AUTOMATED CELL NUCLEUS DETECTION FOR LARGE-VOLUME ELECTRON MICROSCOPY OF NEURAL TISSUE

F. Boray Tek[1*] *Thorben Kroeger*[2] *Shawn Mikula*[3] *Fred A. Hamprecht*[2]

[1]Department of Computer Engineering, Işık University, İstanbul, Turkey
[2]HCI, University of Heidelberg, Germany
[3]Max Planck Institute for Medical Research, Heidelberg, Germany

ABSTRACT

Volumetric electron microscopy techniques, such as serial block-face electron microscopy (SBEM), generate massive amounts of image data that are used for reconstructing neural circuits. Typically, this requires time-intensive manual annotation of cells and their connections. To facilitate this analysis, we study the problem of automated detection of cell nuclei in a new SBEM dataset that contains cerebral cortex, white matter, and striatum from an adult mouse brain. The dataset was manually annotated to identify the locations of all 3309 cell nuclei in the volume. We make both dataset and annotations available here. Using a hybrid approach that combines interactive learning, morphological processing, and object level feature classification, we demonstrate automated detection of cell nuclei at 92.4% recall and 95.1% precision. These algorithms are not RAM-limited and can scale to arbitrarily large datasets.

Index Terms— automated nucleus detection, block-face electron microscopy, interactive segmentation, random forest, block-wise connected components, connectomics, soma

1. INTRODUCTION

High-throughput volumetric electron microscopy techniques have great potential for densely reconstructing neural circuitry in exquisite detail [1]. Extraction of useful information from raw electron microscopy data, such as the morphology of individual neurons and their connections, has largely relied on time-consuming manual annotation. In this work, we study the feasibility of automated neuronal and glial nuclei detection and segmentation. The detected nuclei can be used as seed points for subsequent manual or automated tracing of neurites for morphological and circuit reconstructions. Furthermore, nuclear structure is a useful indicator of cell type; and nuclear locations make it possible to gather cellular spatial statistics and perform quantitative cytoarchitectonics for objective regional parcellations.

Past studies on automated cellular nucleus detection have relied on light microscopic detection of nuclei-specific markers such as NeuN [2] and DAPI [3]. In this case, blob-detection algorithms are sufficient for reliable nuclei segmentation because nuclei are the only objects labeled. In contrast, in electron microscopy, all cellular membranes are labeled, including not just nuclei and somas, but also mitochondria, other sub-cellular organelles and an elaborate network of neurites. The extensive membrane labeling coupled with an absence of specific nuclear markers significantly complicates the problem of nucleus segmentation and renders blob-detection algorithms insufficient.

An additional complication is the massive size of volumetric electron microscopy datasets, which can easily exceed terabytes and therefore require segmentation algorithms that do not load the full dataset into RAM. Therefore we implement block-wise and component-wise processing techniques, which enables the proposed nucleus detection algorithm to scale to arbitrarily large datasets.

The main contributions of this study are: 1) the first algorithm to detect neuronal and glial nuclei in large-volume serial block-face electron microscopy (SBEM), and 2) block-wise and component-wise processing of arbitrarily large volumes (code available online).

2. DATASET DESCRIPTION

2.1. Data Acquisition

A 20-week-old male mouse brain was prepared in its entirety for electron microscopy [4]. A sub-volume containing cerebral cortex, white matter, and striatum was extracted from the epoxy-embedded whole brain with a trimmer (Leica) and scalpel blade, and mounted on an aluminum stub. Back-scattered electrons were imaged at $40\mathrm{nm}$ pixel size in high vacuum with SBEM [1] on a QuantaFEG 200 (FEI) and using a heuristic-based algorithm for automated aberration correction [5]. The final stack size for the cortico-striatal dataset was $4382 \times 3435 \times 30464$ voxels, which was subsequently downscaled 4 times for cell nucleus detection.

[*] while visiting the Heidelberg Collaboratory for Image Processing.

2.2. Data Format

The downscaled and cropped SBEM volume is grayscale (8-bit) and $1024 \times 768 \times 7552$ voxels (x, y, z) in size, where each voxel is $160 \times 160 \times 200\mathrm{nm}$. It is available for download as an HDF5 file, which can be easily viewed by HDFView or read from using Python, Matlab or C. An example 2D subset is shown in Fig. 1.

2.3. Ground Truth and Evaluation Criteria

All neuronal and glial nuclei within the volume were annotated by domain experts. This 'ground-truth' contains 3309 nuclei with estimated center coordinates and radius. Since many nuclei are not spherical, the annotated center and radius is only a rough indication of location and size. The annotations are available as a CSV file with x, y, z, and r columns.

In order to exclude boundary effects, we have removed all spherical annotations intersecting the volume border. This resulted in 2935 final annotations (provided as another file). Before evaluation, the same procedure was applied to detection results. A detection is counted as a true positive only once, and only if it overlaps by at least t voxels with a ground-truth (where t is set as the ground-truth radius). If more than one detection overlaps with the same ground-truth nucleus, the detection with the largest overlap volume is counted as a true positive; all other overlapping detections are counted as false positives.

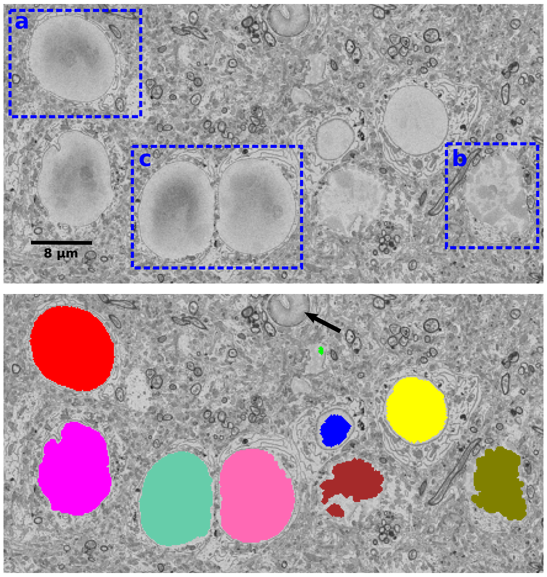


Fig. 1. (Top) An example window (483×227 pixels) from cerebral cortex ($z = 640$) from the original SBEM volume; annotations show (a) neuronal nucleus, (b) glial nucleus, (c) two closely-apposed neuronal nuclei. (Bottom) Output of the neuronal and glial nucleus detection overlayed. The arrow points to an endothelial cell nucleus that was not included in the ground truth and was correctly not detected / segmented.

3. METHODS

Challenges associated with cell nucleus segmentation in SBEM data can be summarized as intra- and inter-regional variability between cell classes, sample staining and imaging variability, and the existence of closely apposed nuclei and cell clusters. Nucleus appearance varies significantly depending on cell type. Neuronal nuclei are generally large, spherical and show a prominent, homogeneous nucleolus with dust-like chromatin granules in a pale nucleoplasm, whereas glial nuclei tend to be smaller with condensed chromatin (Fig. 1) [6]. Nuclei belonging to different sub-classes of neuronal (pyramidal, medium spiny, GABAergic, cholinergic, etc.) and glial (oligodendrocyte, astrocyte, etc.) cells found in our dataset are also known to have distinct structural features [7]. Additionally, the proportion of cell classes varies between different regions of the brain. For example, in our dataset, the dominant cell class is pyramidal cell in cerebral cortex, oligodendrocyte in white matter, and medium spiny neuron in striatum, each displaying a unique nuclear structure.

As a first step towards neuronal and glial nucleus detection, we need a robust classification scheme to distinguish voxels of these nuclei from other voxels (background), where the latter is composed of a variety of neurites, sub-cellular organelles, endothelial cells, and blood vessel lumen, which have similar texture and intensity to the nuclei we aim to detect. Moreover, closely apposed nuclei increase the potential of under-segmented (merged) objects. The classifier should detect as many nucleus voxels as possible, while not causing over / under-segmentation. Subsequent processing steps at component level can deal with false positive voxels.

Our proposed method is motivated by the above requirements and is comprised of three parts: First, we use ilastik [8] to *interactively* train a random forest classifier. Second, we perform thresholding and connected component analysis on the classifier's output probabilities. Filtering components by size followed by morphological processing defines *objects*. Third, we extract shape and intensity based features for each object and train a random forest classifier to obtain final object-level (nucleus/ non-nucleus) decisions.

3.1. Voxel classification

For interactive training of a voxel classifier, we use the free tool ilastik [8], which has already been used for synapse detection on similar data [9].

ilastik computes a set of rotation invariant voxel features (including smoothed raw data, gradient magnitude, Hessian and structure tensor eigenvalues at various scales) in a 3D neighborhood around each voxel. We set up a binary (nucleus vs. background) classification problem. We take a very small subset of data and label some nucleus and background pixels. Then, an immediate classification result is produced, overlayed on original data, and inspected by the user. Further labels are given iteratively until results are satisfactory. After training on the small subset, the classifier is used to predict the nucleus class probability of all voxels in the remaining volume.

3.2. Object detection

Next, we obtain a binary volume image by applying a threshold on the nucleus class probability of each voxel. After a connected component labeling, a size filter removes spurious isolated voxels. Morphological filtering further refines the detection candidates. These operations have to be performed block-wise and component-wise because the data is too large (available online, see end).

Connected component analysis works on blocks which overlap by one voxel in all dimensions. In the first pass, a connected component labeling is computed separately for each block. Adding an offset value to each block's labels ensures a unique range. The resulting label images are compressed and stored in RAM in order to avoid writing to the slower hard disk. In the second pass, a union-find data structure (UFD) is used to build the global connected component labeling. For each pair of adjacent blocks, the two labels for each voxel in the overlap region are merged in the UFD. Finally a dense relabeling (given by the UFD) is applied to each block and the result is written to disk.

For each connected component we compute the centroid, bounding box and its coordinate list by block-wise accumulation. Then, connected components with fewer than V_{th1} voxels are removed because tiny components can be excluded from more expensive further analysis. We discuss the effect of V_{th1} on accuracy in Sec. 4.

To refine and separate merged components we employ a sequence of *3D filtering and morphological operations*. First, we locally reconstruct the binary component using its bounding box and coordinate list. Then, a 3D morphological hole filling operation closes any holes. Next, to separate attached nuclei we apply an 3D opening operation with a sphere-shaped structuring element (SE) of radius $r_s = r_p/K$, where $K = 5$ is a scaling constant, and r_p is the pseudo-radius value of the object $r_p = \sqrt[3]{3v/4\pi}$, and volume v. Any newly created holes are filled again. The local volume is re-labeled to analyze newly emerged components; any new components with volume greater than V_{th2} are added to the output list of *objects*. The effect of V_{th2} on the number of true / false positive objects is discussed in Sec. 4.

3.3. Object classification

For each object in the list we calculate 41 features which include: volume (v), normalized global centroid coordinate (because nuclei vary along z-dimension), pseudo radius (r_p), number of surface voxels (s) (difference of the component from its morphological erosion with a 3D cross SE of size 3×3×3), complexity ($c = s \cdot r_p/v$), aspect ratio of the bounding box, ratio of bounding box volume to object volume, and gray scale intensity histogram of 32 bins.

We train an additional random forest classifier. The labels for the classifier are derived automatically from the overlap of the objects with the ground truth center positions. For each object, the classifier produces the nucleus class probability, which is thresholded to obtain the final detection results. The effect of the threshold value on true / false positive numbers is discussed in Sec. 4.

4. EXPERIMENTS AND RESULTS

First, we evaluated the accuracy of the voxel classifier. In order to obtain a representative training set, we sampled 20 small nucleus blocks (only 0.048% of the total volume) distributed along the z-axis to account for variances among nuclei and background appearance in different parts of the brain. All features of ilastik were utilized.

After thresholding the probability at $p = 0.5$ and connected component labeling (Sec. 3.2), the detected objects were compared against the ground-truth; and true / false detections were counted. The result was 93.22% recall (true positives) at 0.05% precision due to more than five million false positives (Tab. 1, row CC1). Increasing the volume threshold slightly ($V_{th1} \in \{5, 25\}$) reduced the number of false positive detections significantly (CC2 and CC3). Therefore, we set V_{th1}=25 for subsequent experiments.

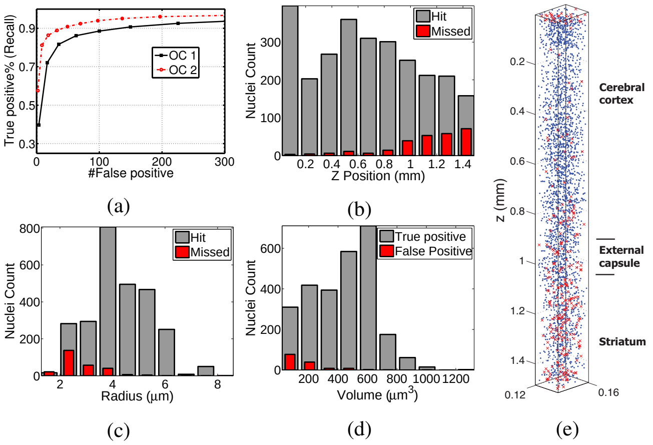
Next, we evaluate the filtering and morphological processing stage for different volume thresholds $V_{th2} = \{$0, 125, 625, 1000, 2500, 5000, 10000$\}$. The results can be seen in the Object Detection (OD) rows of Tab. 1. Even with no volume threshold (OD1), the procedure increased the recall rate but caused a slight drop in precision, due to separation of attached nuclei. The precision improves significantly when V_{th2} was increased to 25 or 1000; however, larger values causes the recall rate to drop faster.

Therefore, we take the 4189 candidates generated by $V_{th2} = 1000$ and evaluate the object classification stage (OC). First, we have limited the training set to be formed only of objects which were included in voxel classification training volumes. This results in a training set of only 154 positive / 62 negative samples; the remaining 2704 positive and 1269 negative samples were left for test. The result of classification (on top of object detection) for prediction threshold of 0.5 is shown in OC1 row of the Table 1. The recall rate dropped to 90.65%; however, the precision has improved significantly (OC1) and beyond the levels that could be achieved by the filtering stage (compare OD5 with OC1). Second, we tested distributing the samples *randomly* and equally into training and test sets, which provided even better results (OC2). Fig. 2 (a) shows Receiver Operating Characteristics (ROC) curve for OC1 and OC2.

The distribution of ground truth hits/misses in Fig. 2 (c) shows that the algorithm mainly misses small nuclei (presumably glial) and nuclei located in striatum Fig. 2 (e). An examination of the results has shown that missed nuclei were mainly glial, whereas almost all neuronal nuclei were detected and segmented completely. This was often caused by the difficulty of separating glial and background voxels which have similar textures. The variability of nucleus volume among true positives is shown in Fig. 2 (d); note that the distribution of volumes among false positives confirms that smaller sized objects mostly contributed to mistakes.

Table 1. Recall and precision performance of the different stages of the method (see text).

Proc.	Parameter	Recall%	Precision%	#False
CC1	$V_{th1}=1$	93.22	0.05	5591508
CC2	$V_{th1}=5$	93.22	0.68	395894
CC3	$V_{th1}=25$	**93.05**	**4.5**	**57826**
OD1	$V_{th2}=0$	99.11	3.74	74846
OD2	$V_{th2}=125$	99.01	18.70	12627
OD3	$V_{th2}=1000$	**97.38**	**68.22**	**1300**
OD4	$V_{th2}=5000$	93.08	87.22	400
OD5	$V_{th2}=10000$	90.21	90.74	284
OC1	$p_{th} \ = 0.5$	$90.65 \pm 0.5\%$	$94.4 \pm 0.4\%$	150 ± 1.1
OC2	$p_{th} \ = 0.5$	$\mathbf{92.43 \pm 0.6\%}$	$\mathbf{95.1 \pm 0.6\%}$	$\mathbf{70 \pm 8.6}$



(a) (b) (c) (d) (e)

Fig. 2. (a) ROC plots for the limited (OC1) and equally distributed training and test data (OC2). Distribution of: (b) hit and missed ground truth nuclei w.r.t z position; (c) w.r.t ground truth radius; (d) detection volume for true and false positives. (e) Locations of hit (blue) and missed (red x) nuclei in the imaged part of the brain.

5. CONCLUSION

We have demonstrated accurate and fully-automated detection and segmentation of neuronal and glial nuclei with 92.4% recall and 95.1% precision in membrane-contrasted volumetric electron microscopy datasets. Further improvement in classification performance is expected by using more training data for the final stage classifier. Another avenue for improvement could be to enhance nuclear staining through the use of nuclear-specific heavy-metal stains. Our algorithms, which are not RAM-limited and can scale to arbitrarily large datasets, will prove useful to future electron microscopy studies investigating precise spatial relationships between different cell types throughout the brain.

Acknowledgements

F. Boray Tek's work in Heidelberg Collaboratory for Image Processing was supported by Işık University and by The Scientific and Technological Research Council of Turkey through BIDEP 2209 grant. We thank Winfried Denk for serial block-face electron microscopy resources and Sarah Mikula and Ivo Sonntag for manually annotating neuronal and glial nuclear sizes and locations in our SBEM dataset. Supported by the Max Planck Society.

6. REFERENCES

[1] W. Denk and H. Horstmann, "Serial block-face scanning electron microscopy to reconstruct three-dimensional tissue nanostructure," *PLoS biology*, vol. 2, no. 11, pp. e329, 2004.

[2] M. Oberlaender, V. J. Dercksen, R. Egger, et al., "Automated three-dimensional detection and counting of neuron somata," *J. Neurosci. Meth.*, vol. 180, no. 1, pp. 147–160, 2009.

[3] P. S. Tsai, J. P. Kaufhold, P. Blinder, et al., "Correlations of neuronal and microvascular densities in murine cortex revealed by direct counting and colocalization of nuclei and vessels," *Journal of Neuroscience*, vol. 29, no. 46, pp. 14553–14570, 2009.

[4] S. Mikula, J. Binding, and W. Denk, "Staining and embedding the whole mouse brain for electron microscopy," *Nature methods*, vol. 9, no. 12, pp. 1198–1201, 2012.

[5] J. Binding, S. Mikula, and W. Denk, "Low-dosage maximum-a-posteriori focusing and stigmation," *Microsc. Microanal.*, vol. 19, no. 01, pp. 38–55, 2013.

[6] E. Ling, J. A. Paterson, A. Privat, S. Mori, and C. Leblond, "Investigation of glial cells in semithin sections. I. Identification of glial cells in the brain of young rats," *J. Comp. Neurol.*, vol. 149, no. 1, pp. 43–71, 1973.

[7] M. Matamales, J. Bertran-Gonzalez, L. Salomon, et al., "Striatal medium-sized spiny neurons: identification by nuclear staining and study of neuronal subpopulations in BAC transgenic mice," *PloS one*, vol. 4, no. 3, pp. e4770, 2009.

[8] C. Sommer, C. Straehle, U. Köthe, and F. Hamprecht, "Ilastik: Interactive learning and segmentation toolkit," in *ISBI*, 2011, pp. 230–233.

[9] A. Kreshuk, C. N. Straehle, C. Sommer, et al., "Automated detection and segmentation of synaptic contacts in nearly isotropic serial electron microscopy images," *PloS one*, vol. 6, no. 10, pp. e24899, 2011.

Dataset and ground-truth annotations available at:
`http://hci.iwr.uni-heidelberg.de/somas`