Oriented Markov Random Field Based Dendritic Spine Segmentation for Fluorescence Microscopy Images

Jie Cheng[1], Xiaobo Zhou[1], Eric L. Miller[3], Veronica A. Alvarez[2],
Bernardo L. Sabatini[2], and Stephen T.C. Wong[1,*]

[1] The Center for Bioengineering and Informatics, The Methodist Hospital Research Institute and Department of Radiology, The Methodist Hospital, Weill Cornell Medical College, Houston, TX 77030

[2] Department of Neurobiology, Harvard Medical School, 220 Longwood Ave Boston, MA 02115, USA

[3] Department of Electrical and Computer Engineering, Tufts University, Medford, MA 02155, USA

[*] Corresponding to: stwong@tmhs.org, phone: (713)4415884

Abstract

Recent research has shown that there is a strong correlation between the functional properties of a neuron and the morphological properties of dendritic spines. Morphological analyses typically involve a significant component of computer-assisted manual labor, which is time-consuming and susceptible to operator bias. Existing automatic or semi-automatic approaches can largely reduce user efforts, by automatically detecting the dendritic spines with little human interaction required during the processing. However, problems such as degraded detection performance for images with larger pixel size (e.g. 0.125 μm/pixel instead of 0.08 μm/pixel) still exist. Moreover, the shapes of detected spines are also distorted. For example, the "necks" of some spines are found to be missing. In this article, we present an oriented Markov random field (OMRF) based algorithm that improves the detection of spines as well as their geometric characterization. We begin by the identification of a region of interest (ROI) in the image containing the dendrites and spines. For this, we introduce an adaptive procedure for identifying the background region of the image. Having identified the ROI, the OMRF model is next discussed within a statistical framework and the segmentation is solved as a maximum a posteriori estimation (MAP) problem, whose optimal solution is found by a knowledge-guided iterative conditional mode (KICM) algorithm. The validation results show that the proposed algorithm not only provides a more accurate representation of the spine shape, but also improves the detection performance for existing algorithms by more than 50% with regard to reducing both the misses and false detection.

I Introduction

The dendrites are cellular extensions of a neuron with many branches, where more than 90% of input to the neuron occurs. The dendritic spine is a small (sub-micrometer) membranous extrusion of the dendrites that contains the molecules and organelles involved in the postsynaptic processing of the synaptic information. The remarkable ability of dendritic spines to change shape rapidly, viz. the spine plasticity, is implicated in motivation, learning, and memory [1][2]. The abnormalities in dendritic spine morphologies are believed to be associated with a variety of brain disorders. In particular, neuron morphology is illustrative of neuronal function and can be instructive in the dysfunction seen in neurodegenerative conditions such as Alzheimer's disease and Parkinson's disease [3][4]. Cognitive disorders such as autism, mental retardation and fragile X Syndrome [5][32], may also be resultant from abnormalities in dendritic spines, such as the number of spines and their maturity status.

Modern fluorescence microscopy methods, such as confocal laser scanning microscopy (CLSM) and two-photon laser scanning microscopy (2PLSM), provide powerful tools to study dendritic spine structures. Currently the analysis of neuronal images generated by these microscopes has remained largely manual. The accuracy of manual detection performed by a trained expert is usually higher compared with the automatic results, even with the possible bias problem. However, usually such analyses are still extremely time-consuming even with the computer assistance, which makes it infeasible to manually analyze a great amount of data.

Automation with compatible detection accuracy is thus in great need for dendritic spine analysis. Recently, some automatic dendritic spines analysis algorithms have been provided for both in vitro [6]-[11] and in vivo [33] images. Koh et al. propose a geometric approach for automatically detecting and quantifying the structure of dendritic spines [6]. Xu et al. propose a new attached spine component detection algorithm by using two grassfire propagations [7]. Based on Koh's approach, Weaver et al. describe a package which is capable of morphometry on an entire neuron, by combining the spine detection algorithms with dendritic tracing algorithms [8]. Cheng et al. propose an automatic spine detection pipeline based on the adaptive thresholding and local dendrite morphology analysis [9]. Bai et al. [10] used an unsharp mask filter to partly correct the inhomogeneity of image intensity. A global threshold computed using a histogram-based strategy is used to segment the neuron structures from the background. Zhang et al. [11] apply a curvilinear structure detector to extract the boundaries as well as the centerlines for the dendritic backbones and spines. A classifier using Linear Discriminate Analysis (LDA) is further built to help improve the accuracy of spine detection. Although these algorithms can greatly reduce the human labor, problems still exist. For instance, the efficiency of detection will degrade for images with larger pixel sizes, in which the spines occupy relatively few pixels. Also, the above mentioned algorithms are prone to break the spine necks: pixels in spine neck regions usually have low intensity values and are likely segmented as background, which cause the spines to be broken into several detached components thereby further degrading spine detection performance. To deal with these issues, here we propose a novel maximum a posteriori - oriented Markov random field (MAP-OMRF) framework for dendritic spine segmentation.

Markov Random Field (MRF) models were first proposed for image processing by Geman and Besag [25]-[27]. Since then, they have been intensively studied in this field for various applications, largely because of their ability to encode expected spatial correlations through contextual constraints of neighboring pixels. Usually, the correlation structure is such that pixels are expected to have the same or similar intensities, i.e. the image is piecewise constant or piecewise continuous. In the literature, the MRF based segmentation methods have been widely applied to medical images obtained from many modalities operating across a range of application areas. These include, but not limited to, cardiac imaging [12]-[13], brain imaging [14]-[18],[29]-[30], digital mammography [19]-[20] acquired by magnetic resonance imaging (MRI), ultrasound, positron emission tomography (PET), and electron microscopy.

Despite the successful applications in medical image segmentation, MRF methods have seldom been used in dendritic spine detection, partially because of the need to specialize these models to address the specific problems discussed previously concerning spine segmentation. To deal with these issues, the OMRF model is proposed in this paper. By encoding the location orientation of the spines relative to the dendrite, this model allows us to overcome many of the segmentation challenges mentioned above. To reduce the computation complexity and make the algorithm more robust to the background noise, a region of interest (ROI) estimation algorithm, which is based on an iterative spline background correction, is also proposed. Finally, we present a knowledge-based iterated conditional modes (KICM) algorithm in which biological constraints are incorporated in the transition matrix. Compared with the normal ICM algorithm,

the segmentation performance of KICM is much improved because of its ability to accurately locate the spine base.

The paper is divided into the following parts: how to obtain the images are first discussed in Section II. Then, the algorithms are presented in Section III, where the OMRF model is proposed. We also describe the computation of the orientation map needed by the OMRF, as well as the ROI estimation algorithm. The KICM optimization scheme is discussed next. In Section IV, the segmentation and detection results are compared with other existing methods and validated manually. Finally the conclusion is provided and the future works are discussed in Section V.

II Image acquisition

Brain slices from the hippocampus were prepared from rat pups (P7) and cultured as described in Alvarez et al [21]. Slices were transfected with a green fluorescent protein (GFP) expressing vector and pyramidal neurons were identified based on their characteristic morphology at 7-20 days post-transfection (DPT). Neuronal morphology was studied using 2-photon laser scanning microscopy (2PLSM) and a custom-built microscope with a water immersion objective (Olympus LUMPlanFI/RI 60x, NA=0.9). The excitation wavelength was 910 nm provided by a Verdi 10-V.-Mirra laser (Coherent). Measurements performed on 100 nm diameter yellow-green fluorescent microspheres (FluorSpheres, Molecular Probes) indicated that the point-spread function of the microscope placed a lower limit on measurable widths of 550 nm.

Images (512 x 512 pixels) of apical and basal dendrites of hippocampal pyramidal neurons were acquired at zoom 3 and 5 (image field, 64x64 µm and 42x42 µm, respectively). The 3D image stacks were 16-bit grey-scale with 1 µm optical section spacing with 20-36 slices in each stack. The analyzed dataset includes a variety of genotypes to ascertain how well our algorithm detects spines with a wide distribution of morphologies. The manual analysis of spine density, length, and width was performed by observers who were blind to the genotype by using in-house developed software [21, 31]. Spine lengths were measured from the junction with the dendritic shaft to the tip. To determine head width and primary dendrite thickness, the fluorescence was measured in a line across the structure. The width of the distribution where fluorescent intensity fell to 30% of maximum was calculated as the results.

III Algorithms

Although 3D image stacks were acquired, our proposed algorithms are based on the 2D maximal intensity projection (MIP) images. This choice was made because: the resolution in the z- direction is much lower compared with that in x- and y- directions. For spines which are big enough, the 3D information in neighboring slices is helpful for detection. However, there are many spines which are so small that they only appear in one slice thereby limiting the utility of a full 3D approach. The same is true with the spine necks which are much thinner compared with the head components. Since the purpose of our algorithm is to solve the detection problem of small spines and the spine necks, we only propose 2D algorithms. Actually, because most of

the morphological information of the dendritic spine is included in the MIP image, 2D based automatic algorithms can effectively detect and measure the dendritic spines. This has been proved by several recently published papers [9]-[11].

The overall structure of our proposed algorithms is outlined in Fig. 1: in order to reduce the computational complexity and improve the detection performance, segmentation is performed only over a region of interest (ROI) covering the area around the dendrite. The algorithm for determining this ROI is provided in Sect. 3.1. The calculation of the orientation map used by our OMRF is then described in Sect. 3.2. Based on the ROI and orientation map, as well as the original image and the initial segmentation results, the OMRF model is proposed in Sect. 3.2 for the purpose of dendritic spine segmentation. Finally, the KICM algorithm which can efficiently find the final segmentation is presented in Sect. 3.3.
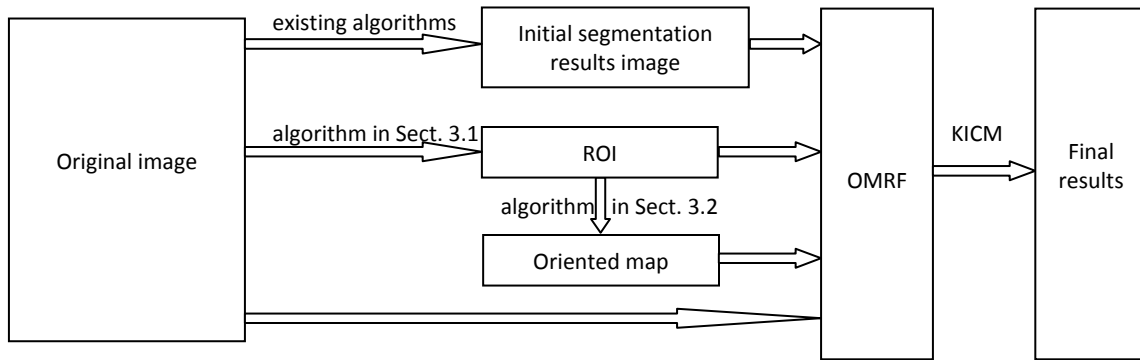

Figure 1: structure of the proposed algorithms

3.1 Region of interest

The main purpose of our ROI algorithm is to reduce the computation cost as well as improve detection performance by limiting the spine segmentation processing to a region surrounding the dendrites which is much smaller than the full image. In existing spine detection algorithms, simple physical constraints are applied to improve the detection results. For example, only those detected blobs within a certain distance to the dendrite are considered to be spines [5]-[11]. However, many background pixels are still contained in the local regions. These noisy background pixels not only increase the processing time, but also potentially increase the risk of misdetection. To deal with this issue, we propose a local ROI estimation method derived from an iterative spline based background correction algorithm (ISBC) [24]. The ISBC algorithm is initially proposed for solving the problem of large scale intensity variations and shading effects in the image caused by uneven illumination, however, it can also be applied to acquire the ROI with minimal modification.

The background is estimated and modeled by a cubic B-spline surface patch S. The value of a surface point of the patch S can be written as [24]:

$$S(u,v) = \sum_{kl} B_k(u)B_l(v)c_{kl} \qquad (1)$$

where (u,v) is position of the pixel; c_{kl} is a control point of the surface (an evenly spaced 5×5 grid of control points are used in this paper); B_k and B_l are the B-spline blending polynomials and the Cartesian product of the two functions is the weight given to the control point. Eq. (1) indicates that the estimated value of each pixel in the patch S is the weighted summation of all control points. The control points are initially estimated by minimizing the Euclidean distance between the spline surface and the original image using least squares regressions. For this process, neither the background nor the foreground has been determined yet, thus *all* of the pixels in the image are used in the fitting procedure. Since the much brighter pixels in dendrites and spines (i.e., the foreground) are included in the background estimation stage, the initially estimated background must be adjusted. This is done by masking out the foreground pixels and applying the spline surface fitting to the refined potential background regions. The foreground pixels are determined through a simple thresholding procedure described below. The process of masking the foreground and recomputing the background spline fit is iterated until the average change of pixel values between two consecutively estimated backgrounds is small enough, e.g. less than the original quantization step of the image, which is set as 1 for our testing. The ROI can then be obtained by thresholding the background compensated image, which is calculated by subtracting the estimated background from the original image. The "compensated image" is calculated by subtracting the iteratively estimated background from the original image. Since the background is initially set as zero, the compensated image is initialized as the original image itself.
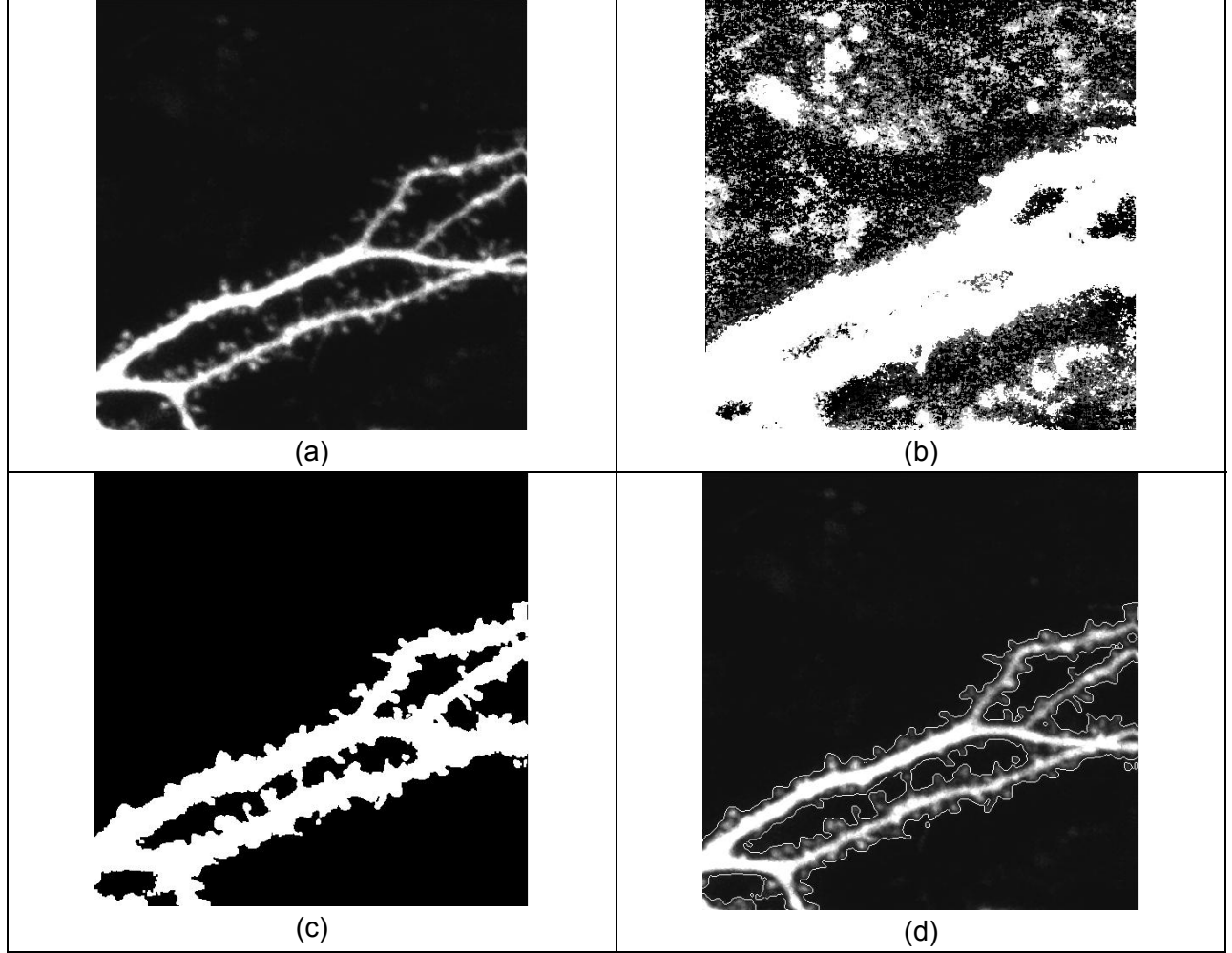
Figure 2: (a) original image; (b) background compensated image scaled to have pixel values between 0 and 255 (c) ROI obtained after segmenting the background compensated image and denoising (d) ROI (regions inside the high-lighted curves) in the original image

At all stages during this process, the masking procedure is accomplished using a simple threshold. Suppose the standard deviation of the pixel values in the background compensated image is σ_c. With mean normalized to 0, the foreground R is defined by:

$$R = \{\text{all pixel } i \mid I_c(i) > k \cdot \sigma_c\} \tag{2}$$

where $I_c(i)$ is the pixel value at i in the background compensated image; k is a factor which is set as 10 during our validation.

Fig. 2 provides an example of this process. For the purpose of better visualization only, Fig. 2 (b) is the final background compensation image scaled so that the pixel values are in the range of 0 to 255. Fig. 2 (c) shows the ROI obtained after thresholding, with small objects removed by applying opening operation (erosion followed by dilation). As we can observe from Fig. 2 (d), all protrusions along the dendrite boundary, which are potential spines, are contained in the obtained ROI. The ROI is a much reduced region compared to the original image which can be

7

obtained in seconds: for the 512×512 image shown in Fig. 2 and codes implemented in Matlab without optimization, the ROI can be obtained in less than 3 seconds by a PC with a 2.5GHz CPU; the number of pixels in the ROI is 49679, which is about 81% reduction of the total pixels in the original image. To further test the effectiveness of the ROI, the processing time with and without using ROI is compared: for ten images being tested, the average processing time for the proposed segmentation algorithm with using ROI is about 2'4.8'', which is about 46% reduction compared with the average 3'52.2'' processing time without using ROI.

3.2 Oriented Markov Random Field

3.2.1 General Model

MRF models are widely applied in various fields of image processing due to their ability to mathematically capture the intrinsic spatial structure of images. Usually for MRF-based methods, segmentation is achieved through an optimization process. The optimization function is derived from probabilistic models of the data and the prior information concerning the spatial structure of the segments which is encoded in the MRF. Indeed, one of the most appealing aspects of the MRF formulation is its flexibility in representing structural prior information across a wide variety of applications. For the problem of interest here, the primary source of prior information concerning spines is related to the structure of the dendrite. Indeed knowing the location and orientation of the dendrite places important geometric and morphological constraints on the associated spines. In this paper we introduce a new MRF model which considers the inherent orientation information in the neuron cells, viz. OMRF model.

We model the observed grey scale image as a random field $\mathbf{y}$, with the intensity of a pixel at location i denoted by a random variable y_i. The distribution of different regions is denoted by a random field $\mathbf{x}$, where the variable $x_i \in \{1, 2, ..., K\}$ means that the pixel i belongs to one of the M region types. For the neuron images, K=3, representing three classes of interest in this application: the dendrite, spine and background.

Given the observed data $\mathbf{y}$, the problem of segmentation amounts to the determination of $\mathbf{x}$, that is, the labels associated with each pixel. We pose this problem in a statistical framework as a maximum a posteriori estimation problem where $\mathbf{x}$ is chosen to maximize the posterior distribution of the label based on $\mathbf{y}$ [23]. Formally, by using Bayes theorem this distribution is described as:

$$p(\mathbf{x} \mid \mathbf{y}) \propto p(\mathbf{y} \mid \mathbf{x}) p(\mathbf{x}) \tag{3}$$

where $p(\mathbf{x})$ is the a priori probability density of the image; $p(\mathbf{y} \mid \mathbf{x})$ is the conditional probability density of the observed image given the distribution of regions. After removal of the shot noise by median filter, the likelihood for each pixel $p(\mathbf{y} \mid \mathbf{x})$ can be described by independent Gaussian [22]. The assumption that the intensity of each pixel is statistically independent can

greatly simplify the processing. Under the independent and Gaussian assumption, the possibly observed image with $\mathbf{x}$ known can be represented as:

$$p(\mathbf{y}|\mathbf{x}) \propto \exp\left\{ \sum_{i \in S} -\frac{1}{2\sigma_{i,x_i}^2}(y_i - \mu_{i,x_i})^2 \right\}$$ (4)

Here S is the collection of all pixels; σ_{i,x_i} and μ_{i,x_i} are the variance and mean value of the intensity of pixel i, which is represented by a random variable y_i. The variance and mean value are class depended, i.e. they are functions of x_i, which means that the intensity distribution for pixels of different types (e.g. spine, dendrite and background) are different. In addition, from a practical perspective the mean values and variances also vary over the image: their values are also related to the pixel location i. This is based on the observation that even in the same image the intensities vary largely for spines or dendrite pieces at different locations. Thus, the mean and variance should be estimated adaptively. Calculation of σ_{i,x_i} and μ_{i,x_i} will be discussed in detail in Section 3.3.1

The a priori density $p(\mathbf{x})$ is described by the OMRF model. According to the Hammersley-Clifford theorem, any Markov random field can be described by a probability distribution of the Gibbs form [25]:

$$P(\mathbf{x}) = Z^{-1}e^{-U(\mathbf{x})}$$ (5)

where $P(\mathbf{x})$ denotes the probability of configurations of the random field $\mathbf{x}$ and $U(\mathbf{x})$ is the energy function and Z a constant needed to ensure that P integrates to one. We notice that the neuron dendrites are oriented structures: the growing direction of each spine is normal to the orientation of local dendrite pieces. This natural orientation information can be combined into the energy function:

$$U(\mathbf{x}) = \beta \cdot \sum_{i \in S}\sum_{j \in N_i} D(x_i, x_j)[1 + \alpha \cdot (1 - |\vec{u}_{ij}^T \vec{v}_i|)]$$ (6)

where $\vec{v}_i$ is the orientation of the dendrite in the vicinity of pixel i and will be described in detail in the following section; $\vec{u}_{ij}$ is the direction of the line connecting two neighboring pixels of i and j; α is the weight factor of the orientation information; β is a parameter which controls how strong the spatial regularization is; S is the collection of all pixels; N_i is the 8-neighbor of pixel i; x_i is the segment associated with pixel i; x_j is the segment of neighboring pixels of i; and $D(x_i, x_j)$ is the similarity function. If $x_i = x_j$, i.e. pixel i and j are of the same type, the value of $D(x_i, x_j)$ is set as 0, otherwise the value is set as 1. Thus, pixel i is intended to be in the same region as most of its neighboring pixels, which will reduce the energy $U(\mathbf{x})$. From Eq. (3) - (6), we have:

$$P(\mathbf{x}|\mathbf{y}) \propto P(\mathbf{y}|\mathbf{x})p(\mathbf{x})$$

$$\propto \exp(-\sum_{i \in S} \frac{1}{2\sigma_{i,x_i}^2}(y_i - \mu_{i,x_i})^2 - U(\mathbf{x}))$$

$$\propto -(\sum_{i \in S} \frac{1}{2\sigma_{i,x_i}^2}(y_i - \mu_{i,x_i})^2 + U(\mathbf{x}))$$

$$= \sum_{i \in S} -\{\frac{1}{2\sigma_i^2}(y_i - \mu_i)^2 + \beta \sum_{j \in N_i} D(x_i, x_j)[1 + \alpha \cdot (1 - |\vec{u}_{ij}^T \vec{v}_i|)]\} \quad (7)$$

3.2.2 Orientation map

The use of orientation is motivated by the intuition that the probability of being segmented as a spine pixel should be related to the directions of both $\vec{v}_i$ and $\vec{u}_{ij}$. Suppose the backbone of the dendrite has already been obtained by thinning and trimming algorithms [9]. Assume c_k is a backbone pixel, we denote by $R_{c_k} = \{c_{k-2}, c_{k-1}, c_k, c_{k+1}, c_{k+2}\}$ the set of local backbone pixels from which we shall determine the orientation at c_k. In greater detail, linear least squares is used to fit a line through these five pixels. Letting γ_{c_k} be the slope of that line, $\vec{v}_i$ is defined as:

$$\vec{v}_i = \{\text{unit vector with slope } \gamma_{c_k} \mid c_k \text{ is the nearest backbone pixel to } i\} \quad (8)$$
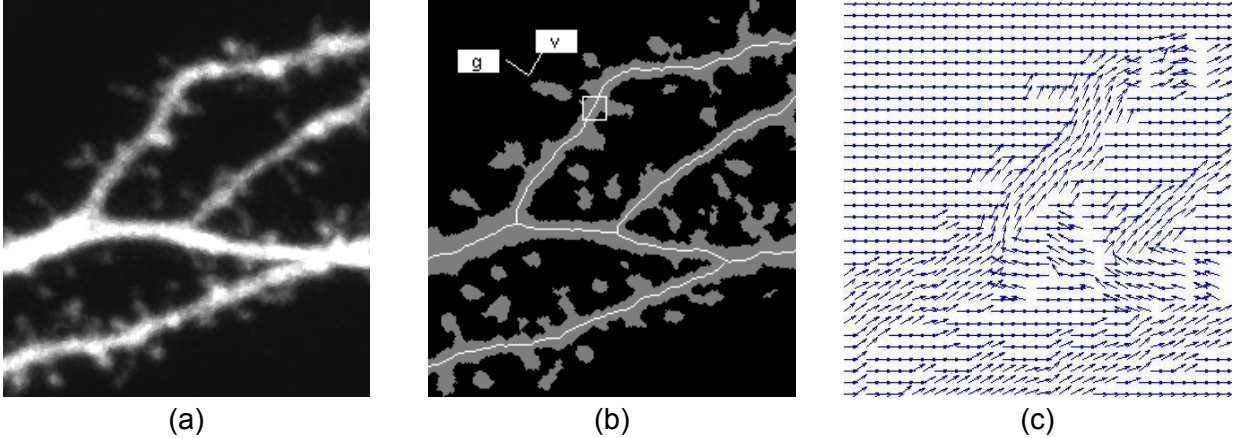

(a) (b) (c)

Figure 3: (a) original image; (b) initial segment result of the neuron without spine detection; the white curves are the extracted backbone pieces; $\vec{v}$ and $\vec{g}$ are the estimated local dendrite orientation and the growing direction of the spine respectively; (c) sampled orientation map

Fig. 3 demonstrates how the orientation map is obtained. The orientation of a pixel is defined as that of the neighboring backbone pixel with the least Euclidian distance, which is estimated in a local window as shown in Fig. 3 (b). Fig. 3 (c) shows the sampled orientation map. The orientation of pixels outside the ROI is set to zero, because we limit our attention to the ROI surrounding the dendrites for segmentation. In Fig. 3 (c) we notice that along the same dendritic trunk, the vectors may abruptly change direction. This is because there are possibly two angles which can describe the orientation of the backbone, α and $\alpha + \pi$. Since only the absolute

values of the scalar product between the vectors are needed in Eq. (7), this ambiguity is irrelevant to the results. In this paper, we assume that the angles are between 0 and π.

3.3 Optimization Algorithm

In neuron images to be processed, there are three different region types, i.e. spine, dendrite and background. For our approach, the segmentation of neuron images is designed as a maximum a posteriori estimation (MAP) problem. Suppose $\mathbf{y}$ is the observed data, the optimal segmentation results $\hat{\mathbf{x}}$ is found by solving the following MAP estimation problem:

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} P(\mathbf{x}\,|\,\mathbf{y}) \qquad\qquad (9)$$

To acquire the optimal solution, an extended iterated conditional modes (ICM) algorithm, viz. KICM algorithm is proposed in this paper. The interleaved ICM and local parameters updating scheme is applied [28], i.e., estimation of the optimal segmentation $\hat{\mathbf{x}}$ and parameter updating are alternatively performed.

3.3.1 Parameter estimation

Here we denote by $\hat{\boldsymbol{\theta}}_1$ and $\hat{\boldsymbol{\theta}}_2$ the optimal parameter estimates for the prior model and the likelihood model respectively. In this paper, $\boldsymbol{\theta}_1 = (\alpha, \beta)$, where α and β are defined in Eq. (6); $\boldsymbol{\theta}_2 = (\varphi_1\,\varphi_2\,...\,\varphi_N)$, with N being the number of total pixels in the image and $\varphi_i = (\mu_{i,x_i}, \sigma^2_{i,x_i})$, $i \in [1,2,...,N]$. As mentioned in Eq. (4), μ_{i,x_i} and σ_{i,x_i} stand for the mean value and variance of the intensity distribution of pixel i respectively, whose values also depend on the segmentation result x_i. For the purpose of simplifying the computation complexity, here $\boldsymbol{\theta}_1$ is set by the user prior to execution of the algorithm. For all images being tested, we have $\alpha = 0.5$ and $\beta = 0.3$. Since the two parameters are not independent to each other, it is hard to manually find the optimal values. Generally, the detection results are good if $\alpha \in [0.3,0.7]$ and $\beta \in [0.2,0.4]$. As for $\boldsymbol{\theta}_2$, ideally it should be estimated within a MAP-type framework, i.e. we maximize $p(\boldsymbol{\theta}_2\,|\,y_i, x_i)$ which is calculated based on the distribution of the prior probability function $p(\boldsymbol{\theta}_2)$ and the conditional probability function $p(y_i, x_i\,|\,\boldsymbol{\theta}_2)$ according to the Bayes theorem. However, such a process would be quite computationally intensive. Thus, in this paper we pursue an ad-hoc approach that appears to work rather well in terms of the ultimate objective of improving our ability to characterize spine number and shape. Specifically, $\boldsymbol{\theta}_2$ is estimated on the assumption that the intensities of the *same type* pixels *in a local window* are independent and identically-distributed (i.i.d.) Gaussians [22]. Since the estimation of segmentation results $\mathbf{x}$ is updated iteratively, $\hat{\boldsymbol{\theta}}_2$ is also iteratively estimated. The parameter of pixel i can be iteratively estimated by the following equation:

$$\hat{\varphi}_i^{(k)} = \arg\max_{\varphi_i^{(k)}} p(\varphi_i^{(k)}\,|\,\mathbf{x}^{(k-1)}, \mathbf{y})$$

$$= \arg\max_{\varphi_i^{(k)}} \frac{1}{(\sqrt{2\pi\sigma_{i,x_i^{(k-1)}}^{2(k)}})^M} \exp(-\sum_{j \in S(x_i^{(k-1)})} (y_j - \mu_{i,x_i^{(k-1)}}^{(k)})^2 / 2\sigma_{i,x_i^{(k-1)}}^{2(k)}) \quad (10)$$

where $\hat{\varphi}_i^{(k)}$ is the estimated parameter of pixel i at step k; $\mathbf{x}^{(k-1)}$ is the segmentation results at step $k-1$, with $\mathbf{x}^{(0)}$ being the initial segmentation result obtained by existing algorithms [6]-[11]; $\mathbf{y}$ is the observed data; $S(x_i^{(k-1)})$ is the set of pixels in a local window around pixel i, with the same class as pixel i, i.e. all pixels belong to class $x_i^{(k-1)}$. The size of the local window is image pixel size dependent and based on the normal size of the dendritic spines. For example, the 15×15 window is chosen for the images being tested with pixel size of 0.125 μm/pixel, in which the spine areas are generally below 100 pixels. Suppose M is the total number of pixels in the set $S(x_i^{(k-1)})$, a necessary condition for maximizing $p(\varphi_i^{(k)} | \mathbf{x}^{(k-1)}, \mathbf{y})$, or equivalently maximizing $\ln p(\varphi_i^{(k)} | \mathbf{x}^{(k-1)}, \mathbf{y})$, is $\frac{\partial \ln p}{\partial \mu_{i,x_i^{(k-1)}}^{(k)}} = 0$ and $\frac{\partial \ln p}{\partial \sigma_{i,x_i^{(k-1)}}^{(k)}} = 0$. Solving this, we can find the adaptive ML estimates:

$$\hat{\mu}_{i,x_i^{(k-1)}}^{(k)} = \frac{1}{M} \sum_{j=1}^{M} y_j \quad (11)$$

$$\hat{\sigma}_{i,x_i^{(k-1)}}^{2(k)} = \frac{1}{M} \sum_{j=1}^{M} (y_j - \hat{\mu}_{i,x_i^{(k-1)}}^{(k)})^2 \quad (12)$$

where $j \in [1, M]$ are all the pixels in a 15×15 window around pixel i with the same class, i.e. all pixels belong to class $x_i^{(k-1)}$.

3.3.2 Estimation with KICM

The optimal segmentation result $\hat{\mathbf{x}}$ is further estimated after the parameters $\hat{\boldsymbol{\theta}}_1$ and $\hat{\boldsymbol{\theta}}_2$ are obtained. Here an extended iterated conditional modes (ICM) algorithm, viz. KICM algorithm, is applied to acquire the optimal solution. ICM [26] is based on two assumptions: 1) The observation components $y_1, ... y_m$ are conditionally independent given $\mathbf{x}$, and each y_i has the same known conditional density function $p(y_i | x_i)$ which depends only on its label x_i, i.e., $p(\mathbf{y} | \mathbf{x}) = \prod_{i \in S} p(y_i | x_i)$; 2) The labeling results satisfy the Markovianity: $\mathbf{x}$ depends on the labels in the local neighborhood.

The algorithm sequentially updates the segment result of every point in the image $x_i^{(k)}$ into $x_i^{(k+1)}$ by maximizing the conditional probability $P(x_i^{(k+1)} | \mathbf{y}, \mathbf{x}_{S-\{i\}}^{(k)})$, which is performed in a raster scan. The iteration stops when the number of pixels that change during a cycle is less than a threshold.

$$x_i^{(k+1)} = \arg\max_{x_i \in L} P(x_i \mid \mathbf{y}, \mathbf{x}_{S-\{i\}}^{(k)}) \tag{13}$$

Here $L = \{1,2,3\}$ represents the three different region types in neuron images. The initial estimate $\mathbf{x}^{(0)}$ can be obtained by any spine detection algorithms [6]-[11].

For real world applications such as dendritic spine detection, a pure statistically optimized solution is not always the best choice for the purpose of image analysis. For example, for neuron images, an intensity distribution based model is likely to segment a dendrite pixel as a spine pixel because of the similar intensity values. To deal with this issue, more constraints with explicit biological meanings must be incorporated into the processing. In our case, these new constraints are not used to modify the energy function. Instead, they are employed to develop a new strategy of searching for ideal solutions. Unlike the ICM algorithm, the proposed KICM algorithm will stop at some point before reaching the statistically optimal solution because of the biological constraints incorporated in the transition matrix. The solution, however, is intended to possess more biological meaning. For instance, spines are protrusions along the boundary of a dendrite. Thus, all pixels inside the dendrite should not be segmented as spine pixels.

To better describe the KICM algorithm, we first rewrite the normal ICM algorithm in a matrix form with the transition matrix A being defined. For each pixel i, suppose that $\mathbf{s}^{(k)} = [s_1^{(k)} \; s_2^{(k)} \; s_3^{(k)}]^{\mathrm{T}}$ is the segment result of the k-th step. Here $\mathbf{s}^{(k)} = [1\,0\,0]^{\mathrm{T}}$, $[0\,1\,0]^{\mathrm{T}}$, or $[0\,0\,1]^{\mathrm{T}}$ means the pixel is segment as the spine, dendrite or background respectively. Then we have

$$\mathbf{s}^{(k+1)} = A \cdot \mathbf{s}^{(k)} = \begin{bmatrix} a(1) & a(1) & a(1) \\ a(2) & a(2) & a(2) \\ a(3) & a(3) & a(3) \end{bmatrix} \cdot \mathbf{s}^{(k)} \tag{14}$$

Here A is the transition matrix,

$$a(m) = \begin{cases} 1, & \text{if } e_m = \min\{e_1, e_2, e_3\} \\ 0, & \text{if } e_m \neq \min\{e_1, e_2, e_3\} \end{cases} \tag{15}$$

and e_1, e_2, e_3 represent the energy if the pixel is segmented as spine, dendrite or background respectively. The energy can be calculated by Eq. (6) and (7):

$$e_1 = E_s = -\log p(y_i \mid x_i \in spine) + U(x_i \in spine) \tag{16}$$

$$e_2 = E_d = -\log p(y_i \mid x_i \in dendrite) + U(x_i \in dendrite) \tag{17}$$

$$e_3 = E_b = -\log p(y_i \mid x_i \in background) + U(x_i \in background) \tag{18}$$

With some modification of the transition matrix A, the KICM algorithm is described as

$$\mathbf{s}^{(k+1)} = \begin{cases} A \cdot \mathbf{s}^{(k)} = \begin{bmatrix} a(1) & a_{12} & a(1) \\ a_{21} & a(2) & a(2) \\ a(3) & a_{32} & a(3) \end{bmatrix} \cdot \mathbf{s}^{(k)}, & \text{if } A \cdot \mathbf{s}^{(k)} \neq [0\,0\,0]^{\mathrm{T}} \\ \mathbf{s}^{(k)}, & \text{otherwise} \end{cases} \tag{19}$$

Here the last column of the transition matrix is unchanged, which means if at the k-th step a pixel is segmented as background, at the next step it will be segmented only based on the lowest energy criterion. However, it is not the case if a pixel is formerly segmented as dendrite or spine.

The value of a_{12}, a_{21}, and a_{32} are calculated based on the prior knowledge (assumption) of the neuron image and the initial segmentation results:

1. In practice incorrectly segmented background pixels are some 'detached' noise blobs, which have relatively high intensity values and thus are likely to be segmented as spine pixels. Thus we assume in the initial segmentation results that no background pixels are segmented as dendrite. Based on this assumption we have $a_{32} = 0$, which means that initially segmented dendrite pixels cannot be segmented as the background later.
2. The spines are the protrusions along the dendrite boundary. If we can estimate the width of dendrite, then spine pixels can only be those pixels whose distance to the center of the dendrite (the backbone) is greater than half of the local dendrite width. The detailed algorithm to estimate the width of local dendrite piece is discussed in [9]. Based on this assumption, the value of a_{12} and a_{21} can be defined:

$$a_{12} = \begin{cases} 1, & \text{if } e_1 = \min\{e_1, e_2\} \text{ and } d_i > \dfrac{t_i}{2} \\ 0, & \text{otherwise} \end{cases} \tag{20}$$

$$a_{21} = \begin{cases} 1, & \text{if } e_2 = \min\{e_1, e_2\} \text{ and } d_i < \dfrac{t_i}{2} \\ 0, & \text{otherwise} \end{cases} \tag{21}$$

Here d_i is the distance between pixel i and backbone, t_i is the estimation of the width of local dendrite

Let x_i be the segment result for pixel i, $x_{S \backslash i}$ be the current segment result elsewhere, $x_{\partial i}$ is the segment result for the 8-neighbor ∂i of pixel i. For MRF models, we have

$$P(\mathbf{x} \mid \mathbf{y}) = P(x_i \mid y, x_{s \backslash i}) P(x_{s \backslash i} \mid y) = P(x_i \mid y, x_{\partial i}) P(x_{s \backslash i} \mid y) \tag{22}$$

We notice that $P(x_i \mid y, x_{\partial i})$ never decreases: x_i is either updated when lower energy is found, or is unchanged when either (a) lower energy can not be found or (b) when $A \cdot \mathbf{s}^{(k)} = [0\,0\,0]^T$ as described in Eq. (19). Besides, $P(x_{s \backslash i} \mid y)$ is kept constant no matter what value x_i is. Thus just like the traditional ICM algorithm, the KICM algorithm performs in a hill-climbing manner and eventual convergence is assured.
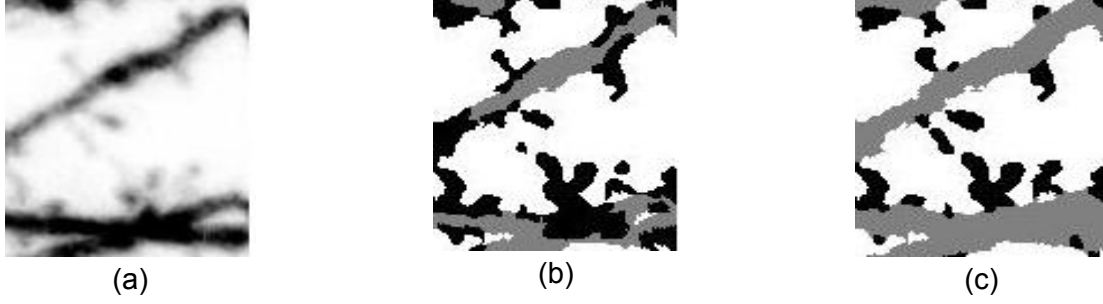
(a)   (b)   (c)

Figure 4: comparison between MRF-ICM and OMRF-KICM, (a) original image; (b) segment result with normal MRF-ICM algorithm; (c) segment result with OMRF-KICM algorithm

The KICM algorithm is a good complement to MRF-based algorithms, which are based on spatial constraints within local neighborhoods. Normally, performance improves as the size of the neighborhood increases. However, larger neighborhoods make the model more complex and greatly increase the computation complexity. That is the reason why 4- or 8- neighborhood models are normally used for MRFs. The KICM algorithm provides a novel perspective on how to combine more local information in a much bigger neighborhood without largely increasing the computation complexity. Together with the orientation information included in the prior models of the neuron images, the segmentation results can be noticeably improved. Fig. 4 shows the comparison of segmentation results between normal MRF-ICM algorithm and the proposed OMRF-KICM algorithm. We can observe from the results that the shapes of the dendritic spines are much better represented with the "weak" components (spine neck) being enhanced and the base of the spines accurately found.

In summary, the main frame of the proposed OMRF-KICM algorithm is described below:

1. Obtain region of interest by background compensation algorithm described in Section 3.3
2. Obtain the orientation map described in Section 3.2
3. Set the value of $\boldsymbol{\theta}_1$
4. Compute the likelihood probability $p(y_i \mid x_i)$ at position i
5. **Initialize** the algorithm with the segment result $\mathbf{x}^{(0)}$, $k = 0$
6. **Repeat**
7.   **Estimate** $\hat{\boldsymbol{\theta}}_2$ with equation (11) and (12)
8.   **Repeat** with $\mathbf{x}^{(k)}$ known
9.   **For** each pixel i in the image
10.   calculate energy e_1, e_2, e_3 using Eq. (16)-(18)
11.   update $a(m)$ using Eq. (15)
12. $\mathbf{s}^{(k+1)} = \mathrm{A} \cdot \mathbf{s}^{(k)}$ using Eq. (19)
13.   **End**
14. $k = k + 1$
15.   **Until** the difference between $\mathbf{x}^{(k)}$ and $\mathbf{x}^{(k-1)}$ is below a threshold

15

16. **Until** N times of iteration

17. **Return** $\mathbf{x}^{(k)}$ as the optimal segmentation results

IV Results and validation

4.1 Validation with existing algorithms

The proposed algorithm is first applied to the initial detection results obtained by existing detection algorithms based on both global thresholding [6] and adaptive thresholding [9]. We can observe from Fig. 5 and 6, the OMRF-KICM method can efficiently improve the spine detection results for neuron images with different pixel sizes.

(a) original MIP images   (b) results using global thresholding   (c) results using adaptive thresholding

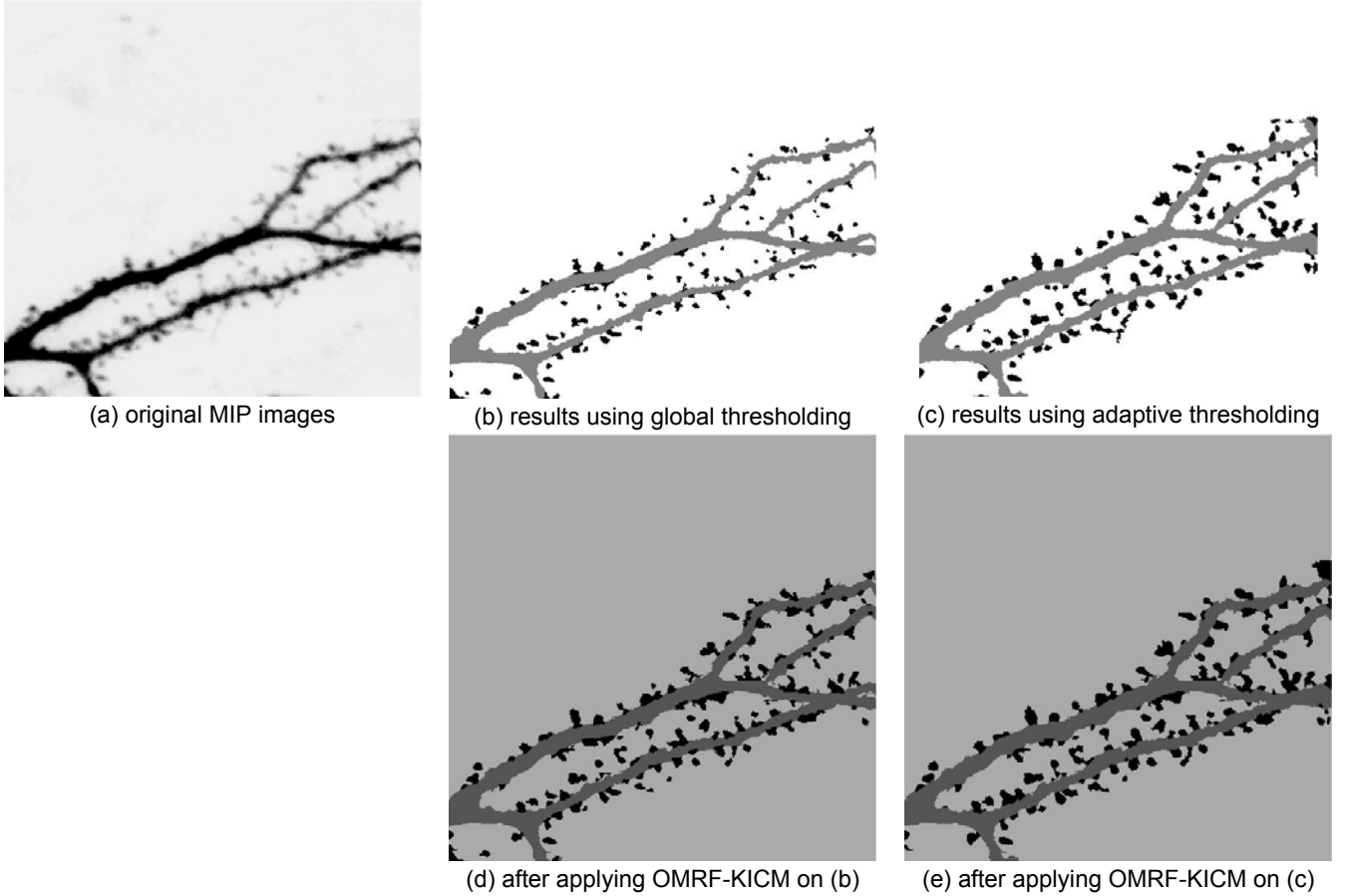(d) after applying OMRF-KICM on (b)   (e) after applying OMRF-KICM on (c)

Figure 5: Detection results for images on with pixel size of 0.08 micron/pixel. (a) MIP images; (b), (d) initial segmentation results using global thresholding based algorithms and results after applying the proposed algorithm model respectively; (c), (e) initial segmentation results using adaptive thresholding based algorithms and results after applying the proposed algorithm

Fig. 5 shows the detection results for neuron images acquired with the pixel size of 0.08 μm/pixel. From Fig. 5 (b) and (d), we can observe that the initial detection results obtained by global thresholding based algorithm are much improved: there are fewer missing spines (especially for the right-upper parts.) Also, the false positives are reduced by combining the

16

broken spine components. The 'weak' regions such as the spine neck areas have been effectively enhanced and the broken spine components are reduced. Compared with the global thresholding based methods, the detection results obtained using the adaptive thresholding-based algorithm [9] show fewer missing spines. However, there are still some false positives caused by the local maximums. The OMRF-KICM method can nevertheless remove those positives as shown in Fig. 5 (c) and (e).

(a) original MIP images   (b) results using global thresholding   (c) results using adaptive thresholding

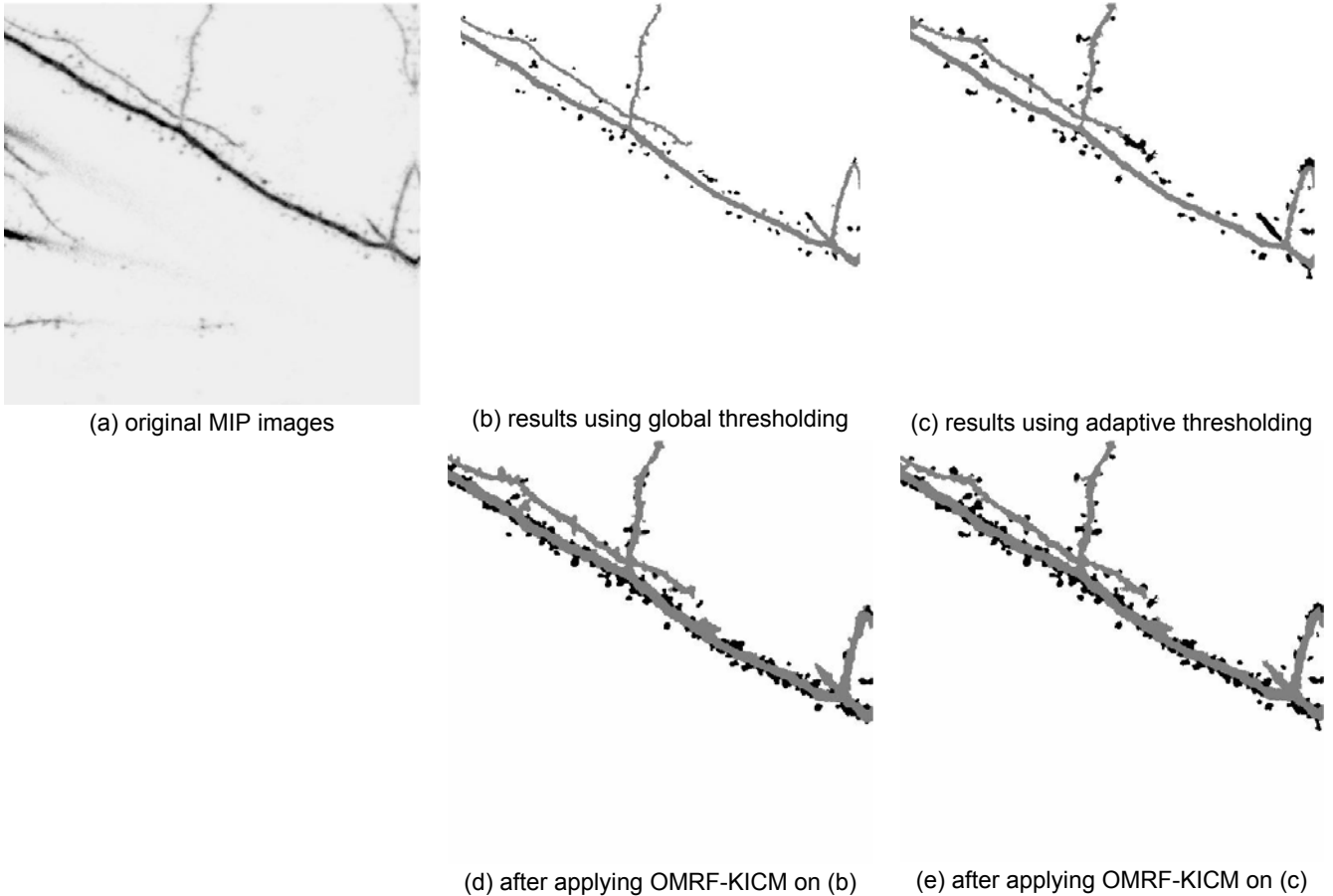(d) after applying OMRF-KICM on (b)   (e) after applying OMRF-KICM on (c)

Figure 6: Detection results for images with pixel size of 0.125 micron/pixel. (a) original MIP images; (b), (d) initial segmentation results using global thresholding based algorithms and results after applying OMRF-KICM method respectively; (c), (e) initial segmentation results using adaptive thresholding based algorithms and results after applying OMRF-KICM method

As shown in Fig. 6, for some neuron images acquired with larger pixel size, both global and adaptive thresholding based algorithms cannot efficiently detect low intensity, 'weak' spines. One easy approach is to include an additional preprocessing step of interpolation so that the image can be processed with a smaller image pixel size. However, problems exist for this method: besides the reduced detection performance caused by the interpolation distortion, the processing time also increases with a bigger image size. For five tested images by using the method described in [9], the average processing time for the interpolated images (762×762) is about 17'18'', which is more than a seven fold increase compared with the average 2'32'' processing time for images (512×512) without interpolation. The proposed algorithm can

nevertheless solve the missing problem well, with higher accuracy and much less processing time (average 2'5'' processing time for 512×512 images).

4.2 Validation with manual results

Automatic dendritic spine detection results, with and without the processing of the proposed algorithm, are first compared with the manual results, which are treated as the gold standard. Initially each automatically detected spine blob is labeled as a spine. However, in some cases a spine seems to be "broken" into several components, which is due to the limited resolution in the z- direction and the small size of spine neck. The broken components of a single spine will cause errors in the quantification of spine structures including spine number and length. To address this problem, a merging algorithm is performed to combine those separated parts into single spines. The details of the algorithm can be found in [9]. Basically the combination is based are two criteria: the broken parts should be close enough and the orientation of the line connecting the centroids of the separated components should be in a certain range of angle, e.g. $[-30^o, 30^o]$. For automatic results, each detected spine component is initially assigned an index number, in an order of the distance to the left edge of the image. After merging, the broken spine head components are reassigned the same index number as the spine base.

Fig. 7 illustrates how the broken spine components are combined and how the automatic detection results are validated using the manual results. Fig. 7 (a) is the segmentation result of part of image 1 by applying method 2 as mentioned in Table 1. We can observe that the broken head parts are successfully combined with the relative spine bases, e.g., spine number 23 and 27. During validation, the spines are manually marked in 3D. Fig. 7 (b) shows how the manually marked image looks in the 2D projected image. For better illustration, the false negatives and false positives are marked as yellow and red respectively. As we can observe in Fig 7, two manually marked spines can not be automatically detected; the automatically detected spine 32 is actually a false result.

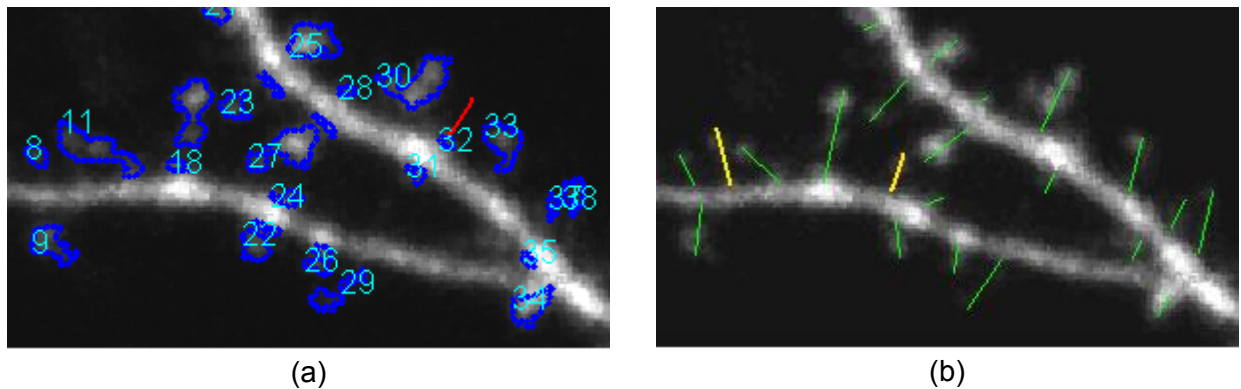(a)   (b)

Figure 7: Comparison of automatic and manual detection results in MIP image (a) labeled automatic results by applying method 2 as mentioned in Table 1, with falsely detected spine marked with red (b) manual results, with spine not detected by the automatic method marked with yellow.

The initial detection results are obtained using both global thresholding (method 1) and adaptive thresholding (method 2) based methods. Five neuron images with different pixel sizes

18

are compared. The manual results are included in the braces. The false positive (FP) and false negative (FN) values before and after applying the proposed method are listed in Table 1. As we can see from the results, the FP (wrong detections) and FN (undetected foreground) are decreased for both methods after the processing of the proposed method. The spine density measurement, however, does not show much improvement. To better illustrate the differences, the precision (P) and recall (R) rates are also provided. According to the paired t-test, the precision rates of methods with and without applying the proposed algorithm show significant difference with p-value 0.0025. The recall rates also show significant difference with p-value 0.001.

Table 1: Comparison of spine detection results. Method 1 and 2 are global and adaptive thresholding based algorithm respectively, after combining broken spine parts. Method 1+ and 2+ are the above two methods with applying the proposed algorithm. Images are acquired with two different pixel sizes, 0.125 micron/ pixel (Image 1, 3, 4) and 0.08 micron/pixel (Image 2, 5). The numbers in the brackets are the total number of spines in the respective images.

	Image 1 (82)		Image 2 (67)		Image 3 (86)		Image 4 (93)		Image 5 (36)	
	FP/P	FN/R	FP/P	FN/R	FP/P	FN/R	FP/P	FN/R	FP/P	FN/R
Method 1	8/90%	19/77%	1/99%	15/78%	12/86%	22/74%	4/96%	11/88%	2/94%	4/89%
Method 1 +	3/96%	9/89%	0/100%	7/90%	5/94%	8/91%	3/97%	6/94%	0/100%	2/94%
Method 2	3/96%	10/88%	2/97%	13/81%	6/93%	15/83%	5/95%	6/94%	0/100%	3/92%
Method 2 +	2/98%	6/93%	0/100%	6/91%	3/97%	5/94%	2/98%	2/98%	0/100%	2/94%

Further validation is performed by comparing the spine length distribution of the proposed method with the manual results. The details of how we automatically measure the length of attached and detached spines can be found in [9]. Spines in different section of the same neuron cell under the same condition (shLUCI in hippocampal pyramidal neurons in rat organotypic slice cultures) are tested. There are altogether 235 spines in three different images. The CDFs of the two distributions are plotted in Fig. 7, from which we can see that the distributions of spine length measured manually and automatically are similar to each other. To quantify the difference, the distributions are tested by two-sample Kolmogorov-Smirnov test. The null hypothesis that the two distributions are the same is not rejected, which means that there is no obvious difference between the two distributions. The probability that the dendrite length distributions of manual and automated results are the same is 99.13%. The biggest difference between these two distributions is 0.075.
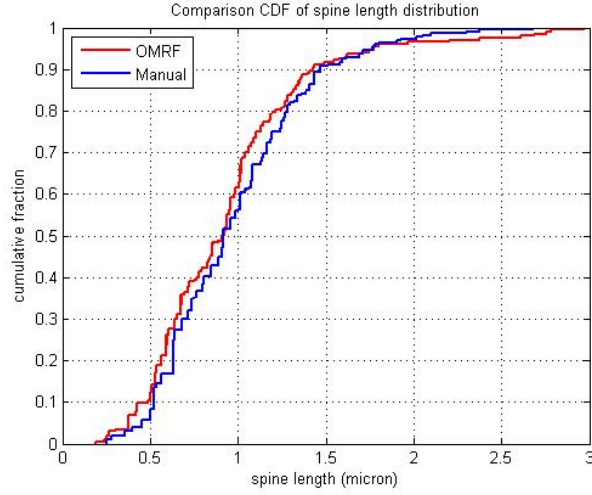
Figure 8: Comparison of spine length distribution of manual results and results obtained by the proposed method.

The proposed method is also validated using published manually segmented data. This recently published paper [21] examines the off-target effects of expressing short hairpin RNAs (shRNA) in neurons. The study shows that expression of shRNA against luciferase, whose coding region is not found in the rat genome, triggers dramatic loss of dendritic spines and simplification of dendritic arbors. Figure 8 shows the difference of average spine densities and average spine length for GUR and shLUCI neurons at 4DPT. By applying the proposed methods, a similar reduction of spine density and length by shLUCI as those identified by the manual analysis can be detected. These results, therefore, validate the proposed algorithm as a valuable tool for automatic analysis of neuronal morphology and the identification of biologically relevant changes in dendritic spine morphology and number.
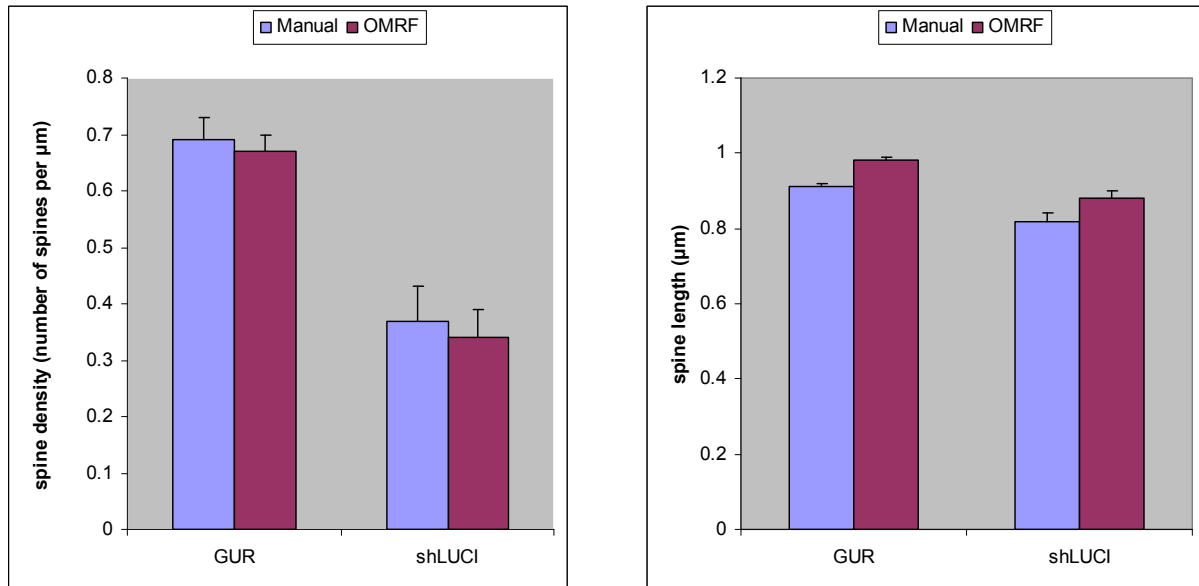

Figure 9: shRNA expression induces retraction of dendrites. Method 1 and 2 are manual and results of porposed method respectively. a). Summary of data of average spine density at 4DPT for GUR and shLUCI neurons. b). Summary of data of average spine length at 4DPT for GUR and shLUCI neurons.

20

V Discussion and conclusion

In this paper we propose a new OMRF-KICM based method which can efficiently improve the automatic detection results of existing algorithms. With the intensity distribution information of background, dendrite and spine pixels, as well as the context and orientation information of the neurons, OMRF-KICM method can obviously improve the detection performance for image with different pixel sizes. Specifically, according to the results in Table 1, we have demonstrated that the false positive and the false negative rates of spine detection can be decreased by more than 50% after applying the proposed method. There are mainly two motivations to improve the detection performance for images with larger pixel sizes. First, larger field of view can be obtained for images with larger pixel sizes. Second, because the already existed large amounts of experimental data with larger image pixel sizes, it would be very time consuming and a waste of resources if all the images have to be acquired again with smaller image pixel sizes.

The spine density measurement nevertheless showed a little improvement compared with the adaptive thresholding algorithm. This is because the total number of automatically detected spines is equal to the number of (actual spines + false positives – false negatives.) We can see from Table 1 that although the false negatives are reduced by 50%, the numbers of total detected spines do not change much because of the simultaneously reduced false positives. The measurement of the spine length however is much improved for images with large pixel size because of the more accurate detection of small spines and the removal of the long false spines.

Furthermore, the approach also holds promise in terms of providing better representation of the true shapes of the spines. The weak parts of the spines, especially the neck regions which are hard to detect, are better recognized. We notice that the dendritic trunks in the segmented images (Fig. 5, 6) being processed by the proposed method are generally thicker than those without the processing. This is partially because the proposed algorithm combines the local spatial information with the intensity distribution information for the segmentation. Generally, the neighboring pixels are prone to be segmented as the same class, although different weights are assigned for neighbors with different orientations. The (2~3 pixel) thicker dendritic trunks make it easier to detect the spine necks, however, to a limited extent: we notice that it is the central part of the spine neck, not the bottom part near to the dendritic trunks, has much lower intensity compared with other parts of the spines, which is the culprit of the broken spine necks.

We want to point out that the proposed automatic method is not designed for providing accurate *absolute* measurements about the spines such as the length. Our neurobiology collaborators are more interested in the comparison between the measurements under difference conditions. As long as the measurements can correctly reproduce the changes, accurate absolute measurements are less of a concern.

The proposed MAP-OMRF method can efficiently improve the automatic detection result. However, its performance can be further enhanced. As we observe from Fig. 7 (a), two

segmented spine blobs are attached to each other. It is a challenging problem to automatically separate the linked spines. Marker based watershed transformation or the level set algorithm can be the two potential choices to solve this problem. Also, the efficiency of constraint 2 of the KICM algorithm depends on the accuracy of the estimated local dendrite width. In cases where the size of one spine neck is abnormally large, the small nearby spines cannot be correctly detected. This problem needs to be addressed in the future work.

Another interesting topic for the future work is the online calculation of $\hat{\boldsymbol{\theta}}_1$ and $\hat{\boldsymbol{\theta}}_2$, which are the optimal estimation of $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ respectively. Suppose $\mathbf{y}$ is the observed data and $\mathbf{x}$ is the distribution of different regions, $\hat{\boldsymbol{\theta}}_2$ can be estimated via the maximization of the posterior probability function $p(\boldsymbol{\theta}_2 \mid \mathbf{y}, \mathbf{x})$, which is calculated based on the prior probability function $p(\boldsymbol{\theta}_2)$ and the conditional probability function $p(\mathbf{y}, \mathbf{x} \mid \boldsymbol{\theta}_2)$ according to the Bayes theorem. How to obtain $p(\boldsymbol{\theta}_2)$ is not straightforward, because of the random location of the dendrite and spine, as well as the inhomogeneous intensity in the images. MRF model based method is one of the possible approaches. Suppose $\hat{\mathbf{x}}$ is the optimal segmentation results, $\hat{\boldsymbol{\theta}}_1$ can be obtained by solving the following MAP estimation problem:

$$(\hat{\mathbf{x}}, \hat{\boldsymbol{\theta}}_1) = \arg\max_{\mathbf{x}, \boldsymbol{\theta}_1} P(\mathbf{x}, \boldsymbol{\theta}_1 \mid \hat{\boldsymbol{\theta}}_2, \mathbf{y}) \tag{22}$$

Assume that $\boldsymbol{\theta}_1$ is uniformly distributed, using the laws of conditional probabilities, equation (22) can be changed to

$$(\hat{\mathbf{x}}, \hat{\boldsymbol{\theta}}_1) = \arg\max_{\mathbf{x}, \boldsymbol{\theta}_1} P(\mathbf{y} \mid \mathbf{x}, \hat{\boldsymbol{\theta}}_2) P(\mathbf{x} \mid \boldsymbol{\theta}_1) P(\boldsymbol{\theta}_1)$$

$$= \arg\max_{\mathbf{x}, \boldsymbol{\theta}_1} P(\mathbf{y} \mid \mathbf{x}, \hat{\boldsymbol{\theta}}_2) P(\mathbf{x} \mid \boldsymbol{\theta}_1) \tag{23}$$

It can be further divided into two sub-problems which can be estimated iteratively.

$$\hat{\boldsymbol{\theta}}_1^{(k)} = \arg\max_{\boldsymbol{\theta}_1} P(\mathbf{y} \mid \hat{\mathbf{x}}^{(k-1)}, \hat{\boldsymbol{\theta}}_2) P(\hat{\mathbf{x}}^{(k-1)} \mid \boldsymbol{\theta}_1) \tag{24}$$

$$\hat{\mathbf{x}}^{(k+1)} = \arg\max_{\mathbf{x}} P(\mathbf{y} \mid \mathbf{x}, \hat{\boldsymbol{\theta}}_2) P(\mathbf{x} \mid \hat{\boldsymbol{\theta}}_1^{(k)}) \tag{25}$$

Many algorithms have been proposed to solve this problem, such as maximum likelihood, coding, pseudo-likelihood, expectation-maximization, mean field approximations, and least squares fit etc [23]. However, how to find a computationally efficient algorithm, which can efficiently combine the prior knowledge of the neuron images, remains a question to be investigated.

INFORMATION SHARING STATEMENT

The image data used in this work are provided by Professor B.L. Sabatini's lab at Harvard Medical School. We are unable to release these data. The algorithm used in this work will be included in the next version of NeruonIQ (Neuron Image Quantitator). NeuronIQ is public available and can be downloaded freely from http://www.cbi-tmhs.org/Neuroniq/index.htm.

ACKNOWLEDGMENTS

REFERENCES

[1] R. Yuste, T. Bonhoeffer, Morphological changes in dendritic spines associated with long-term synaptic plasticity, Annu. Rev. Neurosci., 24:1071–89 (2001)

[2] M.B. Moser, M. Trommald, P. Andersen, An increase in dendritic spine density on hippocampal CA1 pyramidal cells following spatial learning in adult rats suggests the formation of new synapses, Proc. Natl. Acad. Sci. 91:12673–75 (1994)

[3] J. C. Fiala, J. Spacekb, and K. M. Harrisa, Dendritic Spine Pathology: Cause or Consequence of Neurological Disorders, *Brain Resear. Rev.*, June, 39(1): 29-54, (2002)

[4] M. Knobloch, and I. M. Mansuy, Dendritic spine loss and synaptic alterations in Alzheimer's disease, *Molecular Neurobiology*, Feb., 37(1): 73-82 (2008)

[5] C.M. Durand, Mutations in the gene encoding the synaptic scaffolding protein SHANK3 are associated with autism spectrum disorders. Nat Genet 39:25–27 (2007)

[6] I. Y. Koh, W. B. Lindquist, K. Zito, E. A. Nimchinsky, and K. Svoboda, An image analysis algorithm for dendritic spines, *Neur. Comput.* 14: 1283–310 (2002)

[7] X. Xu, J. Cheng, R. M. Witt, B. L. Sabatini, S. T.C. Wong, A shape analysis method to detect dendritic spine in 3D optical microscopy image, *Biomedal Imaging: Macro to Nano*,. 3rd IEEE International Symposium, Apr. 6, pp554 – 557 (2006)

[8] C. M. Weaver, P. R. Hof, S. L. Wearne, W. Lindquist, Automated algorithms for multiscale morphometry of neuronal dendrites, *Neur. Comput.* 16: 1353-83 (2004)

[9] J. Cheng, X. Zhou, E. Miller, R. M. Witt, J. Zhu, B. L. Sabatini, and S. T. Wong, A novel computational approach for automatic dendrite spines detection in two-photon laser scanning microscopy. *J. of Neurosci. Methods* 165: 122-134 (2007)

[10] W. Bai, X. Zhou, L. Ji, J. Cheng, and S. T. Wong, Automatic dendritic spine analysis in twophoton laser scanning microscopy images, *Cytometry A* 71: 818-826 (2007)

[11] Y. Zhang, X. Zhou, R. M. Witt, B. L. Sabatini, D. Adjeroh, and S. T. Wong, Dendritic spine detection using curvilinear structure detector and LDA classifier. *Neuroimag.* 36: 346–360 (2007)

[12] G. Xiao, M. Brady, and J. A. Noble, Segmentation of ultrasound B-mode images with intensity inhomogeneity correction*, IEEE Trans. on Med. Imag.* Jan., 21(1): 48-57 (2002)

[13] A. Juslin, and J. Tohka, Unsupervised segmentation of cardiac PET transmission images for automatic heart volume extraction, *Proc. of the 28th IEEE EMBS ann. int. conf.*, Aug., 1:1077-80 (2006)

[14] S. Ruan, and D. Bloyet, MRF models and multifractal analysis for MRI segmentation, *Proc. of ICSP*, pp. 1259-62 (2000)

[15] Y. Zhang, M. Brady and S. Smith, Segmentation of brain MR images through a hidden markov random field model and the expectation-maximization algorithm, *IEEE Trans. on Med. Imag.*, Jan., 20(1): 45-57 (2001)

[16] V.D. Calhoun, T. Adali, G.D. Pearlson, and J.J.Pekar, Spatial and temporal independent component analysis of functional MRI data containing a pair of task-related waveforms, *Hum. Brain Mapp.*, 13: 43-53 (2001)

[17] X. Descombes, F.Kruggel, and D.Y. von Cramon, Spatio-temporal fMRI analysis using Markov random fields, *IEEE Trans. Med. Imag.*, Dec., 17(6): 1028-39 (1998)

[18] E. Salli, H.J. Aronen, S. Savolainen, A. Korvenoja, and A. Visa, Contextual clustering for analysis of functional MRI data, *IEEE Trans. Med. Imag.*, May, 20(5): 403-14 (2001)

[19] C. D'Elia, C. Marrocco, M. Molinara, G. possi, G. Scarpa, and F. Tortorella, Detection of microcalcifications clusters in mammograms through TS-MRF segmentation and SVM-based classification, *Proc. of the 17th int. conf. on Patt. Recog.* 3: 742-45 (2004)

[20] H.D. Li, M. Kallergi, L.P. Clarke, V.K. Jain, and R.A. Clark, Markov random field for tumor detection in digital mammography, *IEEE Trans. on med. Imag.* Sep., 14(3): 565-576 (1995)

[21] Veronica A. Alvarez, Dennis A. Ridenour, and Bernardo L. Sabatini, Retraction of synapses and dendritic spines induced by off-target effects of RNA Interference, *J. of Neuroscience*, July 2006, 26(30)

[22] T.J. Hebert, Fast iterative segmentation of high resolution medical images, *IEEE Trans. on Nuc. Sci.*, 44(3): 1362-67 (1997)

[23] S.Z. Li, Markov Random Field Modeling in Image Analysis, 2nd ed. Tokyo, Japan: *Springer-Verlag* (2001)

[24] C. Wahlby, J. Lindblad, M. Vondrus, E. Bengtsson and L. Bjorkesten, Algorithms for cytoplasm segmentation of fluorescence labeled cells, *Analytical Cellular Pathology*, 24 (2002)

[25] S. Geman and D. Geman, Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images, *IEEE Trans. on PAMI*, 6(6):721-41 (1984)

[26] J. Besag, On the Statistical Analysis of Dirty Pictures, *J. of the Royal Statistical Society*, Series B, 48(3):259-302 (1986)

[27] J. Besag, Towards Bayesian Image Analysis, J. of Applied Statistics, 16(3): 395-407 (1989)

[28] T. N. Pappas, An adaptive clustering algorithm for image segmentation, *IEEE Trans. on Signal Processing*, 40(4):901-914 (1992)

[29] J.C. Rajapakse, Y. Wang, X. Zheng, J. Zhou, Probabilistic Framework for Brain Connectivity From Functional MR Images, *IEEE Trans. on Med. Imag.*, 27(6):825 - 833 (2008)

[30] S.P. Awate, R.T. Whitaker, Feature-Preserving MRI Denoising: A Nonparametric Empirical Bayes Approach, *IEEE Trans. on Med. Imag.*, 26(9):1242 - 1255 (2007)

[31] S.F. Tavazoie, V.A. Alvarez, D.A. Ridenour, D.J. Kwiatkowski, B.L. Sabatini, Regulation of neuronal morphology and function by the tumor suppressors Tsc1 and Tsc2, *Nat. Neurosci.*, 8(12): 1727-1734 (2005)

[32] T.V. Bilousova, L. Dansie, M. Ngo, J. Aye, J.R. Charles, D.W. Ethell, I.M. Ethell, Minocycline promotes dendritic spine maturation and improves behavioural performance in the fragile X mouse model, J. of Med. Genetics 46:94-102 (2009)

[33] J. Fan, X. Zhou, J. G. Dy, Y. Zhang, S.T. Wong, An automated pipeline for dendrite spine detection and tracking of 3D optical microscopy neuron images of in vivo mouse models, Neuroinformatics, 7:113-130 (2009)