Master 2 in Computer Science - Interaction Specialty

Support of the Interactive Visual Exploration and Classification of Temporal Development in 3D Datasets using the Example of Cell Division Data

Author:
Kejin Jin

Supervisor:
Tobias Isenberg
Alain Trubuil

Hosting lab/enterprise:
Aviz, Inria-Saclay

March 1st – August 31st
2019

Secrétariat - tel: 01 69 15 66 36 Fax: 01 69 15 42 72
email: Murielle.Benard@u-psud.fr

Contents

Summary

In this thesis, I present an interactive tool that is designed and implemented for visualizing the 3D cell division data of plant embryo. In this project within the computer science domains of visualization and human-computer interaction (HCI), we would like to investigate how to best support the visual exploration of temporal developments of biologic organisms. In a collaboration between Inria and Inra, we want to analyze segmented 3D datasets of plant embryos which have developed up to, at least, the 8th stage of cell division. In this project we want to investigate visualization techniques and interactive tools to classify cell types based on their shape, identify sister cells that originated from a single cell in the previous division stage, and thus construct a tree of cell division history from only a single 3D microscopy image.

Keywords

biological data visualization, scientific visualization, interactive design, human-computer interaction

CHAPTER 1

Introduction

This work focuses on 3D visualization of the cell division data. It aims at designing and implementing an interactive tool to help users (which are biologists) to better investigate and examine the plant embryonic cells, classify the cells, estimate the relations among cells, and finally construct a cell division tree. Therefore, by using this tool, biologists can analyze the division pattern of the plant embryos in an interactive and more visually distinct way.

1.1 Motivation

By studying the fundamental unit of life, cell biology represents one of the most exciting areas of biology research. An editorial from Nature Cell Biology once noted that "understanding how cells function and communicate with each other is one of biology's great challenges" [1].

In cell biology, researchers have long had a passion for investigating the cell division process, for example, by cell morphological studies. Cells with countless different shapes not only have aesthetic fascination, but also reflects the functions of cells. With the rapid development of the computer science and technology in recent decades, biologists have already started to apply some software tools to analyze cells and their growth, but there is still quite large room for improvement for these tools, either in functions or in efficiency.

This work, collaborated between the MaiAGE team in INRA (*Institut national de la recherche agronomique*) and the visualization team AVIZ in INRIA (*Institut national de recherche en informatique et en automatique*) is thus a project that aims at assisting biologists to better analyze cell division from a visualization perspective.

1.2 Research goals

Therefore, several research questions are to be examined within the scope of my master internship which fall mainly into the category of visualization and HCI. The following is the three major research goals of this work.

First, we would like to conduct interviews with the biologists to understand their expectations and requirements of this new tool, and listen to their feedback regularly to guide the design and refinement process.

Second, we would like to find a good visual representation of the plant embryo data in order to better capture the features of the dataset and provide the necessary information that our users need, thus supporting decision making.

Third, we would like to figure out how to realize more useful and diversified interactions in order to allow biologists to better navigate, manipulate and analyze the data, and also record their actions, for example when constructing the division tree.

1.3 Contribution

This work has three main contributions.

1) The design of an interactive tool to analyze biological data. We have followed a relatively comprehensive design process as the first and necessary step, from collecting user requirement through interview, defining the design of the visual representations and interactions, to iteratively modifying our user interface, etc.

2) The implementation of the tool to achieve desired functions and visualization. We not only design the prototype on paper, but write codes to implement an excutable version of the tool which biologists can actually run and use it on their laptops.

3) The preliminary qualitative evaluation of the tool afterwards. Even though we do not have enough time to conduct a utility experiment, we have kept gathering feedbacks from the biologist side and also evaluate our tools based on some theoretical criteria.

1.4 Thesis organization

This thesis is divided into seven chapters.

Chapter 2 is about the related work.

Chapter 3 covers the initial work of the project, including the data collections and analysis, user interview, as well as choosing the programming framework.

Chapter 4 focus on the design process of the interactive tool, including the design concept, design space, visual representation design and the function-interaction table.

Chapter 5 describes the interactive tool in detail, showing the major functions of the tool and the important considerations in terms of interaction and visualization. It also explains some technical issues encountered during the implementation process.

Chapter 6 covers the preliminary evaluation work of the interactive tool, based on both theoretical criteria found in literature and the feedback obtained from the biologists.

Chapter 7 concludes the thesis, summarize the contribution of the thesis, and point out the possible future work.

CHAPTER 2

Related Work

2.1 Background

Cell division is one of the most significant processes in all living organisms. During the division process of a cell, DNA is replicated from one cell to another and the organism also grows. A reasonable and proper division pattern is crucial for shaping the architecture of the organism and ensuring its normal functioning. There are both symmetric and asymmetric division, while the latter often leads to the initiation of new cell types [5]. An illustration of asymetric division is shown in Figure 2.1. Since cell division is so important in life science, for years, biologists have been making efforts to try to find the inner mechanisms guiding the cell division patterns, for example, the orientation and position of the division plane.

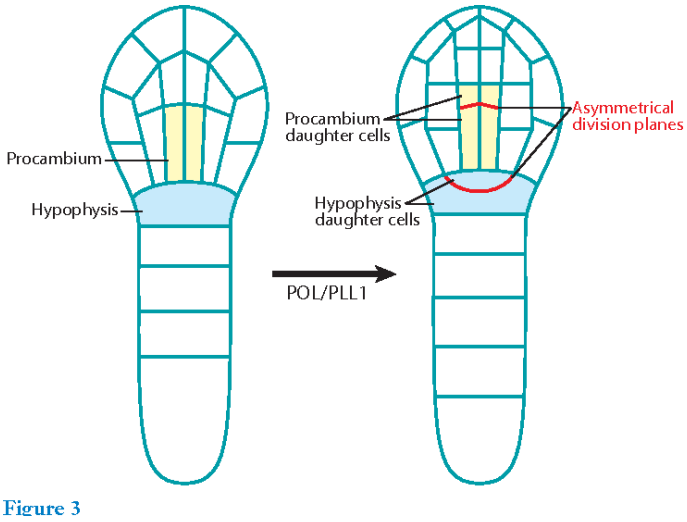

Figure 2.1: Cellular organization of globular stage Arabidopsis thaliana embryos with their asymmetric division planes shown in red [17]

Besides, there are also geometry-based methodologies which are based on the consideration that the orientation and selection of the division plane in animal and plant cells is affected by cell shape [13]. And over the last two decades, the rapid development of computer science and technology has even brought new possibilities for the cell morphological studies that often involve computational biology, thanks to higher imaging resolution, faster processing of the large amount of pixels, and new

graphics techniques, etc. Many biologists have tried to examine the cell geometry in order to find the generic rules that best describes the interplay between cell shape and division pattern, such as 'surface minimization' rule [15, 22]. All these also justify the feasibility and necessity of building such a software to visualize the cell geometry.

While currently most of such cell geometry studies focuses on symmetric divisions in tissues and are mainly analyzed in 2D space, there are limitations of this 2D analysis because the geometrical rule underlying the sequence of 2D cell division patterns may not hold for 3D plant cell division. Therefore, more and more efforts have been made on building some 3D models to check whether those rules also hold in 3D and find out new rules if not [23].

In their work, Moukhtar et al. [10] studied *Arabidopsis thaliana* early embryo, and developed a computer model of cell division to investigate the space of candidate division planes under geometrical constraints in real 3D cell shapes, as illustrated in Figure 2.2.

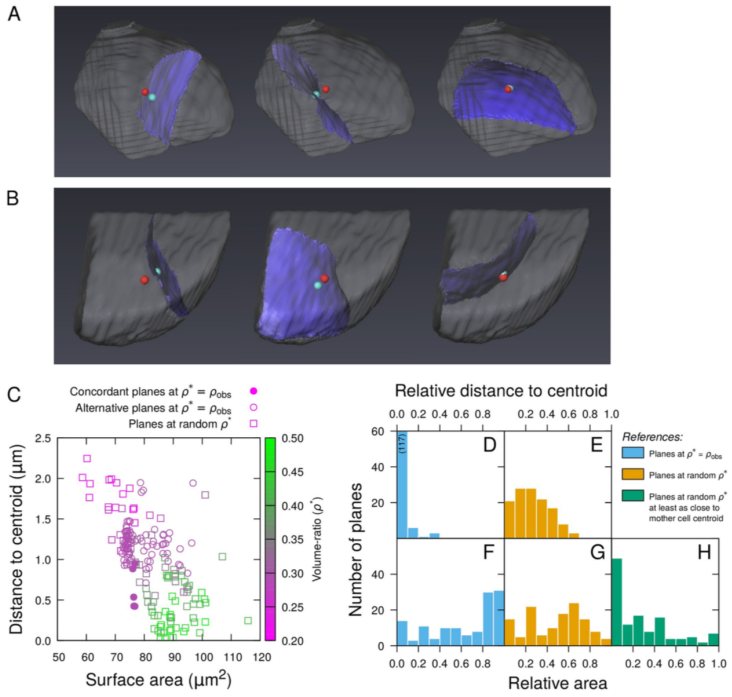

Figure 2.2: Modeling asymmetrical division at the 8C-16C transition

2.2 Visualization in biology

2.2.1 Scientific Visualization

Scientific visualization is branch research and application field of visualization, focusing on the visualization of three-dimensional phenomena such as architecture, meteorology, medicine or biology. It involves research in computer graphics, image processing, high performance computing, mathematics, and other areas to create the visual representations of the complex data in science.

Among them, biological data visualization (BioVis) is an increasingly important application domain. There are some surveys on different areas of biological data visualization such as visualizing time-dependent biological data [20], visualizing live cell imaging [16], visualizing microarray data [6], or visualizing spatial multivariate medical data with glyphs [18].

Information visualization is the other important field of visualization, which is the study of (interactive) visual representations of abstract data to reinforce human cognition. The abstract data include both numerical and non-numerical data, such as text and geographic information.

2.2.2 Cell visualization

Cell visualization falls into the categories of both information visualization and scientific visualization. As for the visualization of the cells, classical ways are to directly capture the image of cells under a microscopy. With the imaging resolution getting higher and higher, it is indeed very important in cell biology. For example, direct and high-definition visualization of some special part of the cell is definitive proof of cell division and particularly important in distinguishing some prominent variations in cell cycle from cell division [21].

Besides, sometimes biologists only need an abstract representation to describe the cell and its division process. Here some rendering techniques in computer graphics can be used to depict the cell shape and structure, such as surface rendering, volume rendering, illustrative rendering, etc. For example, Chawin *et al.* [14] have represent a cell by visualizing several subcellular structures, as shown in Figure 2.3. In this thesis, we choose to combine several rendering techniques to help biologists examine the embryo and the cells in different perspective.

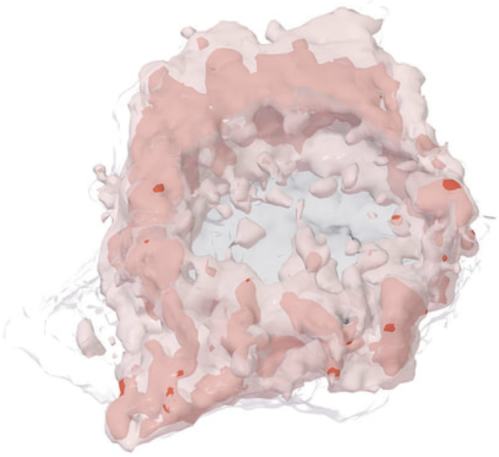

Figure 2.3: illustrative representation of a cell

2.2.3 Tree visualization

Constructing a lineage tree is one of the most important tasks of this thesis. Tree visualization is a technique from InfoVis. One natural way to represent this hierarchical process is to use the binary tree structure. In addition to the traditional tree structure, there are also some other representations such as tree map, tree ring [9], etc,as shown in Figure 2.4. Here in my thesis, the tree ring format has been used to visualize the lineage tree.

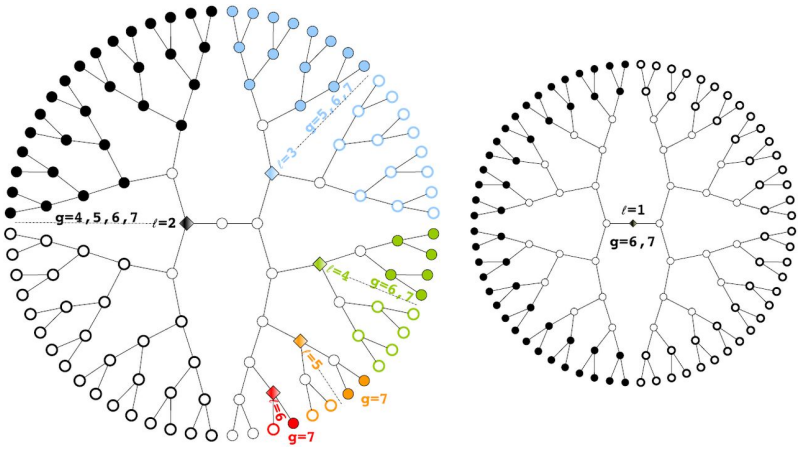

Figure 2.4: Bifurcated subtrees

2.2.4 Brushing and linking

A very significant feature of our system is that we choose to use multiple views linking together to visualize different aspects of the embryo and cells. A change made in one visualization view is automatically reflected in the other views. This is a powerful interaction technique called 'brushing and linking' which combines different visualization techniques together and provides more information than a single visualization [4].

2.3 Current software in 3D visualization

Currently, there are already many successful commercial or open-source software available for visualizing complex 3D data, which cover the functions needed by most researchers for their scientific visualization needs. Some of them are built based on the traditional lower-level 3D graphics APIs like OpenGL, Direct3D, while other are based on some higher-level APIs such as VTK, OpenSceneGraph, etc. The following is just a few examples of them.

1) ParaView: ParaView is an open-source, multi-platform data analysis and visualization application built on top of VTK. ParaView users can quickly build visualizations to analyze their data using qualitative and quantitative techniques. The data exploration can be done interactively in 3D or programmatically using ParaView's batch processing capabilities.

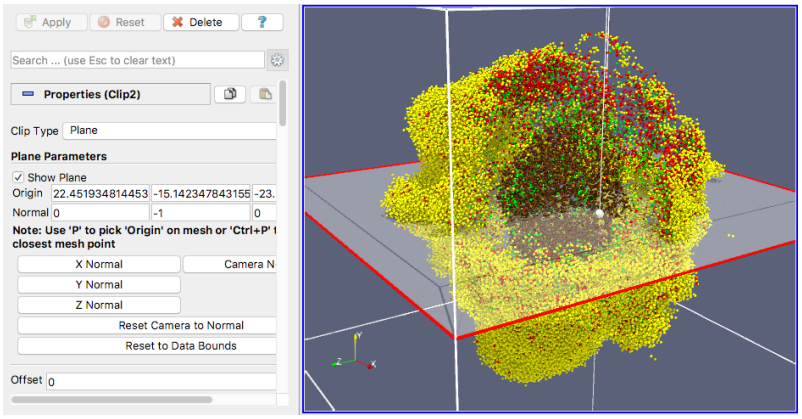

Figure 2.5: Paraview

2) Maya Autodesk: Maya® 3D animation, modeling, simulation, and rendering software provides an integrated, powerful toolset. Use it for animation, environments,

motion graphics, virtual reality, and character creation.

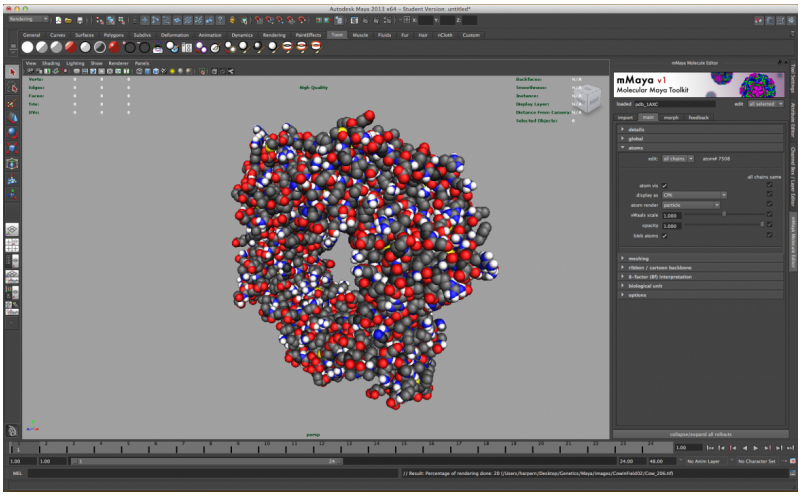

Figure 2.6: Maya Autodesk

3) Ensight: EnSight provides a set of tools to help with many types of engineering analysis, visualization, and communication. With EnSight you can create contours, isosurfaces, particle traces, vector arrows, elevated surfaces, and profile plots. Animation is also supported.

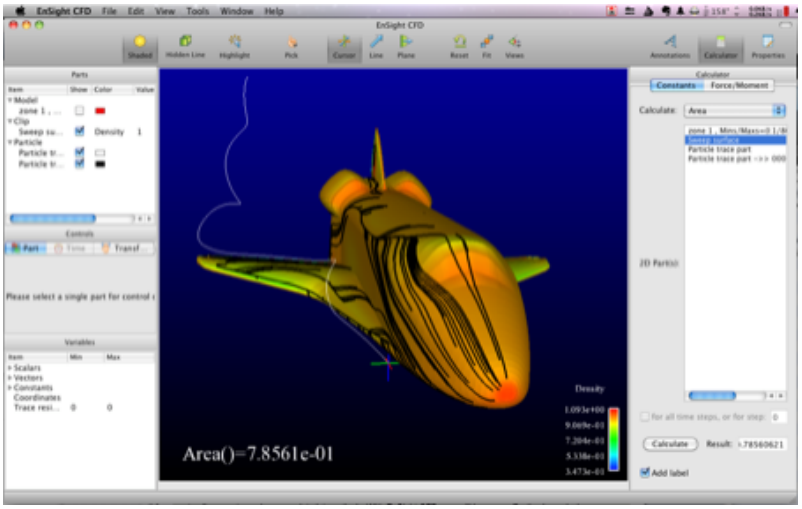

Figure 2.7: Ensight

4) Avizo: Amira-Avizo Software are the leading high-performance 3D visualization and analysis solutions for scientific and industrial data. Avizo Software offer abundant state-of-the-art image data processing, exploration and analysis features within an intuitive workflow and easy-to-use graphical user interface.

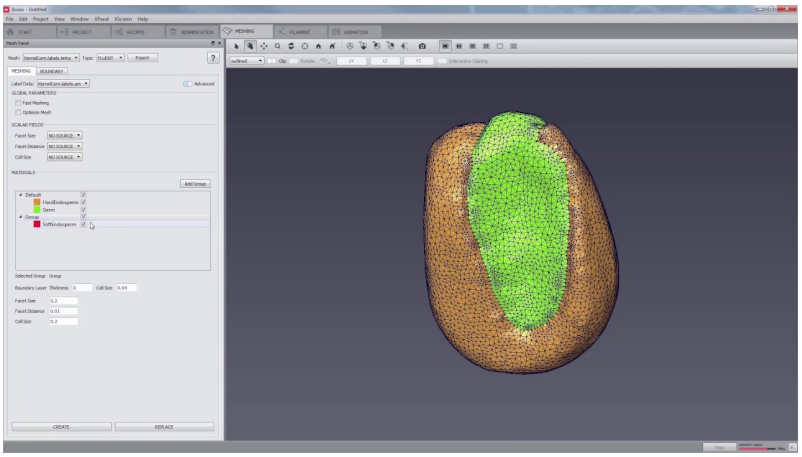

Figure 2.8: Avizo

The tool built in this thesis is based on VTK [19], because it already provides many useful functions to standardize things and achieve rich kinds of visualizations and interactions, and it's easier to learn and implement than lower-level APIs like OpenGL.

CHAPTER 3

Initial work

3.1 User interview

During the implementation process of the tool, I have kept talking with the biologist in a regular basis, with an aim to gather domain knowledge from the users' point of view, specify the users' needs and requirements about the tools, and get their feedbacks.

Studying some papers on user research, I have categorized my interview questions into four aspects: goals, opportunities, priorities and information.

3.1.1 Goals

This part of questions mainly focus on why they need such a tool.

The biologists are studying the determinants of the plant cell division patterns correlated to cell geometry, so they would like to evaluate the existence of geometrical rules in asymmetrical and symmetrical divisions of complex cell shape. Therefore they want to have a tool to analyze the cells and especially the cell division patterns interactively and more efficiently.

3.1.2 Opportunities

This part of questions mainly focuses on what are the drawbacks of the existing tools yet to improve.

According to the biologist, although currently there are already very nice 3d visualization tools widely used in biology field, such as Avizo, there is no such a tool which provides dedicated interaction for the lineage tree construction and So they have to use separate tools in parallel in order to create the tree. Therefore, what we could do is to provide a more integrated platform for them to focus on their study with a single tool.

3.1.3 Priorities

This part of questions mainly focus on what is most important for them when using this tool.

According to the biologist, the biggest challenge facing in front of them now is how to find the sister cells and then construct the lineage tree as efficient and quick as possible, and how to figure out the division patterns of those embryos that they are not very familiar with such as mutant embryos, based on the division history of the already well-studied embryos.

3.1.4 Information

This part of questions mainly focus on what kind of information they want to get from the tool to help them make decisions.

They think the shape of the cells is the most important factor to determine the cell division patterns. Beside, knowing the neighboring relations between cells, the size of their shared interface, and the type of cells (such as apical, central or inner cells) will also help them to examine the potential divisions.

3.2 Data analysis

3.2.1 Data collection

I got the datasets from the biologists in Inra, which are the data about *Arabidopsis thaliana* early embryos. They have been using this plant species as the object of study to investigate geometrical principles underlying plane selection in symmetric and in asymmetric divisions within complex 3D cell shapes [10].

The biologists first got the raw data, that is, the 3D volumetric data of the embryo under the microscope, as shown in Figure 3.1. Then they create the geometric models from the image data by reconstructing the 3D mesh surface using the Avizo software. For surface reconstruction, they first segment the image by assigning labels to image voxels that identify and separate objects in a 3D image. Once the interesting features in a 3D image volume have been segmented, they are able to construct a triangle made surface mesh of the segmented object.

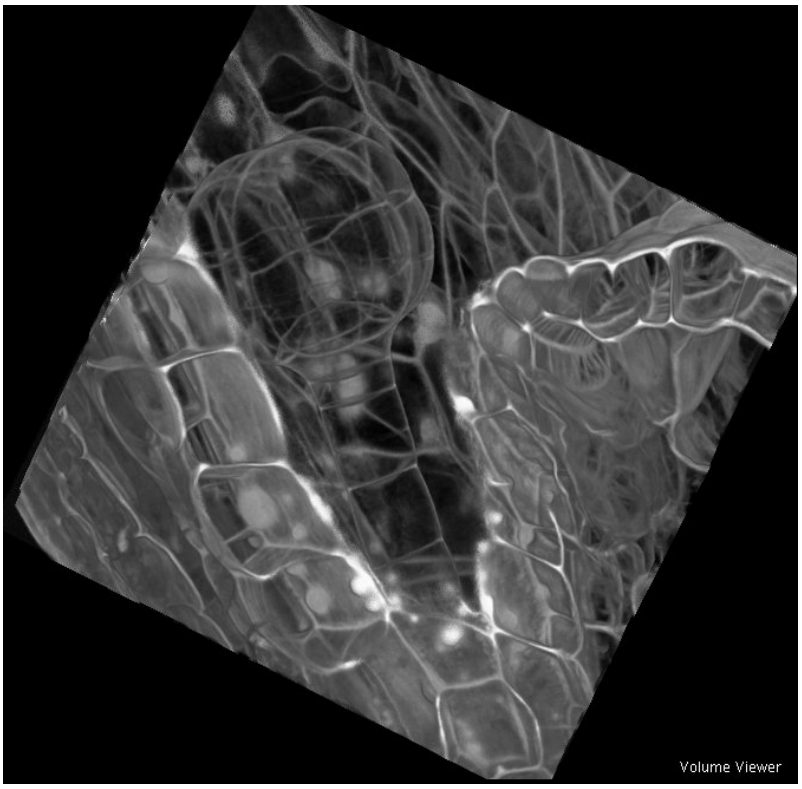

Figure 3.1: The 3D volumetric view of the embryo

3.2.2 Data analysis

Fig 3.2 shows the spatial organization and description of early embryo development. 'A 'shows the 3D representation of cell identity in 8C, 16C and 32C-stage embryos, 'B' shows the volume rendering of 3D confocal images of embryos at all canonical stages from 1C to 32C, and 'C 'shows the segmented cells in embryos from 1C to 32C stages.

Fig. 3.3 explains schematically the cell division process. In our dataset, the plant embryo have developed up to the 8th stage of cell division with 256 cells.

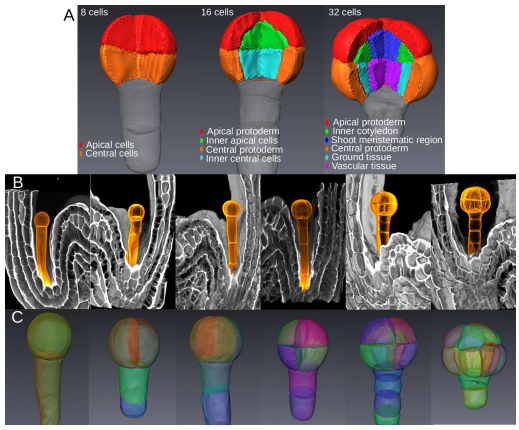

Figure 3.2: Embryo development [10]

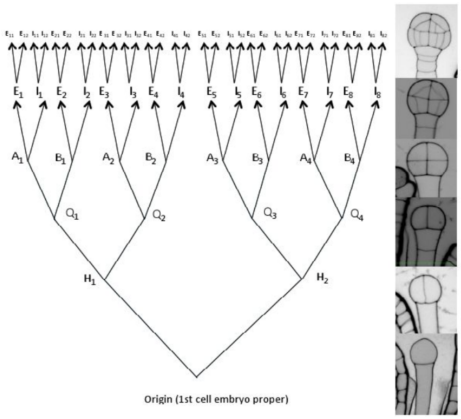

Figure 3.3: Cell division [10]

3.2.3 Dataset analysis

As mentioned above, the 3D surface model of the plant embryo is obtained by labelling and then meshing the raw image data. Besides, the biologists also use some format to store the cell division information of the embryo. Therefore, we have four types of dataset in hand, and each type of dataset contains several files including the plant embryo in different development stage.

1) The raw image data stored in a TIFF file, as shown in the B part in Fig 3.2.
2) The label data also stored in a TIFF file.
3) The surface data stored as a '.surf ' file type, which provides most of the information that we need in this project.
4) The lineage data stored as a '.treeV' file type, which is very important for constructing the cell division tree.

Surface data

The '.surf 'file mainly contains the information about the surfaces and their triangles which we will use to do the 3D reconstruction of the embryo and its cells, as well as finding the neighboring relations between cells.

The information we can get from the surface file is mainly as following.

1) The number of tissues and a corresponding color. Figure 3.4 (A)
2) The number of vertices and their coordinates.Figure 3.4 (B)
3) The number of triangles of each surface and their triangle indices. Figure 3.4 (C)

4) The neighboring relation between two cells. Figure 3.4 (C)

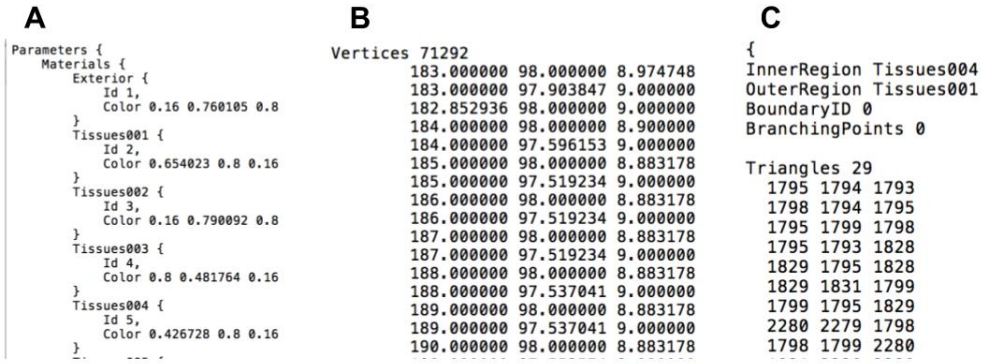

Figure 3.4: The surface file format

Lineage data

The lineage tree is defined in a specific format. Each file contains only one line, like the one below:

6 6 7 7 0 8 8 0

In this array structure, the index of each entry represents the id of each cell, and each number in the entry represents the parent id of this cell. For example, in the array shown above, cell number 1 and cell number 2 are sisters and the number associated to their union(parent) is 6 (this is the greatest number of a cell in the segmented file + 1).

Cells number 3 and 4 are sisters and the number associated to their union (parent) is numbered 7. Cell number 5 does not exist or has no parent, so her parent is 0.

3.3 Programming Tools

Given the nature of our dataset, we should choose a 3D graphics API for the visualization. And since we also need to allow users to explore and manipulate the data, we also should provide a proper interactive interface. Therefore, we have decided to use VTK+QT framework as our programming platform.

3.3.1 VTK

VTK is an open-source API built on OpenGL for 3D computer graphics, image processing and visualization(especially for scientific visualization) with world-class usage.

VTK is easy to learn and use, because it has adopted an object-oriented system design strategy with the access of class and instance data members carefully controlled. Nearly all things(data structures, algorithms, etc) are represented as many objects. So as long as we get familiar with these classes and how to combine them, we can succinctly achieve varies types of visualization.

Fig 3.5 shows VTK's visualization pipeline. The source algorithms generates data by reading or constructing one or more than one data objects. these objects are fed into the filters and one or more data objects are generated in the output. Mappers (or, in some cases, specialized actors) take data and convert it into a visual representation rendered by the rendering engine [19] [3].

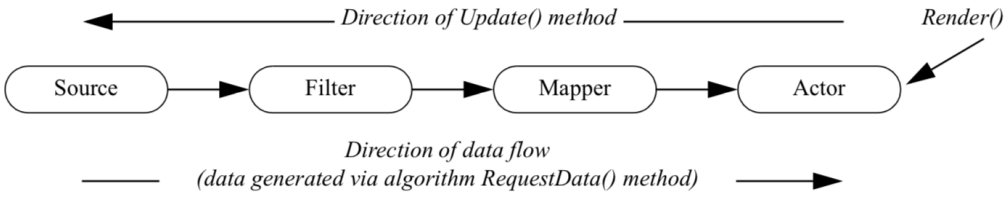

Figure 3.5: VTK rendering pipeline

3.3.2 QT

As this tool is built for traditional PC display, I choose QT as my GUI development tool.

QT is an open-source and cross-platform widget toolkit for designing and creating graphical user interfaces. After years of development, Qt not only has a complete C++ graphics library, but also has recently integrated the database, OpenGL library, multimedia library, network, script library, XML library, WebKit library, etc., and its core library has also included the Inter-process communication, multi-threading and other modules which greatly enrich Qt's ability to develop large-scale and complex cross-platform applications.

Just as its concept reads, "Code Less, Create More, Deploy anywhere.", QT provide

very rich resources that allow users to quickly build the desired user interface and its signal-slot mechanism makes interaction implementation far more easier and effective.

CHAPTER 4

Design process

4.1 Design concept

Before designing how the system will work, it's important to first define what the system will do.

First of all, we should specify the user group of this tool. The users of this tool are biologists who study cell geometry or organ morphogenesis in order to predict the cell division patterns.

Second, we should specify what our system does. Since the biologists would like to predict the cell division plane of the embryo by looking at the geometrical features of the cells, especially for those mutant embryos which they have less experience on, the major purpose of this tool is thus to

1) use different visual representations of the embryo and the cells to help the biologists study them thoroughly in different aspects

2) provide effective operations or interactions to help the biologists construct the division tree more quickly and accurately

Therefore the design concept of this tool can be summarized as follows:

This tool is designed for biologists to analyze the embryo cells, identify sister cells that are originated from a single cell in the previous division stage and thus construct a tree of cell division history.

4.2 Design diagram

The system elements and the design concept about this tool proposed above is shown in the diagram below.

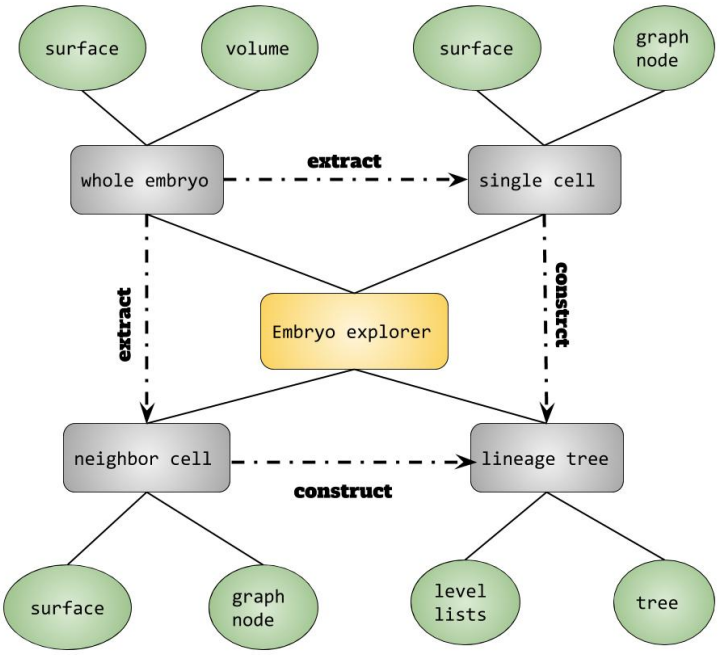

Figure 4.1: Design diagram

4.3 Conceptual model

'*About face 4*' [2], a famous book about interactive design, explains the difference of implementation model and conceptual model, or mental model. As Fig 4.2 shows, The model for how the software actually works is called the implementation model. The way users perceive the works they need to do and how the application can help them do so is their mental model of interaction with the software. It is based on their own ideas about how they do their jobs and how computers might work.

User interface design has to be based on the conceptual model rather than the implementation model. As shown in Fig 4.3, to specify the conceptual model, we have to

1) identify the objects, which refer to what the user wants to manipulate

2) identify the operations, which refer to what the user wants to do with the objects

3) identify the commands, which refer to how the user can activate the operations

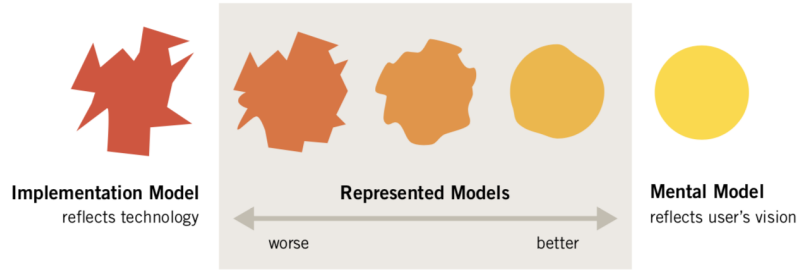

Figure 4.2: Comparison of implementation and mental model

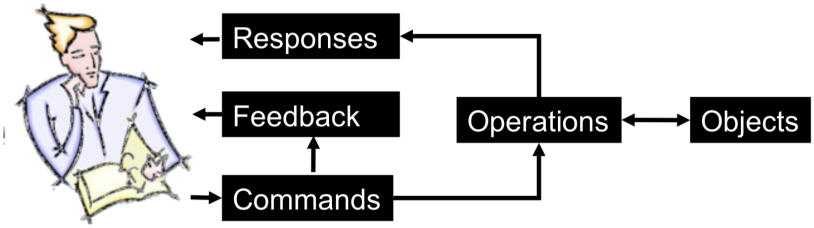

Figure 4.3: Organizing conceptual model

4.4 Visualization design

Mazza [12] introduces the variables that we could consider before we decide which kind of visualization technique to use. Depending on the problem to be solved, the nature of the data and the tasks that the users would do, we should choose a more appropriate way of visual mapping and interaction.

Figure 4.4 shows the categories to consider when designing visual representation introduced in [12]. The items highlighted in red in the suitable variable that fit our project.

Problem	Data type	Dimensions	Data structure	Type of interaction
Communicate	Quantitative	Univariate	Linear	Static
Explore	Ordinal	Bivariate	Temporal	Transformable
Confirm	Categorical	Trivariate	Spatial	Manipulable
		Multivariate	Hierarchical	
			Network	

Figure 4.4: Variables to consider when designing visual representations

4.5 Function-interaction tables

Based on this design concept and conceptual model, as well as the user requirements described in the previous chapter, I have chosen to implement several functions/tasks and come up with some possible interactions to finish these tasks.

The functions and interactions in this project that we have designed so far is summarized into the two tables below.

4.4.1 Table of objects

This table shows the objects that the users want to manipulate, their visual representation, their properties that the users can change, as well as the operations that the users should perform in order to change those properties.

Objects	Representation	Properties	Operations
Embryo	Surface rendering	Orientation	Rotate the embryo
		Size	Resize the embryo
		Shape	Pick, remove/restore certain part
		Color	
	Volume rendering	orientation	Rotate the volume
		Size	Resize the volume
		Shape	Clip the volume
Cells	Surface rendering	Orientation	Rotate the cell
		Size	Resize the cell
		Color	
Neighborhood graph	graph	Orientation	Rotate the graph
		Size	Resize the graph
		Color	Highlight the nodes or edges
			Pick the nodes or edges
Cell lists	Two lists of cell IDs(one for target cell, another for its neighboring cell)	Selected or not	Select on the list(single or multi-selection)
		Clickable or not	
		Background color	
		Marked or not	Marking as sisters
		Item contents	Navigate different levels of lists
Lineage tree	Tree-ring graph	structure	Show/close the tree

4.4.2 Table of functions and interactions

This table shows the interaction considerations of each operation mentioned in the previous table, including the commands that users have to trigger, the feedback obtained in order to tell users that the command has been successfully executed, as well as the results of each command execution.

Operations	Commands	Feedbacks	Responses
Rotate the object	Mouse press and drag	The object rotate following the cursor	The object is rotated
Resize the object	Mouse zoom in/out	The object get bigger or smaller together with the mouse move	The view gets resized
Pick certain cell on the embryo	Right click one cell	The cell is surrounded by a yellow bounding box	The cell is picked
		the corresponding list item is highlighted also in yellow	
Remove/restore certain cell	Right click the item in the tissue list, then a sub menu shows, click the sub menu	A sub menu shows, then click the sub menu	The cell disappears or appears again.

Clip the volume	Drag the range slider either from left or from right	The volume is clipped in three directions.	The new volume after clipping is shown on the screen.
Select cells on the list	Click the item in the tissue list	The list item is selected and highlighted in dark blue	The corresponding cell show in the cell window
			The neighbor ids of this cell show in the neighbor list on the right
			The corresponding node in the neighborhood graph highlighted
	Click the item in the neighbor list	The list item is selected and highlighted in dark blue	The corresponding neighbor cell show in the cell window while the target cell still remain in the window
			The corresponding node in the neighborhood graph highlighted while the target node still remain highlighted
			The shared interface between the two cells shows in the interface window
Mark as sisters	Right click the neighbor list, then a sub menu shows, click the sub menu	The items of these two cells in the tissue list are highlighted in a color the same as the new parent cell	Two sisters is marked a new item is generated in the next-level list
Navigate different levels of lists	Click the ‘next’ or ‘previous’ button	The content of the two list s change	Cells in difference level shows
Highlight the nodes or edges in the neighborhood graph	Mouse hover on the node or the edge	The node or the edge under the mouse cursor turn in red	The node or edge is distinguishable from the rest of nodes and edges.
Pick the nodes or edges in the neighborhood graph	Right click the edge	The edge color change in green	The shared interface between the two cells linked by this edge shows in the interface window
	Right click the node	The node color change and becomes the same as the color of corresponding cell	The corresponding cell and its neighbors show in the cell window
		its neighboring nodes and the linking edges highlighted in brown color.	The corresponding item in the tissue list is highlighted in light blue
Show/close the tree view	Click the ‘show tree’ button	A new pop up window appears	The lineage tree shows

CHAPTER 5

Implementation

After deciding the functions and interactions that could be achieved in this tool, I have implemented an initial executable version of it using VTK+QT framework. For now, this program is compiled on MacOS operation system, but I will make it a trans-platform application in the future.

The following sectors explain the user interface and the main operations of this tool in detail.

5.1 Overview of the user interface

The interface mainly consists of five windows representing five different views of the embryo, two cell lists and several buttons. The layout is shown in Figure 5.1.

As the sketch shows, five windows displays 1) surface of the embryo; 2) volume rendering of the embryo;3) surface of the cells; 4)shared interface between two cells; 5)neighborhood graph respectively.

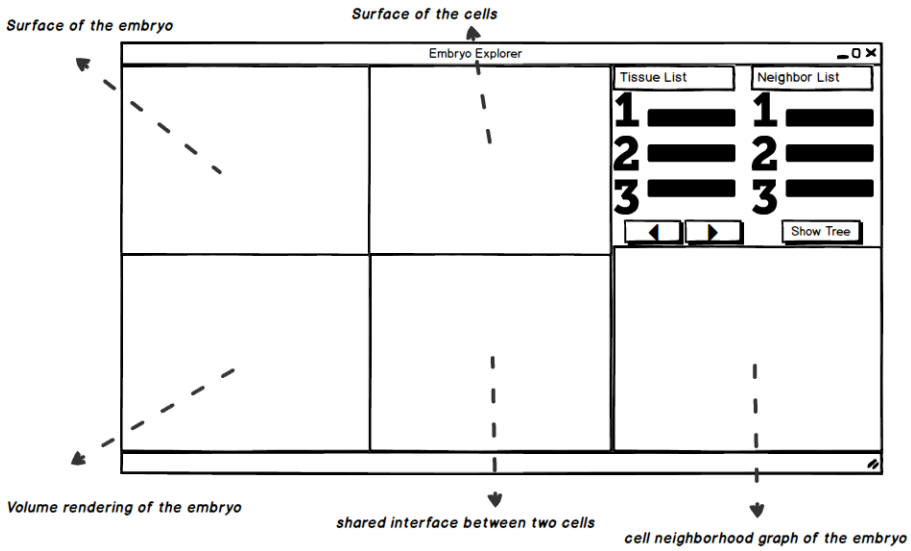

Figure 5.1: Sketch of the user interface

5.2 Observe an embryo

At first, the surface, volume and neighborhood graph of the embryo is shown in three different windows. The first thing the users want to do might be to examine the overall shape and features of the embryo. To do this, users can zoom or rotate them, and for the volume view in particular, users can also clip the volume by controlling three range slider bars. Volume clipping is very useful in observing the inner structure of a 3d Object.

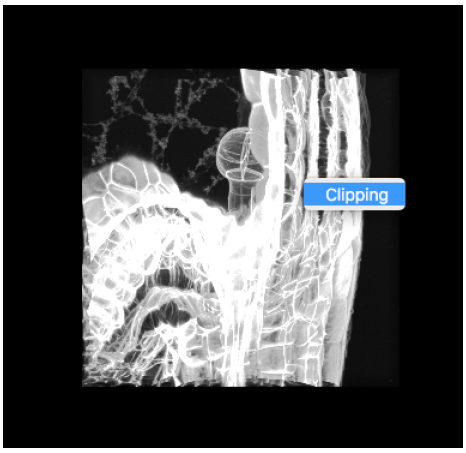

Figure 5.2: Volume view

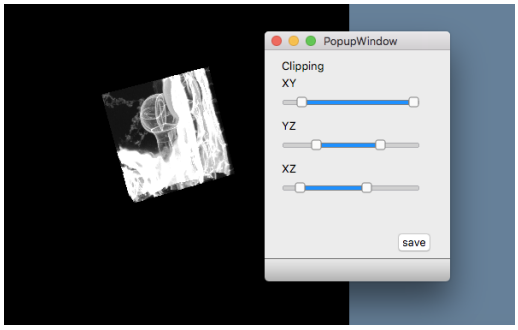

Figure 5.3: Clipping of the volume

5.3 Observe a cell

To observe one or more cells in detail, users can click the items in the tissue list on the right. The list items represent the cell ids.

When a cell is selected, the shape of this cell shows in the forth window. And at the same time, the corresponding part and node in the embryo surface and neighborhood graph are also highlighted with the same color, so that users can not only know the shape but also the position of the cell within the whole embryo. Besides, all the neighbors of this cell will be shown in the second list, called the neighbor list.

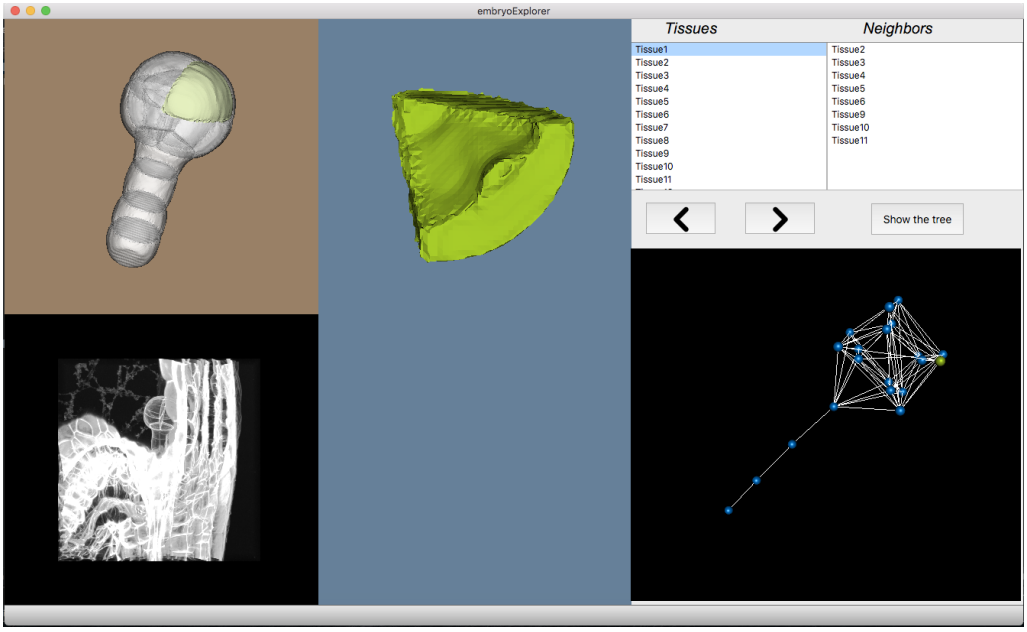

Figure 5.4: Observe a cell

5.4 Remove/restore cells

Sometimes some kinds of cells are not very important object of study, so they can be removed temporarily and restored again when needed. In this way, users can focus on the cells they are more interested in. For example, the 'suspensor' cells are often ignored when studying the cell division, so they can be removed.

Users first right click the cell on the surface view of the embryo, then this cell will be marked and the corresponding item in the tissue list is also highlighted. Then they can right click that item to remove or restore it.

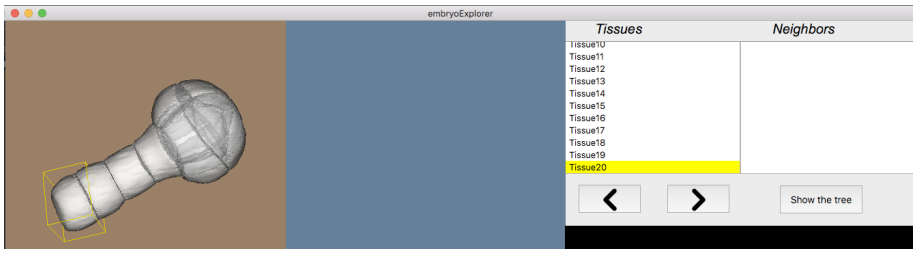

Figure 5.5: Pick cell

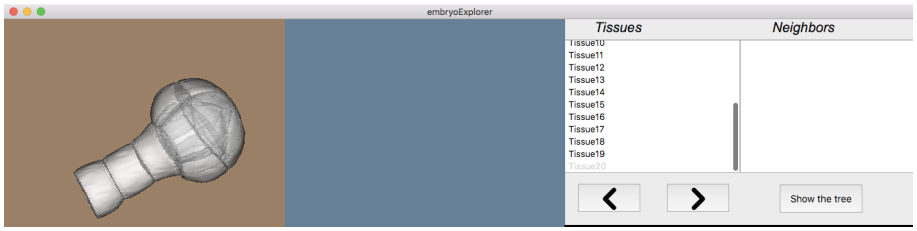

Figure 5.6: Remove the cell

Figure 5.7: Restore the cell

5.5 Examine the relations among cells

Except for the shape and position of a single cell, other importance information that the biologists want to know is the neighboring relations among cells, and the shared interface between the two neighboring cells.

Users can examine the neighboring relations in two ways. First, users can click the items in the neighbor list, so they can see the target cell and this neighbor cell at the same time. Also as figure 5.8 shows, the shared interface between these two cell will be shown in the fifth window from which the biologists can gain some insights about the division plane.

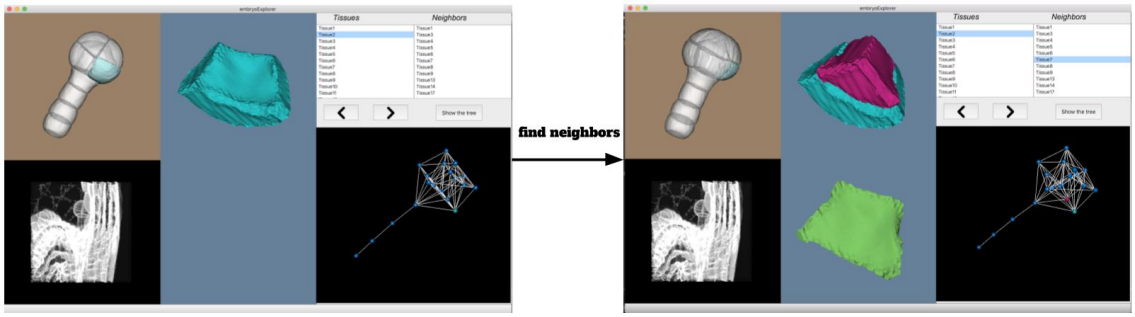

Figure 5.8: Select neighbors

The second way is to directly navigate the neighborhood graph. This graph is shown in the bottom right window of the interface, as marked in Figure 5.9. The node or the edge that the mouse hover on will turn in red and when right clicking on a node, the node and all its neighboring node will be highlighted in the graph and the corresponding cells will also show in the cell window (the shared interface will show if clicking on an edge), as shown in figure 5.9. The details of interactions has already been summarized in sections 4.4.

We construct the neighborhood graph by computing the center of mass of each cell within the embryo and parsing their neighboring relations according to the dataset. So it has the same shape as the embryo and the same relative locations of cells. There are three reasons behind why we decide to use a graph visualization like this.

1) A neighborhood graph can reveal the relative positions among cells more straightforward and clearly than in the surface view, especially in the case when cell numbers are large such as hundreds.

2) The edge-node structure is easier for the users to pick. Imagine when the user wants to pick a cell which is in the very inner part of the embryo, it will get very difficult.

3) Sometimes the users are not very clear about which cell has which name/id, so they can just pick on the graph the cell they are interested in without bothering to figure out which item to click in the cell lists.

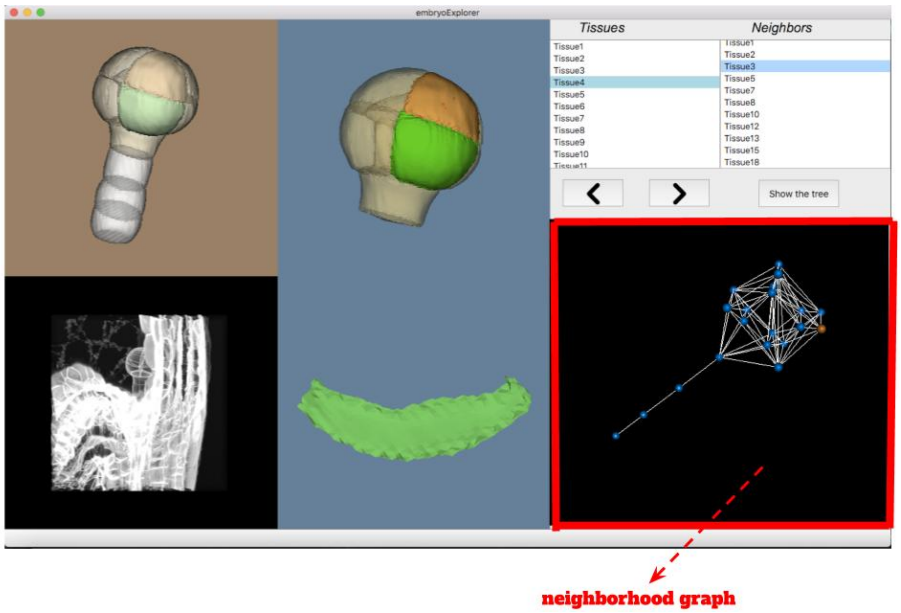

Figure 5.9: Picking on the graph

5.6 Construct a division tree

After the operations mentioned above, now the biologists may already have some ideas about the growing process of the embryo, for example, which two cells are sister cells and originated from a single cell. Next, one of the most important goals of this tool is to help the biologists to record this lineage information and construct a division tree, so that they can use it to do some further studies such as comparing a less-known embryo division process with the well studied ones so as to find some patterns or get some interesting findings.

The iterative process of constructing the tree is explained in figure 5.10. Every time the users mark two cells as sisters cell, a new bigger cell is generated, that is, their parent cell, and the new cell name will show in the tissue list in the next 'level', which are different lists containing cells in different division stage. Users can always check previous or next levels of cells by clicking the two arrow buttons below the lists.

For now, we simply integrate two child cells and assign the same color for them to represent the parent cell. According to the biologists' feedback, this kind of simple concatenation cannot represent the real shape of the parent cell accurately, so in the future when generating the new cell in our tool, the shape has to be computed again

in the backstage.

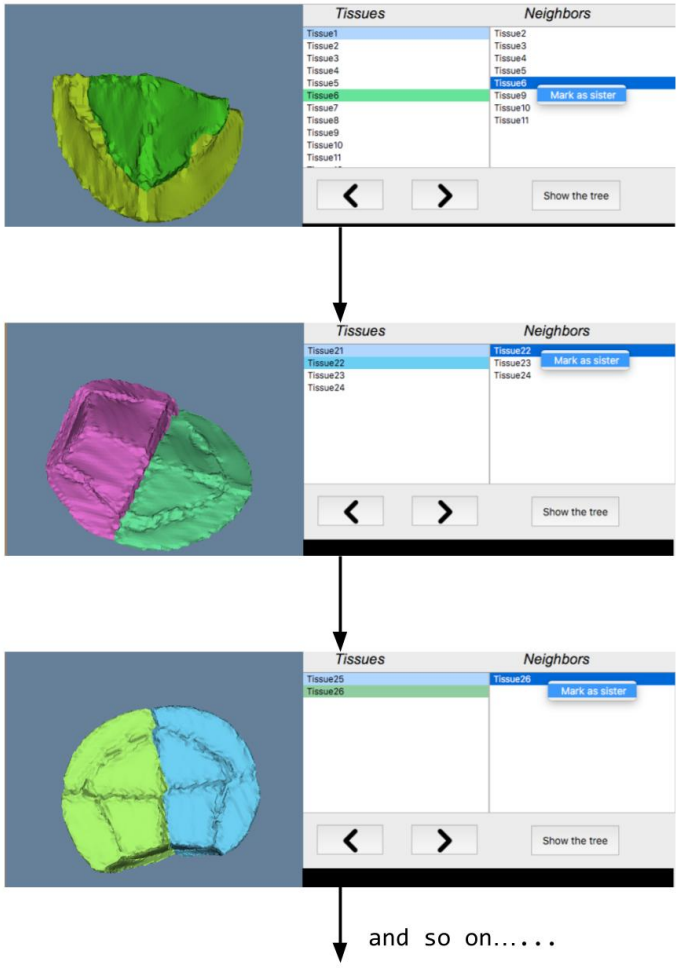

Figure 5.10: Select neighbors

5.7 Observe the tree

During the process of constructing the lineage tree, the users can see the current tree structure at any time by simply clicking the 'Show tree' button. There can be many different ways to represent the tree. Here I have implemented two types of tree, one is the most common tree structure used in computer science, another one is the tree ring structure which saves much space and is useful when the embryo has already gone through many division stages with a large number cells. The trees are shown in figure 5.11 and 5.12.

For now the tree is static, but in the future, more interesting interactions can be added. For example, when clicking one node, the corresponding cell will be shown in place.

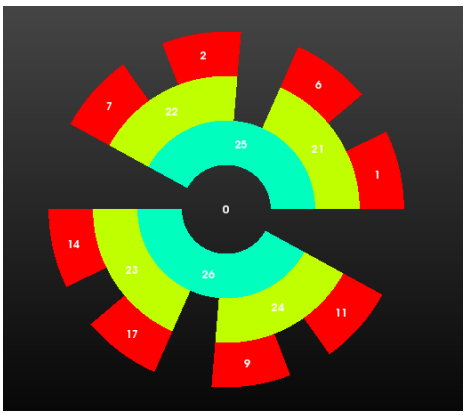

Figure 5.11: Early embryo development

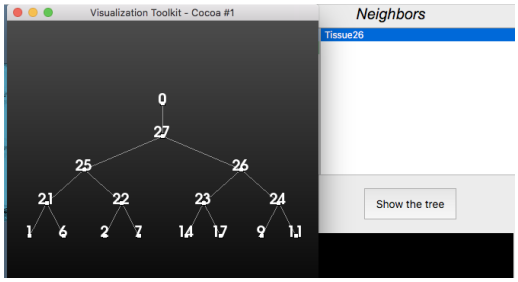

Figure 5.12: Cell division

CHAPTER 6

Evaluation

Due to the limited duration of the internship, we do not have enough time to conduct a comprehensive user experiment to evaluate this system, though there should be in the future. Therefore, we decide to do initial evaluation from two aspects. The first one is based on some papers like [11, 7] on the evaluation criteria of an interactive visualization system, and the other one is to talk with the biologists and listen to their feedback.

6.1 Theoretical evaluation

By linking interface usability knowledge, concepts and methods with evaluation of the visualization techniques, Carla *et al.* [11]established two sets of criteria, with the first being for usability testing of visual representations and the second one for evaluating interactions. Here we use these two sets of criteria to evaluate our system.

6.1.1 Visual representations criteria

Figure 6.1 describes some criteria used to evaluate visualization part of the system.

1) Limitations and cognitive complexity: the geometric or visual constraints like size of the display or maximum number of data elements. In our case, the maximum number of cells is just several hundred, so the dimension and density of the data is not very complex and quite easy for users to see.

2) Spatial organization: the overall layout or distribution of information elements in a visual representation. Here in our case, the spatial locations of the cells are in 3D rather than 2D plane, and some cells are often occluded by others. Although, we already use the neighborhood graph to express the inner structure of the embryo and an extra window to display details of a single cell, it would be better if we could have kind of visualization in the embryo surface window to directly navigate cells and peer into the inner part of the embryo without losing the context, such as using 'focus+context' technique [8].

3) Information coding: in spite of shape and locations, for now, I mainly use color mapping to distinguish different cells and the colors are consistent in all parts of the system like lists, neighborhood graph and cell surface. It would be better if we could have ways to not only distinguish single cell but also different type of cells

(apical, central or suspensor cell, etc)

4) State transition: the result of rebuilding part or the entire visual representation after a user action. Here in our system, the transition between two consecutive representations is discrete without animation.

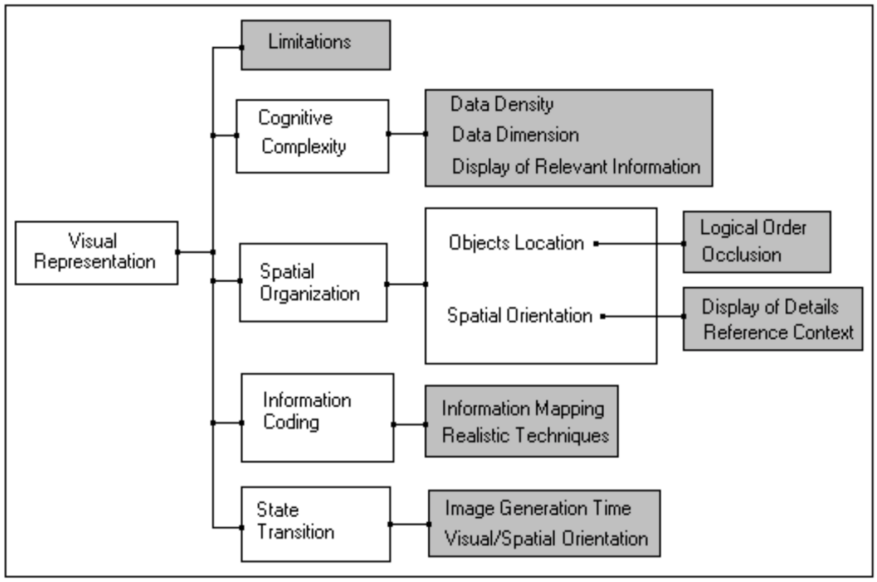

Figure 6.1: Criteria for the evaluation of visual representations of information visualizations techniques

6.1.2 Interaction mechanisms criteria

Figure 6.2 describes some criteria used to evaluate interaction part of the system.

1) Orientation and Help: our system allows control of the level of detail because the users can select a cell or node and see its detail in the cell window. And our system also represents additional information during the process of user action, for example, highlight the list items of two sisters cells with the same color as the parent cell when the users construct the division tree so that the users can have in their mind which two child cells corresponds to which parent cells. Currently, our system does not provide all-round undo/redo operations which should be added in the future.

2) Navigation and Querying: here in our system, most of the features needed for browsing are implemented except for growing and search/query. Cells can be selected by clicking on the lists or pointing at the nodes in the neighborhood graph. Different stages of cells can be navigated by clicking the arrow buttons. Viewpoint

and geometric feature of the visualization can be manipulated by zooming and rotating, etc. Maybe in the future, two marked sister cells can be merged into one node or cell even in the surface view and the neighborhood graph of the embryo and can be hidden/unfolded at any time to achieve the 'growing' feature. Also, querying features can be added which will allow users to quickly find the cell they want to check by name, type, position, etc. Those are very useful interactions especially when the cell number is large.

3) Dataset Reduction: for now, our system only support pruning data by remove certain cells. In the future, there can be filtering function such as only keeping those neighbor cells with the area of the shared interface larger than some threshold, and clustering functions such as classifying cells by their role in the embryo, etc.

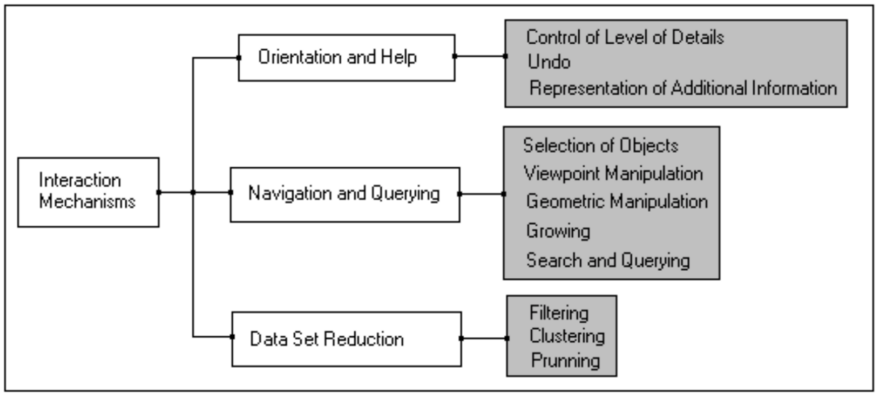

Figure 6.2: Criteria for the evaluation of interaction mechanisms

6.2 User feedback

During this six months of internship, I have kept talking with my supervisor and co-supervisor in a regular basis so as to ensure that my project is on the right track. Besides, near the end of the internship, my supervisor and I have also had a meeting with several other biologists from INRA and I have presented my work in front of them, from which I have also got many instructive feedback.

Blow is some of the feedback got from the biologists, which are also possible future works for this research project.

6.2.1 Time series

For now, we do not have such a visualization that reveals the temporal development process of the plant embryo, but temporal features are a very important aspect of the cell division data. Therefore, maybe we could add time as the forth dimension of the data.

6.2.2 Labeling

Despite shape and location, it's also important to know the biological classification of the cells. For example, it is this complex spatio-temporal organization of cell divisions that leads to diverse cell shapes that are specific to the domains, such as apical vs. central, external vs. internal, etc. Therefore, we could have interactions to allow biologists to label and categorize cells as different types.

6.2.3 Threshold

Currently in our system, given a target cell, those cells that have some shared parts with it are all considered as its neighbor cells, no matter how large the shared part is. This may cause extra workload for the biologists to find the correct sister cell. So we could add some filtering interaction according to the area of the shared interface so as to allow the users to only focus on the cells that are most likely to be sister cells, as mentioned above. Of cause, there can be various kinds of filters such as cell type, location, etc.

6.2.4 Numerics part

Though based on cell geometry, the biologists' study on cell division is still a quantitative study. Therefore, it's not enough to just provide an overall visual representation of the embryo and the cells. We could make it possible to steer the simulation by allowing the users to change model parameters.

6.2.5 Automatic tree generation

When there are hundreds of cells, it would become difficult for the biologists to manually construct the division tree. So they expect that an initial tree could be generated first based on some previous histories of cell division (maybe integrated with machine learning), and then they can examine and modify it later.

6.2.6 Devices

For now, the system only works on traditional PC environment. In the future, it could be integrated in some novel output platforms like tablets, large displays, VR/AR, etc.

CHAPTER 7

Conclusion

In this thesis, I have presented my work during the six-month internship. The goal of this project is to implement an effective interactive tool for studying complex 3D data in biology, more specifically, to design a graphical interactive tool for examining cellular data and explore effective visualization techniques for embryonic plant lineages.

The major work that I have done during the internship include: brief user research with the biologist in the form of interviews; design of the interactive tools; 3D rendering and visualization of embryo and cell division data based on knowledge of computer graphics; implementation of the visualization and the user interface; and the initial evaluation of this tool.

Built from zero, this is just a preliminary version of the system which contains those underlying functions that can meet the users' basic requirements. Many interesting and useful extensions can be made in the future based on this preliminary version, for example some of which have already been mentioned in section 6.2. Besides, in the future, this tool may not only be applied in biological field, but also in many other fields that have the need of visualizing and analyzing, for example, 3D spatially-organized and time-dependent data.

Acknowledgments

I would like to thank my supervisor Tobias Isenberg for his constant encouragement and guidance throughout the project. He has walked me through all the stages of writing the thesis. Without his valuable advice both on higher-level theoretical concepts and on every tiny step in implementation, it would not be possible for me to build such a system completely from zero.

I would like to thank my co-supervisor Alain Trubil for paying so close attention to my work. Thanks for him to come many times to our lab to give me useful feedbacks, organize meetings with other experts, and answering my questions patiently also as future users of this system. His domain expertise helps me ensure that I'm on the right track.

I would like to thank all the members of AVIZ team. We have had many happy memories together during this six-month internship, not only in the office but also in all those outdoor activities. I always feel comfortable with you even from the very first day here. Thanks for your company, friendliness and help.

I would finally like to express my gratitude to my beloved parents who have always been supporting every decision I have made, encouraging me when I'm upset, and giving me advice when I'm unsure what to do. Thank you.

Bibliography

[1] Celebrating 20 years of cell biology. *Nature. Cell Biology 21*, https://doi.org/10.1371/journal.pcbi.1006771:1, 2019.

[2] David Cronin Christopher Noessel Alan Cooper, Robert Reimann. *About face 4*. John Wiley Sons, Inc, 2014.

[3] Lisa S Avila, Sebastien Barre, Rusty Blue, Berk Geveci, Amy Henderson, William A Hoffman, Brad King, C Charles Law, Kenneth M Martin, and William J Schroeder. *The VTK user's guide*. Kitware New York, 2010.

[4] Andreas Buja, John Alan McDonald, John Michalak, and Werner Stuetzle. Interactive data visualization using focusing and linking. In *Proceeding Visualization'91*, pages 156–163. IEEE, 1991.

[5] Rasmussen CG, Humphries JA, and Smith LG. Determination of symmetric and asymmetric division planes in plant cells. *Annual Review of Plant Biology*, 62:387–409, 2011.

[6] Urska Cvek, Marjan Trutschl, Phillip C Kilgore, Randolph Stone II, and John L Clifford. Multidimensional visualization techniques for microarray data. In *2011 15th International Conference on Information Visualisation*, pages 241–246. IEEE, 2011.

[7] Carla MDS Freitas, Paulo RG Luzzardi, Ricardo A Cava, Marco Winckler, Marcelo S Pimenta, and Luciana P Nedel. On evaluating information visualization techniques. In *Proceedings of the working conference on Advanced Visual Interfaces*, pages 373–374. ACM, 2002.

[8] George W Furnas. The fisheye view: A new look at structured files. Technical report, Bell Laboratories Technical Memorandum New York, 1981.

[9] Damien G. Hicks, Terence P. Speed, Mohammed Yassin, and Sarah M. Russell. Maps of variability in cell lineage trees. *bioRxiv*, 2018.

[10] Moukhtar J, Trubuil A, Belcram K, Legland D, Khadir Z, and et al. Urbain A. Cell geometry determines symmetric and asymmetric division plane selection in arabidopsis early embryos. *PLoS Comput Biol 15(2): e1006771*, https://doi.org/10.1371/journal.pcbi.1006771, 2019.

[11] Carla M. Dal, Carla Freitas, Paulo Luzzardi, Ricardo Cava, Marco Winckler, Marcelo Pimenta, and Luciana Nedel. Evaluating usability of information visualization techniques. 12 2002.

[12] Riccardo Mazza. *Introduction to information visualization*. Springer Science & Business Media, 2009.

[13] Nicolas Minc and Matthieu Piel. Predicting division plane position and orientation. *Trends in Cell Biology*, 22(4):193 – 200, 2012.

[14] Chawin Ounkomol, Sharmishtaa Seshamani, Mary M Maleckar, Forrest Collman, and Gregory R Johnson. Label-free prediction of three-dimensional fluorescence images from transmitted-light microscopy. *Nature methods*, 15(11):917, 2018.

[15] Anaëlle Pierre, Jérémy Sallé, Martin Wühr, and Nicolas Minc. Generic theoretical models to predict division patterns of cleaving embryos. *Developmental Cell*, 39(6):667 – 682, 2016.

[16] A Johannes Pretorius, Imtiaz A Khan, and Rachel J Errington. A survey of visualization for live cell imaging. In *Computer Graphics Forum*, volume 36, pages 46–63. Wiley Online Library, 2017.

[17] Carolyn G Rasmussen, John A. Humphries, and Laurie G Smith. Determination of symmetric and asymmetric division planes in plant cells. *Annual review of plant biology*, 62:387–409, 2011.

[18] Timo Ropinski, Steffen Oeltze, and Bernhard Preim. Survey of glyph-based visualization techniques for spatial multivariate medical data. *Computers & Graphics*, 35(2):392–401, 2011.

[19] Will J Schroeder, Bill Lorensen, and Ken Martin. *The visualization toolkit: an object-oriented approach to 3D graphics*. Kitware, 2004.

[20] Maria Secrier and Reinhard Schneider. Visualizing time-related data in biology, a review. *Briefings in bioinformatics*, 15(5):771–782, 2013.

[21] Roberto Therón. Hierarchical-temporal data visualization using a tree-ring metaphor. In *International Symposium on Smart Graphics*, pages 70–81. Springer, 2006.

[22] Fengzhu Xiong, Wenzhe Ma, Tom W. Hiscock, Kishore R. Mosaliganti, Andrea R. Tentner, Kenneth A. Brakke, Nicolas Rannou, Arnaud Gelas, Lydie Souhait, Ian A. Swinburne, Nikolaus D. Obholzer, and Sean G. Megason. Interplay of cell shape and division orientation promotes robust morphogenesis of developing epithelia. *Cell*, 159(2):415 – 427, 2014.

[23] Saiko Yoshida, Pierre Barbier de Reuille, Brendan Lane, George W. Bassel, Przemyslaw Prusinkiewicz, Richard S. Smith, and Dolf Weijers. Genetic control of plant development by overriding a geometric division rule. *Developmental Cell*, 29(1):75 – 87, 2014.