AUTOMATIC DENDRITIC SPINE DETECTION USING MULTISCALE DOT ENHANCEMENT FILTERS AND SIFT FEATURES

*Lavdie Rada**   Ertunc Erdil*   A. Ozgur Argunsah†   Devrim Unay‡   Mujdat Cetin**

* Faculty of Engineering and Natural Sciences, Sabanci University, Tuzla, Istanbul, Turkey
{lavdierada, ertuncerdil, mcetin}@sabanciuniv.edu
† Champalimaud Neuroscience Programme, Champalimaud Centre for the Unknown, Lisbon, Portugal
ali.argunsah@neuro.fchampalimaud.org
‡ Biomedical Engineering Department, Bahcesehir University, Besiktas, Istanbul, Turkey
devrim.unay@bahcesehir.edu.tr

ABSTRACT

Statistical characterization of morphological changes of dendritic spines is becoming of crucial interest in the field of neurobiology. Automatic detection and segmentation of dendritic spines promises significant reductions on the time spent by the scientists and reduces the subjectivity concerns. In this paper, we present two approaches for automated detection of dendritic spines in 2-photon laser scanning microscopy (2pLSM) images. The first method combines the idea of dot enhancement filters with information from the dendritic skeleton. The second method learns an SVM classifier by utilizing some pre-labeled SIFT feature descriptors and uses the classifier to detect dendritic spines in new images. For the segmentation of detected spines, we employ a watershed-variational segmentation algorithm. We evaluate the proposed approaches by comparing with manual segmentations of domain experts and the results of a non-commercial software, NeuronIQ. Our methods produce promising detection rate with high segmentation accuracy thus can serve as a useful tool for spine analysis.

Index Terms— 2-photon microscopy, dendritic spine detection, dot enhancement filter, SIFT features, SVM classifier

1. INTRODUCTION

Dendritic spines are small bulbous cellular compartments recognized as small objects that carry synapses. Morphological changes of dendritic spines are of great interest in neuronal structure and the correlated function at the level of individual spines [1, 2, 3, 4, 5, 6]. Understanding structural changes of spines may provide explanation of how information is stored in the brain as well as contribute to our understanding of several neurodevelopmental disorders [7, 8].

Referring to previous work on automating dendritic spine analysis [9, 10, 11, 12, 13, 14] the problem of correctly detecting spines still remains unsolved and manual interactions are required. We can classify existing methods as centerline extraction based and classification based methods. Centerline extraction based methods consider the spines as small extended objects attached to the dendrites after detecting all dendrite centerlines in the image. The earliest centerline extraction based semi-automatic dendritic spine analysis approaches, proposed by [13, 15], were improved by Zhang et al.

[16] and later by Cheng et al. [17], utilizing the local SNR, which considerably reduces the effect of noise in spine detection. Lastly, to minimize the noise influence and for a better segmentation, local spine detection and geodesic active contour segmentation has been combined by Son et al. [18]. This method fails if there are thinning points into the dendrite due to low resolution and it also has the limitation that geodesic active contour segmentation is not capable of handling noise properly. Classification based methods separate image pixels into different groups with proper classifiers for specific cases. Such a method has been recently introduced by Rodriguez et al. [19]. The method uses point clustering for spine detection improving over Zhang et al. [14]. Since the method uses the distances from the points to the closest point of the surface as the clustering criterion, it may cause spurious spines and even fail due to noise.

In this paper, we propose two methods for automated detection and segmentation of dendritic spines. The methods are able to accurately segment each individual spine without any manual interaction and can handle the noise as well as low resolution of the image. Our first detection technique, introduced in Section 2, extracts the dendritic spines by estimating the second-order directional derivatives, serving as a dot enhancement filter, and eliminates false positives utilizing end points of the dendritic skeleton. In Section 3, we introduce our second method which employs a support vector machine (SVM) classifier with scale invariant feature transform (SIFT) based features. We note that the first approach is unsupervised, whereas the second approach uses training data to learn a classifier for spine detection. To the best of our knowledge, dot enhancement filters and SIFT features have not been adapted to and used for the spine detection before. After spine regions are detected, both methods use a watershed-variational based technique for segmentation which is detailed in Section 4. In Section 5, we present experimental results characterizing the detection and segmentation accuracy of both methods. Finally, we conclude in Section 6.

2. METHOD 1: DOT ENHANCEMENT FILTER AND IMAGE MORPHOLOGY BASED SPINE DETECTION

Our first method combines dot enhancement filters and skeleton end points. They are an attractive tool for spine detection, because of their strong sensitivity response for dots and their good specificity of not producing non dot shapes.

The filters we use in this paper are based on the eigenvalue analysis of the Hessian matrix of the intensities at each pixel of the given

image and have previously been used by Li et al. [20] as parameter-free techniques to improve the sensitivity of nodule detection. In the following, we shortly describe these filters in a $2D$ domain.

Construction of dot enhancement filters in 2D image space: For a given $2D$ image already preprocessed with a median filter, $I(x, y)$, the second derivatives are represented as I_{xx}, I_{yy} and $I_{xy} = I_{yx}$. Then, for each pixel we can construct the Hessian matrix $H = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{yx} & I_{yy} \end{bmatrix}$, which is a real symmetric matrix. We find the eigenvalue of Hessian matrix H determined by the equation:

$$\lambda_1 = K + \sqrt{K^2 - Q^2}, \lambda_2 = K - \sqrt{K^2 - Q^2}, \quad (1)$$

where $K = (I_{xx} + I_{yy})/2$, and $Q = \sqrt{I_{xx} * I_{yy} - I_{xy} * I_{yx}}$.

Without loss of generality, we assume λ_1 is the largest eigenvalue such that satisfy $|\lambda_1| \geq |\lambda_2|$, otherwise we do exchange λ_1 and λ_2. Because we attempt to enhance bright objects in a dark background, the sign of the second derivatives should be negative and for a dot (nodule like object) we expect $\lambda_1 \approx \lambda_2 < 0$. Defining $k(\lambda_1, \lambda_2) = |\lambda_2|/|\lambda_1|$, strict equality of the two eigenvalues leads to $k(\lambda_1, \lambda_2) = 1$. The function $k(\lambda_1, \lambda_2)$ can be used to differentiate between a circle, an ellipse, and a line, providing a value of 1 for a circle, a value between 0 and 1 for an ellipse and 0 for lines.

For filter quality evaluation, the incorporation of the two basic criteria, i.e., sensitivity and specificity, enables a good enhancement filter output. The output $z(\lambda_1, \lambda_2)$ of the dot enhancement filters, considered in [20], is defined by the product of the magnitude function and likelihood function. Likelihood is directly related to the sensitivity and specificity of an enhancement filter, indicating the probability that a pixel belongs to a dot. Li et al. [20] considered the function $k(\lambda_1, \lambda_2)$ as an approximate measure of the likelihood of a dot. A good choice for the magnitude function for a dot enhancement filter is the value of $|\lambda_2|$, providing a value greater than 0, indicating a good sensitivity to whether a bright object exists in a dark background. Therefore, the final enhancement filters for dots expressed as

$$z(\lambda_1, \lambda_2) = \begin{cases} |\lambda_2|^2/|\lambda_1| & \text{for } \lambda_1 < 0, \lambda_2 < 0 \\ 0 & \text{otherwise .} \end{cases} \quad (2)$$

The use of eigenvalues of the Hessian matrix has been previously investigated for construction of blob filters, but this is more complicated than the approach considered here and unknown parameters need to be determined [21].

To improve the performance of the filter described in equation (2) we can take into account the effect of noise and the scale of the objects. This can be done by using multiscale enhancement filters [20, 22]. Convolving the original image with a $2D$ Gaussian function before calculating the second derivatives not only smoothes the image but also serves as a matched filter to preserve an object with an appropriate scale. To enhance all objects in a range of scale $[d_0, d_1]$, we should first employ a Gaussian smoothing filter in the scale range of $[d_0/4, d_1/4]$ and then apply the dot enhancement filter, detailed above. The enhancement filters consist of two steps which should be repeated N times by increasing smoothing scales to provide N enhanced images. The N discrete smoothing scales can be selected as follows: $\sigma_1 = d_0/4$, $\sigma_2 = r\sigma_1, \ldots, \sigma_N = r^N \sigma_1 = d_1/4$, where $r = (d_1/d_0)^{1/N-1}$. The final output of the multiscale enhancement filters at a pixel is the maximum value from the N individual filters. Figure 1 (b) reveals the dot enhancement filter results in a dendritic spine image.

End-point pixels of a skeleton and spine detection: After the dot enhancement process, we exploit the information that spines are attached to the tips of dendritic branches. In particular, we compute the skeleton of the dendritic structures, and combine the information from such skeletons with the enhanced dots for accurate spine detection. To extract the skeleton, we first binarize the input image using an adaptive thresholding algorithm, then apply morphological thinning on this binary image. In order to decide whether a pixel is an end-point, we have to check its connectivity with its neighbors. In $2D$, pixels that have less than 2 neighbors, within their 8-neighborhoods connectivity, are considered as skeleton end-points.

Due to noise, blur and different artifacts in the given image, skeletonization can produce some falsely disconnected pieces in the main backbone. Because of that, we can get false end-points in the dendritic skeleton backbone, see Figure 1 (c), which are not spines. To avoid this false positivity we do check for all skeleton end-points if in their 2×2 neighborhood there is a dot already selected with dot enhancement filters. The combination ensures that the skeleton end-points found have a dot shape as well. In return, the skeleton end points help us eliminate some false positives that would be created by the dot enhancement filters. Hence dot-like skeleton end points are detected as spines by our approach (see Figure 1 (d)).

3. METHOD 2: SIFT FEATURES AND SVM CLASSIFIER BASED SPINE DETECTION

SIFT is a well-known method introduced by [23] to extract local image features invariant to image scaling, translation, rotation, and illumination changes. The method transforms an image into a large collection of local feature vectors invariant to the image and distinguishes keypoints identifying candidate object matches. SIFT features are represented by a 128 dimensional vector and each vector corresponds to a keypoint location in the image. Extracting SIFT features from 2pLSM images reveals that spine centers are generally found as the keypoint of a feature which is the main motivation of our method.

This method can be divided into two different phases: training and test. In the training phase, we extract SIFT features from a set of training images and manually label each feature vector as spine if its keypoint corresponds to a spine center, and as non-spine otherwise. Then, we train an SVM classifier with linear kernel using these pre-labeled feature vectors. More sophisticated kernels can also be used for SVMs, but that may increase the number of parameters to be set. In the test phase of the algorithm, we label each SIFT feature vector obtained from a new test image as spine or non-spine using the classifier constructed in the training phase. Finally, keypoint locations of feature vectors that are labeled as spine become our detected spine regions as shown with red markers in Figure 2 (a).

We can easily notice that the second method is less dependent on the parameters in comparison with the first method. In this way we can avoid the sensitivity of the final results in terms of dendritic morphology. The process is significantly simplified and very fast.

After spines are detected using one of the above methods, we employ a segmentation algorithm to determine spine boundaries accurately, explained in the following section.

4. WATERSHED-VARIATIONAL BASED SEGMENTATION

To segment a spine that have already been detected using above methods, we apply a multi-level segmentation algorithm. As a first level segmentation we use the watershed algorithm, as used in [24]. For each of the methods above, locations of detected spines help to label the watershed segments as spine and non spine regions, Figure

1 (e). Watershed algorithm usually finds larger boundaries than the expert results as stated in [25]. Therefore, we apply a second level segmentation for an accurate segmentation of the spine regions. In the second step, each spine region is refined using a variational based algorithm. This step eliminates the over-segmentation by stopping on the boundary of target spines. To have a variational function which will decrease as soon as we are in the boundaries and which at the same time performs good in presence of noise or fuzzy boundaries we employ a combination of an edge detection function, in particular [26], with a region based one, similar to Chan-Vese (CV) [27].

The minimization equation used for our region-edge based model is:

$$\min_{\Gamma, c_1, c_2} F(\Gamma, c_1, c_2) = \min_{\Gamma, c_1, c_2} \left\{ \mu \int_\Gamma g(|\nabla I_0(x,y)|) ds + \quad (3) \right.$$

$$\left. \lambda \left[\int_{\text{inside}(\Gamma)} |I_0(x,y) - c_1|^2 dx dy + \int_{\text{outside}(\Gamma)} |I_0(x,y) - c_2|^2 dx dy \right] \right\}$$

with μ, λ positive parameters and Γ the contour. The first term is composed by $g(|\nabla I_0(x,y)|) = \frac{1}{1 + \nu |\nabla I_0(x,y)|^2}$, an edge detector function which helps to stop the evolving curve on the edges of the objects in an image. Clearly the $g(|\nabla I_0(x,y)|)$ function is almost 0 on edges where $|\nabla I_0(x,y)|$ is large and 1 in flat regions where $|\nabla I_0(x,y)|$ is small. Our aim is to find a contour Γ such that $g \simeq 0$ along it. The second term is CV [27] region fitting term.

Introducing the level set functions $\phi = \phi(x,y)$ one gets the following Euler-Lagrange equation

$$\delta_\epsilon(\phi) \left\{ \mu \nabla \cdot \left(g \frac{\nabla \phi}{|\nabla \phi|} \right) - \lambda((I_0 - c_1)^2 - (I_0 - c_2)^2) \right\} = 0,$$

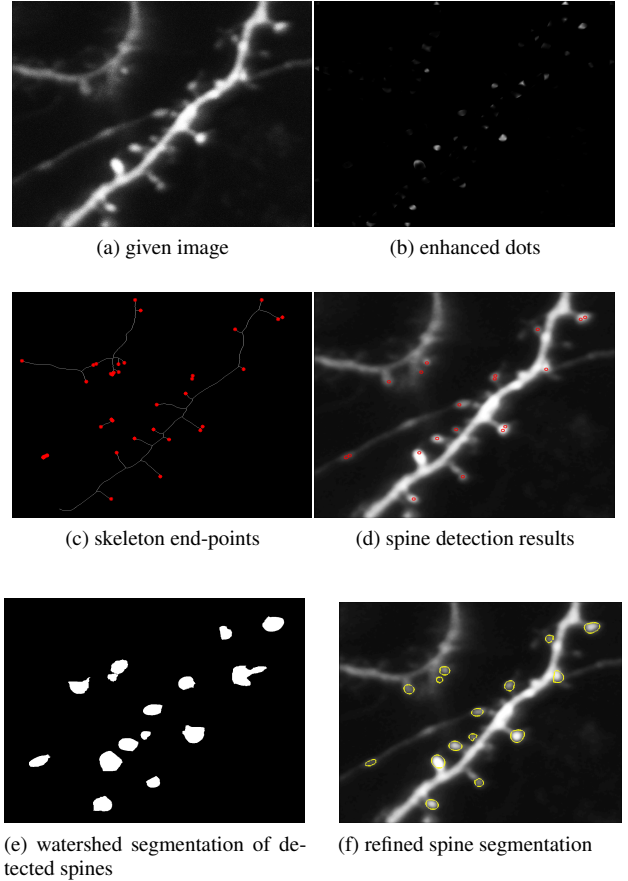with c_1 and c_2 computed similar to the CV model using regularized Heaviside function H_ϵ and the corresponding Delta function δ_ϵ. In this paper we use as Heaviside function $H_\epsilon = \frac{1}{2}(1 + \frac{2}{\pi} \arctan(\frac{z}{\epsilon}))$ while as first initial level set ϕ_0 we use a multi level set with center in the spine key points. Visualization of segmentation results for spines that already have been detected using Method1 and Method2 are shown in Figure 1 (f) and 2 (b), respectively.

5. EXPERIMENTAL RESULTS

In this section, we present experimental results which demonstrate the performance of our spine detection algorithms as well as that of the subsequent segmentation process. We start presenting our experimental results by showing the performance of our spine detection algorithms as well as segmentation accuracy. We use a set of 14 different $2D$ 2pLSM images provided by the Champalimaud Centre Neuronal Structure and Function Laboratory. We compare our results with those of existing work contained in the noncommercial software NeuronIQ [17] (http://www.cbi-tmhs.org/Neuroniq).

Since our goal was to detect of spines with diameters ranging from $4(d_0)$ to $32(d_1)$, the smoothing scales used for dot filters ranged from 1 to 8 in all of our experiments. The parameters μ, λ, ϵ and ν, have been fixed at $\mu = 500$, $\lambda = 1$, $\epsilon = 1$, $\nu = 100$, respectively. As trained data for the second method, we use only 2 images, different from the 14 test images. Optimization of the considered parameters will be our future work.

Analysis 1:- robustness and accuracy of our spine detection algorithms and watershed-variational segmentation. In order to obtain quantitative results for spine detection, we used the average of the decisions of two different domain experts as ground truth. Due


(a) given image   (b) enhanced dots

(c) skeleton end-points   (d) spine detection results

(e) watershed segmentation of detected spines   (f) refined spine segmentation

Fig. 1: Successful segmentation of a dendritic spine image using dot-enhancement-based spine detection followed by watershed-variational segmentation.

to noise and artifacts in 2pLSM images there may be high variations even in the decisions of the experts. In Table 1, we present the average true positive (TP) and false positive (FP) detection results for each image in the data set. We also give the average sensitivity results of all images for both methods.

As shown from Table 1, the second method performs better than the first method. Method1 also contains parameters to choose which is a disadvantage. Therefore, we can conclude that Method2 is preferable when the training data are available. However, there are some cases in which Method1 is superior to Method2. The first case, when there is no training data set, is obvious, since Method2 cannot be used without a training set. The second case is a scenario in which the training set and the test set are obtained from different modalities. To show this, we run both techniques on a confocal microscopy image. For these experiments, we use a training set of 2pLSM images - same training set with the above experiments - and test on a confocal microscopy image shown in Figure 3. The experimental results demonstrate that Method1 performs better than Method2 in this circumstance as shown in Figure 3. Note that Method2 captures some dendrite parts as spine and misses a number of spines around the longer dendrite whereas Method1 captures most spines with fewer false positives. If we construct the trained data using confocal microscopy images we do get better results with

(a) SIFT key-points   (b) segmented spines
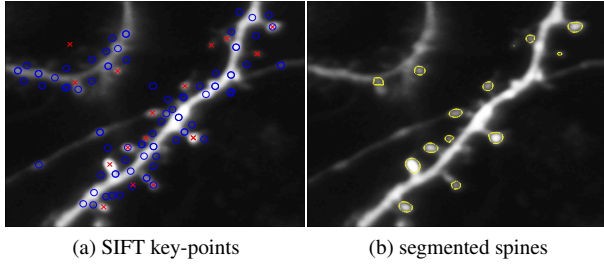
Fig. 2: Successful segmentation using SIFT-based spine detection followed by watershed-variational segmentation.
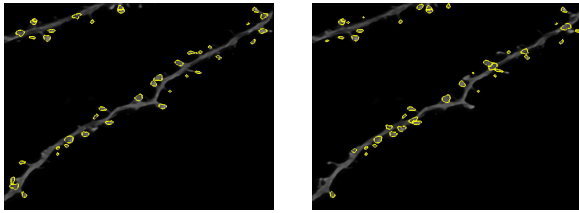
Fig. 3: Results on a confocal microscopy image. Left image: result of dot-enhancement-based spine detection followed by watershed-variational segmentation. Right image: result of SIFT-based spine detection (trained on 2pLSM images) followed by watershed-variational segmentation.

	average ♯ of spines labeled by experts	$Method\ 1$ TP	$Method\ 1$ FP	$Method\ 2$ TP	$Method\ 2$ FP
Image 1	13	12	2	11	1
Image 2	14	10	3	10	1
Image 3	18	14	2	14	1
Image 4	30	22	5	23	2
Image 5	22	9	2	17	1
Image 6	13	12	3	9	2
Image 7	22	12	3	14	3
Image 8	21	8	2	14	1
Image 9	40	27	3	36	4
Image 10	37	25	2	30	2
Image 11	41	27	3	34	10
Image 12	36	31	4	32	12
Image 13	24	14	0	13	1
Image 14	12	9	5	10	5
Average Sensitivity		0.8421		0.8665	

Table 1: Detection performance of the proposed methods
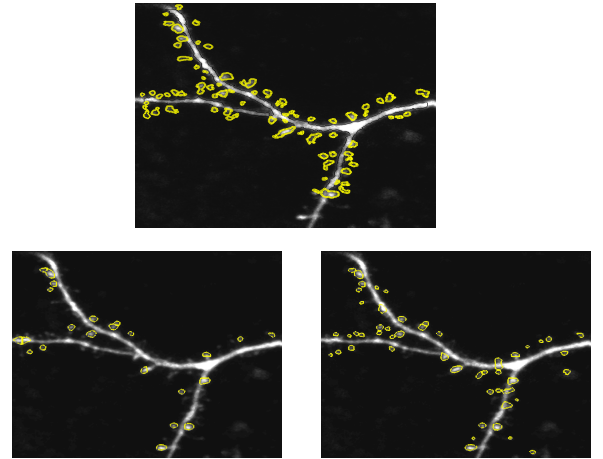
Fig. 4: Results on an image from NeuronIQ web with default parameters. Top image: NeuronIQ software segmentation. Left image: result of dot-enhancement-based spine detection followed by watershed-variational segmentation. Right image: result of SIFT-based spine detection followed by watershed-variational segmentation.

Fig. 5: Result of NeuronIQ on a 2pLSM image from the Champalimaud Centre. Results of our approaches on this image are shown in Figures 1(f) and 2(e).

Method2, which for the sake of brevity we do not show here.

We have evaluated the accuracy of watershed-variational segmentation by comparing the results with manual delineations of a domain expert. We used Dice coefficient [28] as metric which measures the similarity of two sets. Dice coefficient can take a maximum value of 1 which indicates a perfect match. We computed Dice coefficients for 36 different spines and obtained 0.823 on average with minimum 0.574 and maximum 0.968.

Analysis 2:- comparison with NeuronIQ. We compare our methods with a widely used noncommercial software called NeuronIQ developed by The Methodist Research Institute, Center for Bioengineering and Informatics. We first process the images found in the NeuronIQ web. Figure 4 shows the results obtained by our methods and by NeuronIQ. Even though NeuronIQ detects more spines than our methods, we do notice that NeuronIQ will segment spines as well as protrusions or noise and can sometimes split a spine into several fragments depending on the intensity distribution of the image. This is due to the adaptive threshold method used to detect dendritic spines similar to [17].

Moreover, NeuronIQ has many parameters to choose and small changes in parameters appear to cause significant effects on results. While it produces very good results for the images on NeuronIQ web with default parameters, it fails on the images from the Neuronal Structure and Function Laboratory of Champalimaud Centre with the same parameters as shown in Figure 5. Parameters need to be fine-tuned to achieve good results on different images.

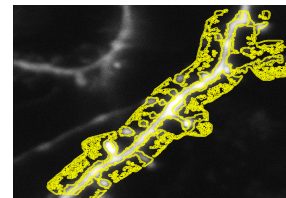## 6. CONCLUSIONS AND FUTURE WORK

We have proposed two spine detection methods and combined them with a powerful approach for spine segmentation. Our first approach is an unsupervised method based on dot enhancement filters and utilizes information from the skeletonization of the dendritic branches. The second method is a supervised method, based on SIFT features and SVM classification. When training data from the imaging modality to be used for the test images are available, the second method should be preferred. The second method avoids the use of several free parameters appearing in the first method as well. When training data are not available, we observe that our first approach provides very good spine detection results purely based on the observed data.

7. REFERENCES

[1] C. D. Harvey and K. Svoboda, "Locally dynamic synaptic learning rules in pyramidal neuron dendrites," *Nature*, , no. 450, pp. 1195–1200, 2011.

[2] H. Hering and M. Sheng, "Dentritic spines: Structure, dynamics and regulation," *Nature Reviews Neuroscience*, vol. 2, pp. 880–888, 2001.

[3] M. Maletic-Savatic, R. Malinow, and K. Svoboda, "Rapid dendritic morphogenesis in ca1 hippocampal dendrites induced by synaptic activity," *Science*, vol. 283, pp. 1923–1927, 1999.

[4] X. Zhou, J. Zhu, K. Liu, B. L. Sabatini, and S. T. C. Wong, "Mutual information based feature selection in studying perturbation of dendritic structure caused by tsc2 inactivation," *Neuroinformatics*, vol. 4, pp. 81–94, 2006.

[5] J. Tanaka, "Protein synthesis and neurotrophin-dependent structural plasticity of single dendritic spines science," *Science*, vol. 319, pp. 1683–1687, 2008.

[6] G. Arvind, I. Inbal, H. Shu-Ying, and T. Susumu, "The dendritic branch is the preferred integrative unit for protein synthesis-dependent ltp," *Neuron*, vol. 69, no. 1, pp. 132–146, 2011.

[7] M. F. Bear, K. M. Huber, and S. T. Warren, "The mglur theory of fragile x mental retardation," *Trends in Neurosciences*, vol. 27, no. 7, pp. 370–377, 2004.

[8] S. M. Weng, F. McLeod, M. E. S. Bailey, and S. R. Cobb., "Synaptic plasticity deficits in an experimental model of rett syndrome: long-term potentiation saturation and its pharmacological reversal," *Neuroscience*, vol. 180, pp. 314–321, 2011.

[9] I. M. Adams, "Structural plasticity of synapses in alzheimer's disease," *Molecular Neurobiology*, vol. 5, pp. 2–4, 1991.

[10] A. Auffret, V. Gautheron, M. Repici, R. Kraftsik, H. T. J. Mount, J. Mariani, and C. Rovira, "Age-dependent impairment of spine morphology and synaptic plasticity in hippocampal ca1 neurons of a presenilin 1 transgenic mouse model of alzheimer's disease," *Journal of Neuroscience*, vol. 29, no. 32, pp. 10144–10152, 2009.

[11] C. R. Gerfen, "Indirect-pathway neurons lose their spines in parkinson disease," *Nature Neuroscience*, vol. 9, pp. 156–158, 2006.

[12] 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, pp. 171–187, 2002.

[13] 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, pp. 1284–1310, 2002.

[14] Y. Zhang, X. Zhou X, R. M. Witt, B. L. Sabatini, D. Adjeroh, and S. T. Wong, "Dendritic spine detection using curvilinear structure detector and lda classifier," *Neuroimage*, vol. 36, no. 2, pp. 346–360, 2007.

[15] C. M. Weaver, P. R. Hof, S. L. Wearne, and W. B. Lindquist, "Automated algorithms for multiscale morphometry of neuronal dendrites," *Neural Computation*, vol. 16, pp. 1353–1383, 2004.

[16] Y. Zhang, X. Zhou, A. Degterev, M. Lipinski, D. Adjeroh, and J. Yuan, "A novel tracing algorithm for high throughput imaging: Screening of neuron-based assays," *Journal of Neuroscience Methods*, vol. 160, pp. 149–162, 2007.

[17] J. Cheng, X. Zhou, E. Miller, R. M. Witt, J. Zhu, B. L. Sabatini, and S. T. C. Wong, "A novel computational approach for automatic dendrite spines detection in two-photon laser scan microscopy," *Journal of Neuroscience Methods*, vol. 165, pp. 122–134, 2007.

[18] J . Son, S. Song, S. Lee, S. Chang, and M. Kim, "Morphological change tracking of dendritic spines based on structural features," *Journal of Microscopy*, 2010.

[19] A. Rodriguez, D. Ehlenberger, D. Dickstein, P. Hof, and S. Wearne, "Automated three-dimensional detection and shape classification of dendritic spines from fluorescence microscopy images," *PLoS One*, vol. 3, no. 4, 2008.

[20] Q. Li, S. Sone, and K. Doi, "Selective enhancement filters for nodules, vessels, and airway walls in two-and three-dimensional ct scans," *Medical Physics*, vol. 30, no. 8, pp. 2040–2051, 2003.

[21] Y. Sato, C. Westin, A. Bhalerao, S. Nakajima, N. Shiraga, S. Tamura, and R. Kikinis, "Tissue classification based on 3d local intensity structure for volume rendering," *IEEE Trans. Vis. Comput. Graph.*, vol. 6, pp. 160–180, 2000.

[22] T. Lindeberg, "Feature detection with automatic scale selection," *International Journal of Computer Vision*, vol. 30, no. 2, pp. 77–116, 1998.

[23] D. G. Lowe, "Object recognition from local scale-invariant features," *Computer Vision. The Proceedings of the Seventh IEEE International Conference on*, vol. 2, pp. 1150–1157, 1999.

[24] E. Erdil, A. M. Yagci, A. Ö. Argunsah, Y. Ramiro-Cortés, A. F. Hobbiss, I. Israely, and D. Unay, "A tool for automatic dendritic spine detection and analysis. part i: Dendritic spine detection using multi-level region-based segmentation," in *Image Processing Theory, Tools and Applications (IPTA), 3rd International Conference on*.

[25] E. Erdil, A.M. Yagci, A.O. Argunsah, Y. Ramiro-Cortes, A.F. Hobbiss, I. Israely, and D. Unay, "A tool for automatic dendritic spine detection and analysis. part i: Dendritic spine detection using multi-level region-based segmentation," in *Image Processing Theory, Tools and Applications (IPTA), 2012 3rd International Conference on*, Oct 2012, pp. 167–171.

[26] C. Le Guyader and C. Gout, "Geodesic active contour under geometrical conditions theory and 3d applications," *Numerical Algorithms*, vol. 48, pp. 105–133, 2008.

[27] T. F. Chan and L. A. Vese, "Active contours without edges," *Image Processing, IEEE Transactions on*, vol. 10, no. 2, pp. 266–277, 2001.

[28] L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.