

# Temporal Matching of Dendritic Spines in Confocal Microscopy Images of Neuronal Tissue Sections

Kishore R. Mosaliganti<sup>1,2</sup>, Firdaus Janoos<sup>1</sup>, Xiaoyin Xu<sup>3</sup>, Raghu Machiraju<sup>1,2</sup>, Kun Huang<sup>2</sup>, Stephen T. C. Wong<sup>3</sup>

<sup>1</sup>Department of Computer Science and Engineering, The Ohio State University, Columbus, OH, USA

<sup>2</sup>Department of Biomedical Informatics, The Ohio State University, Columbus, OH, USA

<sup>3</sup>Harvard Center for Neurodegeneration and Repair, Harvard Medical School, Boston, MA, USA

**Abstract**—Recent research has revealed that morphological characteristics of neuronal structure are closely related to normal neural functions such as learning and memory. Neuronal dendrites and their spines define the inter-connectivity of the neural network and hence, are important predictors of its function. Many neuronal functions are observed to be correlated with the appearance or disappearance of neural structures. In this paper, we temporally track the evolution of spines on the dendrite. Three-dimensional images were acquired by the digitization of neuronal tissue sections using a two-photon laser scanning microscopy with a 40X objective, 0.8 NA, over transgenic mice expressing EGFP. We describe a pipeline of processing algorithms for solving the spine correspondence problem automatically. Our framework requires the segmentation and extraction of the skeleton of the dendrite that is aligned with a rigid registration algorithm. We then build a graph model that is spatially aligned with the dendrite, the nodes of which represent the spine branches. A maximum likelihood estimation framework is employed to match graph models from different time-points. We report the results from the automated matching and validate it with manual measurements. The sensitivity and specificity of the pipeline is evaluated with available ground-truth.

**Index Terms**—neuroinformatics, quantification of dendrites spines morphology, registration, image segmentation, medial axis, skeleton, maximum likelihood estimation.

## I. INTRODUCTION

Recent experiments have revealed that important aspects of cognitive function, such as experience-based learning [1], attention and memory [2] are correlated with variations in dendritic branching morphology and with spine density and distribution [3].

Capowski [4] has provided a detailed history of neuron tracing methods. Many of the current methods are semi-automatic. Optical microscopes are enhanced with a computer imaging system that allows a user to perform segmentation, and conduct various measurements in an interactive fashion [5]–[7]. Koh *et al.* [8] present a geometric approach for automatically detecting and quantifying the three dimensional structure of neurons using a set of heuristics as defined in [9]. More recently, Al-Kohafi *et al.* [10] presented a different approach for neuron tracing that has been inspired by vasculature tracing algorithms.

Our efforts differ from the above approaches in two ways. Firstly, we are interested in structures that exist at a smaller

scale. We are especially interested in the detection of spine structures on the dendrite rather than the entire neuron cell with the dendrites and the soma. Secondly, we wish to track the temporal change in the spine morphological structures in 3D and make comparative measurements. Hence, our problem comprises of detection, tracking and measurements. The segmentation and measurement steps have been well documented in literature. In this paper, we focus on developing a tracking algorithm that draws on a partial segmentation but does not necessitate the complete delineation of the spine structures with exact boundaries.

We now enumerate the three main stages of our pipeline.

**Segmentation:** We present a processing pipeline that eliminates the noise and artifacts in the image. The accomplished segmentation suffices only to the extent that temporal matching may be accomplished. We acknowledge that more robust segmentations using aforementioned techniques is possible. Our tracking method is agnostic of the segmentation method used. Our pipeline includes Otsu’s thresholding, median filtering, contour smoothing filters, medial axis extraction and finally spine skeleton extraction. Locations of the branches are recorded on the skeleton for later use in the matching stage.

**Registration:** Smooth skeletal curves extracted from the segmentation stage are overlapped in a single image plane and rigidly registered using the iterative closest points (ICP) algorithm [11]. The skeletons suffer from large translational and minor rotational movements that are corrected. This initialization is important prior to matching.

**Matching:** Based on the position of the spines on the dendrite, a linear graph model in the image space is constructed. Each node of the graph represents a branch-point in the neuron where a spine possibly emerges from the dendrite. The detection of these branch points or graph nodes is performed in the segmentation stage itself. A graph node encodes a set of attributes consisting of location coordinates, local curvature of the skeleton, orientation of the spine and the location on the parameterized skeleton curve. Given two graph models from different time-steps, a maximum likelihood estimation (MLE) framework assigns probability vectors to each branch point. A dynamic programming algorithm then chooses the pairwise matchings of nodes until all nodes are exhausted. The residual (unmatched nodes) may be analyzed for new emerging spines.

We demonstrate our results on four 3D confocal microscopy data sets. Our framework is compared against manual validation (the correspondence was determined by a pathologist) on

Corresponding author: Kishore Mosaliganti, email: mosaligk@cse.ohio-state.edu

one data set. Sensitivity and specificity ratios are computed and plotted.

Thus, we obtain robust temporal matching of spines on the dendrite. The rest of the paper is organized as follows. In section II, we describe related work in neuronal structure detection, registration and temporal matching. Section III motivates and explains different stages of the image processing pipeline that we employ. We represent the desired neuronal structure using a graph model. In section IV, the MAP framework with dynamic programming for matching graph models is explained, while in section V reports on the implementation and the results obtained. Finally, in section VI, we provide a summary and describe our plans for the future.

## II. RELATED WORK

The advances in modern imaging techniques have required studies to be conducted on large amounts of data. Early research in neuronal structure detection and estimation [12]–[15] has focussed on developing automated techniques that cause the results to be objective and reproducible with little manual intervention.

A common feature in most of the above algorithms is the use of the skeleton structure for making measurements. Rusakov and Stewart [12] applied a medial axis construction to 2D dendritic images. As a result, they were able to compute the length of the neuronal spine in 2D and used that to estimate the lengths in 3D. The medial axis is very sensitive to noise and fidelity of the segmented contour, hence manual intervention was still an issue. Watzel *et al.* [13] extended the above work to a 3D setting. However, the images were restricted to consist of a single dendrite. Candidate spines were detected based on the branching medial axis spurs above a length tolerance. True spines were distinguished from spurious ones using a length tolerance.

Koh *et al.* [8] present a more general skeletonization approach. They assume that the image to be analyzed is of a bi-phase medium, with one phase being the neuronal cytoplasm (dendritic phase) and the other being the background tissue. The method uses a geometric approach, detecting spines as protrusions and contains several parameter settings that require extensive fine tuning. There is no limitation on the number or the structure of the dendrites in the image. In addition to spine length and density, volumetric measurements and spine classifications were obtained. Cohen *et al.* [16] formalized the skeletonization approach by using a graph theoretic representation. The graph was spatially aligned to the neuron in 3D. Connectivity analysis, critical point identification were a part of the framework for constructing the graph. The graph model was sufficient to compute different measurements such as topological analysis, length measurements, diameters and tortuosities.

Other approaches have been inspired by using a model to fit the neuronal structure. Herzog *et al.* [14] employed a 3D reconstruction technique using a parametric model of cylinders with hemispherical ends to fit the shape of the dendrites and the spines. In this method, short spines or spines with thin necks were hard to detect and had to be manually added to

the model. Al-Kofahi *et al.* [10] reported on a state-of-the-art method that uses a generalized 3D cylinder model that follows the neuronal topology recursively for identifying the entire neuron cell including the dendrites and soma portions. Each stage is guided by a set of 4 directional kernels in 3D. Additional modifications, such as dynamic adaptation of the correlation kernels, and adaptive step size estimation, were introduced for achieving robustness to photon noise, varying contrast, and apparent discontinuity and/or hollowness of structures. An approach using neural network recognition for spines [15] has also been suggested.

Meijering *et al.* develop a semi-automatic neurite tracing technique in which individual neurites are detected and traced over a spatial sequence of images. This approach concerns itself with the entire neuron and is not the temporal problem that we seek to solve.

These approaches now place our work in perspective. We wish to model and track the spines on a single dendrite image. Modern approaches such as [10] are superior in terms of speed, automation, and robustness compared to skeletonization-based methods for tracing larger-scale neuronal structure. Smaller structures such as spines still benefit from a skeletonization approach on a localized basis. We also wish to point out that our ultimate goal is to track spine structures on a dendrite as they evolve in time. Hence, a model based representation is essential to understand how the neuron changes with time. Intuitively, the graph theoretic representation similar to [16], that is spatially aligned with the skeleton seems ideal for our purposes. It is easier to think of aligning two graph structures derived from different time-points based on the branching points.

While the temporal tracking of spine/ dendrite structure has been discussed in the neural literature, related work in the tracking of microscopic structures exists in other fields also. Sbalzarini *et al.* [17] presents a computationally efficient, two-dimensional, feature point tracking algorithm for the automated detection and quantitative analysis of particle trajectories as recorded by video imaging in cell biology. The tracking process requires no a priori mathematical model for the motion, is self-initializing, discriminates spurious detections, and can handle temporary occlusion as well as particle appearance and disappearance from the image region. Bonneau *et al.* [18] present an approach for tracking quantum dot (QD) trajectories in semi-conductors. Instead of a frame by frame analysis, it is based on perceptual grouping in a spatiotemporal volume. Individual molecular trajectories of the QD's as unstructured set of points are considered as minimal paths in a Riemannian metric derived from the fluorescence image stack. These paths are computed with a variant of the fast marching method. On the other hand, our image stacks have about 5-6 time-points and therefore require a frame by frame analysis. Furthermore, there is little correlation in the movement of the spines over successive time-frames. Nevertheless, we shall consider an evaluation of these techniques for comparison in our future work.

### III. IMAGE PROCESSING

#### A. Data Acquisition

For high content neuron screening, a fluorescence label, such as green fluorescent protein (GFP), is often used to mark neurons *in vitro*. GFP absorbs blue light and converts it to green light which is of lower energy. The emitted green light can then be captured by an optical microscope such as the confocal laser scanning microscopy (CLSM) or a two-photon laser scanning microscope (2PLSM). Imaging reveals morphological or functional details of the specimen. Three-dimensional images were acquired by the digitization of neuronal tissue sections using a two-photon laser scanning microscopy with a 40X objective, 0.8 NA, over transgenic mice expressing EGFP. Image stacks of 512x512x17 were generated every 24 hours for 5 days.

The nature of the imaging process, photobleaching effects, long time gap between imaging periods and the elastic nature of the tissue sections introduces the following types of artifacts that require additional processing steps in the segmentation process:

- *Shot Noise*: Fluorescence detection in image acquisition is performed by photomultiplier tubes (PMT). PMT are high gain, low sensitivity devices which, combined with the low number of photons emitted by small structures, such as dendritic spines, produces *photon shot noise*.
- *Photobleaching effects*: Heterogenous photobleaching of fluorescence emission during confocal laser scanning is induced by fundamental differences in exposure rates and times at different  $z$ -planes orthogonal to the optical axis.
- *Presence of unrelated structures*: Often, the region being scanned might consist of a portion of another dendrite or unrelated tissue structure. Floating tissue fragments may also be present on the slide at the time of scanning. These cause problems during the medial axis computation.
- *Orientation differences*: There are significant differences in the positions of the dendrite across time-points. This may be due to the non-constant positioning of the scanning head over the slide. The varying position requires rigid registration for alignment.
- *Floating spine heads*: Owing to the photobleaching effect, spine heads are often seen as detached from the main dendrite structure that has a leading protrusion in the direction of the spine head. The spine head has to be accounted for in building the graph model and should not be eliminated as an unrelated structure.
- *Tissue stretching and contraction*: The elastic nature of the tissue causes it to deform locally in response to stimuli. The long gap between successive scans show deformation in the dendrite that needs to be accounted for in the matching process.

#### B. Segmentation

An overview of the image processing pipeline is shown in Figure 1. Our goal is to provide a spatially aligned graph representation to the matching algorithm for each time frame.

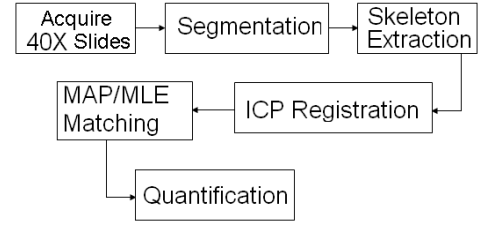


Fig. 1. Image processing pipeline

The graph model, in turn, requires the determination of the skeleton curve with the relative spine branch locations. Hence, the segmentation framework must provide the foreground object (dendrite) whose medial axis and skeleton may be computed. The segmentation consists of the following 2 steps:

1) *Otsu Thresholding*: A maximum intensity projection (MIP) of the image stack is computed along the  $z$ -axis. At this point, we are interested in matching the neurons along a single image plane. This is because of the limited resolution available along the  $z$ -axis (17 slices) does not permit a full 3D delineation of the structure. Moreover, visualization of the individual slices reveals small occurrence of changes. Formally, the MIP image  $P_z$  is expressed as

$$P_z(x, y) = \max_{0 \leq z \leq N} I(x, y, z) \quad (1)$$

Here  $N = 17$ . An automated threshold  $T$  is computed using the Otsu method [19] that separates the image into two classes. The algorithm assumes a bimodal histogram  $H$ . The threshold  $T$  is chosen such that the *weighted within-class variance*  $\sigma_w$  is minimized.

$$\sigma_w^2(t) = q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t) \quad (2)$$

where  $\sigma_1(t)$ ,  $\sigma_2(t)$  are the standard deviations of the classes for a given threshold  $t$  and  $q_1(t)$ ,  $q_2(t)$  are the weights computed from  $H$ . The optimal threshold is selected as:

$$T = \min_{0 \leq T \leq 255} \sigma_w(t) \quad (3)$$

and the final segmented images  $S$  is given by:

$$S(x, y) = \begin{cases} 0, & \text{if } P(x, y) < T \\ 1, & \text{if } P(x, y) \geq T \end{cases}$$

Figure 2a shows an example of a maximum intensity projection image. Note that the neuron is a slender long tissue structure with small protruding spine heads. The spine heads are apparently disjoint from the main structure although that is not so in reality. We would like to obtain a segmentation of this entire structure with no loss in the spine heads. Further more, the entire image consists of only the neuron tissue and the background. For this case, the Otsu thresholding framework provides acceptable results. Figure 2b shows the output of the automated thresholding algorithm. We notice the presence of *photon shot-noise*. In a rigorous implementation, the raw image may be deconvolved based on the point-spread function of the detector. In practice a standard median filter with a window size of 5x5 is used, with acceptable results.

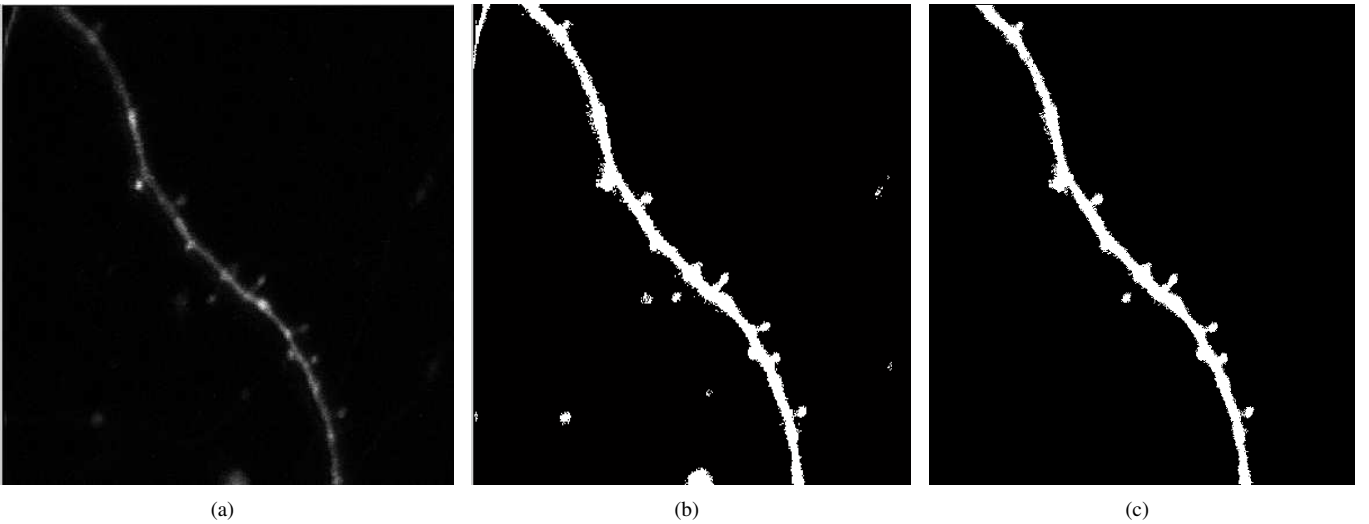


Fig. 2. (a) Maximum Intensity projection along z-axis (b) Otsu thresholding for foreground extraction (c) Connected components image of the main dendrite with a detached spine head

2) *Connected Components*: Earlier, we briefly mentioned the occurrence of unrelated tissue fragments in the images. For example, in Figure 2b, we notice some components that are definitely not associated with the dendrite. These components are detached and located far enough from the neuron. An Euclidean distance map  $D_S$  of the background as a function of the main dendrite component is computed [20]. All structures that lie more than a pre-determined threshold  $T_D$  in the distance map are eliminated. The spine structures remain preserved. Formally, if  $p$  is a pixel in  $S$ , then

$$S_c(p) = \begin{cases} 0, & \text{if } D_S(p) < T_D \\ 1, & \text{if } D_S(p) \geq T_D \end{cases}$$

Figure 2c shows the final segmented image  $S_c$  with the unrelated fragments removed by this method.

### C. Medial Axis Extraction

As we discussed in Section II, the skeleton has been a popular choice for modeling a slender, curved structure such as the dendrite. However, a formal definition of a skeleton is not available in the literature. It is well-known that skeletons computed from heuristics such as "boundary-peeling" have topological artifacts. They are best computed by starting with the medial axis. The medial axis is the locus of the centers of maximally empty balls. Figure 3a shows the medial axis computed from the segmented image  $S_c$ . It is easy to observe that medial axis is replete with noise. It exhibits several protrusions that are generated from the noisy boundary contour. Using a set of hierarchical contour smoothing filters [21] based on a Gaussian kernel, we process the segmented images. The standard deviation  $\sigma$  of the filter is set to 15 pixels. The medial axis from a smooth contour is shown in Figure 3b. At this point, the branches arising from the spine are primarily present, albeit, there are still some spurious ones. We would like to extract the skeletal backbone with the true spine branches. The floating spine heads are re-attached to the

medial axis by joining the centroid of the spine head with the nearest located medial axis branch.

A tree representation of the medial axis is then computed. Each node in the tree has pointers to its neighbors (8-connected) if they belong to the tree. A depth first search is conducted starting from the base node. Each node is assigned a visit number when it is visited. Therefore, the neighbors of each node are either its children or a parent depending on the visit numbers. Once the depth first search is completed, every node is checked to see if it belongs to the skeleton or not. This is accomplished by looking recursively at all its children and proceeding along the longest running branch. The other branches either are spine or spurious branches. This is determined by setting thresholds on the depths ( $T_d = 20$ ). The spurious branches usually have short extensions and may be pruned. Figure 3c shows the obtained skeleton curve. The curve is smooth and continuous. Candidate spine locations are marked out on the skeleton albeit some of them are still spurious. We shall rely on our probabilistic matching framework to identify the spurious locations and not match them with a true spine.

### D. Iterative Closest Point Registration

Figure 4a shows the overlap of color-mapped segmented images from all time-points prior to registration. It is easy to observe that there is considerable shift in their location owing to different scanning positions. We would like to rigidly initialize the images so that spatial correlation is maximized at the matching stage. Automatic landmark selection based methods are also not appropriate since the tissues are elastic and there is a perceptible amount of local deformation. Moreover, the landmarks, which are the spines, are detected along with the spurious ones. Image-based registration algorithms have convergence problems on account of the noise. The metric space is not smooth for optimization convergence. Therefore, we chose to use the Iterative Closest Point (ICP) registration algorithm

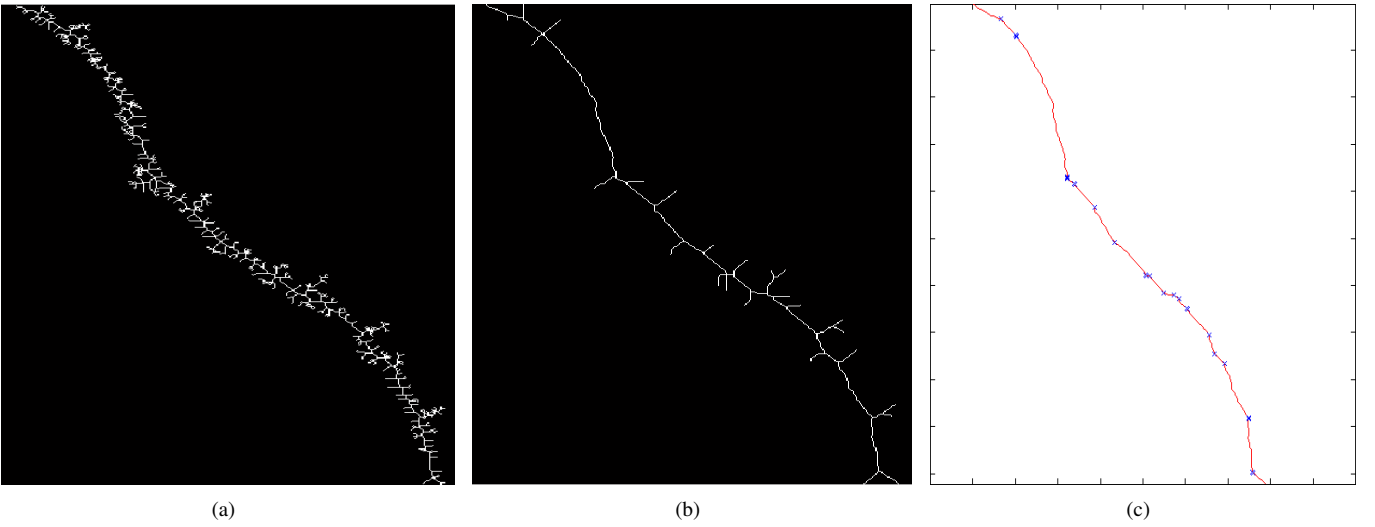


Fig. 3. (a) Medial axis of the segmented image (b) Medial axis after contour smoothing (c) Extracted smooth skeleton with the spine locations marked

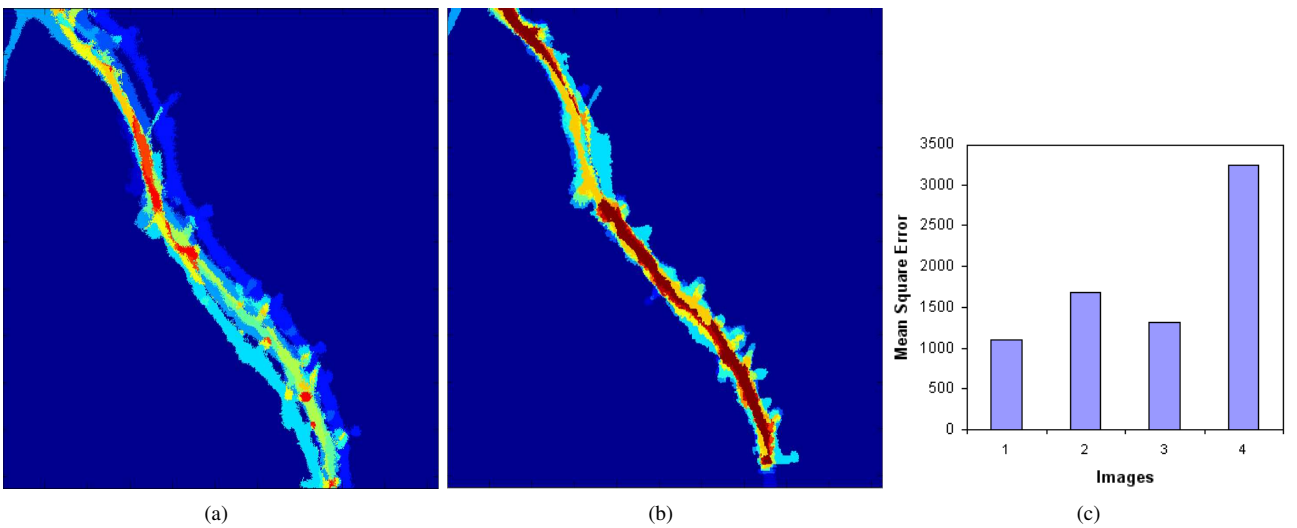


Fig. 4. Overlapped images (a) before and (b) after ICP registration of the skeletons (c) Measurement of the mean square error in the segmented images after ICP registration of the skeletons. The last image suffers from non-linear deformation and therefore has a high value.

[11] for matching skeletons. The matching of skeletons gives us the best localization rigidly prior to the matching stage.

We realize correspondence through an optimization process that searches for a rigid transform that allows the closest similarity between consecutive skeletons. The input is a set of stationary ( $P_S$ ) and moving ( $P_M$ ) points drawn from skeleton curves. For the neuron images, we register each segmented skeleton to its predecessor, and the pairwise transforms were merged sequentially to obtain each image's global transform. We use the Levenberg-Marquardt optimization method in the Euclidean distance metric space. Figure 4b shows the obtained localization in the images across all time-points. The main body of the dendrite seems to overlap well. We now need to match the spines on each of the dendrites.

#### IV. MATCHING FRAMEWORK

To match the spine locations between two dendrites at epochs  $T_1$  and  $T_2$ , we build an attribute graph representation

of the dendrites called the *dendrite graph* from the registered skeleton, as described in section IV-A. We then determine a graph homomorphism between the two dendrite graphs  $\mathbb{D}_1$  and  $\mathbb{D}_2$ , using a maximum likelihood estimation (MLE) based framework.

##### A. Dendrite Graph Model

The **dendrite graph**  $\mathbb{D}(\mathbb{V}, \mathbb{E})$  of a dendrite is an attributed graph constructed from the registered skeleton of the dendrite. Here,  $\mathbb{V}$  is the set of vertices, where each vertex represents the attributes of a branch point. The set of edges  $\mathbb{E}$  represents the adjacency relationship between the branch points. Formally, the set of  $n$  vertices is:

$$\mathbb{V} = \{v_1 \dots v_n | v_i \equiv (\bar{r}_i, \kappa_i, s_i, \theta_i), v_i \in \mathbb{R}^5\} \quad (4)$$

where  $\bar{r} = (x, y)$  is the 2D Euclidean position of the corresponding branch point,  $\kappa$  is the 2D local curvature,  $s$  is the

arc length parametrization of the branch point (see section IV-A.1), and  $\theta$  is the orientation of the spine with respect to the skeleton (see section IV-A.2).

Formally, the set of edges of the dendrite graph is:

$$\begin{aligned} \mathbb{E} &= \{(v_i, v_j)\} \text{ where,} \\ \text{degree}(v_1) &= 1, \\ \text{degree}(v_n) &= 1, \\ \text{degree}(v_i) &= 2, 1 < i < n \end{aligned} \quad (5)$$

Therefore, each non end-point vertex is connected to exactly two other vertices by edges, and each end-point vertex is connected to exactly one other vertex, forming a linear acyclic graph.

1) *Arc Length and Curvature*: We determine arc length and curvature of the branch points along the skeleton according to the method given by Mokhtarian and Mackworth [21]. A planar curve can be represented in the parametric form as  $\mathbf{r}(u) = (x(u), y(u))$ . The function  $r(u)$  is a parametric representation of the curve, and there are infinite distinct representations of the same curve. The natural parametrization of the curve based on arc length  $s$  can be computed from any parametrization using the equation:

$$s = \int_0^u |\dot{\mathbf{r}}(v)| dv \quad (6)$$

where  $\dot{\mathbf{r}}(u) = d\mathbf{r}/du$ .

The curvature  $\kappa$  of the curve at  $s$  is defined as ([22]):

$$\kappa(s) = \frac{\dot{x}(s)\ddot{y}(s) - \dot{y}(s)\ddot{x}(s)}{(\dot{x}(s)^2 + \dot{y}(s)^2)^{3/2}} \quad (7)$$

However, curvature as defined in Equation 7 is susceptible to noise and numerical instabilities. Therefore, in practice, the curvature of a smooth curve is computed. Specifically, Let  $X(s, \sigma) = x(s) * g(s, \sigma)$  and  $Y(s, \sigma) = y(s) * g(s, \sigma)$  where  $g(s, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{s^2}{2\sigma^2}}$  is a Gaussian of width  $\sigma$  and  $*$  denotes convolution. Then,

$$\begin{aligned} \kappa(s, \sigma) &= \frac{X_u(s, \sigma)Y_{uu}(s, \sigma) - X_{uu}(s, \sigma)Y_u(s, \sigma)}{(X_u(s, \sigma)^2 + Y_u(s, \sigma)^2)^{3/2}} \quad (8) \\ X_u(s, \sigma) &= x(s) * \frac{\partial}{\partial s} g(s, \sigma) \\ Y_u(s, \sigma) &= y(s) * \frac{\partial}{\partial s} g(s, \sigma) \\ X_{uu}(s, \sigma) &= x(s) * \frac{\partial^2}{\partial s^2} g(s, \sigma) \\ Y_{uu}(s, \sigma) &= y(s) * \frac{\partial^2}{\partial s^2} g(s, \sigma) \end{aligned}$$

2) *Orientation*: Orientation of the spine branches refer to their relative location on the skeleton curve. Clearly, the branches on the left/right need to be matched with only the ones on the left/right in the other time-point. The orientation of the branches is computed during the depth first search of the skeleton. The variance for this parameter during matching is set very low since we do not want to match a left spine with a right spine.

## B. Graph Homomorphism

A graph homomorphism is a mapping between two graphs that respects their structure [23]. Formally, a graph homomorphism  $f$  from a graph  $\mathbb{D}_1(\mathbb{V}_1, \mathbb{E}_1)$  to a graph  $\mathbb{D}_2(\mathbb{V}_2, \mathbb{E}_2)$  is a mapping  $f: \mathbb{V}_1 \rightarrow \mathbb{V}_2$  from the vertex set of  $\mathbb{D}_1$  to the vertex set of  $\mathbb{D}_2$  such that  $\{f(u), f(v)\} \in \mathbb{E}_2$  whenever  $\{u, v\} \in \mathbb{E}_1$ . Ideally, we would like to determine an exact mapping from every branch point in  $\mathbb{D}_1$  to every branch point in  $\mathbb{D}_2$ . However, the number of estimated branch points in the dendrites at the two time steps varies because of the evolution of spines and due to spurious branch points. Therefore, we use maximum likelihood estimation (MLE) based techniques to find the best homomorphism between the two graphs.

Let  $[i, k]$  be the condition that  $v_k \in \mathbb{V}_2$  (a branch point in  $\mathbb{D}_2$ ) is the evolved version of branch point  $v_i \in \mathbb{V}_1$  in  $\mathbb{D}_1$ . We wish to find a mapping such that  $p([i, k] | v_k = f(v_i))$ , is maximum, where  $v_k \in \mathbb{V}_2$  and  $v_i \in \mathbb{V}_1$ . By Bayes law,

$$p([i, k] | v_k = f(v_i)) = \frac{p(v_k = f(v_i) | [i, k]) \cdot p([i, k])}{p(v_k = f(v_i))} \quad (9)$$

Assuming a uniform distribution for  $p([i, k])$ , we pose the problem as one of determining the best  $k$  that maximizes the likelihood probability  $p(v_k = f(v_i) | [i, k])$ , that is to find  $\arg \max_k p(v_k = f(v_i) | [i, k])$ . We model this distribution by a 5 dimensional normal distribution  $\mathbf{N}(0, \Sigma)$

$$p(v_k = f(v_i) | [i, k]) = \frac{1}{(2\pi)^{5/2} |\Sigma|^{1/2}} e^{(-\frac{1}{2}(v_k - v_i)^T \Sigma^{-1} (v_k - v_i))} \quad (10)$$

where  $\Sigma$  is the covariance matrix of the feature vector  $(x, y, \kappa, s, \theta)$ . We assume the cross-covariance terms to be zero, and only the auto-covariance terms to be non-zero (i.e.  $\Sigma$  is a diagonal matrix). The parameter  $\Sigma$  is not sensitive as long as it is chosen at about the right scale. The following  $\Sigma$  was found to be optimal in our experiments:

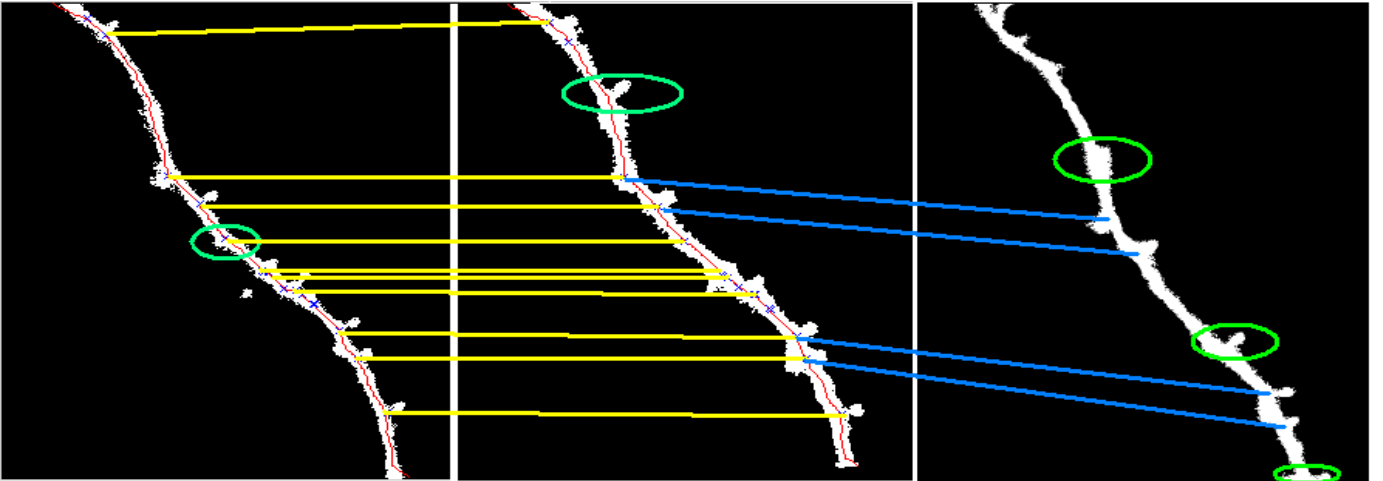
$$\text{trace}(\Sigma) = [ 10 \quad 10 \quad 0.02 \quad 0.0025 \quad 0.0001 ] \quad (11)$$

We use a dynamic programming algorithm to compute the graph homomorphism that maximizes the likelihood probabilities  $p(v_k = f(v_i) | [i, k])$  for all the spines on the two dendritic sections. Let  $n$  be the number of spines in  $\mathbb{D}_1$  and  $m$  be the number of spines in  $\mathbb{D}_2$ . Then

- i Let  $p(i, j)$  be the probability  $p(v_j = f(v_i) | [i, j])$  as computed by Equation 10
- ii Let  $P(i, j)$  be the score of the best correspondences of spines 1 to  $i$  in  $\mathbb{D}_1$  with spines 1 to  $j$  in  $\mathbb{D}_2$
- iii **Boundary Conditions** Let  $P(n, j) = p(n, j)$  and  $P(i, m) = p(i, m)$
- iv **Recursion Formula**

$$P(i, j) = \begin{cases} P(i-1, j-1) & \text{if } P(i-1, j-1) + p(i, j) \geq \\ \quad + p(i, j) & \quad \max\{P(i-1, j), P(i, j-1)\} \\ P(i-1, j) & \text{if } P(i-1, j) > P(i, j-1) \\ P(i, j-1) & \text{otherwise} \end{cases} \quad (12)$$

- v **Termination Condition** To find  $P(n, m)$ , i.e. the best correspondence between spines 1 to  $n$  in  $\mathbb{D}_1$  with spines 1 to  $m$  in  $\mathbb{D}_2$



(a)

Fig. 5. Correspondence (yellow, blue lines) observed among images from different time steps(1, 2 and 5). Note that most of the observable spines have been matched. The time-points sometimes present spines that are not present in the other time-point (green circles). These spines are not matched by our framework and may be detected as a new emergent spine. The correspondence with the third image is poor owing to large elastic deformation.

## V. RESULTS AND VALIDATION

This section reports on a number of experiments we conducted to evaluate our matching framework for spine tracking. Experiments were conducted with a validated temporal dataset having five epochs (four pairs). Our goals were three-fold: (i) We demonstrate the ability of the framework to match existing spines correctly and new spines are left unmatched. (ii) We show that spurious branch points that were earlier detected are not matched. (iii) We demonstrate that real datasets benefit from the availability of automated tools for objective analysis.

All our experiments were compiled and executed on an Intel P4 3.2 GHz processor equipped with 1GB RAM. The registration framework was developed in C++ using the *The Insight Segmentation and Registration Toolkit* (ITK) libraries [24].

In order to validate our algorithm, five images were reviewed and all pertinent spine locations on the skeletons were marked up manually. The matched pairs can be separated and counted in four categories:

- i  $a$  counts the number of *true-positive* results.
- ii  $b$  counts the number of *false-positive* results.
- iii  $c$  counts the number of *false-negative* results.
- iv  $d$  counts the number of *true-negative* results.

Thus, one can define the sensitivity and specificity of our framework as follows:

$$\text{sensitivity} = \frac{a}{a+c} \quad \text{specificity} = \frac{d}{b+d} \quad (13)$$

$$\text{error} = \frac{b+c}{a+b+c+d} \quad (14)$$

Sensitivity measures the effectiveness of the algorithm in detecting the true-positives. A high value indicates that it is easy for the algorithm to rule out a spurious match with a high probability. On the other hand, a high value of high specificity means that the method can identify a *bona fide* match with a high probability of success. A graph of sensitivity

and specificity values for the four pairs of image matchings is plotted in Figure 6. From the graphs, we observe that framework has a good sensitivity ( $> 0.85$ ), specificity ( $> 0.85$ ) and with a low error rate ( $< 0.1$ ) for the first three image pairs. The performance for the fourth image pair is expected to be low. The correspondence of Image 4 with Image 5 has been plotted in Figure 5. In Figure 4c we plot the mean square error metric in the images after registration. We observe a high value in the mean square error for the fourth image after rigid registration. This is because a large amount of non-rigid stretching in the image that is not accounted for by the ICP registration algorithm. All the four parameters in the MLE framework require some amount of localization that is missing in this pair.

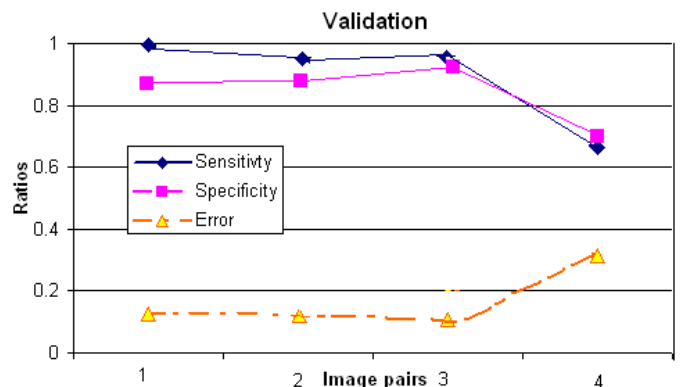


Fig. 6. Plots of the sensitivity, specificity and error ratios for the matching on 5 time-points (4 pairs).

On further investigation of the matching errors in fourth image pair, we found the the spines in image 5 were tightly clustered around each other, as a result of which the skeletonization step combined several spines into a single spine. This caused misidentification of spine correspondences.

## VI. CONCLUSIONS AND FUTURE WORK

In this paper, we described methods to temporally track the evolution of spines on a dendrite using a graph-theoretic representation of the dendrite. A suite of image processing algorithms including adaptive thresholding, filtering, medial axis extraction and contour smoothing are used to obtain an initial skeletal curve with spine locations. We then build a dendrite graph, where each spine emerging from the dendrite is represented as a node in the graph. Attributes of the spine such as its location coordinates, local curvature of the spine segment, location on the parameterized curve are stored as attributes to be matched. An MLE framework is constructed that performs the matching of the two graphs so that the probabilities are maximized. The framework was successful in matching a high percentage of corresponding spines. New spines that were present in some of the time-points were detected in the residues and remained unmatched. Using available ground-truth, the sensitivity of our matching algorithm was measured at  $> 85\%$  with a specificity of  $> 85\%$ .

Our work therefore extends current literature on 3D neuron structure tracing by incorporating a tracking component. In future, we would like deploy our methods with existing ones to explore the full pipeline of detection, matching and making comparative measurements in an automated and objective manner.

## REFERENCES

- [1] F. Engert and T. Bonhoeffer, "Dendritic spine changes associated with hippocampal long-term synaptic plasticity," *Nature*, vol. 399, pp. 66–70, 1999.
- [2] M. B. Moser, M. Trommald, and P. Anderson, "An increase in dendritic spine density on hippocampal CA1 pyramidal cells following spatial learning in adult rats suggests the formation of new synapses," in *Proc. Natl. Acad. Sci., USA*, 1994, vol. 91, pp. 12673–12675.
- [3] L. A. Glantz and D. A. Lewis, "Decreased dendritic spine density on prefrontal cortical pyramidal neurons in schizophrenia," *Archives of General Psychiatry*, vol. 57, no. 1, pp. 65–73, 2000.
- [4] J. J. Capowski, *Computer Techniques in Neuroanatomy*, New York: Plenum, USA, 1989.
- [5] C. F. Garvey, J. Young, W. Simon, and P. D. Coleman, "Automated three-dimensional dendrite tracking system," *Electroencephalogr. Clin. Neurophysiol.*, vol. 35, pp. 199–204, 1973.
- [6] J. Young P. D. Coleman, C. F. Garvey and W. Simon, "Semiautomatic tracing of neuronal processes," *Computer Analysis of Neuronal Structures*, vol. R. D. Lindsay, Ed. New York: Plenum, pp. 91–109, 1977.
- [7] J. R. Glaser and E. Glaser, "Neuron imaging with neuroLucidaA PC-based system for image combining microscopy," *Comput. Med. Imaging Graphics*, vol. 14, pp. 307–317, 1990.
- [8] I. Y. Y. Koh, W. B. Lindquist, K. Zito, E. A. Nimchinsky, and K. Svoboda, "An image analysis algorithm for dendritic spines," *Neural Computation*, vol. 14, no. 6, pp. 1283–1310, 2002.
- [9] K. M. Harris, F. E. Jensen, and B. Tsao, "Three-dimensional structure of dendritic spines and synapses in rat hippocampus (CA1) at postnatal day 15 and adult ages: Implications for the maturation of synaptic physiology and long-term potentiation," *Journal of Neuroscience*, vol. 12, pp. 2685–2705, 1992.
- [10] K. A. Al-Kofahi, S. Lasek, D. H. Szarowski, C. J. Pace, G. Nagy, J. N. Turner, and B. Roysam, "Rapid automated three-dimensional tracing of neurons from confocal image stacks," *IEEE Transactions on Information Technology in Biomedicine*, vol. 6, no. 2, pp. 171–187, 2002.
- [11] P. Besl and N. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 239–256, 1992.
- [12] D. A. Rusakov and M. G. Stewart, "Quantification of dendritic spine populations using image analysis and a tilting disector," *Journal of Neuroscience Methods*, vol. 60, pp. 11–21, 1995.
- [13] R. Watzel, K. Braun, A. Hess, H. Scheich, and W. Zuschratter, "Detection of dendritic spines in 3-dimensional images," in *DAGM-Symposium Bielefeld*, 1995, pp. 160–167.
- [14] A. Herzog, G. Krell, B. Michaelis, J. Wang, W. Zuschratter, and K. Braun, "Restoration of three-dimensional quasi-binary images from confocal microscopy and its application to dendritic trees," in *Proceedings of SPIE, Three-Dimensional Microscopy: Image Acquisition and Processing IV*, 1997, pp. 146–157.
- [15] K. Kilborn and S. M. Potter, "Delineating and tracking hippocampal dendritic spine plasticity using neural network analysis of two-photon microscopy," *Soc. Neurosci. Abstr.*, vol. 24, pp. 422–425, 1998.
- [16] A. R. Cohen, B. Roysam, and J. N. Turner, "Automated tracing and volume measurements of neurons from 3-D confocal fluorescence microscopy data," *Journal of Microscopy*, vol. 173, no. 2, pp. 103–114, 1994.
- [17] I. F. Sbalzarini and P. Koumoutsakos, "Feature point tracking and trajectory analysis for video imaging in cell biology," *Journal of Structural Biology*, vol. 151, pp. 182–195, 2005.
- [18] S. Bonneau and L. D. Cohen M. Dahan, "Single quantum dot tracking based on perceptual grouping using minimal paths in a spatio-temporal volume," *IEEE Transactions in Image Processing*, vol. 14, no. 9, pp. 1384–1395, 2005.
- [19] N. Otsu, "A threshold selection method from gray level histograms," *IEEE Trans. Systems, Man and Cybernetics*, vol. 9, pp. 62–66, Mar. 1979, minimize inter class variance.
- [20] P. E. Danielsson, "Euclidean distance mapping," *Computer Graphics and Image Processing*, vol. 14, pp. 227–248, 1980.
- [21] F. Mokhtarian and A. Mackworth, "Theory of multiscale, curvature based shape representation for planar curves," *IEEE Pattern Anal. Mach. Intell.*, vol. 14, pp. 789–805, 1992.
- [22] A. Goetz, *Introduction to Differential Geometry*, Addison-Wesley, Reading, MA, 1970.
- [23] P. Hell and J. Nešetřil, "Graphs and homomorphisms," in *Oxford Lecture Series in Mathematics and its Applications 28*, 2004, Oxford University Press.
- [24] L. Ibáñez and W. Schroeder, "The ITK Software Guide," *The Insight and Registration Toolkit [www.itk.org]*, Kitware, Inc., 2003.