

A Comparative Study of Feature Descriptors for Mitochondria and Synapse Segmentation

Kendrick Cetina, Pablo Márquez-Neila and Luis Baumela
Departamento de Inteligencia Artificial
Universidad Politécnica de Madrid
Boadilla del Monte, Spain

Abstract—Full understanding of the architecture of the brain is a long term goal of neuroscience. To achieve it, advanced image processing tools are required, that automate the the analysis and reconstruction of brain structures. Synapses and mitochondria are two prominent structures with neurological interest for which various automated image segmentation approaches have been recently proposed. In this work we present a comparative study of several image feature descriptors used for the segmentation of synapses and mitochondria in stacks of electron microscopy images.

I. INTRODUCTION

Deciphering the architecture of the brain is a key challenge of science [6]. In the last years we have seen advances in the automated acquisition of large series of images of brain tissue [7], [11]. The analysis of these images enable the construction of detailed maps of neuron structures from which we will better understand the basic cognitive functions of the brain, such as learning, memory and its pathologies [9]. There are tools to manually analyze and segment the structures in such images [22]. However, the complexity of these images and the high number of neurons in a small section of the brain, makes the automated analysis the only practical solution.

Mitochondria and synapses are two cell structures of neurological interest that are suitable for automated processing. Synapses are the fundamental mechanism of communication between neurons. Quantification of synapses, and the identification of its types and their distribution is critical to understand how the brain works [2]. Besides providing energy to the cell, mitochondria play an important role in many essential cellular functions including signaling, differentiation, growth and death. The morphology and distribution of Mitochondria has great importance in cellular physiology [3] and synaptic function [14]. Also atypical morphologies or mitochondria distributions are indicative of abnormal cellular states or the existence of neurodegenerative diseases [4].

Recent works have proposed algorithms for synapse [12] and mitochondria segmentation [15], [8] employing various discriminating features. To extract these features some approaches use general texture operators [12], where as others employ specifically designed measurements [15]. In this paper we will study and compare the features used in these works for the problem of joint segmentation of synapses and mitochondria. In Figure 1 we show a slice of one of the images used in our study and its associated labels.

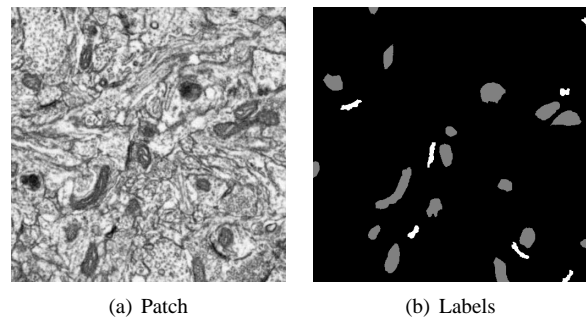


Fig. 1. A patch from the stack (a) and its labels (b), class mitochondria in gray, class synapses in white.

II. PREVIOUS WORK

A. Synapse segmentation

In [1], an Ada-Boost classifier is used to find context cues from a set of standard texture descriptors such as image gradients, Laplacians of Gaussians and eigenvalues of structure tensors. In [18], texture and shape-based features are extracted from small image patches around potential objects of interest. The approach used shows high accuracy and outperforms the features based on the correlation of pixels in [21] and the gray levels and connectivity features with automatic techniques designed to detect synapses in [16].

B. Mitochondria segmentation

In [25], a Gentle-Boost classifier is trained with histograms of gray scale and Gabor filter responses computed in neighborhood windows around each pixel. In [10], a Random Forest classifier was trained with the Haar-like descriptors to detect membranes in individual images of the stack. Graph cut algorithms were used to regularize the results and satisfy the 3D continuity constraints. In [17], mitochondria of melanoma cells were segmented with texton-based features obtained by convolving training images with a filter-bank to generate filter responses, a variety of classifiers were used among which are k-NN, SVM and Ada-Boost. Finally [23] introduces the Ray features and trains an Ada-Boost classifier outperforming the Haar-like features and Histograms of Oriented Gradients when applied to detecting irregularly shaped neuron nuclei and mitochondria.

III. FEATURES

In this section we briefly describe the feature descriptors considered in this study. We begin describing the simple general purpose descriptors and proceed in order of increasing sophistication.

A. Simple Window and Histogram

We construct a simple window based descriptor ordering and storing a vector of the $n \times n$ neighborhood of the pixel that we want to describe. This naive descriptor has proved to be an excellent source of information for texture segmentation [24].

A histogram based descriptor takes for each pixel an $n \times n$ neighborhood on which a gray level histogram is computed. In [15], a histogram and the *Ray* features [23] are used as elements of the feature vector for mitochondria segmentation. In this paper we tested the histogram and the *ray* features separately.

A scheme of these two characteristics is shown in Figure 2.

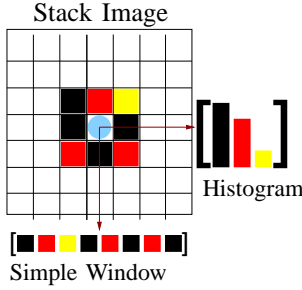


Fig. 2. Simple window and histogram descriptors.

B. Local Binary Patterns

The local binary patterns (LBP) [19], generate a binary code with k digits taking into account for each pixel p a set of k neighbor points at an r distance, where r is the radius from the central pixel p to its neighbors. If the value of p is higher than a neighbor k_i then we insert a 0 in the binary code, or 1 if the value is lower. The feature vector is obtained from the histogram of the LBP binary codes converted to its real values in a $n \times n$ neighborhood. This process is outlined in Figure 3.

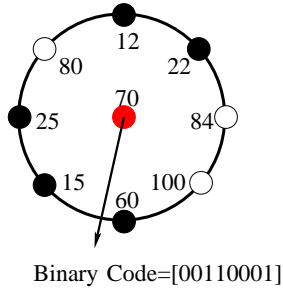


Fig. 3. Depending on the values of the neighboring pixels, the LBP generates a binary code that can be converted to a real value. In this case the value would be: $0 \times 1 + 0 \times 2 + 1 \times 4 + 1 \times 8 + 0 \times 16 + 0 \times 32 + 0 \times 64 + 1 \times 128 = 148$.

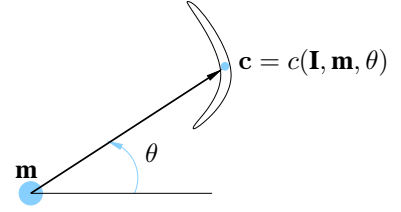


Fig. 4. Function \mathbf{c} returns the position \mathbf{c} from the nearest edge or contour of the image \mathbf{I} to the position \mathbf{m} in direction defined by the angle θ .

C. GRIMS

The GRIMS (Gaussian Rotation Invariant and Multi Scale) descriptors apply to each image in the stack a set of linear Gaussian filters at different scales to compute zero, first and second order derivatives. These linear operators are:

$$\left\{ G_\sigma * \sigma \cdot G_\sigma * \frac{\partial}{\partial x}, \sigma \cdot G_\sigma * \frac{\partial}{\partial y}, \sigma^2 \cdot G_\sigma * \frac{\partial^2}{\partial x^2}, \sigma^2 \cdot G_\sigma * \frac{\partial^2}{\partial xy}, \sigma^2 \cdot G_\sigma * \frac{\partial^2}{\partial y^2} \right\}, \quad (1)$$

where G_σ is a Gaussian filter with standard deviation σ and $*$ is the convolution operator. We will call the result of applying these operators to the image: s_{00} , s_{10} , s_{01} , s_{20} , s_{11} and s_{02} , where the subscript denotes the order of the derivatives.

The feature vector calculated for each pixel in the image at scale σ is $\{s_{00}, \sqrt{s_{10}^2 + s_{01}^2}, \lambda_1, \lambda_2\}$, where $\sqrt{s_{10}^2 + s_{01}^2}$ is the gradient magnitude and λ_1 y λ_2 are the eigenvalues of the Hessian matrix calculated as follows:

$$\lambda_1 = \frac{1}{2} \left(s_{20} + s_{02} + \sqrt{(s_{20} - s_{02})^2 + 4s_{11}^2} \right) \quad (2)$$

$$\lambda_2 = \frac{1}{2} \left(s_{20} + s_{02} - \sqrt{(s_{20} - s_{02})^2 + 4s_{11}^2} \right) \quad (3)$$

This procedure is repeated with various scales $\sigma_0, \dots, \sigma_{n-1}$, and since there are 4 features for each scale, we obtain a feature vector of size $4n$. In our experiments we use $n = 4$ scales, therefore, we obtain a feature vector with 16 dimensions.

D. Ray Descriptors

Ray features [23] are used in [15] for mitochondria segmentation, they are good at extracting shape features in objects with strong edges, such as mitochondria. The features of the Ray descriptors depend on the function:

$$\mathbf{c} = \mathbf{c}(\mathbf{I}, \mathbf{m}, \theta), \quad (4)$$

that computes the position \mathbf{c} from the nearest edge or contour of the image \mathbf{I} to the position \mathbf{m} in direction defined by the angle θ . See Figure 4.

The feature vector is:

$$\mathbf{f}^{\text{Ray}}(\mathbf{I}, \mathbf{m}, \theta) = [f_{\text{ndist}}, f_{\text{norm}}, f_{\text{ori}}], \quad (5)$$

where

$$f_{\text{ndist}}(\mathbf{I}) = \|c(\mathbf{I}, \mathbf{m}, \theta) - \mathbf{m}\| \quad (6)$$

$$f_{\text{norm}}(\mathbf{I}) = \|\nabla \mathbf{I}(c(\mathbf{I}, \mathbf{m}, \theta))\| \quad (7)$$

$$f_{\text{ori}}(\mathbf{I}) = \frac{\nabla \mathbf{I}(c(\mathbf{I}, \mathbf{m}, \theta))}{\|\nabla \mathbf{I}(c(\mathbf{I}, \mathbf{m}, \theta))\|} \cdot (\cos \theta, \sin \theta)^\top \quad (8)$$

- f_{ndist} , is the Euclidean distance to the point \mathbf{c} from the point \mathbf{m} , where \mathbf{c} is the nearest edge.
- f_{norm} , returns the gradient norm in point \mathbf{c} .
- f_{ori} , take into account the orientation of the point in the nearest edge in direction θ . When a point \mathbf{m} close to the center of a closed blob is evaluated, f_{ori} usually returns values close to 1.

The final step is aligning the features to a canonical orientation to make the feature vector invariant to rotations. This is solved by sorting the vector starting with the feature that is aligned with the direction of maximum variance.

E. Difference of Gaussian's

The difference of Gaussians can be utilized to increase the visibility of edges in an image and also for blob detection. This feature subtracts one blurred image from another blurred with a different scale. A Gaussian kernel suppresses only high frequency spatial information and by subtracting one image from the other, it discards all but a handful of spatial frequencies that are present in the original gray-scale image. In our approach we used four subtractions of blurred images at different scales.

F. Laplacian Of Smoothed Image

Due to the elliptical shape of the structures we want to segment, we tested a blob detector based on the Laplacian of the Gaussian smoothed image. This is one of the first and also most common blob detectors which usually results in strong positive responses for dark blobs and strong negative responses for bright blobs of similar size. We used different scales for this approach because the output is strongly dependent on the relationship between the size of the blob structures in the image domain and the size of the Gaussian kernel used for pre-smoothing. Therefore to identify different unknown sizes of structures we used four scales which we describe later on.

G. Eigenvalues of the structure tensor

The structure tensor is a matrix derived from the gradient of a function. It represents the predominant directions of the gradient in a specified neighborhood of a point. It is often used in image processing due to the properties of its eigenvalues λ_1, λ_2 and eigenvectors e_1, e_2 : If $\lambda_1 = \lambda_2$, the gradient in the patch has no predominant direction which indicates that the patch has rotational symmetry and if $\lambda_1 > \lambda_2$, then $\pm e_1$ is the direction that is maximally aligned with the gradient within the patch. For a image \mathbf{I} the 2D structure tensor is given by

$$\begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix},$$

where x and y are the variables in the image and I_x and I_y are the partial derivatives of I with respect to x and y respectively.

H. Histogram of Oriented Gradients

The histogram of oriented gradients descriptors were first described in [5] for pedestrian detection in static images, they after were tested for human detection in film and video, as well as to a variety of common animals and vehicles in static imagery. The feature vector for the HOG implementation is obtained by dividing the image into small connected regions, called cells, and for each cell compiling a histogram of gradient directions for the pixels within the cell. To build our HOG feature vector we used an 18 bins histogram representing the signed directions of the gradient, a 9 bins histogram representing the unsigned directions, 4 features representing the sum of the gradients in each direction and the gray value of the pixel. This makes a vector with 32 features per cell.

I. Radon-Like features

It has been shown empirically that trying to segment the structures in connectome images using only geometric or textural features is not very effective. The Radon-Like features were proposed as a remedy to this problem as they are designed to leverage both the texture and the geometric information present in the connectome images to segment structures of interest. As a first step, Radon-Like features use the edge map of a connectome image as a means to divide it into regions that are defined by the geometry of the constituent structures. Next, for each pixel, line segments with their end points on the closest edges are computed in all directions. Finally, for each pixel, a scalar value is computed along each direction using the information in the original image along these line segments using a so-called extraction function defined in [13].

IV. EXPERIMENTS

We used an image stack obtained from the somatosensory cortex of a rat, with a resolution of $3.686\mu m$ per pixel. The thickness of each layer is $20\mu m$. We used 60 images of the stack for training and 10 for testing. We ran our experiments on a *16GB RAM* computer with 4 cores at *3.2GHz*.

For the descriptors that use a Gaussian kernel, we experimentally selected the scales $\sigma_0 = 4, \sigma_1 = 5.65, \sigma_2 = 8, \sigma_3 = 11.31, \sigma_4 = 16$ for our tests. In our HOG implementation we used 4×4 pixels per cell and 6×6 cells per block. Due to the block division required, we assigned the computed histogram to the central pixel in each cell.

For the histogram and simple window descriptors, we tested with several window and bin sizes. A 20×20 pixel window with 10 bins for the histogram and a 15×15 box for the simple window had the best performance.

For the LBP we used a *radius* of 10 pixels with 25 sample points from which we obtain the LBP code for each pixel. With such codification we obtain a new image stack on which we compute a 10 bins histogram on a 20×20 pixels window, from which we build our feature vector.

For classification purposes we used two algorithms. A Gaussian and a Random Forest Classifier. The Gaussian classifier is a generative parametric classifier that assumes Gaussian class

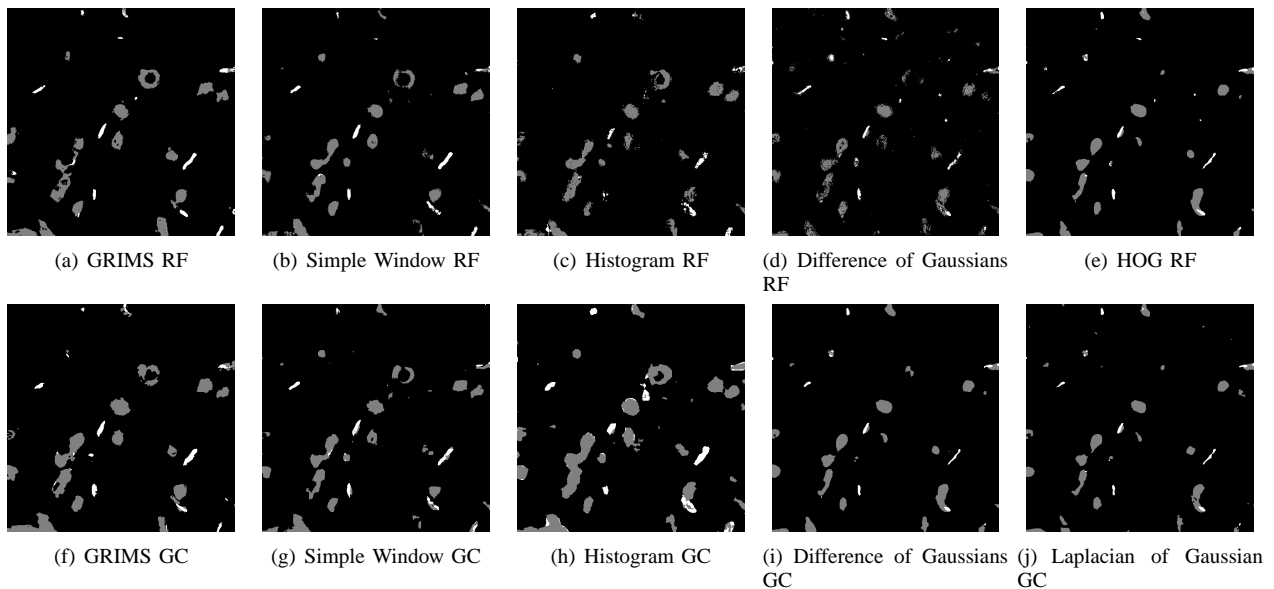


Fig. 5. Segmentation results. In the first row the best results with Random Forest Classifier and in the second row the best results with Gaussian Classifier

conditional distributions. We chose this classifier because of its simplicity and speed. Random forests operate by training a multitude of decision trees and selecting the class label that is the mode of the classes output by individual trees. We used the *scikit-learn* [20] implementation of the Random Forest classifier with 100 decision trees.

The high dimensionality of the feature vectors produced by the HOG and Radon-Like feature descriptors, given the number of training samples, produce degenerate Gaussian distributions with singular covariance matrices. For this reason in our experiments we do not use a Gaussian classifier with HOG and Radon-Like features. Similarly, we do not show the results of the Random Forest classifier with LBP features, given its poor performance.

In our experiments we use the ROC curve of each class against the rest and the Jaccard Index as comparison indices. We have performed an extensive set of experiments involving different feature configuration. We compared our work with the Radon-Like features [13] and the work in [15] but testing the descriptors of their feature vector individually, i.e., the histogram and Ray descriptors were tested separately. For brevity, in Figures 6 and 7 we only show the best segmentation results for the ROC and Jaccard Index respectively. In these figures **GC** stands for *Gaussian Classifier* and **RF** stands for *Random Forest Classifier*. In Figure 5 we show sample results of the best segmentation for some of the features studied.

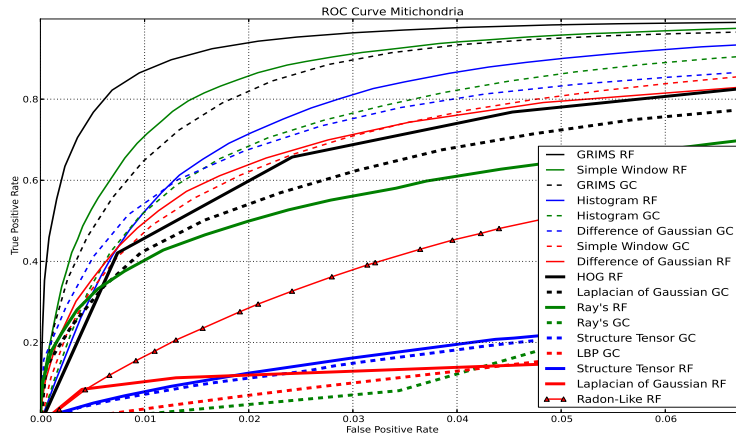
The results of our experiments show that the Random Forest classifier achieves the best performance for both mitochondria and synapse segmentation. However, as shown in Table I, the time it takes to train and classify is roughly one order of magnitude larger than the Gaussian classifier. On the other hand, the Gaussian Classifier is significantly faster than the Random Forest at expense of a marginal loss in performance. We can see this performance loss in Figure 6, where, for

TABLE I
LEARNING AND PREDICTION TIMES

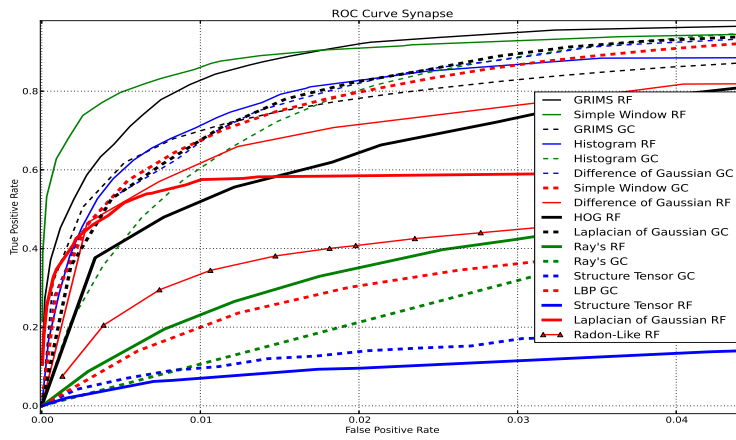
Random Forest Classifier		
Descriptor	Learning	Prediction
LBP	27min	36s
Simple Window	126.1min	29.4s
Histogram	12.3min	13.29s
GRIMS	35.2min	15.54s
Ray	53.1min	19s
HOG	54.4min	21s
Laplacian of Gaussian	18.4min	11.3s
Difference of Gaussian	18.6min	11.9s
Gaussian Classifier		
Descriptor	Learning	Prediction
LBP	1.4s	1.3s
Simple Window	3s	1.2s
Histogram	3s	0.5s
GRIMS	1.2s	1s
Ray	2s	0.4s
Eigenvalues of Structure Tensor	2.1s	0.2s
Laplacian of Gaussian	2.3s	1s
Difference of Gaussian	2.3s	1.1s

example, the synapse ROC Curve for the GRIMS descriptors with Random Forest is the second best curve followed by that for the with Gaussian Classifier. Similarly, in the mitochondria plot, the ROC curves of any feature set always achieve worse performance for the Gaussian classifier than for the Random Forest.

The GRIMS descriptors provide the best performance for mitochondria segmentation, closely followed by the simple window. On the other hand, for the segmentation of synapses,



(a) Mitochondria



(b) Synapse

Fig. 6. ROC Curves. The top figure shows the class Mitochondria and the bottom figure shows the class Synapse. Dotted lines are the experiments done with a Gaussian classifier (GC) and continuous lines with a random forest classifier (RF).

the simple window descriptor provides the best performance, immediately ahead of the GRIMS. The Jaccard index and the ROC curve provide similar evaluation results.

The random forest classifier provides an importance measure for each feature, in our experiments the most discriminative feature is the gray level and we can see that the descriptors that use the gray level and its variations are the best descriptors in this work.

Although Ray descriptors [23] were specifically designed to describe structures with strong edges, such as mitochondria, they achieve their best performance when used in combination with gray value histograms [15]. Similarly, LBP's have been extensively used for describing textures, one of their features being their robustness against illumination changes. However, as our experiments suggest, they do not performed well in EM images possibly because they have a very strong background noise.

We would also like to remark the good results achieved

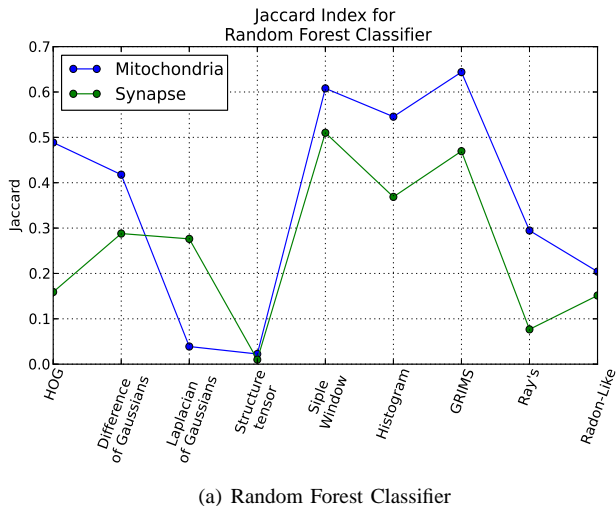
by the simple window descriptor. There are various reasons that justify this result. First, it is well known that texture classification using the joint distribution of intensity values over extremely compact neighborhoods outperforms classification using sophisticated filter banks with large support [24]. Second, both mitochondria and synapses are dark regions on a clear background. Hence, they can be well represented with such a feature. Finally, this feature works slightly better for synapse than for mitochondria segmentation. The reason for this is that in our images an average synapse is around 10 pixels thick. So, in most cases it can be fully enclosed within a 15×15 window. On the other hand, mitochondria are structures much larger. Very seldom they fit entirely within that window. Hence they are more difficult to represent with a compact neighborhood. However, they have a well defined texture that can be described invariantly to rotations with GRIMS.

VI. ACKNOWLEDGEMENTS

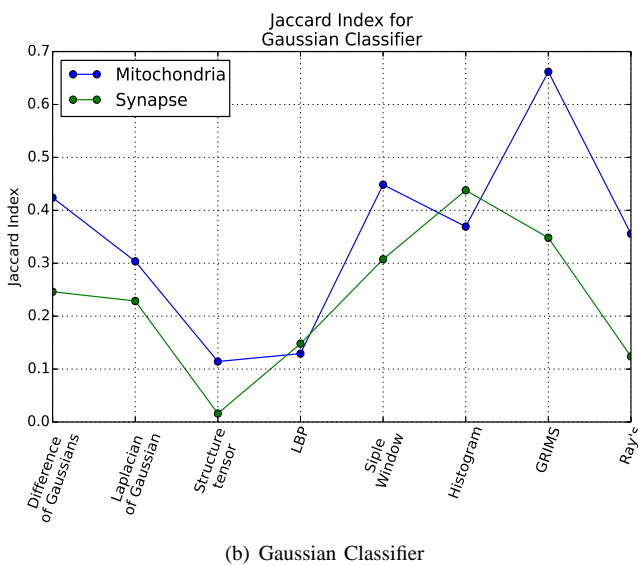
This research was funded in part by the Cajal Blue Brain Project. Kendrick Cetina was funded by CONACYT México. The authors are grateful to Ángel Merchán and the Laboratorio de Circuitos Corticales UPM-CSIC for providing the images used in the paper.

REFERENCES

- [1] C. Becker, K. Ali, G. Knott, and P. Fua, "Learning context cues for synapse segmentation," *Medical Imaging, IEEE Transactions on*, vol. 31, no. 2, pp. 474–486, 2012.
- [2] L. Blazquez-Llorca, A. Merchán-Pérez, J. R. Rodríguez, J. Gascón, and J. DeFelipe, "FIB/SEM technology and alzheimer's disease: Three-dimensional analysis of human cortical synapses," *Journal of Alzheimer's Disease*, vol. 34, no. 4, pp. 995–1013, 2013.
- [3] S. Campello and L. Scorrano, "Mitochondrial shape changes: orchestrating cell pathophysiology," *EMBO Reports*, vol. 11, no. 9, pp. 678–684, 2010.
- [4] D.-H. Cho, T. Nakamura, and S. Lipton, "Mitochondrial dynamics in cell death and neurodegeneration," *Cellular and Molecular Life Sciences*, vol. 67, no. 20, pp. 3435–3447, 2010.
- [5] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. Int. Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 886–893.
- [6] J. DeFelipe, "From the connectome to the synaptome: An epic love story," *Science*, vol. 330, no. 6008, pp. 1198–1201, 2010.
- [7] W. Denk and H. Horstmann, "Serial block-face scanning electron microscopy to reconstruct three-dimensional tissue nanostructure," *PLoS Biol*, vol. 2, no. 11, p. e329, 10 2004.
- [8] R. Giuly, M. Martone, and M. Ellisman, "Method: automatic segmentation of mitochondria utilizing patch classification, contour pair classification, and automatically seeded level sets," *BMC Bioinformatics*, vol. 13, no. 1, p. 29, 2012.
- [9] N. Kasthuri and J. W. Lichtman, "Neurocartography," *Neuropsychopharmacology*, 2010.
- [10] V. Kaynig, T. Fuchs, and J. M. Buhmann, "Neuron geometry extraction by perceptual grouping in stem images," in *Proc. Int. Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2902–2909.
- [11] G. Knott, H. Marchman, D. Wall, and B. Lich, "Serial section scanning electron microscopy of adult brain tissue using focused ion beam milling," *Journal of Neuroscience*, vol. 28, no. 12, pp. 2959–2964, 2008.
- [12] A. Kreshuk, C. N. Straehle, C. Sommer, U. Koethe, M. Cantoni, G. Knott, and F. A. Hamprecht, "Automated detection and segmentation of synaptic contacts in nearly isotropic serial electron microscopy images," *PLoS ONE*, vol. 6, no. 10, p. e24899, 10 2011.
- [13] R. Kumar, A. Vázquez-Reina, and H. Pfister, "Radon-like features and their application to connectomics," in *Proc. Int. Conference on Computer Vision and Pattern Recognition Workshops*, 2010, pp. 186–193.
- [14] D. Lee, K.-H. Lee, W.-K. Ho, and S.-H. Lee, "Target cell-specific involvement of presynaptic mitochondria in post-tetanic potentiation at hippocampal mossy fiber synapses," *The Journal of Neuroscience*, 2007.
- [15] A. Lucchi, K. Smith, R. Achanta, G. Knott, and P. Fua, "Supervoxel-based segmentation of mitochondria in em image stacks with learned shape features," *IEEE Transactions on Medical Imaging*, vol. 31, no. 2, pp. 474–486, 2012.
- [16] J. Morales, L. Alonso-Nanclares, J.-R. Rodríguez, J. DeFelipe, Á. Rodríguez, and Á. Merchán-Pérez, "Espina: a tool for the automated segmentation and counting of synapses in large stacks of electron microscopy images," *Frontiers in neuroanatomy*, vol. 5, 2011.
- [17] R. Narasimha, H. Ouyang, A. Gray, S. W. McLaughlin, and S. Subramaniam, "Automatic joint classification and segmentation of whole cell 3D images," *Pattern Recognition*, vol. 42, no. 6, pp. 1067–1079, 2009.
- [18] S. Navlakha, J. Suhan, A. L. Barth, and Z. Bar-Joseph, "A high-throughput framework to detect synapses in electron microscopy images," *Bioinformatics*, vol. 29, no. 13, pp. i9–i17, 2013.
- [19] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.



(a) Random Forest Classifier



(b) Gaussian Classifier

Fig. 7. Jaccard index of synapse and mitochondria segmentation. The top figure shows results with a random forest classifier, the bottom figure shows results with a Gaussian classifier.

V. CONCLUSIONS

We tested nine feature descriptors with two classifiers in an EM stack for the joint segmentation of mitochondria and synapses. Our results show that GRIMS and simple window descriptors exhibit the best performance.

Although the Random Forest classifier achieves better precision, we suggest the use of the Gaussian Classifier given the large size of the typical EM image stacks and the gains in speed provided by this classifier.

As a future work we would like to consider the segmentation of other prominent neuronal structures, such as vesicles, myelin or membranes. We would also like to use feature selection techniques to study the combination of several image features.

- [20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [21] A. Roseman, "Findema fast, efficient program for automatic selection of particles from electron micrographs," *Journal of structural biology*, vol. 145, no. 1, pp. 91–99, 2004.
- [22] B. Schmid, J. Schindelin, A. Cardona, M. Longair, and M. Heisenberg, "A high-level 3D visualization api for java and imagej," *BMC Bioinformatics*, vol. 11, no. 1, p. 274, 2010.
- [23] K. Smith, A. Carleton, and V. Lepetit, "Fast ray features for learning irregular shapes," in *Proc. Int. Conference on Computer Vision*, 2009, pp. 397–404.
- [24] M. Varma and A. Zisserman, "Texture classification: Are filter banks necessary?" in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2003, pp. 691–698.
- [25] S. Vitaladevuni, Y. Mishchenko, A. Genkin, D. Chklovskii, and K. Harris, "Mitochondria detection in electron microscopy images," in *Workshop on Microscopic Image Analysis with Applications in Biology*, 2008.