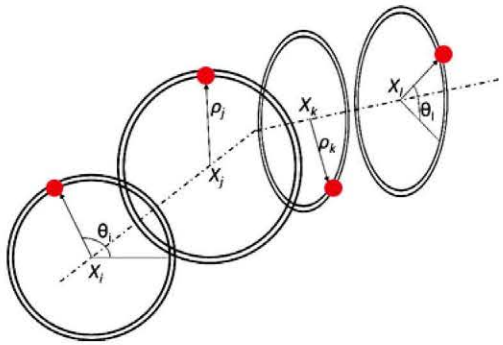


Fig. 3 Parameterization of the spines' markers positions (*red points*): axial direction (X), angular direction (θ) and radial direction (ρ)

From the three parameters that specify the position of a spine's insertion point on the dendrite's surface, the radial distance is the least relevant. There are two reasons for this: first, small variations on the distance to the medial axis may be due to changes in the dendrite's radius or to inaccuracies introduced during the manual marking process. Second, they are very small in comparison to the total spine length. Based on such consideration, this paper provides an alternative and simplified 2D representation for the insertion points, which preserves their relative positions but discards the radial component of their coordinates, therefore simplifying the visual analysis of point distributions. This alternative representation domain requires two transforms, straightening and unrolling, which ensure consistency with the original data sequences.

In order to illustrate the advantages provided by the method proposed here, two artificial distribution patterns of spines have also been considered, following helicoidal and purely random distribution patterns. On the one hand, helical arrangements of spines have been described in fish and mammalian Purkinje cells (O'Brien and Unwin 2006). On the other hand, dendrites of cortical pyramidal neurons have been studied using the tools presented in this paper, searching for helicoidal patterns. However, the results obtained show that the arrangement of spines is closer to a random distribution, discarding the hypothesis of a more structured format.

The following sections describe the straightening and unrolling geometrical transforms that allow the representation of spine data in a simplified domain.

Straightening Transform

The first operation to be performed on the dendrite segment is to straighten it, in order to eliminate the possible irregularities that could appear on its surface along the axial and radial directions. The straightening process must keep relevant points along the dendrite's medial axis, such as the spines' insertion points, in the same relative order as in their original positions.

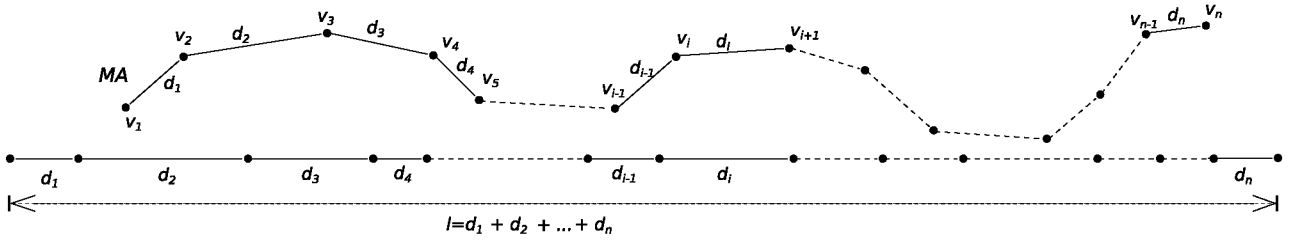
Starting at one end of the dendrite, it is possible to follow the medial axis that users have previously marked using one of the tools mentioned in Section "Introduction" (the medial axis could also be computed automatically by means of some skeleton or medial axis transforms (Borgefors et al. 1999; Lakshmi and Punithavalli 2009) and a final pruning step, if required).

Figure 4 shows a sketch of the operations involved in each step of the algorithm. These steps are described below:

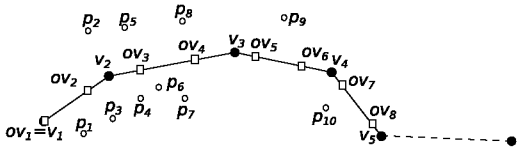
1. Step (a) computes the length value l of the medial axis MA (the blue axis shown in Fig. 6). The parameters $v_i(x, y, z),\ i = 1, \ldots n$ are the spatial positions of the set V of n vertices introduced by neuroscientists to define the MA polyline. The MA length is defined as:

$$l = \sum_{i=1}^{n-1} d_i(v_i, v_{i+1}), \tag{1}$$

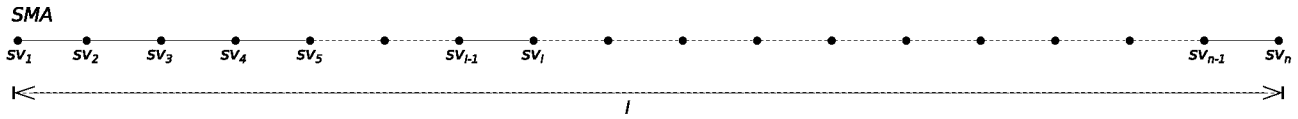
   where $d_i(v_i, v_{i+1})$ is the Euclidean distance between vertices v_i and v_{i+1}.
2. Step (b) introduces additional vertices along the dendrite's MA in order to facilitate the computation of the transformed coordinates of any point p_j located on the dendrite's surface after applying the straightening transform. The purpose of this step is to obtain a dense, evenly spaced collection of vertices v_i that can be associated with the spines' insertion points p_j, in order to displace them smoothly while the dendrite is being straightened. In consequence, this stage can be seen as a preprocessing step, necessary to minimize the geometric error introduced in the position of relevant points after the application of the straightening transform.
   Ideally, there would be no errors if the straightening transform were computed using the actual analytical function that maps the medial axis onto a straight line. This function could be used to compute the appropriate transform that would take every point in the MA into its final position (and together with every associated point point from the dendrite's surface corresponding to that slice of the medial axis, obtaining thus a straightened dendrite). But this process is unfeasible from the point of view of computational cost, and it has been approximated by the transformation of a discrete set of points placed along the MA. Clearly, a denser set of points will approximate the continuous,



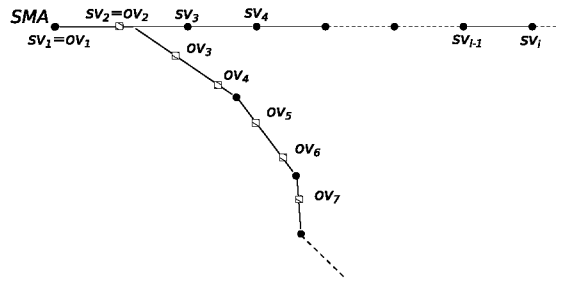
(a) Compute the length value l of the medial axis MA defined by vertices v_i introduced while manually tracing the dendrite (Eq. 1).



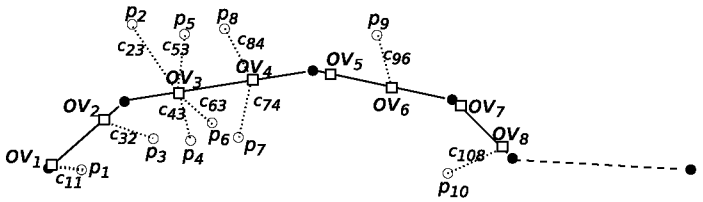
(b) Oversample the polyline by introducing new vertices in the original domain in order to create a dense collection of vertices that are evenly spaced along the original medial axis polyline. These oversampled vertices, labelled ov_i, will be used as references in the straightening process. Figure 4(b) shows the set of vertices ov_1 to ov_8 introduced in the polyline segment comprised between v_1 and v_5.



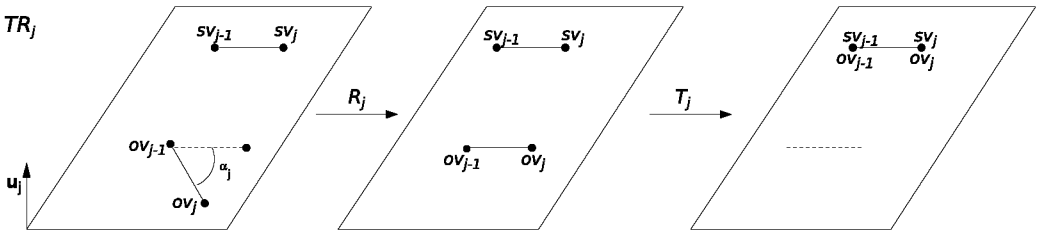
(c) Create the straightened medial axis SMA of the same length l and with the same number of vertices sv_i than the oversampled vertices ov_i, located at the corresponding intervals (Eq. 5).



(d) Translate the axes to match their origins (the selected end of the dendrite; Eq. 6).



(e) Relate the points p_i on the dendrite's surface to the closest oversampled vertex ov_j of the medial axis (Eq. 7) .



(f) For each of the vertices ov_i that characterizes the polyline within the dendrite's medial axis, compute the transformation TR_j (rotation + translation) that brings each vertex ov_i to its final position sv_i within the straightened medial axis, applying it to every point associated with each of the oversampled vertices (Eq. 8–10).

Fig. 4 Diagram summarizing all the steps involved in the straightening transform

analytical transform better. Therefore, rather than applying the transform to the vertices v_i which are placed along the medial axis polyline, and then, applying it to the set of points p_j associated with each of the vertices v_i, the transform's accuracy can be incremented by increasing the density of vertices

on the medial axis by using an oversampling factor w. Nevertheless, the number of new vertices has to remain bounded in order to keep accuracy and computational cost well balanced.

Figure 5 and Table 1 show the dependence of the average straightening errors and processing time on the total number of vertices used for computing the straightening transform. The original number of vertices was 45, and the errors are computed by comparing the relevant points' final positions obtained using the stated number of vertices with those obtained using the densest (from a practical point of view) population of oversampled *MA* vertices. In this data set, the oversampling factors range between 0.55 and 11.11.

Experimental tests with a variety of marked dendrites have shown that introducing an oversampling factor of $w = 2.5$ yields acceptable results, achieving a good balance between accuracy and computational cost. In consequence, after the oversampling process, the original n vertices v_i will be replaced with a new set of m oversampled vertices ov_i, which are uniformly spaced along the *MA*. The total number of vertices introduced, m, and the number of vertices introduced between each original vertex pair, m_i, are given by the following expressions:

$$m_i = \left\lceil w \cdot \frac{d_i(v_i, v_{i+1})}{l} \cdot n \right\rceil \tag{2}$$

$$m = \sum_{i=1}^{n-1} m_i \tag{3}$$

From now on, all these vertices form the new set of oversampled vertices *OV*:

$$OV = \{ov_k\},\ k = 1, m \tag{4}$$

3. Step (c) performs the straightening process, substituting the original dendrite's hand-marked points v_i of the *MA* by a set of straightened and uniformly oversampled points sv_i. The result is the straight medial axis *SMA*, a straight line of the same length l and with the same number m of oversampled vertices ov_i that were introduced in step (b), keeping their relative distances:

$$d_i(sv_i, sv_{i+1}) = d_i(ov_i, ov_{i+1}),\ ov_i \in OV,\ _{i=1,\ldots,m} \tag{5}$$

4. Step (d) translates the *SMA* for matching both its origin ($sv_1 \in SV$) and the direction of the first linear sector ($\overrightarrow{\mathbf{sv_1}} = (sv_1, sv_2)$) with the corresponding elements of the original *MA* ($ov_1 \in OV$, $\overrightarrow{\mathbf{ov_1}} = (ov_1, ov_2)$):

$$SMA = T \cdot MA \mid sv_1 = ov_1,\ \overrightarrow{\mathbf{sv_1}} = \overrightarrow{\mathbf{ov_1}} \tag{6}$$

Table 1 Tabular data of Fig. 5 considering absolute values for execution time (s) and average error (μm)

Number of resampled vertices	Execution time (s)	Average error (μm)
500	2.90	0.026
475	2.74	0.017
450	2.61	0.024
425	2.47	0.030
400	2.33	0.028
375	2.20	0.039
350	2.06	0.033
325	1.90	0.037
300	1.80	0.044
275	1.62	0.031
250	1.49	0.051
225	1.35	0.057
200	1.22	0.045
175	1.07	0.080
150	0.92	0.087
25	0.79	0.099
100	0.65	0.118
75	0.51	0.108
50	0.37	0.244
25	0.22	1.326

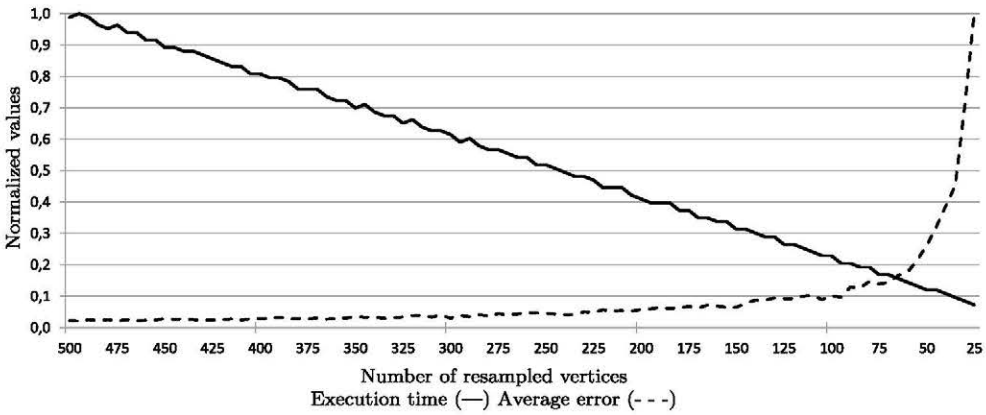
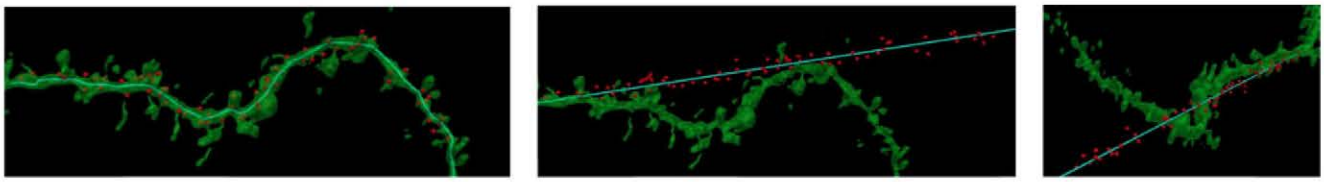
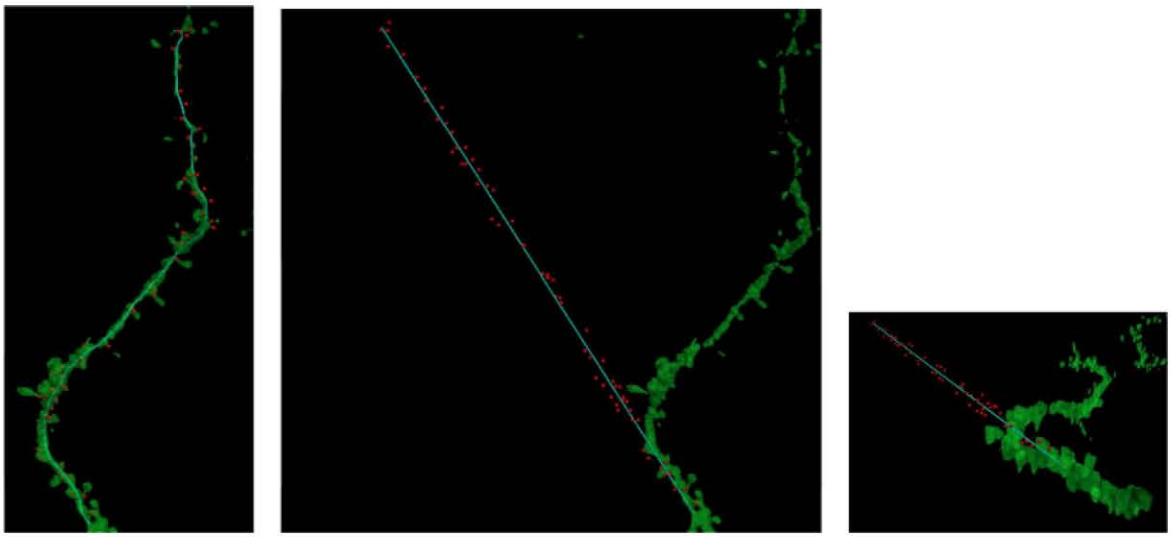


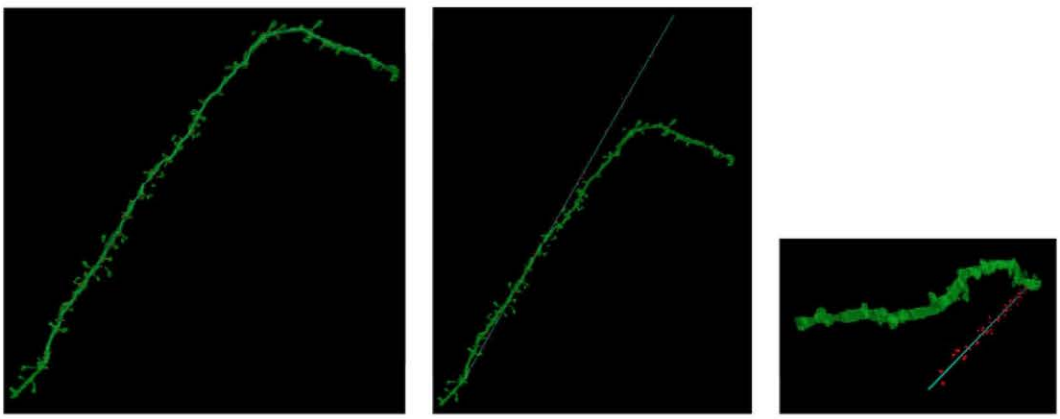
Fig. 5 Typical curves obtained when normalizing the approximation error and the execution time obtained with the straightening transform



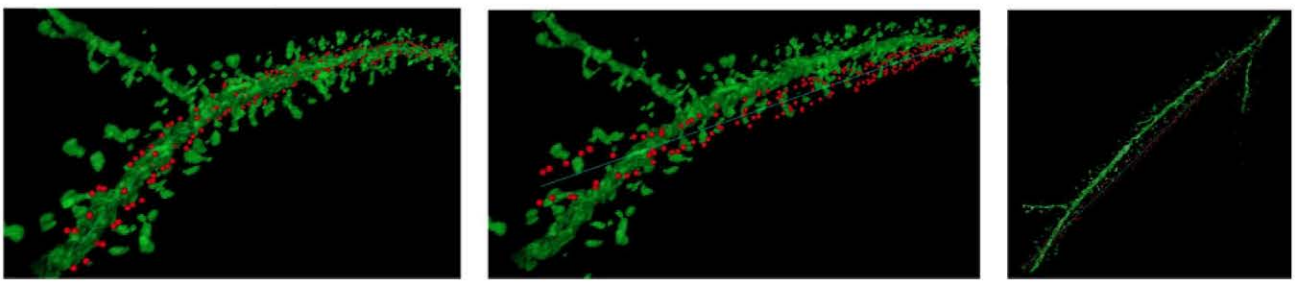
(a) Example of a tortuous dendrite.



(b) Example of another tortuous dendrite.



(c) Example of a third tortuous dendrite.



(d) Dendrite presented in Fig. 2.

Fig. 6 *Left*: dendrites with the original curvature; center, *right*: different views of the results achieved after applying the straightening transform

Bold letters have been used for denoting vectors, being T the required translation.

5. Step (e) establishes correspondences c_{ij} among the elements of the oversampled original medial axis OV and the points at the dendrite's surface, following the criteria of minimizing the Euclidean distance.

$$c_{ij} = (p_i, ov_j) \mid$$
$$d(p_i, ov_j) = \min(d(p_i, ov_j)) \; \forall p_i \in P, \; \forall ov_j \in OV \quad (7)$$

6. Step (f) computes for each vertex ov_j with at least one correspondence c_{ij} established in step (e) the transform $TR_j = T_j R_j$ required for moving ov_j to the position of the straightened MA's vertex sv_j associated with it. Being α_j the angle among the vectors $\overrightarrow{\mathbf{sv_j}}$ and $\overrightarrow{\mathbf{ov_j}}$ (those linking vertices ov_{j-1} and ov_j, and sv_{j-1} and sv_j), and $\overrightarrow{\mathbf{u_j}}$, the unitary vector perpendicular to the plane containing $\overrightarrow{\mathbf{sv_j}}$ and $\overrightarrow{\mathbf{ov_j}}$:

$$\overrightarrow{\mathbf{u_j}} = \frac{\overrightarrow{\mathbf{sv_j}} \times \overrightarrow{\mathbf{ov_j}}}{|\overrightarrow{\mathbf{sv_j}} \times \overrightarrow{\mathbf{ov_j}}|}, \quad (8)$$

Rotating vector $\overrightarrow{\mathbf{ov_j}}$ can be performed in an elegant way using Rodrigues' rotation formula:

$$\overrightarrow{\mathbf{ov_j^{rot}}} = \overrightarrow{\mathbf{ov_j}} \cos\alpha_j + \left(\overrightarrow{\mathbf{u_j}} \times \overrightarrow{\mathbf{ov_j}}\right) \sin\alpha_j + \overrightarrow{\mathbf{u_j}} \left(\overrightarrow{\mathbf{u_j}} \cdot \overrightarrow{\mathbf{ov_j}}\right) (1 - \cos\alpha_j) \quad (9)$$

Then we apply the translation:

$$\overrightarrow{\mathbf{T_j}} = \overrightarrow{\mathbf{sv_j}} - \overrightarrow{\mathbf{ov_j}} \quad (10)$$

Finally, the same transformation TR_j is applied to the set of points p_i of the dendrite's surface that belong to the set of correspondences c_{ij}, that is, those that are closer to the vertex ov_j.

The transform is only applied to the relevant points extracted by the user because those points are the only ones that will be analyzed later on. The result is a distribution of surface points aligned along the dendrite's central axis, ready for the unrolling transform.

Figure 6 illustrates the importance of this transform, showing several dendrites that are far from the ideal, straight model. Problems such as sharp bends, strong local curvature and the presence of irregularities are common, preventing users from getting an accurate idea of the 3D distribution of insertion points across the dendrite's surface. Straightening the dendrite facilitates the analysis task, allowing users to apply either the methods described in this paper, or regular interaction techniques, or else just comparing visually different dendrite sections (as they show a similar geometry). Any of them could be used as an example for describing the unrolling transform and DISPINE's functionality, which is done in the following sections.

Unrolling Transform

After the straightening transform, the relevant points are processed sequentially, arranged according to their X coordinate, starting again at one end of the dendrite. Then, a cylindrical transform to each of the points is applied:

$$(X, Y, Z) \rightarrow (X, \theta \cdot \overline{\rho}, \rho) \quad (11)$$

where θ is the azimuth or steering angle of the point around the sheath axis, ρ is the radial distance from the point to the central axis SMA, and $\overline{\rho}$ is the mean radial distance of the cloud of points (Fig. 7).

The unrolling transform maps points between two different 3D domains, but the variability of the radius ρ, the third coordinate in the transformed domain, is very small. Therefore, the unrolled points can be mapped easily to a 2D domain by using the X and $\theta \cdot \overline{\rho}$ coordinates. Please note that using the average radius in the unrolling transform instead of its value at each relevant point is equivalent to applying a scale factor to the angular coordinate; using local values for the radius would just distort the unrolled representation.

Figure 8 shows an example of applying the unrolling transform to the spines' insertion points of the straightened dendrite displayed in Fig. 6d. Figure 8a to d show the unrolled insertion points from the straightened dendrite in green, and their 3D spatial positions before

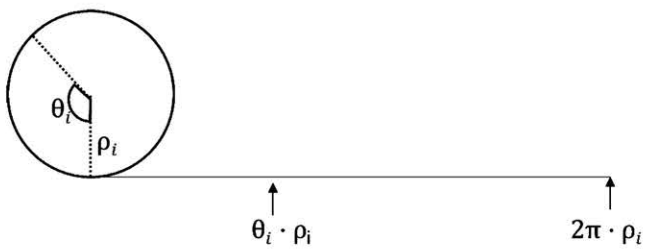
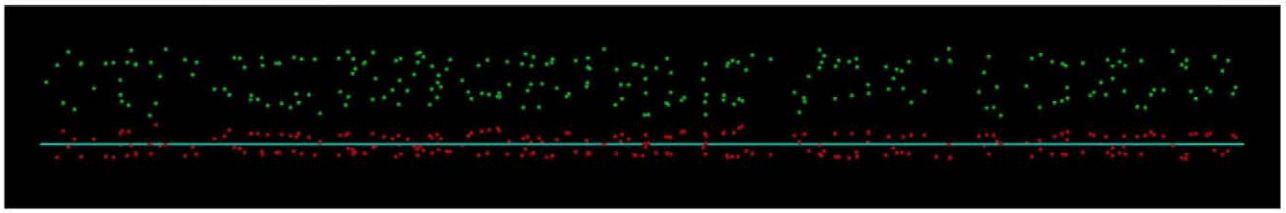
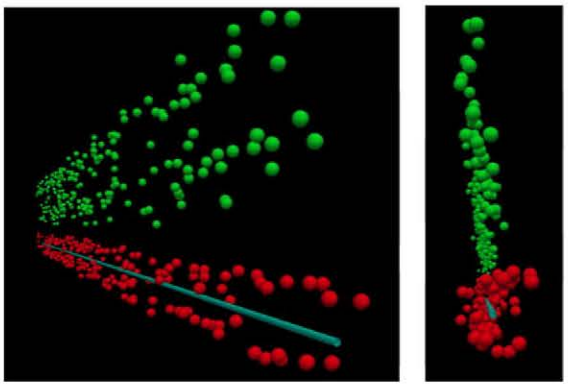


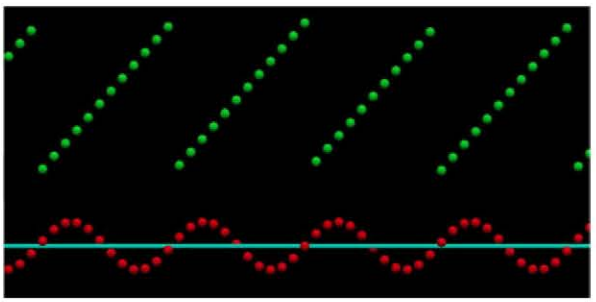
Fig. 7 For each sample i, conversion of the angular coordinate to a tangential distance in the unrolled domain is performed by multiplying θ_i by a scale factor $\rho_i = \overline{\rho}$. Please note that using an average radius has two advantages. First, it limits the effects of local variations in ρ. And second, if a fixed value $\overline{\rho}$ were not used for translating radial values to ordinates, the unrolled representations of thick and thin dendrites sections would show wide scale variations in the Y axis



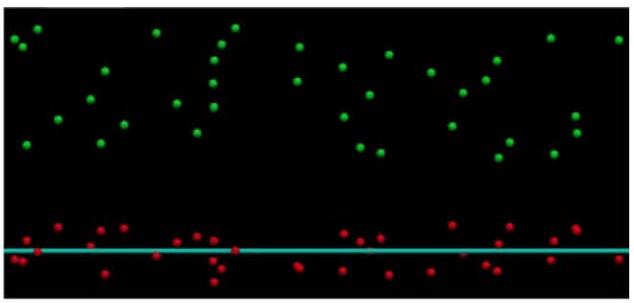
(a) Frontal view of the unrolling transform. In green, the unrolled points. In red, the same points before unrolling. The medial axis is displayed in light blue.



(b) Perspective and perpendicular or axial views of the points shown in (a) illustrating how the user perceives the straightened (almost tubular) and the unrolled data (almost planar) from different points of view.



(c) Same as (a) but for a synthetic helicoidal pattern.



(d) Same as (a) but for a random pattern.

Fig. 8 Three views of the unrolling transform applied to the data shown in Fig. 6, together with two examples of a synthetic helicoidal and random patterns. The views show the unrolling-transformed insertion points in *green*, alongside the corresponding data before straightening, plotted as a cloud of *red points*. The medial axis is displayed in *light blue*

unrolling in red (together with the straightened medial axis, which is plotted in light blue). Figure 8b displays slanted and axial perspective views of the straightened insertion points, in their original and unrolled representations. The unrolled points show a relatively small amount of radius variability that can be appreciated best in the axial view (right part of Fig. 8b). The original points in red obviously show additionally a larger scattering in angular direction, since they are placed around the light blue medial axis. Finally, Fig. 8c and d present the unroll-transformed and original insertion positions for synthetic helicoidal and random point distributions.

DISPINE's Tool

This section presents DISPINE, a prototype developed in close collaboration with the neuroscientists who are actually using it. The prototype functionality and technologies used are described below.

System Functionality

In addition to implementing the straightening and unrolling transforms, DISPINE also allows performing the following tasks that fulfill all the demands posed so far by neuroscientists for analyzing spine distributions.

3D Navigation for Point Distribution Exploration

3D Navigation for point distribution exploration is the basic functionality of the tool and the motivation behind this work, because direct exploration in 3D around raw microscopy data is not enough for extracting reliable conclusions about the distribution of spines in dendrites, as presented in Figs. 2–8.

In order to improve the analysis process, users can mix raw data together with the information produced during previous analysis stages. For example, the user may display markers such as the spines' insertion points, the spines' apex location, the dendrites' axis, a 3D cubic spline interpolating the spines' insertion points, etc. Moreover, users can display the results of applying the straightening and unrolling transforms to the raw data on a separate window.

When both views are simultaneously selected, only the insertion points are presented in the 3D Cartesian view. This simplifies the whole scene, facilitating the exploration of the original data by offering interactive panning and zooming over any of the available views.

Synthesis of Three-Dimensional Point Distribution According to User-Defined Patterns

Synthesis of three-dimensional point distribution according to user-defined patterns can be defined both in the 3D Cartesian domain and in the transformed domain. In our case, we have implemented two typical patterns that can be found in tubular structures: a random pattern and a helicoidal pattern (Fig. 8c and d). For the random pattern case, the algorithm implemented follows the standard uniform distribution, introducing a pseudo-random distortion over the 3D marker coordinates. For the spiral case, the user can select the number of cycles, the angular offset of the first loop, the radius, the axis length, and the number of samples regularly distributed along the whole spiral.

Visual Mapping of Relevant Features Associated with Markers

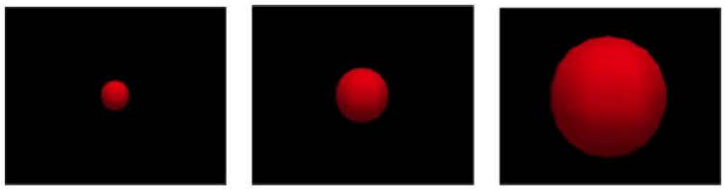
For spines, typical features under consideration are the spine size, length or orientation. We have chosen 3D shapes based on deformed spheres for representing these three variables as glyphs, assigning spine size to the shape volume, spine length to the ratio among the major and minor axis of the resulting deformed sphere, and spine orientation to the directional orientation of the 3D shape. The average values for spine size (measured as spine volume) and length have been computed for each dataset. Individual spines have been compared with this average value in order to know whether they are bigger or longer than the average (Fig. 9). As for size, those spines that are above or below the mean value are represented by a larger or smaller sphere, proportional to the deviation from the mean (Fig. 9a). For length, spheres representing spines are deformed to obtain an ellipsoid, elongated if the spine value is longer than average, or flattened otherwise (Fig. 9b).

Spine orientation is defined as the difference between the radial direction obtained when joining the insertion point with the closest point of the medial axis, denoted as $\overrightarrow{\mathbf{ro}_\mathbf{i}}$, and the orientation of the spine's medial axis extracted from the spines manually segmented, denoted as $\overrightarrow{\mathbf{so}_\mathbf{i}}$ (Fig. 10 illustrates this concept). If no deviation is found, the deformed shape is vertical in the unrolled plane (almost normal to the dendrite's surface).

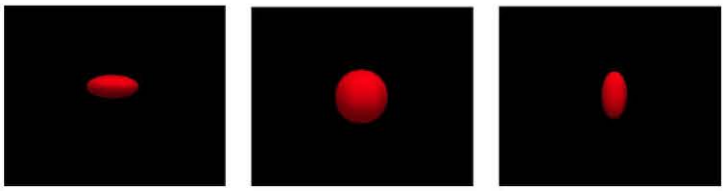
Spine orientation is the most difficult feature from the point of view of data representation, because of its high dimensionality. First, spine insertion points are represented in the original domain by their 3D coordinates, and in the unrolled domain, by two parameters (distance along the *MA* and ρ-scaled azimuth). Representing spine orientation from the insertion point outwards requires three additional parameters in the original domain, and two in the unrolled domain.

When visualizing spines in the original domain, the presence of spine and dendrite morphological data gives the user enough cues to get a clear idea about spine orientation, even though the user might not properly grasp the magnitude of other parameters which are obscured by dendrite and spine clutter. In consequence, a simple solution for maximizing visualization efficiency would be just to superpose the glyphs described in the previous paragraphs and displayed in Fig. 9a to c on top of the spine representations. The left and central images of Fig. 9d show this solution. As it can be appreciated, placing the glyphs on the spines' heads results in improved data interpretability. Regarding user navigation, DISPINE permits users to explore only raw data, only glyph data, or both, as shown in Fig. 9d.

Visualizing spine orientation in the unrolled domain is hard, because the user does not have the morphological cues mentioned above. Since eccentricity cannot always be used to represent spine direction because there might be spherical glyphs (remember that eccentricity is also used for representing spine length), spine direction is represented by the presence of a non-

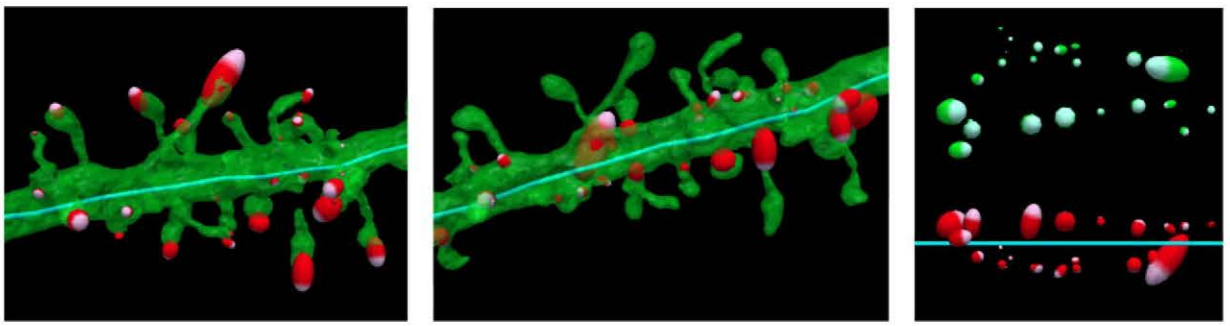


(a) Spheres deformed to represent spine size. From left to right, spines smaller, equal or larger than the mean value.



(b) Spheres deformed to represent spine length. From left to right, spines shorter, equal or longer than the mean value.

(c) 3D Shapes oriented in the unrolled space. Vertical orientation with respect the unrolling plane means no deviation from the radial direction. From left to right: 0°, 45° and 90° (approximately).



(d) Visual mapping applied on the original system of reference (left,center) and on the unrolled domain (right). Visual mapping orientation is best illustrated in left, placing the deformed shaded sphere at the spine heads instead of using their insertion points, as it has been done so far (center). Please notice that the dendrite segment in the unrolled domain has been rotated.

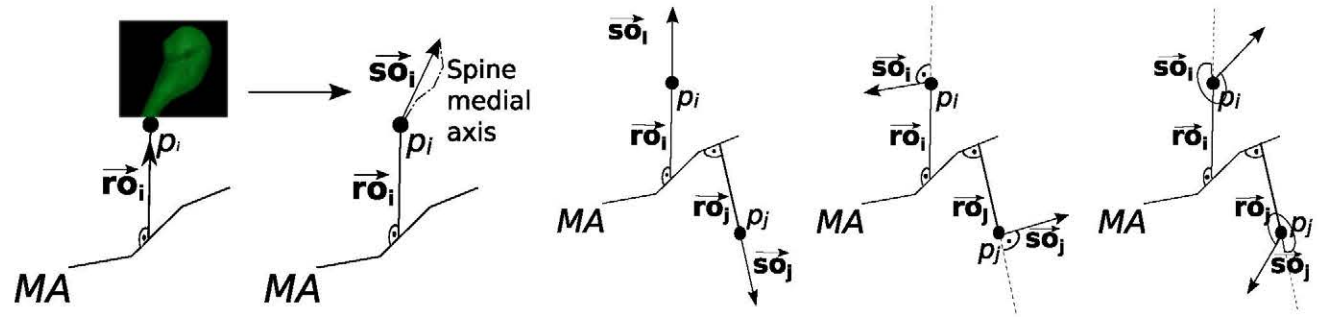
Fig. 9 Examples of glyphs used for visually mapping features such as spine size (**a**), length (**b**) and orientation (**c**). A dual view is also available for this tool functionality (**d**)

saturated color region, oriented in the same spatial direction as the spine (Fig. 9c and d).

Point Filtering

As it can be seen in Fig. 11, the user can select the data to be displayed by setting up different values. In this manner, the data which does not meet the selected parameter ranges may be filtered out. For example, users may enable or disable the visualization of spines according to their size, length, angular position in which they grow, etc. Only the selected set of spines that fall inside the interval specified by the upper and lower thresholds will be drawn. Local spatial exploration can also be performed if the user selects a region of interest defined from the straightened medial axis.

Any combination of parameter values is user-selectable for filtering out the undesired data. In consequence, the filter options provide additional help for simplifying the amount of data and the complexity of the representation, allowing the user to focus his/her attention on a specific set of samples (Fig. 11). In this particular example, regarding the spatial position, the user has selected a central region of the dendrite segment, and for the azimuth, only the samples included in the $[180^{o}, 360^{o}]$ interval.



(a) Extraction of the spine medial axis and the corresponding vector orientation $\overrightarrow{so_i}$.

(b) $\widehat{\overrightarrow{ro_i}\overrightarrow{so_i}} = 0°$.

(c) $\widehat{\overrightarrow{ro_i}\overrightarrow{so_i}} = 90°$.

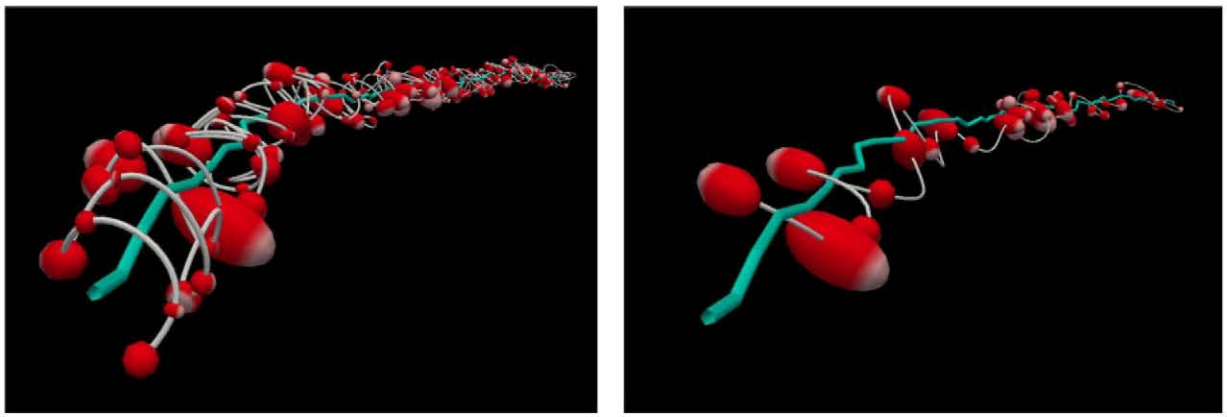
(d) $\widehat{\overrightarrow{ro_i}\overrightarrow{so_i}} = 315°$.

Fig. 10 Different cases of spine orientation defined by the angle between the radial orientation vector $\overrightarrow{ro_i}$ and the orientation of the spine's medial axis extracted from the spines, represented by the vector $\overrightarrow{so_i}$. The value of the angle computed will be the orientation of the glyph used for mapping visual features in the unrolled domain (Fig. 9d *right*). The azimuth angle is not shown, since it can be easily perceived from the glyph in the unrolled representation

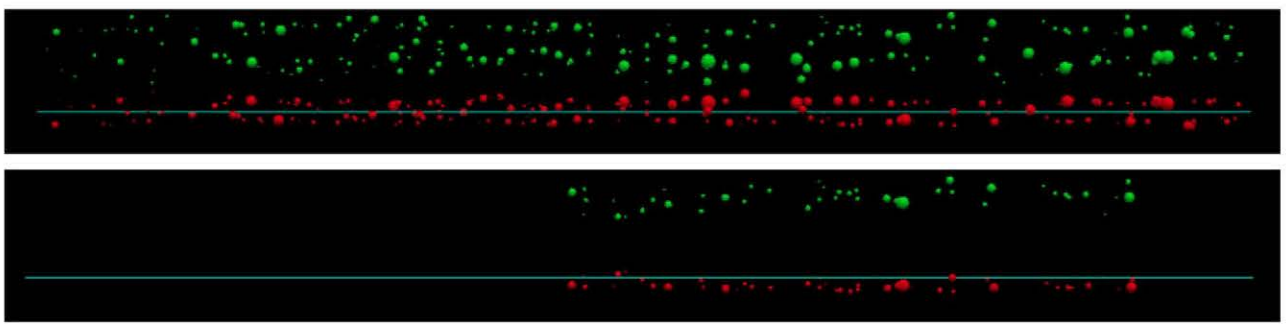
Data Clustering for Periodic Data

DISPINE offers tools for searching for repetitive patterns in the position of spine insertion points. Given the human ability for detecting patters visually, this is done by offering users the possibility of superposing data from different dendrite areas. The tool offers two options:

1. Fixing a particular origin and period with respect to X (medial axis position), and comparing spine insertion points from several different dendrite



(a) Filtering spine samples by size and length using the visual mapping representation. Left: original set of samples. Right: selected subset.



(b) Filtering spine samples by angular and spatial position. Top: original set of samples. Bottom: selected subset (only those spines located between two specific axial locations and growing between two azimuth directions are displayed).

Fig. 11 Filtering out insertion points according to spine size, length, angular position, and spatial position. The whole unfiltered data set is shown in Fig. 2

fragments at that specific period. This has been labeled *multi-period clustering*.
2. Fixing a region origin, selecting different period lengths, and comparing points from two adjacent periods. This has been denoted by *intergroup folding*.

Figure 12 shows an example of this functionality. The user has selected four cycles in the reference pattern, delimited by yellow bars, and the corresponding data samples on the unrolled domain are grouped into one cluster. Figure 12a shows several samples coming from the four different cycles taken as the period of reference. Notice the four colors assigned to the points identifying their original cycles. Inter-group folding, shown in Fig. 12b, allows to group two cycles inside the initial interval of interest. In this case, only two colors are required for representing the original pair of cycles.

For multi-period clustering, clusters can be defined by varying the frequency f of the current helicoidal pattern reference (with $f = 1/lc$, being lc the helix pitch) or by varying the number of cycles n that belong to each cluster. Defining UP as the set of unrolled points, cluster i is formed by displacing the selected set of samples to the current cycle defined as the reference:

$$pc_j(x, y, z) = p_j(x - (k \cdot lc), y, z),$$
$$\forall j \in UP, k = 0, \ldots, \left\lfloor \frac{l}{lc \cdot n} \right\rfloor - 1 \tag{12}$$

being k the ordinal cycle inside cluster i and $lc \cdot n$ the total length of the dendrite fragment under consideration.

The tool also performs inter-group folding as a help for period determination, if users want to analyze different clustering periods. Just like before,

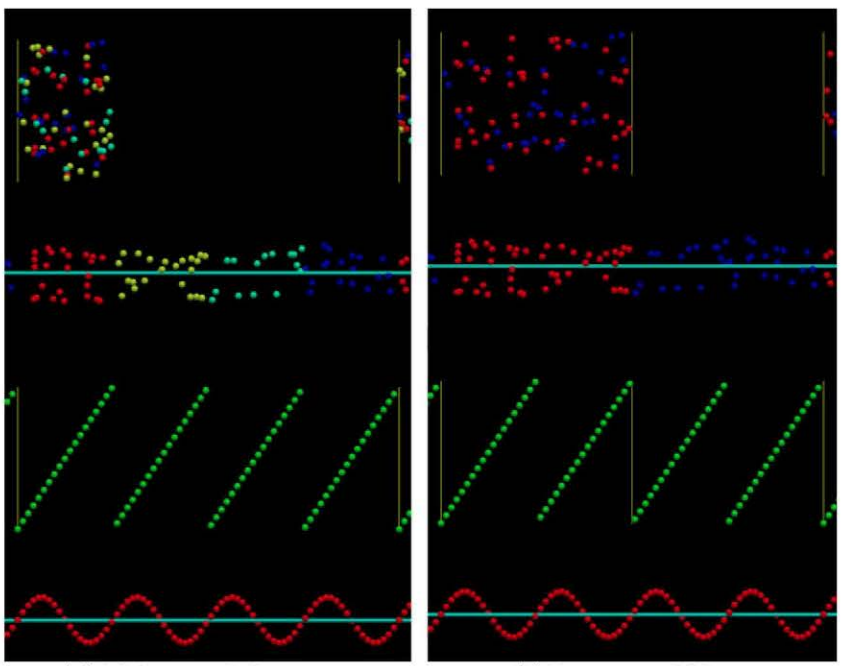
$$pc_j(x, y, z) = p_j(x - (k \cdot lc \cdot n), y, z), \tag{13}$$
$$\forall j \in UP, k = 1 - \left(\left\lceil \frac{t}{n} \right\rceil \bmod 2 \right) \tag{14}$$

being *mod* the module function, t the ordinal of the cycle that contains p_j, and i the target cluster.

Users can visualize at the same time the target helix as a reference pattern and the clustering result. They are able as well to specify an offset for the cycle starting point in the samples' analysis as a percentage value of the basic cycle length along the medial axis. Additionally, they can select a uniform color for the grouped samples, or different color codes depending on their original positions. Transparency can also be used for simultaneously visualizing some overlapping samples, because the sample density increases when different dendrite regions are grouped.

Visual Animation of the Raw Data and the Transforms Applied

Occasionally, 3D spatial interpretation is a difficult task for certain users. Animating the transforms offers thus a very helpful support for this set of experts, because



(a) Multi-period clustering. (b) Intergroup folding.

Fig. 12 Data clustering for spiral analysis. Color codes in the gathered samples are assigned depending on their original positions with respect to four different cycles (*multi-period*), or two pairs of cycles (*intergroup*), grouped in the current spiral pattern taken as the reference

it illustrates easily all the steps of the visual analysis method proposed in this work.

Standalone animations free users from taking care of interaction controls, allowing them to focus their attention on the data being examined. This visual effect is synchronized in both representation domains if the dual viewing mode is selected.

Detection of Periodic Patterns Using Sound Feedback as an Optional Aid

Sound can also be used for giving users additional cues about the data being analyzed. On account of human sensitivity to audible frequencies and rhythms (temporal repetition patterns of audible frequencies), this feedback proves particularly useful when users are trying to detect the presence of periodic patterns hidden within the analyzed data (Mansur et al. 1985). In this framework, we have selected the distance among points along the dendrite's medial axis as well as their angular position as the key parameters for translating spatial information into sound:

- Each of the chromatic scale notes has been associated with a range of values of the azimuth parameter θ by dividing its range (360°) among the twelve semitones of this scale
- The time when a particular note is being played depends on the spines' position along the medial axis.

The activation instant of the notes is determined by the spatial position of spines along the medial axis length, translating therefore axial spatial information to the time domain and fixing $t = 0$ for the axis origin. In DISPINE, the user may select both sound parameters (tones and time instants) for exploring dendrites along the medial axis, or just tone, in order to analyze the sequential distribution of the spines' insertion point angle along the spines' medial axis. Other more complex mappings for assigning audio stimuli to circular pitch are also possible (Deutsch 2010), but are beyond the goals of this work.

Technologies Used

The development method followed is very similar to *Scrum* because it is very well adapted to our environment and to the user-centered design philosophy followed in this application (Rising and Janoff 2000). This methodology enables software to be developed rapidly on the basis of small sprints, the result of each representing a further improvement in the prototype.

Another issue taken into consideration was to ensure the portability of DISPINE across a wide range of systems. We chose a set of stable and widely tested free distribution tools that made this objective easier. Thus, we chose Python vs. 2.6 as the programming language, Qt for user interface development, VTK vs. 5.4 as the visualization library, and NumPy vs 1.5.1 for linear algebra. Accordingly, DISPINE has already been successfully installed on a variety of systems with Linux and Windows.

With regard to the GUI architecture, it was conditioned both by the selected toolkits and by the required prototype functionality. The design pattern that best fitted our purposes was the observer-pattern, because we needed to synchronize multiple views of the same model taking into account the changes introduced in the model by user interaction (Gamma et al. 1995). Moreover, Qt and VTK made available a large set of widgets which enabled the establishment of view-hierarchies in order to define new specific component and container hierarchies, or else, to reuse generic ones. View-hierarchies were also applied to the management of the input, output and DISPINE layout.

Case Study: Preliminary Analysis of Cortical Dendritic Spine Distributions

The system presented here has been used for the analysis of cortical dendritic spine distributions. The use of the straightening and unrolling transforms for analyzing data has been decisive for guiding the neuroscientists' work, as well as for reaching new conclusions about the spatial patterns found in these distributions.

In the present work we intracellularly injected cells in cytoarchitectonically identified layer III of the human cingulate cortex, but the proposed method can be applied to any other type of neuron in the human brain or in any other species. The fluorescent marker Lucifer Yellow (LY) was applied to each injected cell by continuous current until the distal tips of each cell fluoresced brightly (Fig. 1a), indicating that the dendrites were completely filled and ensuring that the fluorescence did not diminish at a distance from the soma (for a detailed description of the cell injection methodology, see Elston and Rosa 1997; Elston et al. 2001; Ballesteros-Yáñez et al. 2010).

Following the intracellular injection of pyramidal neurons, the sections were stained using rabbit antisera against LY. The antibody was visualized with biotinylated goat anti-rabbit IgG and streptavidin-conjugated Alexa fluor 488. The sections were then studied with the aid of a Leica TCS 4D argon/krypton mixed-gas

confocal scanning laser attached to a Leitz DMIRB fluorescence microscope. Fluorescent labeling profiles were imaged using an excitation peak of 491 nm to visualize Alexa fluor 488.

Stacks of images were acquired at high magnification (63× with an oil-immersed objective) to build volumetric reconstruction of a dendrite (Semwogerere and Weeks 2005).

The reader may find a video at http://www.datsi.fi.upm.es/~arodri/cbb/ninf11/dispine_video.wmv showing a typical working session where the user runs the following sequence of actions after loading the data file:

1. Interactively explore the dendrite, looking for relevant features in the image stack.
2. Animate the visualization, adding rotational motion to the elements of the 3D scenario.
3. Add insertion points and additional visual references (medial axis of the dendrite and interpolation curve) in the 3D view of the stack.
4. Obtain the straightening transform.
5. Compute the unrolling transform.
6. Define a new distribution pattern for synthetic data.
7. Interactively examine and filter out points on both domains, 3D Cartesian and transformed, either with static views or with animated visualizations, according to different criteria: spine size, length, angular position or spatial position along the medial axis.
8. Perform visual mappings of the morphological features extracted from spines.
9. For periodic patterns, perform visual data analysis in the unrolled domain taking into account spatial frequency sampling.
10. Animate the path of the spines along the medial axis receiving audio feedback about the spatial distributions examined.

Once neuroscientists have completed the exploration of the dendritic spine distribution under study, they can extract hypothesis about the morphological features of the structure being analyzed. Considering the example presented in Fig. 2, we can visually compare the two synthetic patterns defined as references with the data extracted from the segment shown. Examining Figs. 8–11 in the way in which a domain expert would do, it seems at first sight that this distribution is closer to a random distribution, discarding the hypothesis of a structured pattern such as the helicoidal distribution. If we now focus our attention at the clusterings presented in Fig. 12, we can easily observe again that the gathered samples do not resemble the helicoidal pattern, but they look rather like the random distribution pattern. However, if we look at the original 3D distribution shown in Figs. 2 and 6 (left column), a neuroscientist has no evidence about the existence or not of any pattern, preventing him to reach a firm conclusion with such data.

Discussion and Future Work

At times, it may be difficult to detect 3D features when the object geometry is masked by non-rigid transformations such as those found in real dendrites. In these cases, moving to a simpler transformed domain can be an effective way to simplify the analysis process.

This paper presents a methodology for assisting neuroscientists to study the 3D organization of spine insertions in dendrites, a task that is important for studying dendritic spine distribution patterns. This may prove critical not only for better understanding the brain's synaptic organization, but also for detecting differences among neuronal circuits under different circumstances.[2] Thus, we propose that the method described here can be of general interest because it represents a substantial help for the analysis of the dendritic spine organization.

The method at issue is based on two transforms, straightening and unrolling, which have been applied to spine insertion points scattered along the dendrites' surface. These two transforms were used to map the 3D position of points into a two dimensional space.

The creation of an unfolded arrangement such as the one proposed enables a straightforward visualization and analysis of these structures. As pointed out in the introduction, it manages to avoid the difficulties of performing the analysis directly in 3D while definitely improving the user's insight. The result of both transforms allows users to enhance the interpretability of data without losing any of the properties present in the original tubular data, such as its spatial ordering, or the spatial relationships among neighbor points or point attributes. Neuroscientists working with this type of data have argued that, from their experience, exploring these properties directly in 3D is not possible because visual references are very difficult to establish using point-based representations. This can be noticed in the video accompanying Fig. 1 or in the examples presented

[2]For example, the morphology, number and density of dendritic spines are altered in many brain diseases and under several conditions such as malnutrition, alcohol or toxin exposure (Fiala et al. 2002).

throughout the paper. In any case, as it has been shown in the previous sections, the original data remains available, so the users contrast their hypotheses within both representation domains at all times.

Central to the methodology presented here is also the design of DISPINE, an interactive environment that permits exploring 3D data in various reference systems, filtering out insertion points according to different spine features, both morphological and spatial. Additionally, DISPINE supports the following options:

- Pattern synthesis (random and helicoidal distributions).
- Periodic spatial pattern analysis applied to spirals.
- Simple visual mapping creation (spine size, length and orientation).
- Audible feedback.

All these options increase the resources available to users for analyzing such complex 3D data.

The tests performed with end users, together with the feedback provided by them, show that this tool can effectively help analyzing complex tubular structures such as spines' insertion points in dendrites. It can also be applied to different brain regions, species, and experimental conditions (healthy vs. abnormal, young vs. old, etc.), and it can additionally be applied to other scientific fields, where this kind of tubular point distributions can be found. Further experiments with other datasets, like the one presented in this paper, will allow neuroscientists to easily and quickly explore new hypotheses about 3D neuron morphology. In turn, this will provide support in deciding whether additional statistical analysis will be required for demonstrating or refuting specific hypothesis (Morales et al. 2012). Furthermore, this tool will help improving productivity by incorporating new interactive aids in Neuroscience; for example, the use of this tool has been fundamental for guiding the statistical analysis performed in Morales et al. (2012).

DISPINE has been tested in several laboratories from different institutions, located in a number of countries. The feedback provided by users has been very positive, supporting the evidence that this tool has opened new possibilities for analyzing spine distributions in ways that were not possible before because of a lack of tools for performing this kind of analysis. In general, users have evaluated also very positively DISPINE's functionality and ease of interaction.

Future work will be devoted to the improvement of the tool usability by introducing local interactions for selected regions of interest, and to the development of additional tools for characterizing spine distributions, such as the inclusion of stereoscopic rendering. It must be noticed that we consider stereoscopic vision just an additional aid for improving depth discrimination, although the complexity of examining the distribution patterns of dendritic spines requires new methods to simplify this task, and not just new technologies for visualizing data in 3D.

Also, future work will consider the possibility of including user-defined azimuth references in order to keep the orientation consistency between the straightened, unrolled projections, and the 3D domain (its effect over the straightened dendrite will be equivalent to perform a torsion along its medial axis in order to assign azimuth uniform values to spines directed towards the same region in 3D space).

Information Sharing Statement

The software will follow a free distribution policy under the GPL licence. It can be downloaded together with the dataset shown in the paper from http://cajalbbp.cesvima.upm.es/dispine. All the videos referenced in this work will be also available at the same url.

Acknowledgements This work was supported by grants from the following entities: Centre for Networked Biomedical Research into Neurodegenerative Diseases (CIBERNED, CB06/05/0066) and the Spanish Ministry of Education, Science and Innovation (grants BFU2006-13395; SAF2009-09394 to Javier DeFelipe; TIN2010-21289 and the Cajal Blue Brain Project, Spanish partner of the Blue Brain Project initiative from EPFL).

Arellano, J. I., Benavides-Piccione, R., Defelipe, J., & Yuste, R. (2007). Ultrastructure of dendritic spies: Correlation between synaptic and spine morphologies. *Frontiers in Neuroscience, 1*, 131–143.

Ballesteros-Yáñez, I., Benavides-Piccione, R., Bourgeois, J. P., Changeux, J. P., & DeFelipe, J. (2010). Alterations of cortical pyramidal neurons in mice lacking high-affinity nicotinic receptors. *Proceedings of the National Academy of Sciences of the United States of America, 107*(25), 11,567–11,572. http://www.biomedsearch.com/nih/Avlterations-cortical-pyramidal-neurons-in/20534523.html.

Bitplane (2011). Imaris. Web. http://www.bitplane.com/go/products/imaris.

Borgefors, G., Nyström, I., & Baja, G. S. D. (1999). Computing skeletons in three dimensions. *Pattern Recognition, 32*(7), 1225–1236. doi:10.1016/S0031-3203(98)00082-X.

DeFelipe, J. (2010). From the connectome to the synaptome: An epic love story. *Science, 330*(6008), 1198–1201. doi:10.1126/science.1193378.

Defelipe, J., & Fariñas, I. (1992). The pyramidal neuron of the cerebral cortex: Morphological and chemical characteristics

of the synaptic inputs. *Progress in Neurobiology, 39*(6), 563–607. doi:10.1016/0301-0082(92)90015-7.

Deutsch, D. (2010). The paradox of pitch circularity. *Acoustics Today, 6*(3), 8–14. doi:10.1121/1.3488670. http://link.aip.org/link/?ATC/6/8/1.

Donohue, D. E., & Ascoli, G. A. (2011). Automated reconstruction of neuronal morphology: An overview. *Brain Research Reviews, 67*(1–2), 94–102.

Eick, S. G., & Karr, A. F. (2002). Visual scalability. *Journal of Computational and Graphical Statistics, 11*(1), 22–43.

Elston, G. N., & Rosa, M. G. (1997). The occipitoparietal pathway of the macaque monkey: Comparison of pyramidal cell morphology in layer III of functionally related cortical visual areas. *Cerebral Cortex, 7*(5), 432–452. http://www.biomedsearch.com/nih/occipitoparietal-pathway-macaque-monkey-comparison/9261573.html.

Elston, G. N., Benavides-Piccione, R., & DeFelipe, J. (2001). The pyramidal cell in cognition: A comparative study in human and monkey. *Journal of Neuroscience, 21*(17), RC163(1–5). http://view.ncbi.nlm.nih.gov/pubmed/11511694.

Fiala, J. C., Spacek, J., & Harris, K. M. (2002). Dendritic spine pathology: Cause or consequence of neurological disorders? *Brain Research Reviews, 39*(1), 29–54. http://view.ncbi.nlm.nih.gov/pubmed/12086707.

Fuchs, R., & Hauser, H. (2009). Visualization of multi-variate scientific data. *Computer Graphics Forum, 28*(6), 1670–1690.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design patterns: Elements of reusable object-oriented software*. Reading, MA: Addison-Wesley Professional.

Kasai, H., Fukuda, M., Watanabe, S., Hayashi-Takagi, A., & Noguchi, J. (2010). Structural dynamics of dendritic spines in memory and cognition. *Trends in Neurosciences, 33*(3), 121–129. http://www.ncbi.nlm.nih.gov/pubmed/20138375.

Lakshmi, J. K., & Punithavalli, M. (2009). A survey on skeletons in digital image processing. In *2009 international conference on digital image processing* (pp. 260–269). doi:10.1109/ICDIP.2009.21.

Mansur, D. L., Blattner, M. M., & Joy, K. I. (1985). Sound graphs: A numerical data analysis method for the blind. *Journal of Medical Systems, 9*(3), 163–174 (1985). doi:10.1007/BF00996201.

Meijering, E. (2010). Neuron tracing in perspective. *Cytometry. Part A, 77*(7), 693–704. doi:10.1002/cyto.a.20895.

Morales, J., Benavides-Piccione, R., Dar, M., Rodríguez, A., Bielza, C., Fernaud, I., et al. (2012). *Dendritic spine organization on the human cerebral cortex* (p. 18, under review).

O'Brien, J., & Unwin, N. (2006). Organization of spines on the dendrites of purkinje cells. *Proceedings of the National Academy of Sciences of the United States of America, 103*(5), 1575–80. http://www.biomedsearch.com/nih/Organization-spines-dendrites-Purkinje-cells/16423897.html.

Portera-Cailliau, C., Pan, D. T., & Yuste, R. (2003). Activity-regulated dynamic behavior of early dendritic protrusions: Evidence for different types of dendritic filopodia. *Journal of Neuroscience, 23*(18), 7129–7142. http://www.jneurosci.org/cgi/content/abstract/23/18/7129.

Rising, L., & Janoff, N. S. (2000). The Scrum software development process for small teams. *IEEE Software, 17*(4), 26–32 (2000). doi:10.1109/52.854065.

Semwogerere, D., & Weeks, E. (2005). Confocal microscopy. In G. Wnek, & G. Bowlin (Eds.), *Encylopedia of biomaterials and biomedical engineering*. New York: Taylor & Francis.

Spruston, N. (2008). Pyramidal neurons: Dendritic structure and synaptic integration. *Nature Reviews Neuroscience, 9*(3), 206–221. http://www.ncbi.nlm.nih.gov/pubmed/18270515.

Yuste, R. (2010). *Dendritic spines*. Cambridge, MA: MIT Press.