Master's Thesis

A visual analytics system for neuronal morphology analysis of zebrafish brain

Jung Min Moon

Department of Computer Science and Engineering

Graduate School of UNIST

2019

A visual analytics system for neuronal morphology analysis of zebrafish brain

Jung Min Moon

Department of Computer Science and Engineering

Graduate School of UNIST

ULSAN NATIONAL INSTITUTE OF SCIENCE AND TECHNOLOGY

A visual analytics system for neuronal morphology analysis of zebrafish brain

A thesis/dissertation
submitted to the Graduate School of UNIST
in partial fulfillment of the
requirements for the degree of
Master of Science

Jung-Min Moon

01. 11. 2018 of submission
Approved by

Advisor
Won-Ki Jeong

ULSAN NATIONAL INSTITUTE OF SCIENCE AND TECHNOLOGY

A visual analytics system for neuronal morphology analysis of zebrafish brain

Jung-Min Moon

This certifies that the thesis/dissertation of Jung-Min Moon is approved.

01.11.2018 of submission

signature

Advisor: Prof. Won-Ki Jeong

signature

Committee Member : Prof. SungAhn Ko

signature

Committee Member : Prof. Jae-Young Sim

Abstract

The electron microscope is capable of capturing tissue in three dimensions up to the nanometer scale. This makes it possible to analyze the shape of brain cells beyond conventional analysis throughout the brain region. Morphological analysis of brain regions requires tracking the morphological features of each cell and applying techniques for visualizing electronic microscope (EM) volume data and brain cells at the same time, as well as brain region information such as diencephalon and mesencephalon. However, morphological analysis of cells in the brain region has been challenging. Because terabytes of EM volume data must be processed, brain cell segmentation and accurate brain subregion information down to 120 nm are required. In this thesis, I present a novel system for visualizing and analyzing EM volume data and extracted brain cell regions. The system shows an EM volume data view and a histogram of the morphological features, and it provides a graph-based user interface. This system classifies the whole brain cell cluster into various brain cell selection methods according to the type and expresses statistical information of the selected cell cluster. This allows the user to mask only the desired brain regions or desired brain cells and to perform a morphological analysis of the brain cell population. Using this system, users can quickly analyze the brain cell type and the role of the group by using the user interface. By using this system, I can know the shape of the brain cell by brain area and analyze brain cell morphology and function by known brain area function. I also propose an efficient data structure of terabytes of data to construct a system. I performed several case studies, including zebrafish whole brain cells analysis, diencephalon and habenula analysis, mesencephalon tectum neuropil analysis, and optic nerve analysis with brain expert using 120 nm/px isotropic zebrafish EM data.

Contents

List of Figures

I Introduction

1.1 Motivation

In recent years, data have increased exponentially in various fields, and the size and diversity of data have gone beyond terabytes to petabytes. Typical factors for this phenomenon are the development of better measurement methods and measurement equipment. Recently, serial section electronic microscopy (ssEM) produced three-dimensional data of isotropic terabytes on a nanometer scale. These data focus on the high-resolution representation of biological specimens using three-dimensional (3D) microscopy techniques. Such high-resolution biology samples can be used for a variety of studies. In recent research [2], connectome studies were carried out using these data. This is a study of the distribution and shape of nerve cells using the entire zebrafish brain through high-resolution ssEM. This analysis requires a high-throughput application. In addition, large amounts of data should be simultaneously analyzed and compared through several visualization methods, enabling effective research. The cell nucleus of Zebrafish Brain count is more than 100,000, and automated tracking is required to track all of them. Moreover, it has been difficult to trace what type of brain cells are located in a specific region from a large amount of data and in which region the brain cells of a specific morphology are distributed.

To solve these problems, we need efficient analysis methods such as visual analytics. With recent developments in cognitive science, information systems, graphic design, interaction design, and mathematical-statistical methodologies, users can quickly and easily discover patterns hidden in large amounts of data. Such patterns are being studied in several visualization groups. Because visualization techniques are used in various fields, there are various kinds of visualization techniques. Visualization includes scientific visualization, information visualization, infographics, creative visualization, and data visualization. Visual analytics is visualization of data, which allows the analyst to find patterns, differentiation, and outliers of the data through visualized data. Through this process, the analyst can find out the insights inherent in the data, use the analysis results to quickly understand what to do, and access large amounts of data.

This thesis presents a theory on how to track and analyze small areas (cells) in high-resolution three-dimensional data. I effectively visualize the nucleus in biological high-resolution three-dimensional data and propose an analysis method that utilizes features of the zone. It also measures and accurately filters cell nucleus type. This thesis includes the preprocessing of high-resolution data and the visual analytics model. It also applies the model to actual connector research to demonstrate its usefulness and to provide new insights.

1.2 Background

Connectomics

Existing connectome research has focused on connectivity and has been trying many functional studies in anatomical areas. In this study, the researcher has focused on creating new insights for brain scientists by utilizing fMRI data, visualizing connectivity with brain visualization, and showing functional areas related to brain regions or activation regions according to biological conditions. This process involves comparing various fMRI data and finding insights on a very complex network. In [3], we compared different data efficiently by using group visualization and topology visualization techniques. In [4], we studied visualization in the AR/VR domain using fMRI data. In the field of visualization, it is research that overcomes the complex network of the brain with effective visualization. There are many approaches to researching connectomes. The case study of this theory is based on the analysis of serial section EM segmentation data and [2] zebrafish whole brain analysis, one of the first goals of which is analysis of the nucleus morphology in high resolution. This process requires an automated nucleus segmentation process such as FusionNet [5]. On the ssEM, the brain nucleus is dense and complex in a narrow area of more than 100,000. Because of this problem, obtaining the nucleus analysis of the whole brain is challenging. However, through learning a deep-running image like CNN, we can get a probability map that a nucleus exists. This probability map can be thresholded and connected components can be computed to produce separate nucleus regions.

Visual analytics

The visual analytics process is beneficial for effectively navigating and understanding high-dimensional, high-volume data. Visualization analysis is not just about visualization; it also provides a process for users to infer data using a user interface.

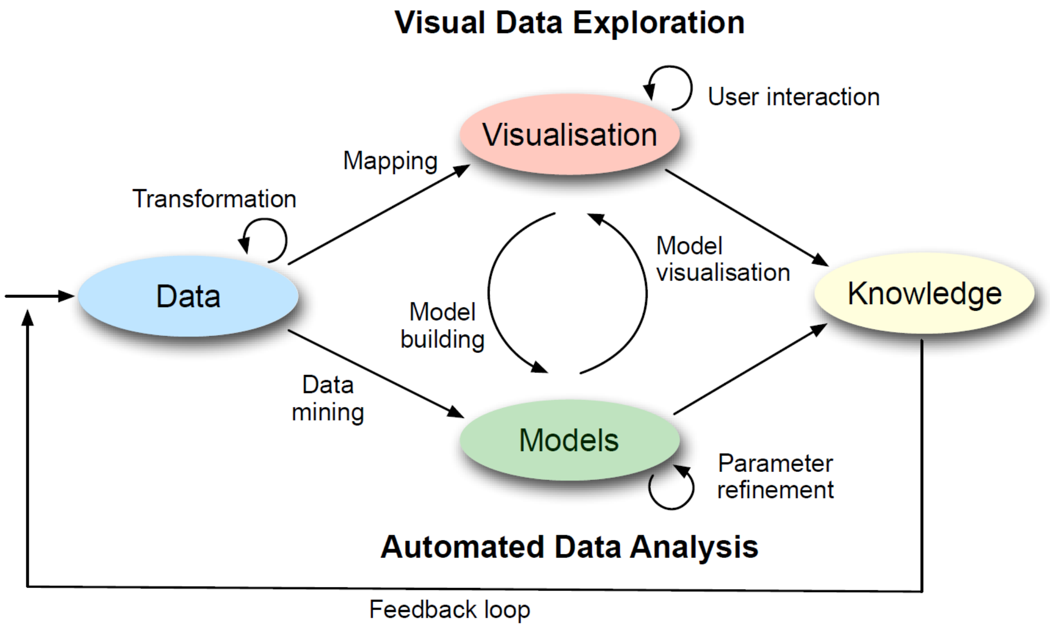


Figure 1: Visual analytics process [1]

According to [6], visual analytics is a kind of interactive approach in which users visualize data and infer data through various model visualizations. The process of making new biological discoveries using high-resolution biology data should be tried in as many different ways as possible. In addition, the attempt should be intuitive to give feedback to the user. These searches fit well with the visual analytics process.

Larval zebrafish EM data

The larval zebrafish serial-section EM data analyzed here were captured from a 5.5 days post-fertilization larval zebrafish. This specimen was cut into 18,000 serial sections and collected onto a tape substrate with an ATUM [7]. A series of images spanning the anterior quarter of the larval zebrafish was next acquired at a nearly isotropic resolution of 56.4×56.4×∼60 nm vx^{-1} from 16,000 sections in the resulting serial section library using scanning EM. All image planes were then co-registered into a three-dimensional volume with an FFT signal whitening approach [8].

For training, two small sub-region dataset crops were extracted from a near-final iteration of the full volume alignment to avoid ever deploying the segmentation on the training data. Two volumes were chosen to train on different features in the dataset. One volume was 512×512×512 and the other was 512×512×256. The features of interest, neuronal nuclei, were manually segmented as area-lists in each training volume using the Fiji image processing package [9] with TrakEM2 plug-in [10]. These area-lists were exported as binary masks used in the training procedure. For accuracy assessments, an additional 512×512×512 sub-volume was manually segmented (test dataset).

Cell Segmentation

Electron microscopic connectomics is an ambitious research direction with the goal of studying comprehensive brain connectivity maps by using high-throughput, nanoscale microscopy. One of the main challenges in connectomics research is developing scalable image analysis algorithms that require minimal user intervention. Recently, deep learning has drawn much attention in computer vision because of its exceptional performance in image classification tasks. For this reason, its application to connectomic analyses holds great promise. We used a novel deep neural network architecture, FusionNet, for the automatic segmentation of neuronal structures in the connectomics data. FusionNet leverages the latest advances in machine learning, such as semantic segmentation and residual neural networks, with the novel introduction of summation-based skip connections to allow a much deeper network architecture for a more accurate segmentation. We deployed the trained network to the complete set of 2000 sections of the larval zebrafish brain imaged at 112.8×112.8×∼120 nm vx^{-1} resolution, which is around 150 megabytes in raw data size.

1.3 Related Work

Visual anomaly detection

Anomaly detection has been studied in various fields. However, visually expressing an ideal region is difficult because the boundaries are ambiguous and the process of obtaining such data is complicated. In addition, detecting anomalies based on data that changes over time is difficult, mostly regarding visual representation. This anomaly detection may lead the user to make erroneous judgments. In actual traffic or urban areas, in studies such as those by [11], [12], [13], and [14], features are extracted statistically through multidimensional data that is updated in real time and anomaly detection is performed. In other areas, much research such as that by [15] is being done to detect anomalies such as network traffic. In the biomedical field, such as our research field, application studies such as that by [16] can analyze the size and shape of organs like and analyze the degree of recovery or deterioration of the disease.

Unlike the studies mentioned above, the current system focuses on the process of user inferring data more interactively. In the process, abnormal cells or groups are visualized, and the user can modify or maintain the analysis pipeline.

Data flow system

This study is a visual programming application that takes the morphological feature of the brain nucleus of zebrafish as input and makes a subset of this data using various group creation operations. Much research has been done on visual programming through data flow in the visualization field. There are also many approaches to comparing and analyzing groups at the same time, such as in [17] and [18]. This approach has been studied for a long time and enables extensive data analysis of tabular data. In the biological field, it is important to visualize the whole data in a batch, but it is also necessary to analyze the region of interest. From this point of view, it is important to focus on what biologists need as you create a subset of data, such as in [19]. This process enables comparison between different groups. In [20], the authors conducted research using this approach in the area of biology, where connectivity is more important. The above studies and approaches are the same. However, the current study provides data-based operations such as anatomical subregions in domain-specific fields.

1.4 Problem definition and contribution

The analysis of the brain nucleus through conventional ssEM has been limited for several reasons:

- Morphological features of whole brain cells could not be extracted. To generate this data, more than 100,000 cells had to be identified manually, and previous studies were only possible for small areas within the brain.

- Also, existing connectome studies have focused on brain anatomical inter-domain connectivity or function.

Such an attempt was made to visualize a small region through ssEM image and mesh rendering in applications such as CATMAID, but the function was limited because it was not for nucleus-type analysis.

Therefore, this study proposes an interactive visual programming method using an automated nucleus tracking system and dataflow. This system attempts to analyze the nucleus of the whole brain based on ssEM, nucleus segmentation, and anatomical subregion data. The case study also evaluates the usefulness of the system.

1.5 Task Description and Requirement

The system described herein is designed to provide brain nucleus analysis through actual ssEM. Its purpose is to visualize cell segmentation results and ssEM together. Specific requirements have been formulated in close collaboration with domain experts with expertise in brain science data analysis. Regular meetings were held for about a year, detailed system design was discussed, and the prototype system was demonstrated to experts for feedback. Three prototypes were built and the improvements were repeated. Below we describe the goals of this research(G1, G2) and four requirements (R1–R4).

G1 Biologists will use ssEM data to analyze the morphology of the nucleus at a glance. Also, a zebrafish's brain will be divided anatomically into several regions. They would like to know if there are differences in the shape or distribution of the brain nuclei in these areas.

G2 In contrast to **G1**, they want to know the location of cells with a specific morphology in ssEM space.

We developed this application for the above goals. Based on these goals, we collaborated with experts to define specific requirements. The essential requirements for analyzing brain nuclei are as follows:

R1 **Various visualization**: The original ssEM data and segmentation results are XY image sequences. Preprocessing is required to show these data through various visualization methods. An optimized rendering pipeline for fast visualization with high-resolution is also required.

R2 **Cell selection**: Select cells in a variety of ways and provide statistics on them. There may be a selection method through the shape of the cells and a cell selection method through anatomical regions. The brain of a zebrafish has anatomical subregions such as habenula and mesencephalon. The proposed system must be able to select and isolate cell data through this subregion mask. This feature allows biologists to compare and analyze cells in the desired areas.

R3 Comparison between groups: The distribution of selected cell groups can be confirmed and intergroup comparisons should be possible. In biological data, the process of analysis is carried out by making rapid and repeated comparisons and inferring phenomena. Therefore, it is important to be able to distinguish the differences visually.

R4 Analytical process of selecting cells: The process of analyzing morphology through brain nucleus segmentation requires various features and various statistical approaches. Therefore, the analysis process should be able to try various methods and be logical. So, we need a more flexible system. The cells of interest should be easy to understand. Also, analysis pipeline comparisons are required.

Based on these requirements, we have developed an application to analyze brain cell nucleus morphology within ssEM. It is an interactive system that allows users to generate various cell groups within a project to analyze and infer data. The system includes data preprocessing modules, visualization modules, and analysis modules.

In the data preprocessing module, ssEM volume, cell segmentation volume, and anatomical subregion volume are converted into multi-resolution data blocks, and labels are given to each cell. At the same time, this module computes the morphological feature of the cells. The entire data is terabyte-scaled and processed through a distributed system to accelerate this process. This module is a preprocessing process for securing various visualization methods and for selecting cells through the morphological features in cell selection. This module indirectly satisfies **R1** and **R2**.

Visualization modules include XY-view, YZ-view, ZX-view, and volume-view. The center coordinates of the visualization through each axis are shared. With one cell in the center, the user can look at the cells in the three viewports. It also allows users to identify groups of cells easily selected via the volume-view easily. This module has been developed to meet **R1** and **R3**.

Through the graphical interface, the analysis module roughly identifies the cells selected by the user and creates data groups. In this module, cell groups can be created through thresholding in the histogram through the morphological feature of cells, or cells can be selected by adding anatomical subregions. These groups of cells can be joined or subtract via set-operation. This analysis model satisfies **R2** and **R4**.

1.6 Thesis Overview

Integrating the entire zebrafish brain nucleus with the visual analytics model is not easy and requires many engineering strategies. First, you need a way to find valid data in Big Data, a way to efficiently visualize data in 3D space, and a pipeline that can be easily analyzed intuitively by non-experts. What is needed, therefore, is high-speed preprocessing of terabyte-size datasets, visualization strategies for ssEM data, and compelling visual analytics design. The remaining portions of this thesis are structured as follows. In chapter 2, I introduce these data preprocessing, visualization strategies, and analytics design. Chapter 3 shows the results of applying these methods to actual case studies. Contributions to the thesis are as follows:

- Ingest large datasets(ssEM, nucleus segmentation, anatomical subregion volume) that apply high-performance analytics.
- Intuitive, usable, effective visualization suitable for domain experts.
- Intuitive workflow pipeline.

II Application

2.1 Architecture and Overview

This research is a brain cell analysis system that utilizes a zebrafish's ssEM data and the segmentation volume of its data. The system converts to a multi-resolution data structure and computes the morphological features of each cell. After that, I visualized them together and analyzed them. This application is Windows-based with Qt and includes OpenGL, CUDA-based visualization, and computation acceleration modules.

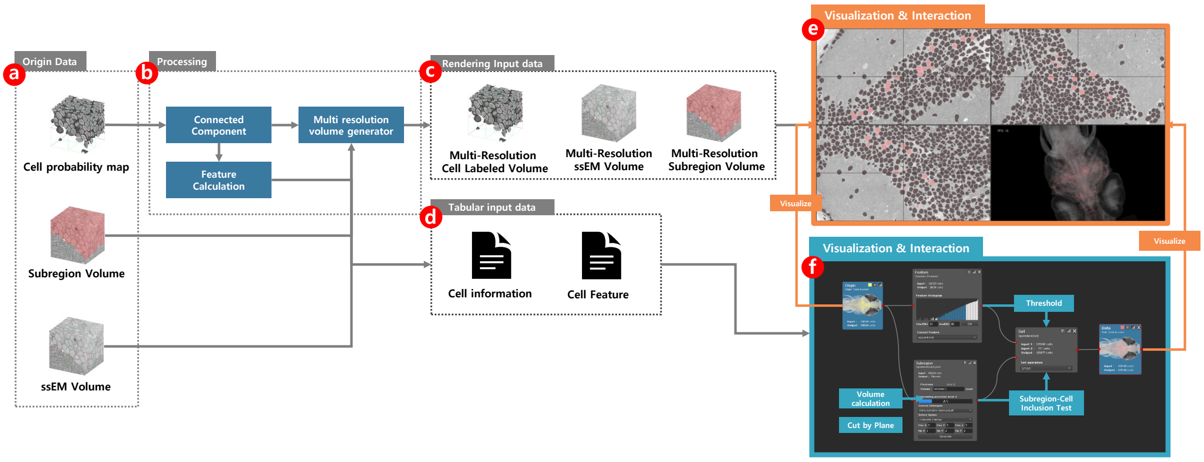


Figure 2: Overview of system: (a) original datasets, (b) preprocessing modules, (c) rendering input datasets, (d) analysis input datasets, (e)multi-axis view and volume view, and (f) graphical interface for analysis

2.2 Data Extraction and Preprocessing

Data structure

The structure of the data in this analysis process is critical because it determines the speed and quality of visualization and the flexibility of the analysis process. Mostly, for this analysis, em volumes, cell volumes, and subregion volumes must be dynamically loaded and rendered. Also, to render simultaneously from different angles of requirements, it is necessary to make the data into three forms of XY, YZ, ZX. The data used in this research is 5120 x 4608 x 8000 (px). The entire data is managed in blocks of 512 x 512 x 512 (px), and when visualized, 512 x 512 image tiles are loaded. Also create all blocks for XY, YZ and ZX planes for more efficient rendering. Each plane is managed with 4 levels of resolution. Each level goes down to 1/2 size. The depth in each axis does not downsample.

The structure of the data in this analysis process is critical because it determines the speed and quality of visualization and the flexibility of the analysis process. Mostly, for this analysis, em volumes, cell volumes, and subregion volumes must be dynamically loaded and rendered.

Also, to render simultaneously from different angles of requirements, it is necessary to render the data into three forms of XY, YZ, and ZX. So, the file structure I propose is as follows Fig. 3:

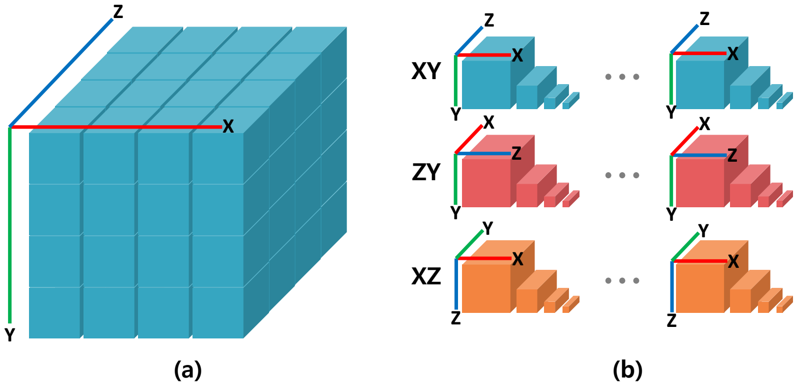


Figure 3: (a) The data used in this research is 5120 x 4608 x 8000 (px). The entire data is managed in blocks of 512 x 512 x 512 (px), and when visualized, 512 x 512 image tiles are loaded. (b) We also created all blocks for the XY, YZ, and ZX planes for more efficient rendering. Each plane was managed with four levels of resolution. Each level went down to half-size. The depth in each axis did not downsample.

This structure makes the data size larger than the initial data, but concentrates on IO speed in the application and makes it possible to do an efficient analysis. If there is data in three directions, it is not necessary to rotate the block in the YZ or ZX directions when loading data, and thus unnecessary operations can be reduced. By utilizing this structure, we avoided the problem of dropping the frame rate with the IO bottleneck when we continued to move and load the data in the analysis process. Subregion volume and cell volume are then calculated by sequentially calculating from level 4 to level 0 when calculating the cells included in the volume measurement or subregion. This allows the user to quickly know the approximate value and continue to update it in the future to improve accuracy.

All processes were created quickly in a cluster environment, as shown Fig. 4. Afterwards, the cell, subregion volume was not a required block for the entire volume. Because only the part where each cell or mask was present would be used, all the empty blocks were removed.

Distributed connected components

The result of the cell segmentation was calculated as a probability map of the cells in space. To analyze the shape of each cell, we first needed to create a binary volume through thresholding for the entire cell volume. After this process, cell labeling was attempted on the entire volume. Because the entire data was in terabytes, it was difficult to do all the processing with one computing device. Therefore, I solved this problem as shown in Fig. 5. The entire volume data was split into clusters in the z direction, and each node computed only its own data. Each node processed the connected component algorithm, and each process performed depth first search

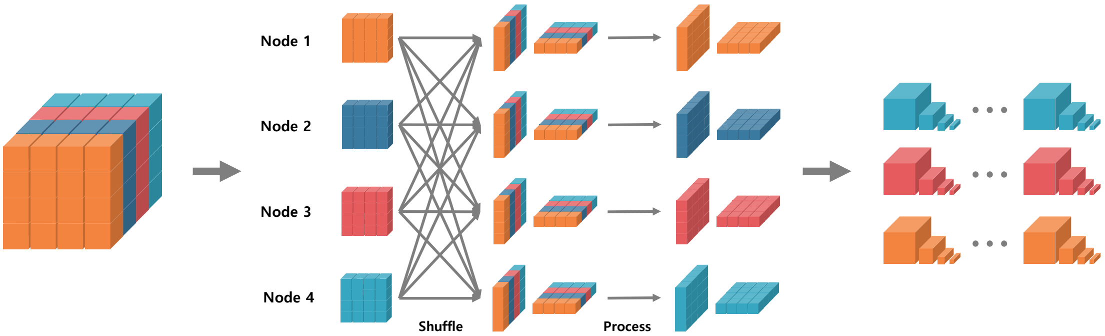


Figure 4: To create the entire data structure, proceed in a cluster environment. First, I divided the data in the z direction and placed it on each node; each node created an XY-multi-resolution block. Then, I shuffled 0-level blocks to each other. Each node that received the data generated the data in the YZ and ZX planes.

(DFS) on different cells. Then, to combine the results, I first obtained the cell label maximum of each node's output so that the output of each node did not use the same cell index. I then created a cell index tree for the cells at the boundary and updated the entire data.

Cell morphology feature

The cell morphology feature is calculated based on the bounding box of each cell in the cell label volume. The morphology features used in this research were volume, surface area, sphericity, eccentricity, intensity, and position. The volume and the area of each cell were calculated by counting the voxels in the cell volume data. Intensity is the intensity of the EM volume at each cell location. The position information uses the center x, y, z coordinates of the cell bounding box. Sphericity was calculated on the pre-computed volume and surface area. I then calculated the surface area of the sphere that the current volume can have and took the ratio of this value to the currently measured surface area. The closer this value is to 1, the more spherical. The closer to 0, the more the shape is distorted. Eccentricity was used to calculate how much the shape of the cell was deflected. I calculated the covariance matrix using the coordinates of all voxels in the cell and calculated the eccentricity using the eigenvalues, as shown in (2). through the semi-major axis and the semi-minor axis. the closer to 0, the closer to a circle.

$$Sphericity = \frac{S_i}{S_m} \tag{1}$$

where S_m is the measured surface area and S_i is the surface area of the ideal sphere of the currently measured volume.

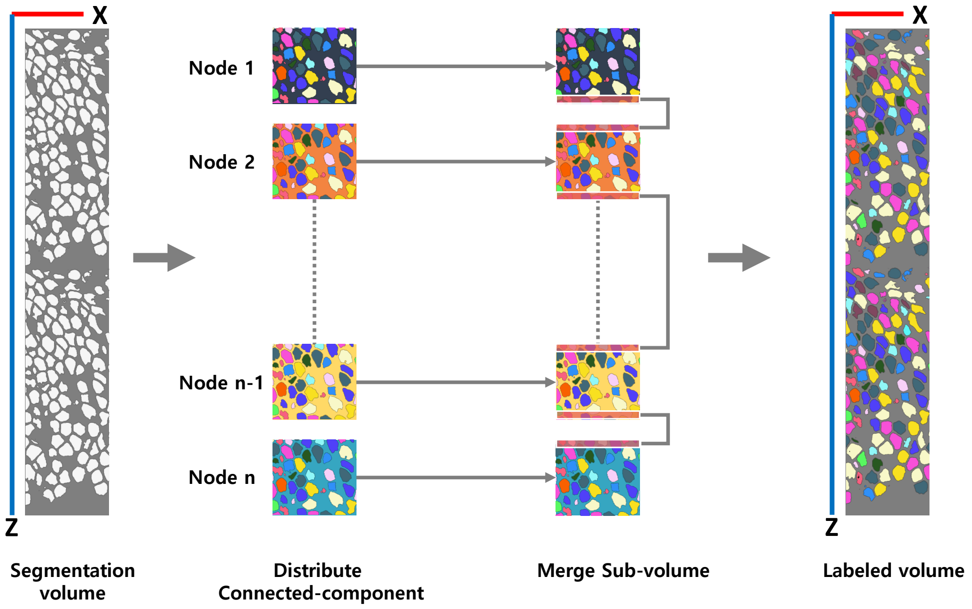


Figure 5: Distributed connected component pipelines in a clustered environment

$$Eccentricity = \sqrt{1 - \frac{{\lambda_2}^2}{{\lambda_1}^2}} \tag{2}$$

where λ_1 is semi-major axis length and λ_2 is semi-minor axis length

2.3 Visual Interface

Cell visualization

In this research, there are two main ways to visualize the brain nucleus of zebrafish. One method is to highlight the nucleus together with the EM image based on ssEM, visualize the entire ssEM volume by the volume-rendering method, and express the cell position by a point in 3D space.

Multi-axis visualization

The visualization showing the cross section on three coordinate systems is composed of three layers in total. Each layer renders ssEM, anatomical subregion, cell. This method of visualization is the most understandable way for biologists familiar with microscopic images. Also, because this visualization technique does not use a sampling method, biological data can be analyzed with the highest resolution. The data used in this rendering was preprocessed into a multi-resolution block and visualized using the currently visible resolution block. The cell layer and the subregion layer can be compared with the actual ssEM by adjusting the opacity independently. In particular, because the cell part was rendered on the EM, the intensity information of the

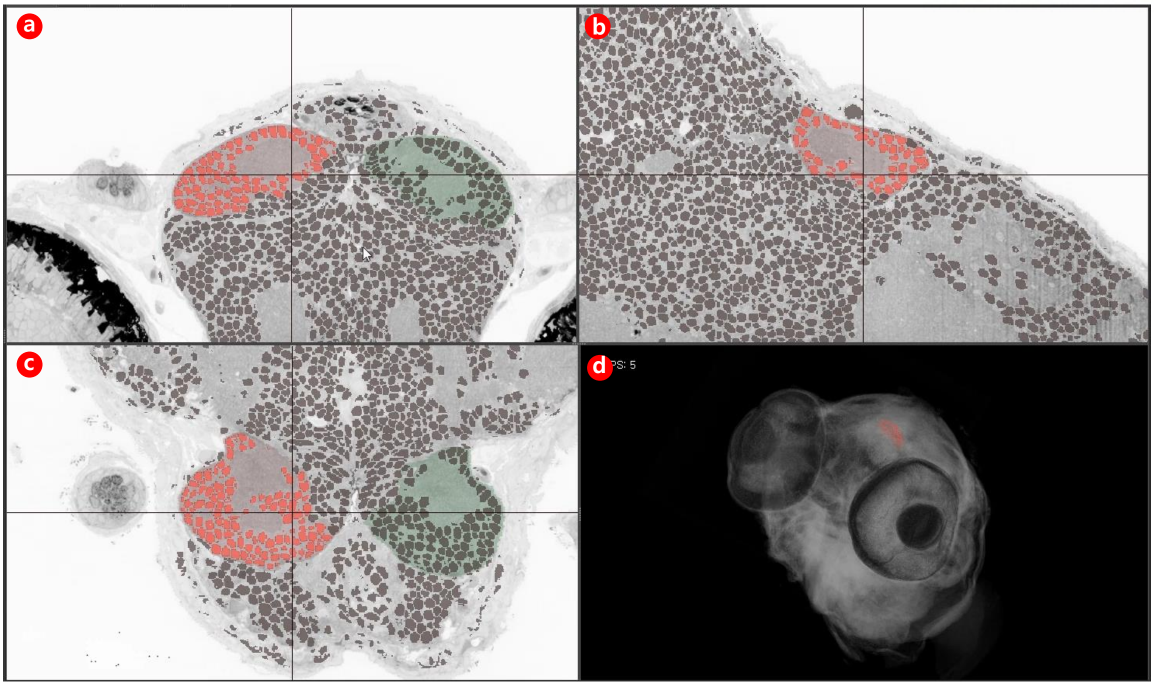


Figure 6: (a) The ssEM volume, cell volume, and anatomical subregion of the XY coordinate system are rendered together. (b), (c) show the YZ and ZX coordinate systems, respectively. (a), (b), and (c) show the same coordinates being rendered. (d) utilizes the entire volume to perform volume rendering, and points represent only the selected cell group.

microscope image cannot be seen. Because it hides the characteristics of the cell, it controls the opacity.

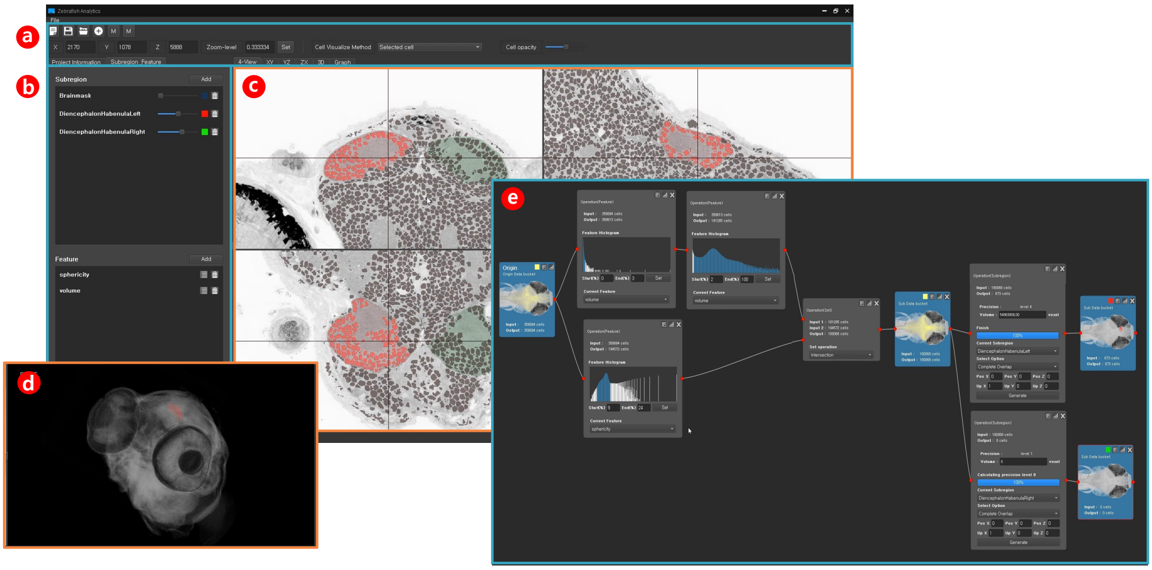
The user can also render each cell with a random color. This function can be used to verify that nearby cells are the same connected component. The morphology of the cells is ambiguous and can therefore be verified through this function.

Volume visualization

When the system visualizes only three coordinate system sections, the user will not be able to know at which coordinates the entire cell position or group of selected cells is located. Therefore, users can understand and analyze more quickly by checking the location of cells in 3D space. The data of this visualization generates the volume data by preloading the block with the lowest resolution when loading the analysis project. It also shows the zebrafish effectively using the predefined transfer function. When the data bucket is selected in the graph interface, the point of the cell position is created, and the user can visualize the desired cell group.

Graph interface

The analysis process is the primary analysis method in which the user creates the desired group of cells in the whole brain nucleus and confirms the morphological characteristics of the group. In



X 2170 Y 1078 Z 5888 Zoom-level 0.333334

Cell Visualize Method: Selected cell

Cell opacity

Subregion: Brainmask, DiencephalonHabenulaLeft, DiencephalonHabenulaRight

Feature: sphericity, volume

Origin — Input : 358894 cells, Output : 358894 cells

Operation(Feature) — Input : 358894 cells, Output : 358613 cells; Start(%) 0 End(%) 3; Current Feature: volume

Operation(Feature) — Input : 358613 cells, Output : 181285 cells; Start(%) 2 End(%) 100; Current Feature: volume

Operation(Feature) — Input : 358894 cells, Output : 194572 cells; Start(%) 9 End(%) 24; Current Feature: sphericity

Operation(Set) — Input 1 : 181285 cells, Input 2 : 194572 cells, Output : 180068 cells; Set operation: Intersection

Sub Data bucket — Input : 180068 cells, Output : 180068 cells

Operation(Subregion) — Input : 180068 cells, Output : 673 cells; Precision : level 4; Volume : 54863808.00 voxel; Finish 100%; Current Subregion: DiencephalonHabenulaLeft; Select Option: Complete Overlap; Pos X 0 Pos Y 0 Pos Z 0; Up X 1 Up Y 0 Up Z 0

Sub Data bucket — Input : 673 cells, Output : 673 cells

Operation(Subregion) — Input : 180068 cells, Output : 0 cells; Precision : level 1; Volume : 0 voxel; Calculating precision level 0 100%; Current Subregion: DiencephalonHabenulaRight; Select Option: Complete Overlap; Pos X 0 Pos Y 0 Pos Z 0; Up X 1 Up Y 0 Up Z 0

Sub Data bucket — Input : 0 cells, Output : 0 cells

Figure 7: Application layout, (a) Top toolbar: Load the analysis project or add a node for the graph interface. It also adjusts the current x, y, and z coordinates, and zoom level. The user can choose the rendering method of the cell or adjust the opacity. (b) Add an anatomical subregion and add morphological features. The user can also change the opacity of the subregion or the rendering color. (c) Cell visualization: This area can be changed through (e) and tab change. The user moves the tab and proceeds with the analysis. (d) Visualization of a cell using volume-rendering technique. (e) Graph interface: Add nodes in this area, connect, and analyze.

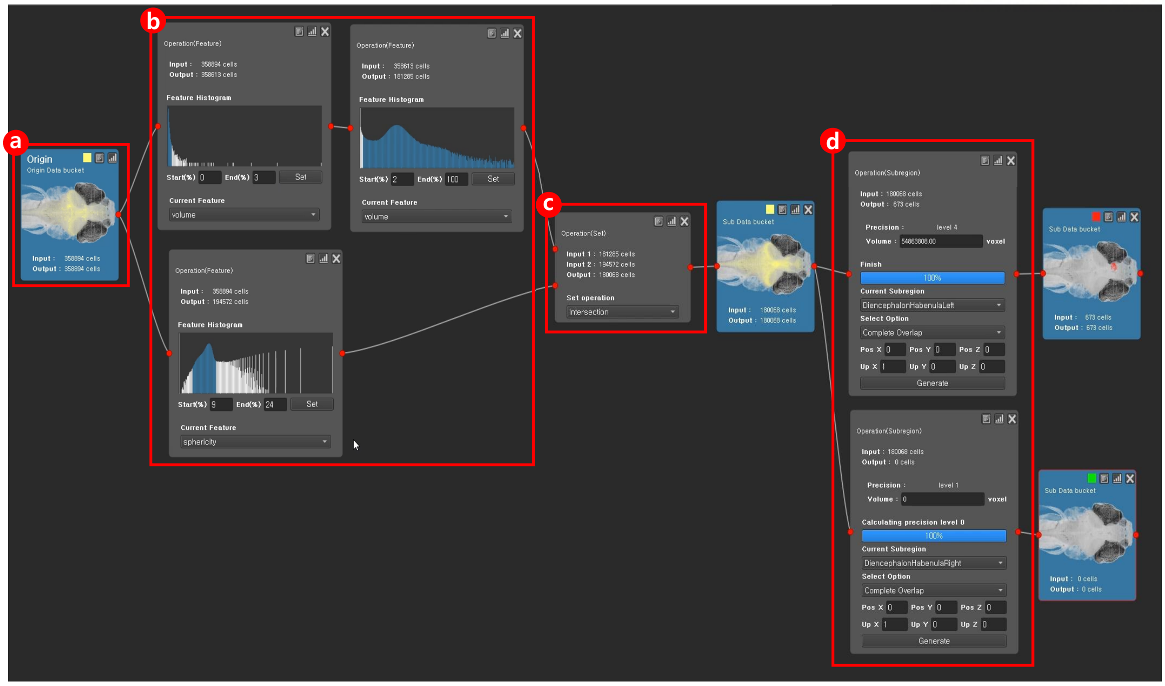
the analysis process, the user can check the selected cells at any time using the cell visualization technique. For more flexible cell selection methods, this study offers three selection methods and one data buffer.

This interface uses the tree structure. Nodes represent data and operations and edges represent this data flow. Modification of the parent node affects the child nodes in turn. This makes it easier for users to understand the data and manage the data flow.

All blocks also specify the number of input and output cells. The user can check the morphological feature statistic report of the current input group by using the statistic function. This function includes maximum, minimum, average, sum, variance, and five number summaries.

Data bucket

The data bucket at the beginning of the analysis represents the data of the entire nucleus generated by the cell segmentation. As the analysis proceeds, the nucleus contained in this block is displayed through the central XZ projection heatmap visualization. The user can customize the colors of the cells to be displayed in the heatmap. This visualization roughly represents the location of the cell and represents the number of cells in the cell group. Then the user creates a new data group by connecting to the necessary operations through the edge. The pipeline of the analysis process moves from left to right, making cell groups more precise selections. When the



Origin Data bucket — Input : 358894 cells, Output : 358894 cells

Operation(Feature) — Input : 358894 cells, Output : 358613 cells, Start(%) 0, End(%) 3, Current Feature: volume

Operation(Feature) — Input : 358613 cells, Output : 181285 cells, Start(%) 2, End(%) 100, Current Feature: volume

Operation(Feature) — Input : 358894 cells, Output : 194572 cells, Start(%) 9, End(%) 24, Current Feature: sphericity

Operation(Set) — Input 1 : 181285 cells, Input 2 : 194572 cells, Output : 180068 cells, Set operation: Intersection

Sub Data bucket — Input : 180068 cells, Output : 180068 cells

Operation(Subregion) — Input : 180068 cells, Output : 673 cells, Precision : level 4, Volume : 54863808,00 voxel, Finish 100%, Current Subregion: DiencephalonHabenulaLeft, Select Option: Complete Overlap, Pos X 0, Pos Y 0, Pos Z 0, Up X 1, Up Y 0, Up Z 0

Sub Data bucket — Input : 673 cells, Output : 673 cells

Operation(Subregion) — Input : 180068 cells, Output : 0 cells, Precision : level 1, Volume : 0 voxel, Calculating precision level 0 100%, Current Subregion: DiencephalonHabenulaRight, Select Option: Complete Overlap, Pos X 0, Pos Y 0, Pos Z 0, Up X 1, Up Y 0, Up Z 0

Sub Data bucket — Input : 0 cells, Output : 0 cells

Figure 8: (a) Data bucket, (b) feature operation, (c) set operation, and (d) Subregion operation

data bucket is selected, the cell visualization tab shows the cell group of the currently selected block.

Feature operation

Using the morphological feature added in (b) of 7, the histogram of the input cell group can be confirmed. This histogram also allows the user to select the desired data by dragging the mouse. The selected cell group is the output of this operation. By default, the selection in the histogram can create a period of 0-100%. Therefore, if the data is densely populated, the user can overlay this operation node to view the data. This operation node can be used to select data, but it is also easy to use the histogram to determine the distribution of morphological features of the input data.

Subregion operation

The subregion operation selects cells through the subregion added in (b) of 7. In this process, the user can add plane information. This operation selects the cells contained in the subregion among the cells in the normal vector direction of the plane. It also calculates the volume of a subregion that is cut into planes. This volume can be used in biology research to compare the density of cellular bodies in anatomically divided regions.

This application offers three options for calculating cell inclusion. The user can select cells that are fully contained, cells that intersect a subregion, and cells that touch a subregion.

The cell inclusion test and volume calculation process update the results by counting from the block with the lowest resolution. This approach allows users to see rough results quickly. If the calculations are more accurate in the future, the user can see the accurate data.

Set operation

The set operation combines the two inputs to produce one output. This allows the user to create a cell group by combining a feature-based operation and a subregion-based operation. Th set operation makes the above two operations more robust. Feature operations threshold one-dimensional morphological features. By combining two feature operations through the set operation, thresholding of more than one dimension is possible. This operation works by selecting the union, intersect, and subtract options.

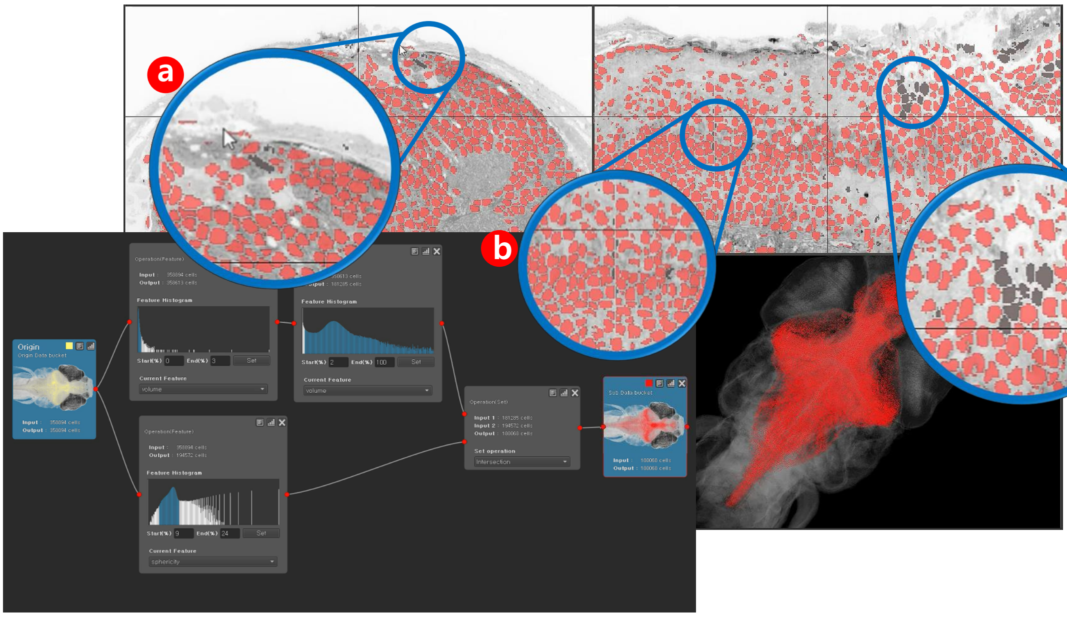
III Result

3.1 Case Study 1: Anomaly Detection

Accuracy is essential in brain science research through ssEM. There are also issues in detecting abnormal cells. However, segmentation has significant accuracy but requires a way to control the error. This application combines morphological features and thresholding to detect abnormal cells, which can then be removed from the analysis. The use cases below have been removed using volume and sphericity features. Overall, there were three types of errors: There was a very small nucleus in the volume, which was mostly segmentation noise. Merge errors caused by over-segmentation and errors owing to ssEM quality. Unlike normal nuclei, these errors were small or very large in volume. The shape will also be distorted. Volume with small cells, thresholding relatively large cells. In addition, high-sphericity cells were removed, and two groups of intersection cells were selected. The number of cells in the whole segmentation result was 358,894, but after eliminating the errors, the number of total nuclei was 180,068, which was about 49.83%, and normal cells were found.

3.2 Case Study 2: Symmetric Comparison

This use case compared the left and right symmetries of the brain. We compared the brain cell nucleus density and counts of the left and right brain of habenula and a whole zebrafish brain. We used data that minimized segmentation error through previous use cases before analysis. Analysis of the entire brain divided the left and right through the plane passing through the center of the brain and calculated the volume of the brain mask and the cells contained inside. The left and right habenula calculated the sum of the number of nuclei included and the volume of the nucleus each and calculated the density of each region by calculating the mask volume.



Origin
Origin Data bucket
Input : 358894 cells
Output : 358894 cells

Operation(Feature)
Input : 358894 cells
Output : 358613 cells
Feature Histogram
Start(%) 0 End(%) 3 Set
Current Feature
volume

358613 cells
Output : 181285 cells
Feature Histogram
Start(%) 2 End(%) 100 Set
Current Feature
volume

Operation(Feature)
Input : 358894 cells
Output : 194572 cells
Feature Histogram
Start(%) 9 End(%) 24 Set
Current Feature
sphericity

Operation(Set)
Input 1 : 181285 cells
Input 2 : 194572 cells
Output : 180068 cells
Set operation
Intersection

Sub Data bucket
Input : 180068 cells
Output : 180068 cells

Figure 9: a) The intensity of the outer edge of ssEM is high, and the contrast is very low. Therefore, the segmentation quality is not good. b) There is an error that the cells separate when there is an intermediate segmentation error slice. c) There is an error when cells are clumped owing to over-segmentation.

3.3 Case Study 3: Non-brain analysis

In this study, we tried to analyze not only the brain nucleus but also the cells of the Zebrafish eye. There are very low intensities around the eyes, and the shape is also different from that of a normal brain nucleus. Therefore, we analyzed the previous brain cell nucleus error control and another pipeline. In this process, we thresholded large cells with low volume and low intensity. After that, we classified the cells into left and right eyes through both eye masks and compared the number, volume, and density of cells in each eye.

	Count (cells)	Cell volume sum (voxel)	Mask volume (voxel)	Density (%)
LB	86,953	2,334,413,346	7,882,081,604	29.6167
RB	88,469	2,322,888,604	7,810,132,604	29.7420
WB	180,068	4,657,301,950	15,712,320,344	29.6411
LH	671	20,206,899	54,863,808	36.8310
RH	530	15,448,743	43,136,724	35.8134

Table 1: Compare properties of the region of cells.

LB(left sides of brain) RB(right sides of brain) WB(whole brain) LH(left sides of habenula) RH(right sides of habenula)

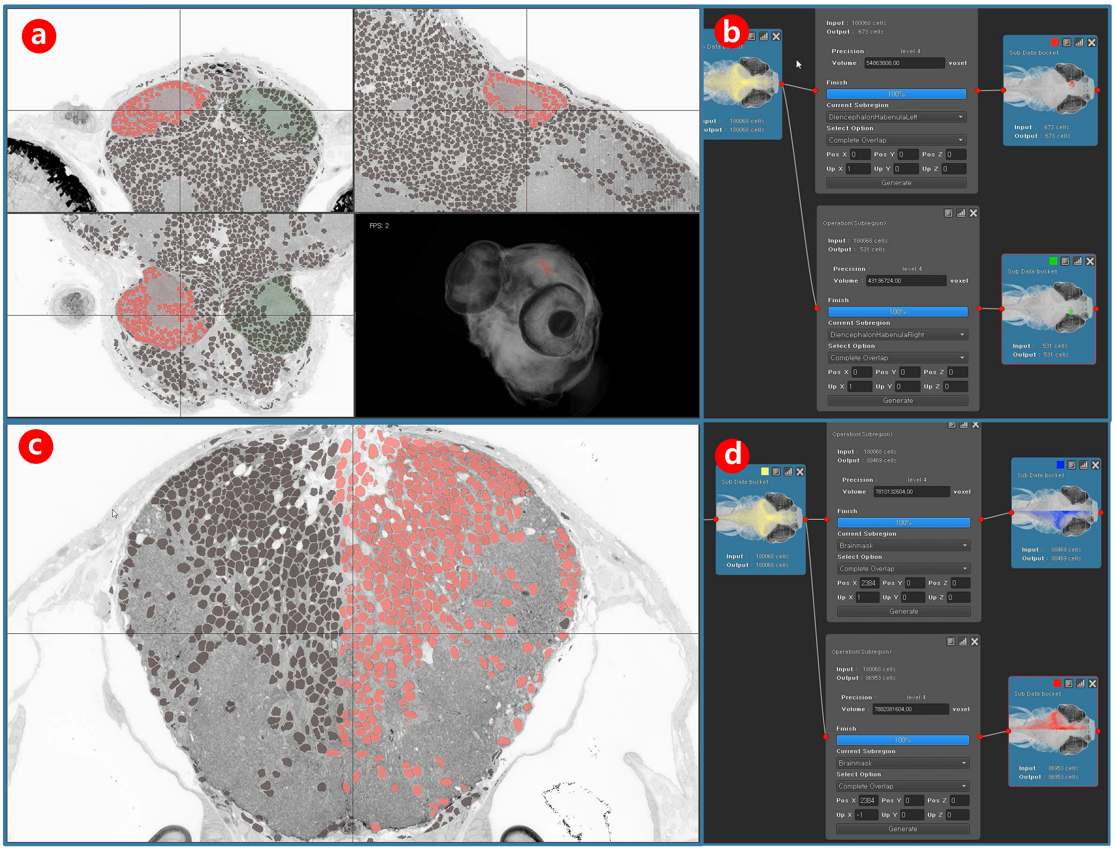
	Count (cells)	Cell volume sum (voxel)	Mask volume (voxel)	Density (%)
LE	97,478	1,191,058,926	4,908,468,744	24.2654
RE	99,330	1,261,839,368	5,137,648,144	24.5606

Table 2: Comparison of properties of cells of the eye.

LE(Left sides of eye) RE(Right sides of eye)

IV Conclusion and Future work

In this research, I introduced a visual analysis system for neuronal morphology analysis of the zebrafish brain. The system can generate cell groups based on the morphology of the nucleus of the zebrafish and can see the statistics of those groups. The user can select cells based on an anatomical subregion and see the morphology statistics for them. This analysis process can be combined through the graph interface to enable more powerful analysis. Using the 120-nmpx zebrafish ssEM data, we evaluated the effectiveness and usefulness of this system through the biological fact-checking and expert interviews. New studies can be added to the system to allow users to select cells flexibly. Users can also visualize data such as the axon membrane skeleton and study the relationships of adjacent cells.



Data bucket
Input : 180068 cells
Output : 180068 cells

Input : 180068 cells
Output : 673 cells
Precision : level 4
Volume : 54863808,00 voxel
Finish
100%
Current Subregion
DiencephalonHabenulaLeft
Select Option
Complete Overlap
Pos X 0 Pos Y 0 Pos Z 0
Up X 1 Up Y 0 Up Z 0
Generate

Sub Data bucket
Input : 673 cells
Output : 673 cells

Operation(Subregion)
Input : 180068 cells
Output : 531 cells
Precision : level 4
Volume : 43136724,00 voxel
Finish
100%
Current Subregion
DiencephalonHabenulaRight
Select Option
Complete Overlap
Pos X 0 Pos Y 0 Pos Z 0
Up X 1 Up Y 0 Up Z 0
Generate

Sub Data bucket
Input : 531 cells
Output : 531 cells

Sub Data bucket
Input : 180068 cells
Output : 180068 cells

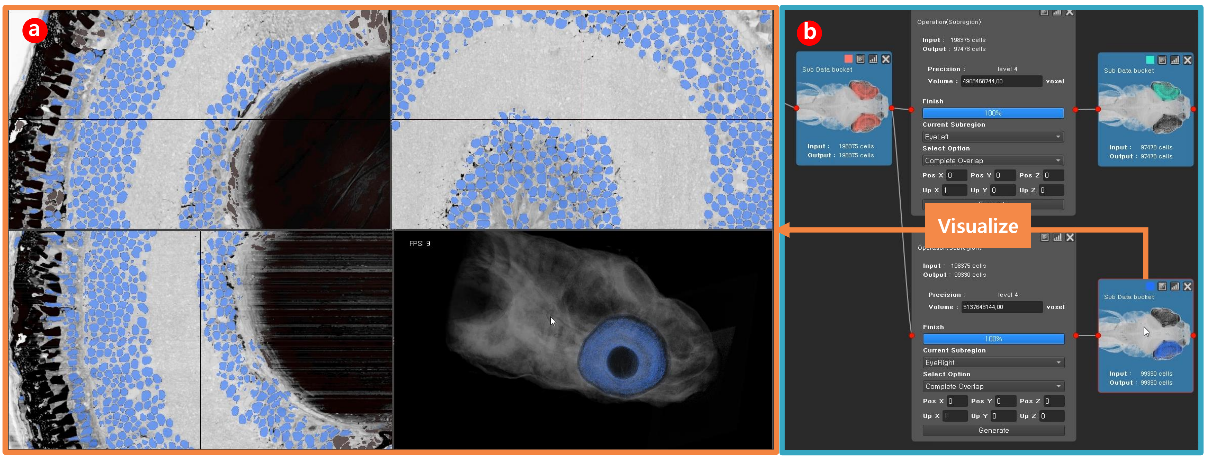
Operation(Subregion)
Input : 180068 cells
Output : 88469 cells
Precision : level 4
Volume : 7810132604,00 voxel
Finish
100%
Current Subregion
Brainmask
Select Option
Complete Overlap
Pos X 2384 Pos Y 0 Pos Z 0
Up X 1 Up Y 0 Up Z 0
Generate

Sub Data bucket
Input : 88469 cells
Output : 88469 cells

Operation(Subregion)
Input : 180068 cells
Output : 86953 cells
Precision : level 4
Volume : 7882081604,00 voxel
Finish
100%
Current Subregion
Brainmask
Select Option
Complete Overlap
Pos X 2384 Pos Y 0 Pos Z 0
Up X -1 Up Y 0 Up Z 0
Generate

Sub Data bucket
Input : 86953 cells
Output : 86953 cells

Figure 10: Symmetric comparison analysis and visualization results. Selected nuclei are rendered in red. a) The left and right sides of Habenula are visualized through three axes. It also shows the volume rendering of ssEM and the location of the nucleus in the currently selected Habenula. b) We created two pipelines for the graph interface for this analysis, and analyzed left and right, respectively. c) The whole brain of the zebrafish was divided into left and right areas. This process is performed using the cut plane function of d). The calculated result can be confirmed in d).



Sub Data bucket
Input : 198375 cells
Output : 198375 cells

Operation(Subregion)
Input : 198375 cells
Output : 97478 cells
Precision : level 4
Volume : 4908468744,00 voxel
Current Subregion: EyeLeft
Select Option: Complete Overlap
Pos X 0 Pos Y 0 Pos Z 0
Up X 1 Up Y 0 Up Z 0

Sub Data bucket
Input : 97478 cells
Output : 97478 cells

Operation(Subregion)
Input : 198375 cells
Output : 99330 cells
Precision : level 4
Volume : 5137648144,00 voxel
Current Subregion: EyeRight
Select Option: Complete Overlap
Pos X 0 Pos Y 0 Pos Z 0
Up X 1 Up Y 0 Up Z 0

Sub Data bucket
Input : 99330 cells
Output : 99330 cells

Figure 11: a) It is the visualization of the cells in the eye part of the zebrafish. The user selects cells through the pipeline of b).

References

[1] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon, "Visual analytics: Definition, process, and challenges," in *Information visualization.* Springer, 2008, pp. 154–175.

[2] D. G. C. Hildebrand, M. Cicconet, R. M. Torres, W. Choi, T. M. Quan, J. Moon, A. W. Wetzel, A. S. Champion, B. J. Graham, O. Randlett *et al.*, "Whole-brain serial-section electron microscopy in larval zebrafish," *Nature*, vol. 545, no. 7654, p. 345, 2017.

[3] J. J. Keiriz, L. Zhan, O. Ajilore, A. D. Leow, and A. G. Forbes, "Neurocave: A web-based immersive visualization platform for exploring connectome datasets," *Network Neuroscience*, pp. 1–18, 2018.

[4] M. de Ridder, Y. Jung, R. Huang, J. Kim, and D. D. Feng, "Exploration of virtual and augmented reality for visual analytics and 3d volume rendering of functional magnetic resonance imaging (fmri) data," in *Big Data Visual Analytics (BDVA), 2015.* IEEE, 2015, pp. 1–8.

[5] T. M. Quan, D. G. C. Hildebrand, and W.-K. Jeong, "Fusionnet: A deep fully residual convolutional neural network for image segmentation in connectomics," 2016.

[6] G. Ellis and F. Mansmann, "Mastering the information age solving problems with visual analytics," in *Eurographics*, vol. 2, 2010, p. 5.

[7] 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 in Neural Circuits*, vol. 8, Jun. 2014. [Online]. Available: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4073626/

[8] 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 *Proceedings of AIPR Workshop 2016.*

[9] J. Schindelin, I. Arganda-Carreras, E. Frise, V. Kaynig, M. Longair, T. Pietzsch, S. Preibisch, C. Rueden, S. Saalfeld, B. Schmid, J.-Y. Tinevez, D. J. White, V. Hartenstein, K. Eliceiri, P. Tomancak, and A. Cardona, "Fiji: an open-source platform for biological-image analysis," *Nature Methods*, vol. 9, no. 7, pp. 676–682, Jun. 2012.

[10] A. Cardona, S. Saalfeld, J. Schindelin, I. Arganda-Carreras, S. Preibisch, M. Longair, P. Tomancak, V. Hartenstein, and R. J. Douglas, "TrakEM2 Software for Neural Circuit Reconstruction," *PLOS ONE*, vol. 7, no. 6, p. e38011, Jun. 2012. [Online]. Available: http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0038011

[11] G. Andrienko, N. Andrienko, G. Fuchs, and J. Wood, "Revealing patterns and trends of mass mobility through spatial and temporal abstraction of origin-destination movement data," *IEEE Transactions on Visualization & Computer Graphics*, no. 1, pp. 1–1, 2017.

[12] N. Cao, C. Lin, Q. Zhu, Y.-R. Lin, X. Teng, and X. Wen, "Voila: Visual anomaly detection and monitoring with streaming spatiotemporal data," *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 23–33, 2018.

[13] Z. Liao, Y. Yu, and B. Chen, "Anomaly detection in gps data based on visual analytics," in *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on.* IEEE, 2010, pp. 51–58.

[14] M. Riveiro, M. Lebram, and M. Elmer, "Anomaly detection for road traffic: A visual analytics framework," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 8, pp. 2260–2270, 2017.

[15] F. Zhou, W. Huang, Y. Zhao, Y. Shi, X. Liang, and X. Fan, "Entvis: A visual analytic tool for entropy-based network traffic anomaly detection," *IEEE computer graphics and applications*, vol. 35, no. 6, pp. 42–50, 2015.

[16] I. Gutenko, K. Dmitriev, A. E. Kaufman, and M. A. Barish, "Anafe: Visual analytics of image-derived temporal features—focusing on the spleen," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 1, pp. 171–180, 2017.

[17] P. E. Haeberli, "Conman: a visual programming language for interactive graphics," in *ACM SigGraph Computer Graphics*, vol. 22, no. 4. ACM, 1988, pp. 103–111.

[18] B. Yu and C. T. Silva, "Visflow-web-based visualization framework for tabular data with a subset flow model," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 1, pp. 251–260, 2017.

[19] M. Gleicher, D. Albers, R. Walker, I. Jusufi, C. D. Hansen, and J. C. Roberts, "Visual comparison for information visualization," *Information Visualization*, vol. 10, no. 4, pp. 289–309, 2011.

[20] H. M. Jamil, "Designing integrated computational biology pipelines visually," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 10, no. 3, pp. 605–618, 2013.