

Received April 22, 2021, accepted May 8, 2021, date of publication May 26, 2021, date of current version June 4, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3084066

ZeVis: A Visual Analytics System for Exploration of a Larval Zebrafish Brain in Serial-Section Electron Microscopy Images

JUNYOUNG CHOI¹, DAVID GRANT COLBURN HILDEBRAND², JUNGMIN MOON¹,
TRAN MINH QUAN³, TRAN ANH TUAN¹, SUNGAHN KO¹,
AND WON-KI JEONG⁴, (Member, IEEE)

¹School of Electrical and Computer Engineering, Ulsan National Institute of Science and Technology, Ulsan 44919, South Korea

²Laboratory of Neural Systems, The Rockefeller University, New York, NY 10065, USA

³College of Engineering and Computer Science, VinUniversity, Hanoi 12400, Vietnam

⁴Department of Computer Science and Engineering, College of Informatics, Korea University, Seoul 02841, South Korea

Corresponding author: Won-Ki Jeong (wkjeong@korea.ac.kr)

This work was supported in part by the Brain Research Program under Grant NRF-2017M3C7A1047904, in part by the Bio and Medical Technology Development Program of the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (MSIT) under Grant NRF-2019M3E5D2A01063819, in part by the Korea Health Technology Research and Development Project through the Korea Health Industry Development Institute (KHIDI) funded by the Ministry of Health and Welfare, Republic of Korea, under Grant HI18C0316, and in part by the ICT Creative Consilience Program of the Institute for Information & Communications Technology Planning & Evaluation (IITP) funded by MSIT under Grant IITP-2021-0-01819. The work of David Grant Colburn Hildebrand was supported in part by the Fellowship in Neuroscience from the Leon Levy Foundation, and in part by the National Alliance for Research on Schizophrenia and Depression (NARSAD) Young Investigator Grant from the Brain & Behavior Research Foundation.

ABSTRACT The automation and improvement of nano-scale electron microscopy imaging technologies have expanded a push in neuroscience to understand brain circuits at the scale of individual cells and their connections. Most of this research effort, called ‘connectomics’, has been devoted to handling, processing, and segmenting large-scale image data to reconstruct graphs of neuronal connectivity. However, connectomics datasets contain a wealth of high-resolution information about the brain that could be leveraged to understand its detailed anatomy beyond just the connections between neurons, such as cell morphologies and distributions. This study introduces a novel visualization system, *ZeVis*, for the interactive exploration of a whole larval zebrafish brain using a terabyte-scale serial-section electron microscopy dataset. *ZeVis* combines 2D cross-sectional views and 3D volumetric visualizations of the input serial-section electron microscopy data with overlaid segmentation results to facilitate the analyses of various brain structures and their interpretations. The system also provides a graph-based data processing interface to generate subsets of feature segmentation data easily. The segmentation data can be filtered by morphological features or anatomical constraints, allowing statistical analysis and comparisons across regions. We applied *ZeVis* to actual data of a terabyte-scale whole-brain larval zebrafish and analyzed cell nucleus distributions in several anatomical regions.

INDEX TERMS Cell morphology, larval zebrafish, serial-section electron microscopy, visual analytics.

I. INTRODUCTION

Connectomics is a branch of neuroscience that aims to study the densely connected neuronal circuits embedded within the brain. To examine densely packed neuronal tubes having diameters of only tens of nanometers [1], thin serial sectioning of a tissue sample and high-resolution electron

microscopy (EM) imaging are commonly used. The resolution of serial-section electron microscopy (ssEM) is typically 4–50 nanometers per voxel edge. Hence, the imaging of even small tissue samples (e.g., 1 mm³) can result in petabyte-scale raw data sizes containing about 100,000 nerve cells with 1 billion synapses [2]. One of the main challenges in analyzing connectomics datasets lies in developing scalable and automatic image processing and data analysis algorithms to cope with the ever-increasing data size and complexity.

The associate editor coordinating the review of this manuscript and approving it for publication was Ashish Mahajan¹.

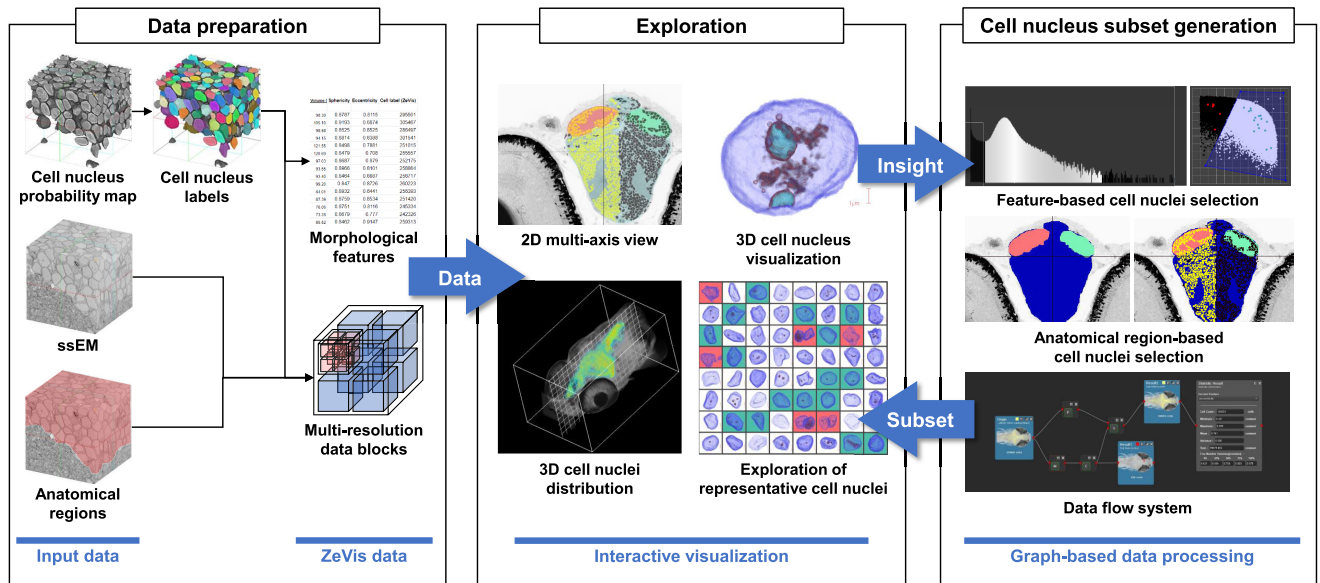


FIGURE 1. Visual analysis workflow and summary of each data processing step in ZeVis.

Over the past decade, significant research progress has been made in computer vision and visualization domains to address challenges in connectomics. Many prior studies focused on developing image processing algorithms, including image enhancement and segmentation, for extracting neuronal connectivity graphs from ssEM datasets. Most previous visualization research in connectomics [3]–[7] focused on building interactive data exploration systems that provide enhanced visual cues or metaphors to help users better determine brain connectivity. While a vast number of computational studies have focused on connectomics [8]–[12], few have sought to leverage the abundant information available in large-scale connectomics datasets to examine anatomical and cell biological details—e.g., cell distributions, cell densities, or morphological variations—across brain regions.

Our primary objective here is to begin to fill this void by producing a unified visual analysis workflow to examine the distribution and shape of segmented cell nuclei throughout an entire brain. We approach this problem using a recently generated terabyte-scale whole-brain larval zebrafish ssEM dataset [8], [13]. In this study, we propose a novel visualization system, ZeVis, which includes an interactive visualization technique for inspecting cell nucleus morphology and distribution in 2D cross-sectional and 3D volumetric views. This system also offers graph-based data processing for generating subsets of nuclei with specific properties or restricted to specific brain regions, while allowing users to filter data and perform comparative visual analysis easily and interactively. We demonstrated the usability of ZeVis via two case studies: selection of target cell nucleus subsets and comparison of brain symmetry. In these case studies, we generated normal cell nucleus subsets by filtering out objects segmented incorrectly. We then analyzed the difference in cell nucleus morphology, density, and distribution across the

left and right hemispheres of the larval zebrafish brain. We also demonstrated that the proposed system could handle multi-terabyte ssEM data.

II. METHODS

The overview of the proposed system is shown in Figure 1. First, the data is processed into a ZeVis format that enables interactive exploration of terabyte-scale volumetric data and effective analysis of cell nuclei. With the formatted data, specific subsets of cell nuclei can be generated and analyzed through the *interactive visualization* and *graph-based data processing* steps in the ZeVis system.

The interactive visualization step includes: 1) a 2D multi-axis view for observing slices through the ssEM data, anatomical subregions, and selected cell nucleus subsets; 2) a 3D visualization for observing the shape of each cell nucleus; 3) a representative cell nucleus explorer that displays the range of cell nucleus shapes contained within the selected subset; and 4) a 3D distribution viewer that shows the locations of the cell nuclei contained within the selected subset.

The graph-based data processing step includes: 1) feature-based cell nucleus selection, 2) anatomical subregion-based cell nucleus selection, and 3) a data flow system that can combine multiple selection criteria for flexibly generating cell nucleus subsets and effectively managing multiple analyses.

Through exploration of terabyte-scale data in combination with the user’s interactions and graph-based data processing, the interactive visualization step aims to accurately analyze morphological features of each target cell nucleus and reveal the distribution of generated cell nucleus subsets. Multiple selection methods can be used to create subsets of target cell nuclei in order to facilitate a variety of analyses. In the subsequent sub-section, we explain how each element of the ZeVis system achieves these goals.

A. DATA PREPARATION

The input volume data, consisting of a ssEM image stack, anatomical subregions, and a cell nucleus segmentation probability map, must be converted into multi-resolution data blocks. The cell nucleus segmentation probability map is converted to cell nucleus labels by thresholding to create a binary mask (for our experiments, threshold = 0.5) and then performing connected component analysis (Fig. 2). Morphological features of each cell nucleus are then computed from these labels. The six morphological features used in this study are listed in Table 1.

TABLE 1. Morphological features for cell nucleus analysis.

Morphological feature	Definition
Intensity	Average voxel value of the ssEM volume data contained within a cell nucleus region
Volume (v)	Number of voxels in a cell nucleus region
Surface area (s)	Estimated total area on the surface of a cell nucleus region using the Crofton formula and run-length encoding [14]
Sphericity	Describes how close a cell nucleus region shape is to a sphere, with values closer 1 indicating a more spherical shape: $Sphericity = \frac{4\pi(\sqrt[3]{3v/4\pi})^2}{s}$
Eccentricity	Describes how elongated a cell nucleus region is by calculating the difference in length between the major axis (λ_1) and the minor axis (λ_2) created by principal component analysis (PCA), with values closer to 1 indicating more elongated shapes: $Eccentricity = \sqrt{1 - \frac{\lambda_2^2}{\lambda_1^2}}$
Position (x, y, z)	The centroid of a cell nucleus region

Because the input data has a terabyte-scale size, it far exceeds the size that can be handled by a conventional workstation with only gigabytes of random-access memory. We utilize a streaming method that interactively loads the required data from disk to overcome this problem. We also employ a multi-resolution data structure for the ssEM data, subregion volumes, and cell nucleus labels based on three axial views to ensure interactive rendering and rapid exploration of the multi-terabyte data. Figure 3 shows an example in which the level 0 block (original resolution) and the corresponding blocks are converted to the next level (lower resolution). To accomplish this for the entire dataset, the original-resolution data is first split into $512 \times 512 \times 512$ voxel blocks. Then, the next level blocks, with lower resolution and a wider view, are generated by downsampling along two axes, so that even when the level of the block increases, the resolution of one axis is maintained to permit smooth viewing of the data at fine increments. By iterating this process, we create blocks of ssEM data, cell nucleus labels, and anatomical subregions from level 0 to level 5. When displaying, the blocks required for the user-requested rendering scale and region are identified and then 2D image tiles of size 512×512 pixels are loaded from the blocks and rendered. Recently used tiles are cached and searched first to enable smooth navigation (Fig. 4).

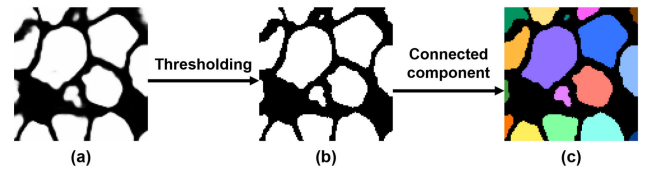


FIGURE 2. Process of cell nucleus labeling | The cell nucleus probability map (a) is converted to a binary mask (b) by thresholding. The cell nucleus labels (c) are then generated by connected components analysis.

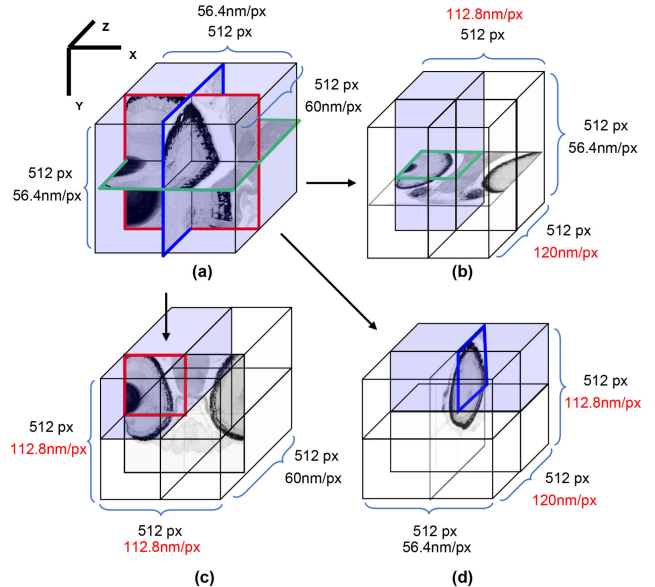


FIGURE 3. Multi-resolution data block conversion | Each block contains 512^3 voxels. Each level 0 block (a) has the original resolution ($56.4 \times 56.4 \times 60\text{nm}^3 \text{vx}^{-1}$). Separate level 1 blocks maintain full resolution along the y axis (b), the z axis (c), or the x axis (d) and have half the resolution along the x and z axes, the x and y axes, or the y and z axes, respectively. The higher-level blocks represent a wider range of data with a lower resolution for each plane.

B. INTERACTIVE VISUALIZATION

1) 2D MULTI-AXIS VIEWS

The visualization showing the cross-section of each axis consists of three layers: the ssEM data layer, the anatomical subregion layer, and the cell nucleus label layer. For the user-specified field of view, the ssEM data layer shows the EM image data, the anatomical subregion layer shows restricted areas in the brain defined by masks, and the cell nucleus label layer shows each individual cell nucleus. In the cell nucleus label layer, each cell nucleus is assigned a color representing the individual cell nucleus index (random color) or the subset to which it belongs (single color per subset). These three layers are alpha-blended together with a user-defined transparency for the final rendering (bottom of Fig. 4).

2) 3D VISUALIZATION OF CELL NUCLEUS DISTRIBUTIONS

With only 2D cross-sectional views, it is difficult to represent the spatial distribution of cell nuclei in the 3D space. Therefore, directly visualizing the cell nucleus distribution in 3D can help the user understand and analyze the data more

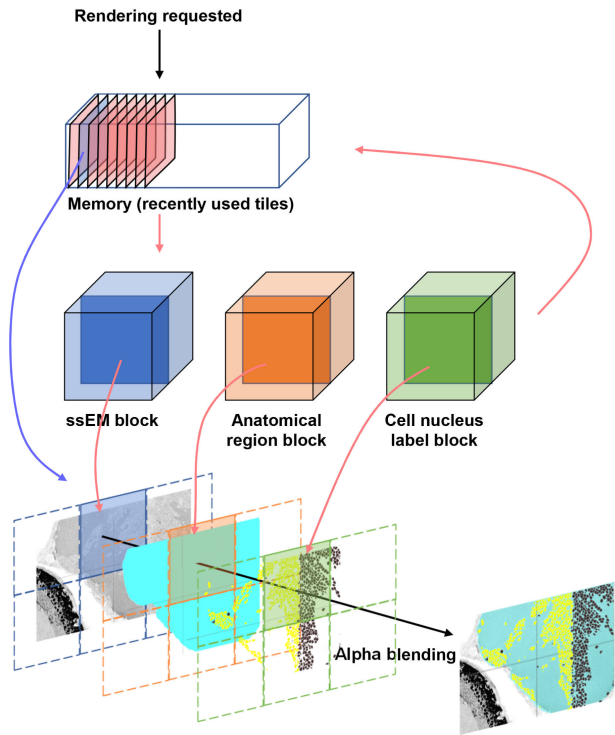


FIGURE 4. Image rendering process | When a rendering is requested, the required data for the position and scale is first searched for in memory. If found in memory, the data is referenced directly. Otherwise, the data is loaded from data blocks on the disk. The loaded data are then rendered by alpha blending to produce the final result.

intuitively. To generate a 3D representation of the cell nucleus distribution, a 3D histogram for the spatial distribution of the selected cell nucleus subset is computed and rendered with the ssEM image volume by direct volume rendering (Fig. 5). When the user specifies the scale of the 3D visualization, a 3D grid that fits the scale is formed and the cell nucleus count within each grid bin is calculated. When the scale is small, the grid is created sparsely to efficiently depict the overall distribution (Fig. 5a). When the scale is increased, the a denser grid is created to show the distribution with more detail (Fig. 5b). Additionally, a range of cell nucleus counts can be selected interactively so that only grid bins with the corresponding values are displayed (Fig. 5c).

3) 3D VISUALIZATION OF INDIVIDUAL CELL NUCLEI

It can also be useful to visualize and analyze the 3D shape of individual cell nuclei. To increase the information available to the user, we generate a visualization that combines the ssEM data and cell nucleus label (Fig. 7). First, a 3D volume is created by loading the segmented cell nucleus region from the segmentation label block. Next, a 3D masked ssEM volume is created by loading the ssEM image data belonging to this same segmented cell nucleus region from the ssEM data block, thus separating the data of interest from complex surrounding structures. The ssEM volume is then visualized using direct volume rendering to display the detailed internal structure of the cell nucleus, blended with

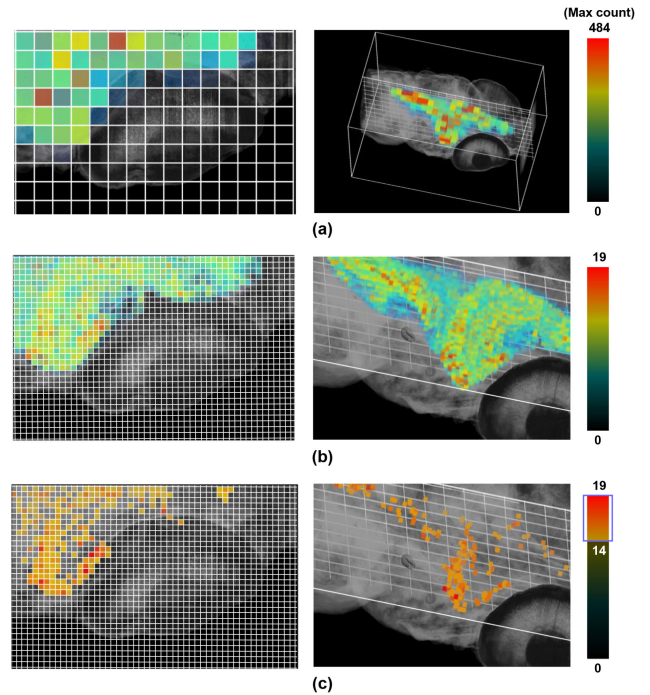


FIGURE 5. Interactive 3D histogram for intuitive observation of cell nucleus distribution | Grids are generated at the user-selected scale for binning cell nucleus counts (left panels). The grids are then visualized in 3D (right panels). The grid is generated sparsely at the lower scale (a) and densely at the higher scale (b). If a specific range of cell nucleus counts is specified, only the grids with counts in the specified range are displayed (c).

the surface rendering of the segmented cell nucleus region for better representation of its 3D shape (Fig. 7d).

4) EXPLORATION OF REPRESENTATIVE CELL NUCLEI

To extract a representative set of example cell nuclei from a given subset, we use clustering in the 5-dimensional cell nucleus morphology feature space (intensity, volume, surface area, sphericity, and eccentricity). Through k-means clustering, 64 clusters are created. The cell nuclei closest to the centers of each cluster are then selected as the representative cell nuclei for the subset. 3D visualizations for the representative cell nuclei are created and displayed as an 8×8 grid (bottom of Fig. 6). This makes it possible to analyze or check the shape distribution of cell nuclei in a subset without checking each individually. In one basic use case, real cell nuclei (green background grid elements in Fig. 6) can be extracted from unwanted segmentation errors (red background grid elements in Fig. 6). These results can also be marked on the 2D view (green and red dots on the 2D plot in Fig. 6), allowing intuitive and precise selection of cell nucleus subsets. These extracted representative subsets can also be used for training machine learning classifiers (see Section II-C1).

C. GRAPH-BASED DATA PROCESSING

1) FEATURE-BASED CELL NUCLEUS SELECTION

The simplest method of selecting cell nuclei is to specify a particular range of cell nucleus feature values while observing

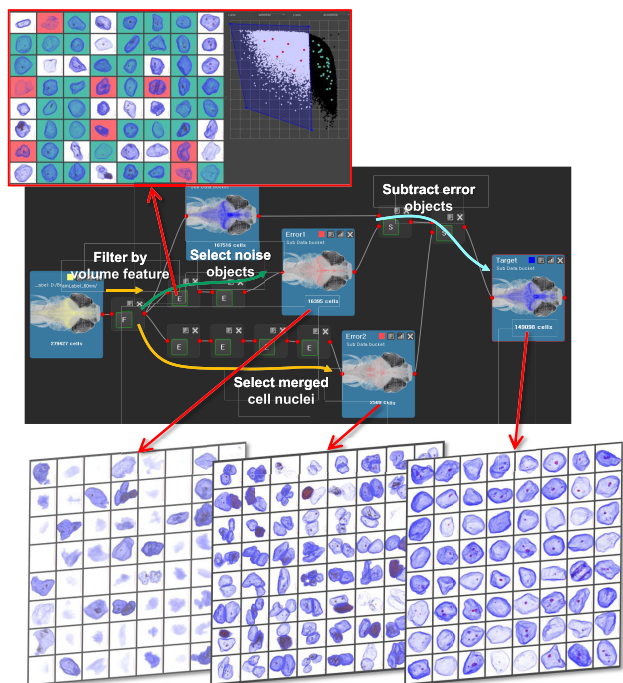


FIGURE 6. Process for creating cell nucleus subsets | Three analysis flows are used to generate a normal cell nucleus subset (shown in the upper part) from the initial cell nucleus segmentation set, which contains some noise. The bottom left, middle, and right panels depict the results of selection for non-nucleus objects, merged cell nuclei, and normal cell nuclei, respectively.

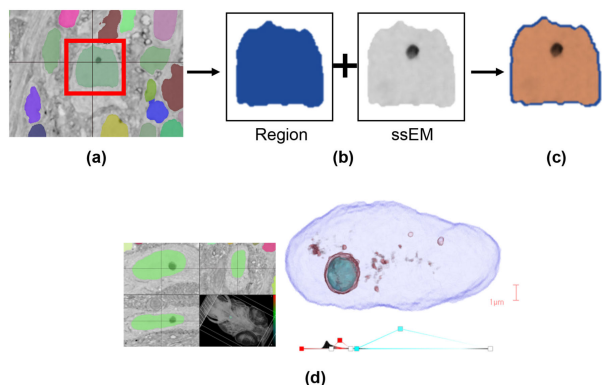


FIGURE 7. 3D visualization of individual cell nuclei | The cell nucleus boundary (a) is found and the ssEM image data and cell nucleus label region are retrieved (b). The final visualization is generated by combining the ssEM volume rendering and the region surface rendering (c). An example of this visualization is shown for a Mauthner cell (d).

the resulting distribution in a 1D feature histogram or a 2D feature plot. This method can create cell nucleus subsets that belong to specific feature ranges using information obtained either through exploration or prior knowledge.

However, this approach is insufficient when the exact feature distribution of the target cell nuclei is unknown or there is no single feature value threshold that can select the target cell nuclei accurately. An intelligent cell nucleus selection can instead be used to overcome this problem, for example by utilizing a machine learning approach to categorize cell

nuclei (Fig. 8). To accomplish this, target cell nuclei (foreground seed) and non-target cell nuclei (background seed) are selected from the 64 representative cell nuclei extracted from the input cell nucleus subset, then used to train a random forest classifier to categorize the nuclei in real-time. The trained classifier generates a new cell nucleus subset by performing classification on all cell nuclei in the input subset.

2) ANATOMICAL SUBREGION-BASED CELL NUCLEUS SELECTION

Cell subsets belonging to a specific region of the data can be created through anatomical region data. Moreover, new subregions can be created interactively by defining cut-planes, for example to divide the larval zebrafish into right and left sides along the body midplane. This process is computationally intensive and consume substantial amounts of time because the anatomical subregion can span large portions of the data. To minimize this burden, we create a subregion tree that represents the anatomical subregion as an octree [15]. The octree can then be used to perform cell nucleus filtering, subregion volume calculation, and cut-plane operations (Fig. 9). The subregion tree is created for each data block spanning the desired region and is divided into eight child nodes starting from a root node of size 512^3 until all voxels in the node are included or excluded in the subregion. Leaf nodes that do not contain all voxels in the region are removed. That is, the region can be expressed entirely as leaf nodes of the subregion tree.

To check whether each cell nucleus in the input subset is included in the subregion, we check that a leaf node at the cell nucleus location exists. The subregion volume can be calculated by summing the volumes of all leaf nodes. For creating a new subregion through a cut-plane operation, each node of the input subregion tree is checked to determine if it should be included in the new region, with only the included nodes being retained.

3) DATA FLOW SYSTEM

For flexible cell selection and efficient analysis, we employed a data flow system, which is an intuitive and dynamic technique that allows users to perform tasks by constructing flow diagrams [16], [17]. All selection and subregion generation methods can be chained in the data flow system for generating the desired subsets of cell nuclei from the entire brain set. This flow system uses a directed acyclic graph (DAG), where each node represents either data or an operation and each edge represents the data flow between nodes. Due to the nature of directed graphs, any modification to a node affects its descendants.

For visualization or analysis, the user can create an analysis flow starting from the whole brain cell nucleus set to the desired cell nucleus subset by connecting several selection method nodes. Additionally, a new cell nucleus subset can be created by combining different flows using set operations such as union, intersection, and subtraction between cell nucleus subsets.

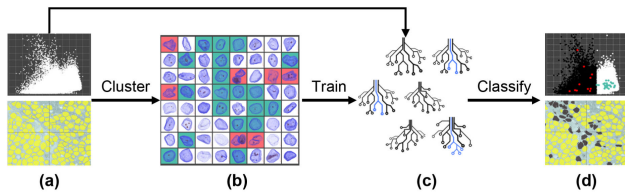


FIGURE 8. Intelligent cell nucleus selection | From the input cell nucleus subset (a), a representative cell nuclei (b) are found using a clustering approach. A random forest classifier (c) is then trained with the foreground and background selection information to re-classify all input cell nuclei and generate a new cell nucleus subset (d).

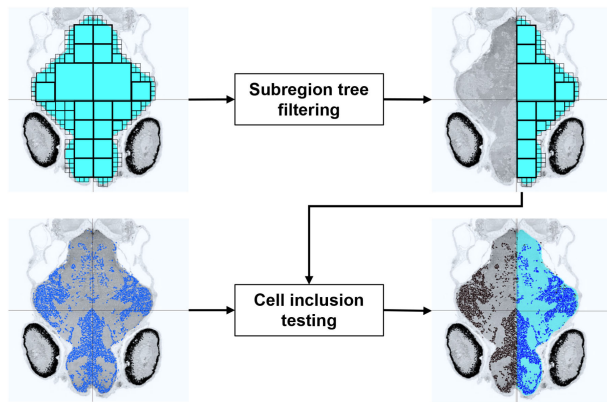


FIGURE 9. Selection of cell nuclei within an anatomical subregion | The input subregion tree for the brain region can be filtered by a cut-plane operation (top) to separate the left and right sides of the body. The new cell nucleus subset is generated by cell inclusion testing (bottom) using the generated subregion tree.

Figure 6 shows an example of the data flow system. In the upper graph interface, starting from the entire brain cell nucleus set (left-most node), a subset is generated by thresholding based on the value of the volume feature for each nucleus before being divided into three different analysis flows. The middle analysis flow (green arrow) creates a subset of objects that are not cell nuclei (artifacts). The bottom analysis flow (yellow arrow) creates a subset of cell nuclei which were incorrectly merged during segmentation. Finally, a subset of normal cell nuclei is generated by combining these analysis flows using the subtraction operation.

III. RESULTS

In this section, we demonstrate the usability and performance of the *ZeVis* system for examining a terabyte-scale larval zebrafish brain ssEM dataset.

A. DATASET

The input data used in this experiment are a larval zebrafish brain ssEM image volume, cell nucleus probability map, and subregion mask volumes for the brain, habenula, and optic tectum neuropil, each with size $10240 \times 9216 \times 16000$ voxels.

The larval zebrafish ssEM dataset was captured from a 5.5 days post-fertilization larval zebrafish. This specimen was

cut into $\sim 18,000$ serial sections and collected onto a tape substrate using an ATUM device [18]. A series of images spanning the anterior quarter of the larval zebrafish was acquired at a nearly isotropic resolution of $56.4 \times 56.4 \times \sim 60 \text{nm}^3 \text{vx}^{-1}$ from 16,000 sections using a scanning EM. All image planes were then co-registered into a 3D volume with a FFT signal whitening approach [19], forming an image volume that spans $2.28 \times 10^8 \mu\text{m}^3$ with 1.12×10^{12} voxels and occupies 2.4 terabytes [8], [13]. This volume was converted to the *ZeVis* multi-resolution data structure, which resulted in a total size of 5.68 terabytes.

To obtain cell nucleus labels, we used an automatic segmentation approach. Automatic nucleus segmentation is particularly challenging in the larval zebrafish brain because many of its cell nuclei are densely packed into a band along the body midplane. Many studies have tried to solve similar challenges with deep learning approaches, including deep neural networks [20], deep contextual networks [21], and deep fully residual convolutional neural networks (FusionNet [22]). Among the methods we investigated, FusionNet achieved better segmentation accuracy by leveraging recent advances in machine learning, including semantic segmentation and incorporation of residual neural network layers with summation-based skip connections to enable a much deeper network architecture. Therefore, we used FusionNet segmentation results to delineate cell nucleus regions automatically in the larval zebrafish ssEM dataset. The cell nucleus probability map obtained from these methods was converted to a cell nucleus label volume and multi-resolution data structure resulting in total size of 3.29 terabytes.

B. TARGET CELL NUCLEUS SELECTION

ssEM image segmentation is prone to errors due to various image artifacts and noise. Therefore, it is necessary to find and filter out abnormal structures such as non-nucleus objects and merged cell nuclei for accurate image analysis.

To exclude non-nucleus objects that are impractically small or large, lower and upper bounds on the volume feature are set. The upper bound can be determined by examining known large neurons that are likely to also have large cell nuclei. Figure 10 shows the volume histogram of all cell nucleus labels. The volume of the right-side Mauthner cell (denoted by the cyan line in Fig. 10), which has the largest volume among the known large neurons (denoted by the blue lines in Fig. 10), was defined as the upper bound. To determine the lower bound, the smallest cell nucleus volume among known large neurons (MiV2, green line in Fig. 10) was mirrored (red line in Fig. 10) across the median volume of the all cell nucleus labels (black dashed line in Fig. 10). Therefore, the lower and upper bound volumes of the normal cell nuclei were defined as $13.72 \mu\text{m}^3$ and $236.78 \mu\text{m}^3$, respectively.

It is possible that non-nucleus objects or merged cell nuclei may exist in the normal cell nucleus volume range. To exclude these, a non-nucleus object subset (bottom left in Fig. 6) and merged cell nucleus subset (bottom middle in Fig. 6) were

TABLE 2. Comparison of cell nucleus distributions across the left and right sides of the brain. WB: whole brain, L: left side of the brain, R: right side of the brain, HL: left habenula, HR: right habenula, OTNL: left optic tectum neuropil, and OTNR: right optic tectum neuropil.

	Subregion volume (μm^3)	Nucleus count	Total volume of nuclei (μm^3)	Median nucleus volume (μm^3)	Mean nucleus volume (μm^3)	Nucleus count density (%)	Nucleus density diff (%)	Nucleus diff (%)
WB	23,454,062.53	144,489	6,356,796.66	43.11	44.0	27.10		
L	11,729,463.30	69,053	3,071,339.216	43.60	44.48	26.18		
R	11,738,775.54	72,143	3,136,548.68	42.64	43.48	26.72	-4.47	-2.04
HL	83,741.71	604	29,433.39	48.47	48.73	35.15		
HR	65,804.85	537	25,321.53	46.93	47.15	38.48	11.09	-9.48
OTNL	925,803.96	395	20,123.58	48.84	50.95	2.17		
OTNR	841,212.06	443	20,658.9	45.02	46.63	2.46	-12.15	-12.98

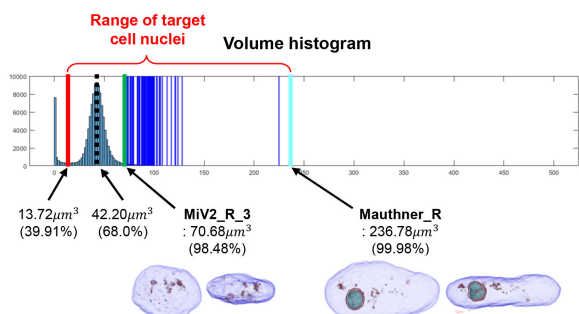


FIGURE 10. Lower and upper bound selection for realistic volume feature values using known large neurons | The blue lines on the volume histogram represent volumes associated with known identified large neurons. The upper bound (cyan line) was defined by the largest value among the known neurons. The smallest value among the known neurons (green line) was mirrored across the median of all cell nucleus labels to define the lower bound (red line).

generated by intelligent cell nucleus selection and excluded using a set operation.

From a total of 279427 label objects, 167516 objects were selected by volume thresholding. Among these, a normal cell nucleus subset with 149098 objects was created (bottom right in Fig. 6) by excluding 16395 non-nucleus objects and 2509 merged cell nucleus labels (486 labels were in both exclusion groups).

C. SYMMETRY COMPARISON

We compared the distribution of cell nuclei between the left and right sides of the brain. The brain subregion mask used in this study spans the entire brain without separating the left and right sides. Part of the spinal cord is also included in the brain mask. Hence, we make new subregions for the left and right side of the brain by applying vertical cut-planes to the entire brain subregion mask along the body midplane (Fig. 11). By combining this anatomical refinement with the normal cell nucleus subset, we generated cell nucleus subsets for each side of the brain.

For each side, the cell nucleus count, volume, and density were calculated from these subsets, as well as the difference between the left and right sides for the cell nucleus count and the cell nucleus density. Similarly, we also generated the left

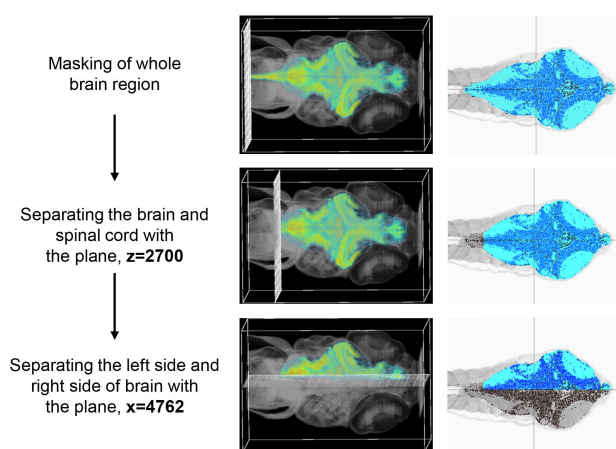


FIGURE 11. Subregion generation for the left side of the brain | A new subregion is created by applying a cut-plane operation that removes the spinal cord and a cut-plane operation that divides the left and right sides of the brain.

and right side subregions for other anatomical regions such as the habenula and the optic tectum neuropil and performed the same calculations. The results are shown in Table 2.

D. PERFORMANCE ANALYSIS

We measured the memory consumption of the aforementioned experiments to verify the scalability of the system. Additionally, to verify if the terabyte-scale data is processed interactively, time consumption for each operation was measured while the user performed the experiments. This measurement was performed on a workstation containing an Intel i7-6700 CPU, 64GB RAM, RTX 2080 Ti GPU, and Windows 10 OS. The complete dataset consisted of 10.15 terabytes and was streamed from a 12 terabyte capacity SATA 7.2K RPM hard drive.

Upon initial loading of the dataset, 2023.7 megabytes was consumed and it took 6.8 s. For data exploration in the 2D multi-axis view, the average rendering performance was 5.69 frames per second and an average of 162.3 megabytes of data was cached in memory. For representative cell nucleus exploration, a total of 407.3 megabytes were consumed to

create 64 visualizations of cell nuclei in 3D and this took 58.9 s (0.9 s for each object). Subregion tree generation for anatomical region analysis required 1447.7 megabytes and 253.8 s for the brain region, 6.9 megabytes and 19.8 s for the habenula region, and 347.2 megabytes and 57.5 s for the optic tectum neuropil region. Filtering the subregion tree using the cut-plane operation consumed 1444.5 megabytes and 21.5 s for removing the spinal cord region. For partitioning the brain into separate sides, 1256.2 megabytes and 56.5 s were consumed for creating the left side region, while 639.9 megabytes and 49.6 s were used for the right side. More time is required to create a subregion tree for the first time than a subsequent subregion filtering task because the former requires data blocks to be referenced from disk. However, because this operation is performed through multi-threading, the user can proceed with other analyses while these tasks are being completed.

IV. CONCLUSION

In this study, we introduced a visual analysis system for studying cell nuclei across the entire larval zebrafish brain. Unlike existing tools for analyzing ssEM datasets, which mainly focus on connectivity analysis, our system is designed to examine the distribution and shape of brain cell nuclei. To achieve this goal, our system provides interactive visualization of cell nucleus morphology and cell nucleus spatial distributions with a unified workflow management interface for efficient data processing. Our experiments with the terabyte-scale larval zebrafish dataset showed that our system can effectively analyze cell nucleus morphology and distribution. While we focused on only a few anatomical regions here, we expect that several other regions can be studied in future work. Furthermore, we considered all cell nuclei in the brain, but future studies could examine specific cell types such as neurons, glia, or progenitors. We believe that this tool serves as a starting point for analyzing the wealth of additional information available through ssEM datasets.

The ZeVis system uses automatic cell nucleus segmentation results and subregion masks as input data in addition to the ssEM images. Such data pre-processing is usually time-consuming, which can be a bottleneck in the analysis workflow. In the future, a more efficient analysis might be possible by producing the necessary segmentation and masks interactively in the system (e.g., applying segmentation only to the region of interest at a specific zoom level in real-time).

REFERENCES

- [1] J. W. Lichtman and W. Denk, "The big and the small: Challenges of imaging the brain's circuits," *Science*, vol. 334, no. 6056, pp. 618–623, Nov. 2011.
- [2] M. Helmstaedter, "Connectomics at cellular precision," *e-Neuroforum*, vol. 7, no. 3, pp. 45–47, Sep. 2016.
- [3] S. Saalfeld, A. Cardona, V. Hartenstein, and P. Tomancak, "CATMAID: Collaborative annotation toolkit for massive amounts of image data," *Bioinformatics*, vol. 25, no. 15, pp. 1984–1986, Aug. 2009.
- [4] J. Beyer, A. Al-Awami, N. Kasthuri, J. W. Lichtman, H. Pfister, and M. Hadwiger, "ConnectomeExplorer: Query-guided visual analysis of large volumetric neuroscience data," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 12, pp. 2868–2877, Dec. 2013.

- [5] A. K. Al-Awami, J. Beyer, H. Strobel, N. Kasthuri, J. W. Lichtman, H. Pfister, and M. Hadwiger, "NeuroLines: A subway map metaphor for visualizing nanoscale neuronal connectivity," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 12, pp. 2369–2378, Dec. 2014.
- [6] D. Haehn, J. Hoffer, B. Matejek, A. Suissa-Peleg, A. Al-Awami, L. Kamensky, F. Gonda, E. Meng, W. Zhang, R. Schalek, A. Wilson, T. Parag, J. Beyer, V. Kaynig, T. Jones, J. Tompkin, M. Hadwiger, J. Lichtman, and H. Pfister, "Scalable interactive visualization for connectomics," *Informatics*, vol. 4, no. 3, p. 29, Aug. 2017.
- [7] K. M. Boergens, M. Berning, T. Bocklisch, D. Bräunlein, F. Drawitsch, J. Frohnhofen, T. Herold, P. Otto, N. Rzepka, T. Werkmeister, D. Werner, G. Wiese, H. Wissler, and M. Helmstaedter, "WebKnossos: Efficient online 3D data annotation for connectomics," *Nature Methods*, vol. 14, no. 7, pp. 691–694, Jul. 2017.
- [8] D. G. C. Hildebrand et al., "Whole-brain serial-section electron microscopy in larval zebrafish," *Nature*, vol. 545, no. 7654, pp. 345–349, 2017.
- [9] F. Drawitsch, A. Karimi, K. M. Boergens, and M. Helmstaedter, "FluoEM, virtual labeling of axons in three-dimensional electron microscopy data for long-range connectomics," *eLife*, vol. 7, Aug. 2018, Art. no. e38976.
- [10] A. Motta, M. Berning, K. M. Boergens, B. Staffler, M. Beining, S. Loomba, P. Hennig, H. Wissler, and M. Helmstaedter, "Dense connectomic reconstruction in layer 4 of the somatosensory cortex," *Science*, vol. 366, no. 6469, Nov. 2019, Art. no. eaay3134.
- [11] C. Eschbach and M. Zlatic, "Useful road maps: Studying *Drosophila* larva's central nervous system with the help of connectomics," *Current Opinion Neurobiol.*, vol. 65, pp. 129–137, Dec. 2020.
- [12] R. W. Friedrich and A. A. Wanner, "Dense circuit reconstruction to understand neuronal computation: Focus on zebrafish," *Annu. Rev. Neurosci.*, vol. 44, no. 1, Jul. 2021.
- [13] D. G. C. Hildebrand, "Whole-brain functional and structural examination in larval zebrafish," Ph.D. dissertation, Graduate School Arts Sci., Harvard Univ., Cambridge, MA, USA, 2015.
- [14] G. Lehmann and D. Legland, "Efficient N-dimensional surface estimation using Crofton formula and run-length encoding," *Insight J.*, 2012.
- [15] D. Meagher, "Geometric modeling using Octree encoding," *Comput. Graph. Image Process.*, vol. 19, no. 2, pp. 129–147, Jun. 1982.
- [16] T. B. Sousa, "Dataflow programming concept, languages and applications," in *Proc. Doctoral Symp. Informat. Eng.*, vol. 130, Nov. 2012, pp. 1–12.
- [17] M. Carkci, *Dataflow and Reactive Programming Systems*. Scotts Valley, CA, USA: Create Space Independent Publishing Platform, 2014.
- [18] K. J. Hayworth, J. L. Morgan, R. Schalek, D. R. Berger, D. G. C. Hildebrand, and J. W. Lichtman, "Imaging ATUM ultrathin section libraries with WaferMapper: A multi-scale approach to EM reconstruction of neural circuits," *Frontiers Neural Circuits*, vol. 8, p. 68, Jun. 2014.
- [19] A. W. Wetzel, J. Bakal, M. Dittrich, D. G. C. Hildebrand, J. L. Morgan, and J. W. Lichtman, "Registering large volume serial-section electron microscopy image sets for neural circuit reconstruction using FFT signal whitening," in *Proc. IEEE Appl. Imag. Pattern Recognit. Workshop (AIPR)*, Oct. 2016, pp. 1–10.
- [20] D. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Deep neural networks segment neuronal membranes in electron microscopy images," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2843–2851.
- [21] H. Chen, X. Qi, J. Cheng, and P. Heng, "Deep contextual networks for neuronal structure segmentation," in *Proc. AAAI*, 2016, pp. 1167–1173.
- [22] T. M. Quan, D. G. C. Hildebrand, and W.-K. Jeong, "FusionNet: A deep fully residual convolutional neural network for image segmentation in connectomics," *Frontiers Comput. Sci.*, vol. 3, p. 34, May 2021.



JUNYOUNG CHOI received the B.S. degree from the Ulsan National Institute of Science and Technology (UNIST), South Korea, in 2017, where he is currently pursuing the Ph.D. degree with the School of Electrical and Computer Engineering. His research interests include visualization, machine learning, and extended reality.



DAVID GRANT COLBURN HILDEBRAND received the B.S. degree in mathematical biochemistry from Northeastern University and the Ph.D. degree in neurobiology from the Harvard Medical School. He is currently a Postdoctoral Fellow with the Laboratory of Neural Systems, The Rockefeller University, where he is studying social perception in marmoset monkeys. His graduate work largely focused on applying serial-section electron microscopy or connectomics techniques to examine neuronal connectivity across the larval zebrafish brain.



TRAN ANH TUAN received the B.S. degree in computer science from the Ulsan National Institute of Science and Technology (UNIST), Ulsan, South Korea. He is currently pursuing the M.Sc. degree in visual computing with Saarland University, Saarbrücken, Germany. From 2017 to 2020, he was a Research Assistant with the High Performance Visual Computing Laboratory, UNIST. His research interests include the application of deep reinforcement learning in medical imaging, object tracking, object segmentation, realistic image synthesis, and volume rendering.



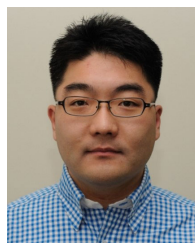
JUNGMIN MOON received the B.S. degree in computer engineering from Kyungpook National University, in 2016, and the M.S. degree in computer science, focused on visual analytics, from UNIST, in 2019. He is currently a Graphics Engineer with NCSOFT working on visualizing game performance and optimizing graphics. His research interests include graphics optimization, visual analytics, and high-performance visual computing.



SUNGAHN KO received the Ph.D. degree in electrical and computer engineering from Purdue University, in 2014. He is currently an Associate Professor with the School of Computer Science and Engineering, Ulsan National Institute of Science and Technology (UNIST), Ulsan, South Korea. His research interests include visual analytics, information visualization, and human-computer interaction.



TRAN MINH QUAN received the B.S. degree in electrical engineering from KAIST, South Korea, in 2012, and the Ph.D. degree in computer science, focused on GPU computing, from UNIST, South Korea, in 2019. He is currently an Affiliated Lecturer with VinUniversity and also a member of VinGroup, the largest enterprise in Vietnam by capitalization. His research interests include image-to-image translation, inverse problems, and high-performance visual computing, specialized for bio-medical image processing and applications.



WON-KI JEONG (Member, IEEE) received the Ph.D. degree in computer science from The University of Utah, in 2008. He was a member of the Scientific Computing and Imaging (SCI) Institute. He was an Assistant Professor and an Associate Professor with the School of Electrical and Computer Engineering, Ulsan National Institute of Science and Technology, from 2011 to 2020, a Visiting Associate Professor of neurobiology with the Harvard Medical School, from 2017 to 2018, and a Research Scientist with the Center for Brain Science, Harvard University, from 2008 to 2011. He is currently a Professor with the Department of Computer Science and Engineering, Korea University. His research interests include visualization, image processing, and parallel computing.

...