A Tool for Automatic Dendritic Spine Detection and Analysis. Part I: Dendritic Spine Detection Using Multi-Level Region-Based Segmentation

Ertunc Erdil[1], A. Murat Yagci[2], A. Özgür Argunsah[4], Yazmín Ramiro-Cortés[4], Anna F. Hobbiss[4], Inbal Israely[4] and Devrim Unay[3]
[1] Computer Engineering Department, Bahcesehir University, Besiktas, Istanbul, Turkey
e-mail: ertunc.erdil@bahcesehir.edu.tr
[2] Computer Engineering Department, Bogazici University, Bebek, Istanbul, Turkey
e-mail: murat.yagci@boun.edu.tr
[3] Biomedical Engineering Department, Bahcesehir University, Besiktas, Istanbul, Turkey
e-mail: devrim.unay@bahcesehir.edu.tr
[4] Champalimaud Neuroscience Programme, Champalimaud Centre for the Unknown, Lisbon, Portugal
e-mail: ali.argunsah@neuro.fchampalimaud.org, yazmin.cortes@neuro.fchampalimaud.org, anna.hobbiss@neuro.fchampalimaud.org, inbal.israely@neuro.fchampalimaud.org

***Abstract*— We propose an image processing pipeline for dendritic spine detection in two-photon fluorescence microscopy images. Spines of interest to neuroscientists often contain high intensity regions with respect to their surroundings. We find such maxima regions using morphological image reconstruction. These regions facilitate a multi-level segmentation algorithm to detect spines. First, watershed algorithm is applied to extract initial rough regions of spines. Then, these results are further refined using a graph-theoretic region-growing algorithm which incorporates segmentation on a sparse representation of image data and hierarchical clustering as a post-processing step. We compare our final results to segmentation results of the domain expert. Our pipeline produces promising segmentation results with practical run times for monitoring streaming data.**

***Keywords*— Neural image processing, fluorescence microscopy, dendritic spines, image segmentation, clustering**

I. Introduction

Recent advancements [1] in the field of imaging have allowed researchers to investigate how neuronal structure and function are correlated at the level of individual spines [2], [3], [4] . It has been shown that dendritic spine head volume is correlated with the amplitude of the uncaging excitatory postsynaptic currents (μEPSCs) evoked at the corresponding spine [2], [3], [1], [5], [6]. It has also been shown that activity-induced plasticity at multiple spines leads to bidirectional structural plasticity, growth and shrinkage of spines, that is spatially and temporally constrained. Spine volume changes are dynamic and varied following activity, and changes in the shape of the spine neck have also been implicated as important mediators of biological signals [7], [8] . Therefore, understanding how activity correlates with structural changes of spines may provide an important link to elucidating how information is stored in the brain, as well as to contribute to our understanding of several neurodevelopmental disorders, such as Fragile X and Rett syndrome [9], [10], which show abnormalities in both spine structure and function. The increased number of studies focused on neuronal structure leads to the collection of vast amount of data to be analyzed and hence the need for automatic image analysis tools. Also, current methods for analyzing structural changes on a spine by spine basis are innefficient, and restricted to changes in spine head volume. An ideal tool is expected to detect all the spines on a dendritic branch, calculate the relative positions of spines, estimate the volumes and shapes of each individual spine. Although there are proposed solutions for quantifying some of these parameters individually, such as detecting spines on a dendritic branch [11] or estimating the volumes of spines [12], there is no automatic standalone tool attacking all aspects of the problem.

Here we introduce an automatic multi-level region based segmentation method to detect dendritic spines from two-photon laser scanning microscopy images which is an initial step of our automatic dendritic spine detection and analysis framework.

This paper is outlined as follows; the following section describes our approach to the dendritic spine detection and segmentation problem. Then, we present and evaluate our experimental results. Finally, we discuss conclusions and future work.

II. Method

An overall look at our image processing pipeline is given in Fig. 1. The input to the pipeline is a 2-D maximum intensity projection [13] along z-axis, of a 3-D image stack obtained from two-photon fluorescence microscopy. A preprocessing step is necessary because of the artifacts introduced by the imaging modality. In this work, all our preprocessing is performed in 2-D. Apart from median filtering to eliminate speckle noise, using an edge preserving smoothing filter [14] is useful. Due to image acquisition, the image histogram sometimes indicates a low contrast, for which an adaptive

histogram equalization proves useful most of the time. Histogram equalized image is then binarized using Otsu's method to separate dendrite and its background.

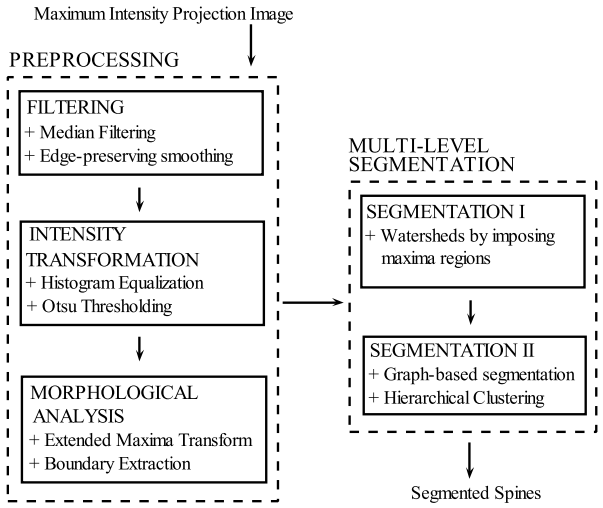


Fig. 1. Image processing pipeline for dendritic spine detection and segmentation

A. Morphological Analysis

Mathematical morphology provides powerful tools for filtering and extracting region, boundary, and texture information in binary and greyscale images with extensions to multichannel images. Most of its operations depend on set-theoretic primitives like erosion, ε, and dilation, δ, and the concept of a structuring element, S. An extensive treat of morphological image analysis can be found in [15]. We use morphology for detecting dendrite boundaries and also for finding maxima regions on the dendrite using greyscale image morphology. These methods provide good results and also in an efficient way.

Boundaries of objects in a binary image can be defined as difference of image itself from its eroded version. As given in (1), prior to boundary extraction, we slightly dilate the Otsu thresholding result, f_b, to make sure that the dendrite is fully included in the region of interest, where β represents the binary image containing boundaries of this region.

$$\begin{aligned} f_{b'} &= \delta_{S_1} f_b \\ \beta(f_{b'}) &= f_{b'} - \varepsilon_{S_2}(f_{b'}) \end{aligned} \tag{1}$$

An important branch of morphology is geodesic transformations which lead to powerful morphological reconstruction algorithms when iterated until stability. In geodesic transformations, a so called mask image controls evolution of a marker image by geodesic transformations. We apply these ideas to grey scale dendrite images with the assumption that heads of dendritic spines at full view, should include maxima regions in such images. Extraction of these regions may not be possible by simple thresholding since as also evident in Fig. 2(d), their intensities differ with respect to their fluorescence levels after absorbing the fluorescent substance. Instead, we apply h-maxima transformation, which reconstructs an image function, f, from $f - h$ where h is a contrast criterion and the transformation suppresses all maxima whose depth is lower than or equal to h. It uses reconstruction by dilation formulated as

$$HMAX_h(f) = R_f^{\delta}(f - h) \tag{2}$$

The *extended maxima* are defined as the regional maxima of the corresponding h-maxima transformation [15]. Fig. 2 shows effect of h-maxima transform on a 1-D signal and also on a portion of dendrite image intensity plot.

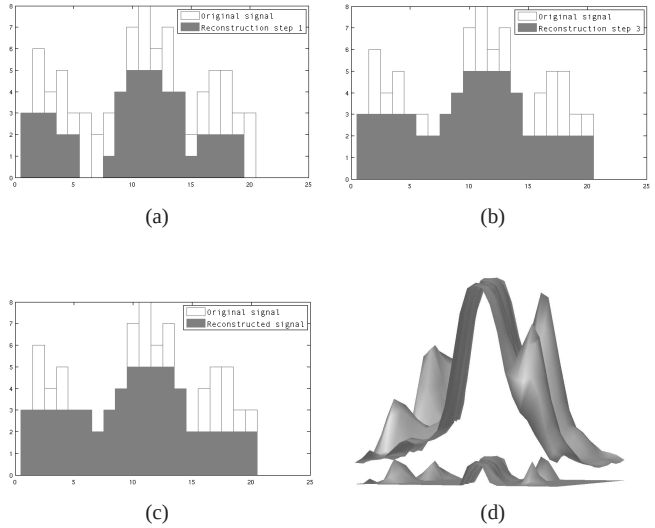


Fig. 2. *Figures (a)-(c) show first, third and final iterations of h-Maxima transformation simulated on a 1-D signal and with $h = 3$. Figure (d) shows effect of the transformation on a portion of dendrite image with the transformed signal locating 4 spines and the dendritic arbor as given in lower part. Although actual intensities of maxima are very different, the transformation is according to their contrast with the surroundings.*

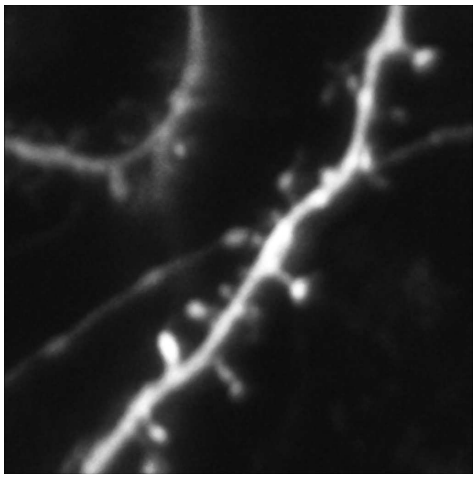
The result of all morphological operations applied to a dendrite image yields a feature image as given in Fig. 3.

B. Multi-level segmentation

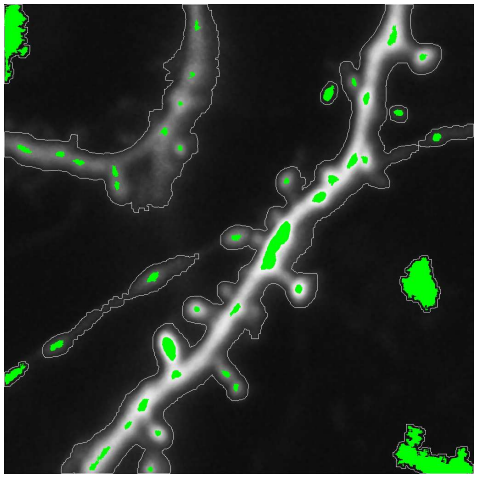
We apply a multi-level segmentation approach to detect and segment spines using the above-mentioned feature image. We describe the levels of segmentation in subsections Segmentation I and II respectively.

1) Segmentation I: Finding maxima regions in the image provides a perfect basis to run watersheds algorithm [16] as an initial segmentation process. We invert the image and then impose all the detected maxima as well as the regions outside the detected dendrite boundaries as minima. These become the deep regions of all possible basins in the image so that watersheds algorithm can start filling them. Eventually, dams are constructed at the object boundaries.

Without restriction, watersheds algorithm usually finds larger boundaries than the expert results. Therefore, a second



(a) Preprocessed image



(b) Feature image

Fig. 3. *Feature image showing result of morphological reconstruction and boundary extraction (b) on a preprocessed image (a). Note that green blobs indicate maxima regions for $h = 12$ and white boundaries mark the image foreground containing dendritic structures.*

level of segmentation is necessary to further refine results of watersheds segmentation. Result of Segmentation I is given in Fig. 4.

2) Segmentation II: This level of segmentation takes each previously found region of interest, possibly containing a spine, and refines it using a modified version of a graph-theoretic algorithm for arbitrary shape detection [17] together with hierarchical clustering to improve segmentation results.

Each region of interest (ROI) in Fig. 4 can be represented as a graph using κ-neighbor sparsification where similarity between two pixels is defined as;

$$s(p_i, p_j) = e^{-\frac{(f(p_i)-f(p_j))^2}{\sigma^2_{ROI}}} \tag{3}$$

This definition produces a good transient similarity function, although we do not weigh this measure with distance as described in [18]. Furthermore, we use $\kappa = 8$ in our tests

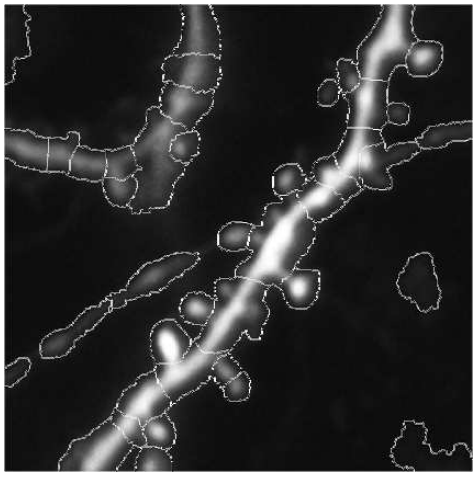


Fig. 4. Result of Segmentation I

and σ^2_{ROI} refers to the variance of ROI intensity levels.

An undirected graph is constructed such that its vertices represent pixels in the ROI and edges represent above-mentioned similarities between pixels. The original algorithm defines a property called $attachment(v_i) = \frac{sumOfWeights(v_i)}{degreeOfFreedom(v_i)}$, where $sumOfWeights(v_i)$ is that of all edges linking to a vertex v_i, and $degreeOfFreedom(v_i)$ is the number of vertices adjacent to v_i. An unlabeled vertex with the highest attachment is considered a good starting point (seed) for region growing, since it may be the center of a homogeneous region. Starting with the seed vertex, the algorithm finds regions automatically in a breadth-first search fashion. A vertex is included in the region if it has no stronger connection to another vertex than its neighboring vertex in the region. Furthermore, if a region cannot be enlarged, then the next unlabeled vertex with the highest attachment is selected as seed to start a new region. The segmentation process terminates when all pixels are labeled.

As is, this graph-theoretic algorithm creates oversegmentation, which can be solved up to some level, by using a relaxation criterion defined in the original paper and formulized in (4), where ρ represents a relaxation factor between 0 and 1. If this criterion is satisfied, then the pixel, p_j, is included in the region of p_i. Since the relaxation factor is very sensitive and proved hard to tune in our experiments, we chose $\rho = 0$ and instead used hierarchical clustering [19] to merge oversegmented regions by defining an inter-cluster similarity measure and cutting the final dendrogram to obtain more meaningful segmentations. We define the cluster similarity simply as absolute differences of cluster means. The resulting dendrogram is cut into a number of hard clusters which define segmented regions.

$$s(p_i, p_j) + s(p_i, p_j).\rho \geq s(p_m, p_j), \forall p_m \in kneighborhood(p_j) \tag{4}$$

Fig. 5(a) shows result of graph-theoretic segmentation on

a certain spine region. Then, in the following subfigures, hierarchical clustering with different initializations are shown as a post-processing step. Hierarchical clustering facilitates elimination of oversegmentation by forming quasi-concentric connected components. The intuitive idea of providing $k = 2$ and separating the region as foreground and background does not work well and creates undersegmentation most of the time. Instead, choosing larger k values gives the algorithm the ability to slowly shrink the region into a more refined segmentation. We decided to choose empirically $k = 20$ in all our experiments.

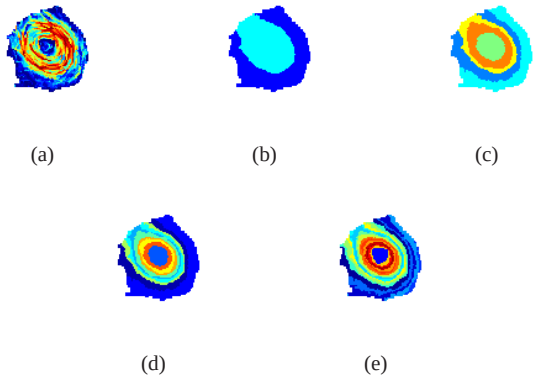


Fig. 5. *Spine region segmented using graph-theoretic clustering (a), post-processed using hierarchical clustering with $k = 2$ (b), $k = 5$ (c), $k = 10$ (d), $k = 20$ (e). Note that each color represents a different cluster.*

Finally, Fig. 6 shows the result of Segmentation II, hence the whole pipeline. Evaluation of these results are provided in the next section.

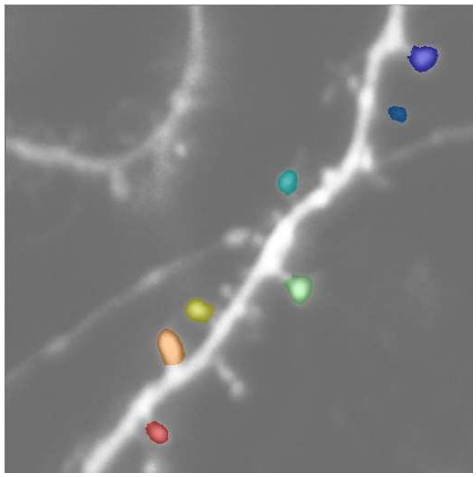


Fig. 6. Final result of multi-level segmentation which shows successful automatic detection and segmentation of all spines of interest to the domain expert.

III. Experimental results

In this section, we provide accuracy and run-time results of the proposed method.

A. Data set and experimental setup

Mouse hippocampal organotypic slice cultures were prepared from P7 to P10 animals [20], and AFP or Dendra [21] was sparsely introduced via biolistic gene transfection. Two-photon laser scanning microscopy images of CA1 pyramidal neurons collected using Ultima Multiphoton Microscopy System (Prairie Technologies). Each image in the series is a 3-D ($1024 \times 1024 \times z$) 16-bit intensity dendrite image stack with 30-50 z-slices. We took one random image from each of three data sets and created 2-D maximum intensity projections in z direction. These images are 8-bit greyscale. Domain experts provided us as well with the ground truth segmentations of spines at full view which are of interest to them. Each image contains 7 to 8 such spines. Our method also segments many of the spines which are occluded due to rotations in 3-D. However, we simply ignored such spines, because they are not of interest to neuroscientists.

Our experimental setting is a 2.0 Ghz dual-core PC with sufficient main memory. We do not use any explicit software-based parallelism. Our implementations are Matlab-based.

All of our results in this section are obtained with the same test settings, i.e., 7×7 median filtering, $h = 12$ for extended maxima transform and $k = 20$ for hierarchical clustering. Furthermore, these input parameters can be fine tuned by the neuroscientists for their own image characteristics.

B. Evaluation of results

We evaluate the accuracy of our segmentation results with respect to ground truth provided by the domain expert. We use Dice coefficient for the evaluation. This coefficient measures similarity between two sets X and Y, which are in our case the binary images representing the expert' s segmentation decision and the algorithm' s decision. Dice coefficient takes values between 0-1 and it is formulated as follows;

$$\mathcal{D}(X, Y) = \frac{2|X \cap Y|}{|X| + |Y|} \tag{5}$$

The accuracy results are tabulated in Tables 1, 2 and 3. In each experiment, all spines of interest were automatically detected and segmented. The tables contain evaluation of each spine segmentation as well as their average. Improvement achieved by multi-level segmentation approach can be seen by comparing Segmentation II results to Segmentation I results. These results are promising for initial experiments.

A few spines in images CA1-196 and CA1-223 match with only 60-70% of the expert manual segmentations. This is due to the current accuracy of our method for automatically deciding which of concentric connected components formed by hierarchical clustering should belong to foreground. Improving this automatic decision will be one of our future works.

We also provide run time results in Table 4 which indicate practical run times for streaming dendrite image processing in many situations. Note that currently, our method also finds,

Table 1. Accuracy results on image CA1-196

	$\mathcal{D}$ after	
	Segmentation II	**Segmentation I**
Spine 1	**0.930**	0.816
Spine 2	**0.890**	0.852
Spine 3	**0.881**	0.818
Spine 4	**0.845**	0.749
Spine 5	**0.931**	0.876
Spine 6	**0.697**	0.686
Spine 7	**0.605**	0.582
Spine 8	**0.896**	0.832
AVERAGE	**0.832**	0.776

Table 2. Accuracy results on image CA1-223

	$\mathcal{D}$ after	
	Segmentation II	**Segmentation I**
Spine 1	**0.666**	0.516
Spine 2	**0.733**	0.557
Spine 3	**0.813**	0.773
Spine 4	**0.829**	0.644
Spine 5	**0.878**	0.759
Spine 6	**0.920**	0.597
Spine 7	0.840	**0.862**
AVERAGE	**0.811**	0.672

Table 3. Accuracy results on image CA1-665

	$\mathcal{D}$ after	
	Segmentation II	**Segmentation I**
Spine 1	**0.877**	0.673
Spine 2	**0.888**	0.668
Spine 3	**0.835**	0.570
Spine 4	**0.908**	0.652
Spine 5	**0.924**	0.737
Spine 6	**0.941**	0.682
Spine 7	**0.918**	0.595
AVERAGE	**0.899**	0.654

besides spines of interest, some other components which are occluded spines or spine-like protrusions which affect total run times. It should also be noted that, although we did not exploit it in this work, our method is highly parallelizable.

Table 4. Run time results

Image	# of Components	Time (sec)
CA1-196	33	25.57
CA1-223	38	38.33
CA1-665	11	16.01

IV. Conclusions and future work

We presented an image processing pipeline for detection and segmentation of dendritic spines. The proposed pipeline yields good results in terms of accuracy and run times.

Although 3-D processing presumably creates more accurate results, processing in 2-D is still desirable, because it offers faster operations in commodity computers and still yields meaningful results unless a spine is occluded due to 3-D rotation. It should also be noted that the resolution in z-axis is usually low. Nevertheless, we will extend our method to 3-D.

To improve the accuracy of our results, we will introduce more domain knowledge to our algorihms. For example, we plan to estimate which clusters created by hierarchical clustering belong to image foreground by devising an intelligent learning algorithm which learns from previous expert results.

After further elaboration of our methods and more extensive experiments, our aim is to combine the methods in a standalone software tool for automated volume calculations of image sequences.

References

[1] Matsuzaki M, Honkura N, Ellis-Davies GC, Kasai H. Structural basis of long-term potentiation in single dendritic spines *Nature*, 17;429(6993):761-6., 2004.

[2] Harvey C.D., Svoboda K. Locally dynamic synaptic learning rules in pyramidal neuron dendrites *Nature*, 20;450(7173):1195-200, 2007.

[3] Govindarajan A., Israely I., Huang, S. Y., and Tonegawa S. The dendritic branch is the preferred integrative unit for protein synthesis-dependent LTP *Neuron*, Vol. 69-1, 2011.

[4] Tanaka J. et al., Protein Synthesis and Neurotrophin-Dependent Structural Plasticity of Single Dendritic Spines *Science*, Vol. 319, p. 1683-1687, 2008.

[5] Asrican, B., Lisman, J., and Otmakhov, N. Synaptic strength of individual spines correlates with bound Ca2+-calmodulin-dependent kinase II *Journal of Neuroscience*, Vol. 27, 14007-14011, 2007.

[6] Smith, M.A., Ellis-Davies, G.C., and Magee, J.C. Mechanism of the distance-dependent scaling of Schaffer collateral synapses in rat CA1 pyra- midal neurons *J. Physiol.*, 548, 245-258, 2003.

[7] Noguchi J, Matsuzaki M, Ellis-Davies GC, Kasai H. Spine-neck geometry determines NMDA receptor-dependent Ca2+ signaling in dendrites *J. Physiol.*, 19;46(4):609-22, 2005.

[8] Bloodgood BL, Sabatini BL. Neuronal activity regulates diffusion across the neck of dendritic spines *Science*, 310(5749):866-9, 2005.

[9] Bear M.F., Huber K.M., and Warren S.T. The mGluR theory of fragile X mental retardation *Trends in Neurosciences*, Vol. 27, Issue 7, p. 370-377, 2004.

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

[11] Bai, W., Zhou, X., Ji, L., Cheng, J., and Wong, S. T. C. Automatic dendritic spine analysis in two-photon laser scanning microscopy images *Cytometry Part A*, Vol. 71, p 818-826, 2007.

[12] Y. Zhang, K. Chen, M. Baron, M. A. Teylan, Y. Kim, Z. Song, P. Greengard, and S. T. Wong, A neurocomputational method for fully automated 3d dendritic spine detection and segmentation of medium-sized spiny neurons *NeuroImage*, vol. 50, no. 4, pp. 1472-1484, 2010.

[13] Wallis, J.W., Miller, T.R.. Three-dimensional display in nuclear medicine and radiology *J. Nucl. Med.*, 32(3):534-46, 1991.

[14] Ibanez, L., Schroeder, W. *The ITK Software Guide* 2.4e, Kitware Inc., 2005.

[15] Soille, P. *Morphological Image Analysis: Principles and Applications* 2e, Springer-Verlag, NY, 2003.

[16] Meyer, F. Topographic distance and watershed line *Signal Processing*, Vol. 38, pp. 113-125, 1994.

[17] Mimaroglu S., Erdil E. ASOD: Arbitrary shape object detection *Engineering Applications of Artificial Intelligence*, Elsevier, Vol. 84, p 1295-1299, 2011.

[18] Shi J. and Malik J. Normalized Cuts and Image Segmentation *Pattern Analysis and Machine Intelligence*, IEEE Transactions, Vol. 22, p 888 905, 2000.

[19] Tan, P. and Steinbach, M. and Kumar, V. *Introduction to Data Mining* Addison Wesley, Boston, 2005.

[20] Stoppini, L., Buchs, P.A., and Muller, D. A simple method for organotypic cultures of nervous tissue *J. Neurosci. Methods*, 37, 173-182, 1991.

[21] Gurskaya, N.G., Verkhusha, V.V., Shcheglov, A.S., Staroverov, D.B., Chepurnykh, T.V., Fradkov, A.F., Lukyanov, S., and Lukyanov, K.A. Engineering of a monomeric green-to-red photoactivatable fluorescent protein induced by blue light *Nat. Biotechnol.*, 24, 461-465, 2006.