A Visual Navigation System for Querying Neural Stem Cell Imaging Data

Ishwar Kulkarni[1] Shanaz Y. Mistry[1] Brian Cummings[2] M. Gopi[1]
[1]Department of Computer Science,
[2]Department of Anatomy and Neurobiology
University of California, Irvine

ABSTRACT

Cellular biology deals with studying the behavio r of cells. Current time-lapse imaging microscopes help us capture the progress of experiments at intervals that allow for understanding of the dynamic and kinematic behavior of the cells. On the other hand, these devices generate such massive amounts of data (250GB of data per experiment) that manual sieving of data to identify interesting patterns becomes virtually impossible. In this paper we propose an end-to-end system to analyze time-lapse images of the cultures of human neural stem cells (hNSC), that includes an image processing system to analyze the images to extract all the relevant geometric and statistical features within and between images, a database management system to manage and handle queries on the data, a visual analytic system to navigate through the data, and a visual query system to explore different relationships and correlations between the parameters. In each stage of the pipeline we make novel algorithmic and conceptual contributions, and the entire system design is motivated by many different yet unanswered exploratory questions pursued by our neurobiologist collaborators. With a few examples we show how such abstract biological queries can be analyzed and answered by our system.

Keywords: Neuroscience, stem cell segmentation, tracking, cell imaging, data management, visual analytics, navigation, exploration, query processing.

1 INTRODUCTION

Cellular biological experiments typically include culturing different types of cells in a controlled environment. Observation of such cultures includes observing the life processes of the cells for expected and unexpected developments among the cells. Earlier, images of these cultures were taken at distant time intervals to understand the start and end states of the experiment. Only a few characteristics like cell proliferation can be understood from these images. But experiments such as those concerning drug screening involve altering the conditions of the cultures to affect changes in dynamic responses of the cells. These responses of the cell can include metabolism, motility (motion), mutation, migration, proliferation rate and rate of apoptosis and other higher level details such as abnormal protein aggregation and cell-cell interaction. In order to observe these kinematic and dynamic properties of the cells, the images have to be taken often during the entire course of the experiment. Current confocal laser microscopes can take time-lapse images of the culture at a pre-determined frequency to enable such observations of dynamic responses. But, as a result, the amount of data produced by these devices is so massive (approximately 250GB of image and sensor data per experiment) that it is virtually impossible to manually sift through the data, track many cells, compute statistical quantities like area, speed etc., find conditions when various events happen, or identify patterns and correlations among patterns. Such information is used by neurobiologists to answer questions like how many cells undergo mitosis during a period of time; and to test hypothesis like whether the cell's metabolism decreases during division. Hence automation in terms of data processing, data management, visual interaction for navigating through the data, and querying the system for data exploration, are absolutely unavoidable for scientists to quickly frame and test their new hypotheses on various cellular behaviors. In this paper, we propose such an enabling system that opens up new frontiers for neuroscience research.

IEEE Symposium on Visual Analytics Science and Technology
October 23 - 28, Providence, RI, USA
978-1-4673-0014-8/11/$26.00 ©2011 IEEE

1.1 Main Contributions

We present an end-to-end system to automate different stages of the data capturing, management and usage in the context of experiments on human neural stem cells (hNSC). Following are the main contributions of the paper:

We present novel algorithms for image processing for cell identification, segmentation, and tracking. Our algorithm can also robustly find the boundary of the cells that enables accurate computation of other statistical parameters like area of the cell.

We propose a hybrid data representation, storage, and management technique that can handle statistical data, image data and semantic data. Our data management technique is specifically designed to provide fast query processing, efficient data integration for visualization, navigation, and exploration.

We provide data navigation techniques that take semantics of the data into account and enable the users with contextual responses and navigation. The visualization and navigation techniques involve the user in a tight feedback loop using context menus, tooltips and hyperlinked charts.

We introduce a new querying methodology that is designed based on the objects and attributes that are used in our application, and use set operations to explore patterns and correlations in the data.

In the course of the above process, we take the data through different levels of semantic abstraction for data visualization, correlation and hypothesis formulation.

1.2 Images and the Imaging System

Our imaging system uses Olympus VivaView Incubator Fluorescent Microscope. This system allows time-lapse imaging of cell cultures. It also captures images at different focal depths on the culture dish by illuminating the dish with four different laser wavelengths.

The cells in the images are human neural stem cells cultured in different media and substrates. The cells are 10-20μm in size on an average (excluding the branches). The images capture an area of 433.5μm $\times$ 330μm. The cells are marked with fluorescent proteins so that when an appropriate laser is used during imaging, only those cells are visible in the images, which would help in image processing for detecting and tracking these cells. In our experiments, we use Green Fluorescence Protein (GFP) that exhibits fluorescence when irradiated with light of wavelength 488nm. A typical experiment would have images taken at 2-10 minutes interval, and the experiment can run for one to two weeks. Experimental conditions

can change in the middle - new media can be added, the CO_2 level or temperature can be changed, etc.

The captured images are 16-bit grayscale images of intensity values ranging from 0 to 4096 (12 bits of actual data). The images are of dimensions 1024×1344 pixels. A typical image is shown in 1 in Figure 2.

2 Related Work

2.1 Related work in Segmentation and Tracking

The problem of cell segmentation has been worked upon for many years, and a large number of methods and techniques have been proposed. This is largely because cells are dynamic entities that vary widely in appearance and exhibit varied types of behavior, therefore methods developed for one type of cell are not applicable to other cell types.

A common approach for segmentation and tracking is using edge detection and morphological operators. Various combinations of these techniques are used in [1] and [15], but for neural stem cells these methods do not always produce accurate results due to presence of edges with very low contrast.

Some techniques use the level-set segmentation method. This method used by [16] with success on leukocytes, uses a zero-set of implicit energy functional, that traces a smooth boundary around cells. Another popular approach to cell segmentation is the Active Contour Model, proposed in [7]. These models take a combination of cell features such as cell boundary, internal pixel values etc. and minimize energy to 'wrap' a contour around the cell. Due to lack of sharp contrast at boundaries and non-uniform shapes of neural stem cells, these methods do not produce satisfactory results with our images.

Model based segmentation methods are an important class of segmentation techniques that identify cells in images and image sequences based on certain well defined assumptions about spatial or temporal attributes of cells and their motion. Methods such as [19] use size and shape based contours to identify cells. Methods such as the one proposed in [20] use motion patterns on leukocytes in images of blood streams to identify cells.

The watershed approach is a commonly used segmentation technique and is the one we use as a step in our segmentation procedure. Our method is largely inspired from [23] where the authors proposed a method of segmentation based on *Ultimate Eroded Point*. The method uses two erosion structures one for coarse and other for fine erosion successively. This method makes two assumptions: firstly the shape of structure for erosion captures the shape of the cells, and secondly, the topography of the cell image (or the height field) needs to be smooth enough for erosion to a single point to produce a seed.

Methods that track cells by shape matching predominantly use shape descriptors (proposed in [2]). They represent shapes of cells as a number of binned histograms for each point on the outline of the shape and perform graph matching. This method is computationally expensive and also not suitable for highly concave shapes.

Our segmentation method is largely based on watershed algorithm. The problem with watershed generally comes in the form of under and over segmentation of a region, which is to be avoided if statistically correct segmentation is to be achieved. For this we used the technique of seed-point similar to the one suggested in [23]. We produce the seed points using intensity maxima of an enhanced image while avoiding false detection, rather than *UEP*s as in that paper because cells in our image lack a shape to derive the eroding structure elements from. The method of segmentation is described more fully in Section 4.1.

2.2 Related Work in Visual Analytics

Visual navigation plays an important role as the first step in data exploration and knowledge creation from large multi-dimensional data. Navigation refers to the process of traversal of the user view from one aspect or representation of data to another. Generally, the data collected from biological experiments is multi-modal: it consists of image data as collected from microscopes, textual data from annotation and numerical data from various measurements and calculations. Navigation of the data, thus helps the user correlate such modal data [5]. In general Visual Analytics is the approach of combining visualization, human factors and data analysis [8].

An important aspect of any visualization and navigation system is the existence of the human element in the exploratory process. As stated in [9], the importance of data navigation and exploration comes from the vast repertoire of data created in short period of time by scientific experiments, using large computational power that is at our disposal. The author further states that the raison d'Être of such systems is not to find correlations in data (as [4] does) but to represent the data in a fashion that the existence of correlations becomes apparent to the human user.

Applying visual data navigation and exploration techniques to biological data has seen interesting work such as [10]. The authors developed MassVis, a system to analyze mass spectrometer data on protein complexes. Another important contribution in the direction of cell tracking visualization is [14], where the authors develop a single cell tracking visualization scheme seeCell. A similar tracking visualization method for dendritic cells in stream of microscopy images has been proposed in [22]. We propose a system that not only tracks but also correlates the various attributes of the cell life processes. To the best of our knowledge our system is the first to have an end-to-end module that segments the cells, tracks their motion, identifies interesting events and represents the statistical and semantic data visually.

The work done by [18] explores non-temporal data of breast cancer tumors. In this work, the raw data taken as input for the system is in the form of MRI scans and other 'raw' visual data, and the authors develop a system to extract the 3-Dimensional representation to present to user for exploration. Imaris [3], a commercially available software, works with grayscale cell images to segment and track cells. The segmentation is done via blob detection. The 4D data is analyzed by identifying 3D blobs that have sizes above a user provided threshold. Tracking in Imaris is done by associating cells in spatial neighborhood and by extent of overlap. 'Trails of cells' are provided to the user, who then corrects the mismatch, in contrast to our system, which performs matching based on generic shapes.

Successful visual data navigation and exploration systems generally employ a series of operations as proposed in [21]. The author describes that a visual representation system first gives an overview, then allows for zooming into items of interest from such an overview, filters out irrelevant details and then provides details of the interesting items on demand. The author also expands the idea into exploration of patterns by representing correlation of datasets and maintaining history of actions (annotations) to extract patterns. This technique for visual representation of data is very widely acknowledged in literature as the mantra for a good representation of multi-dimensional data.

Another important feature of a good Visual Analytic system is allowing effective user queries. As far as user queries are concerned, [5] provides a path querying system, where the user draws a desired path and a hyperlabel, which displays information on the objects found around the path. Our system on the other hand allows users to click on individual cells or select multiple cells based on which information pertaining to that cell or group of cells can be generated via a context menu.

The method proposed by [10] provides users with the ability to customize their queries by providing them with drop downs and search boxes. In this case, the user has to manually pick what they wish to view. In our system, we propose hyperlinked charts, where

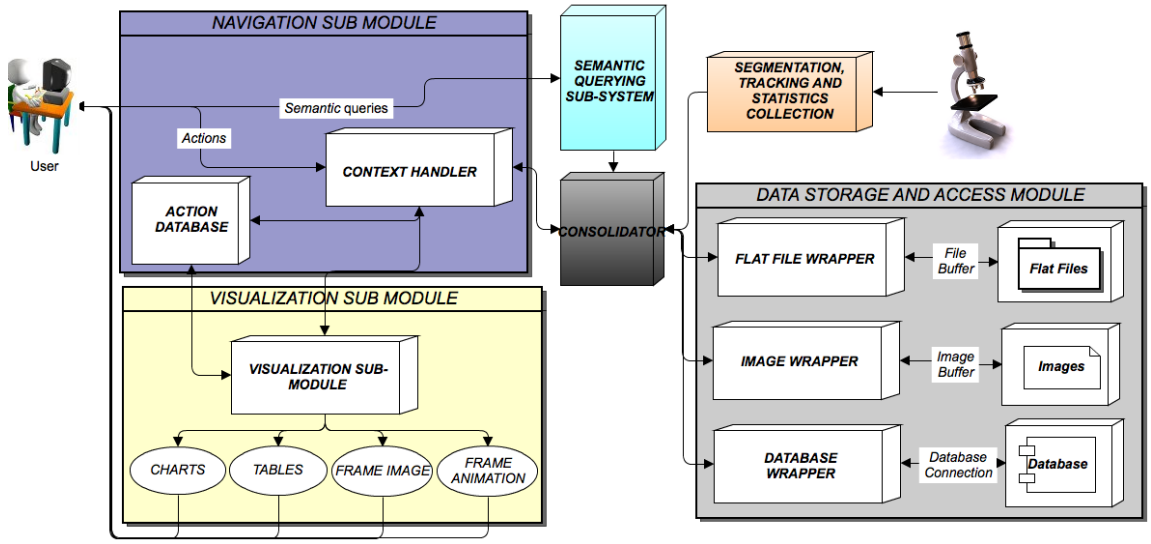

Figure 1. Block diagram of the system. Our system gets the time lapse images of live cellular biology experiments from the microscope. From these images the segmentation and tracking module (in orange) extracts semantic and statistical information from these images. Such collected and computed data is efficiently stored, managed and accessed by the data storage and access module (in light gray). The user is provided with a set of effective visualization entities (in yellow) to pictorially see the data, while the navigation module (in purple) lets the user to visually traverse through the hierarchy of data and the attributes. Finally we also propose a query system (in blue) that the user can use to query the underlying database.

the user can just click on a plot to create a different image associated with the chosen point on the plot. We also allow the capability of drill-down plots, such that a new plot can be created in real-time when the user chooses on a range in a plot he is currently viewing.

3 System Overview

An overview of the system is shown in Figure 1. The visual data from the microscope consists of raw images, void of any semantic information. Thus, segmentation of images is required to identify the parts of images that make up a cell and those that make up the background. *Segmentation* is a process of grouping pixels across boundaries of the cell as foreground and background. Similarly, *tracking* groups such identified regions of cells across time. As the images are time-lapse sequences of the snapshots of the state of cells, associating cells over time is important.

The information generated thus is stored in the database as statistical data related to cells, frames and experiments, and the semantic information is stored in flat files. The data present in the relational database, the image data and flat files are accessed by a central module called the *consolidator*, which uses the wrapper modules to access different types of data. Such a storage of hybrid data allows for easy and quick access. We have described this data storage and access system in Section 6. The consolidator also accesses image files using the Image Wrapper module. The consolidator essentially acts as an interface between the *data storage* and *access module* and the *navigation, visualization and query* modules described below.

In visual analytic systems there exists a loop between visualization and navigation which involves the user accessing and navigating the data through visual representations. This process iterated by a domain expert helps in the discovery of knowledge via hypothesis framing and validation. This tight-knit loop is shown in the *visualization and navigation* module in Figure 1. These actions of the user span a graph where the edges are formed by the actions for navigation initiated by the user and the vertices are the visualization entities. We describe this view for navigation and visualization in Section 7.

The validation of hypothesis also involves the user querying the system at a semantic level. In order to facilitate such an interaction with the system in terms of higher level semantics, we develop a query analysis subsystem depicted as such in the Figure 1. This type of interaction with the system requires storage and access of underlying data in a manner that can satisfy the requests for visualization and semantic queries. This query system has been described in detail in Section 8.

4 Segmentation and Tracking

In this section we describe the process of segmenting and tracking raw microscopy images to derive higher level semantic meaning from the images. This involves marking pixels that form the cell interior and the cell boundary (i.e. segmentation), matching cells between frames (i.e. tracking) and detecting interesting events in the cells. A detailed description of this process and how it differs from existing methods of segmentation and tracking has been given in our recently published work [13].

4.1 Segmentation

Segmentation is the process of dividing the image into cell regions and background. We carry out the segmentation in two steps: robust cell center marking (identified by high brightness) and detection of the cell boundary (around the identified cell centers). The output of the first step is used in the second step. Robustness of detection of cell centers is an important criterion for determining the quality of segmentation. In order to identify the center of the cell, we first identify the regions that are certain to be cells with high confidence by performing a contrast enhancement operation. This gives a consistent high response within the core of cell and a low response to regions outside. We use convolution with the Difference of Gaussians (DoG) kernel as the contrast enhancement technique (2 in Figure 2). Difference of Gaussians is applied by convolving the original image with Gaussian kernels twice – once with large sigma and again with small sigma, then subtract the result of the latter from that of the former. This operation specifically, is invariant to local intensity variations that are common within a cell. The output of this operation is a bright region in the core of the cell. The

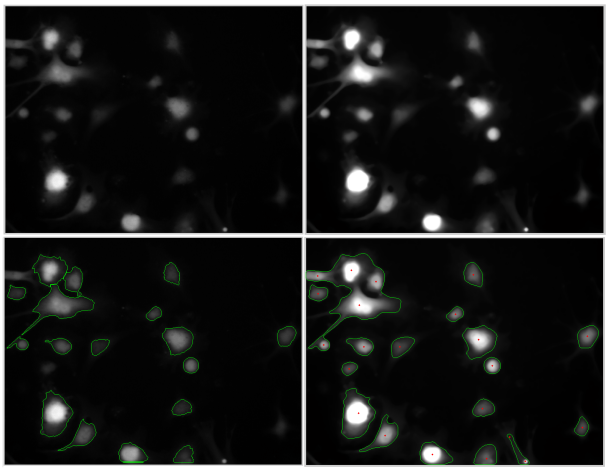

Figure 2. Segmentation: Clockwise from Top-Left: 1. Original Image, 2. Image after DoG contrast enhancement, 3. Cell centers and cell regions (after some morphological operations), 4. Final image after watershed.

boundaries of these bright regions are found by applying a few morphological operators, like *dilation, erosion and open operations*, to remove the effects of spurious intensity maximas, followed by edge detection (3 in Figure 2). The weighted centroid of that region, together with the intensity maxima gives the location of the cell.

With cell centers correctly identified, accurate cell boundaries that encompass the whole cell, including the branches of the neurons, need to be identified. We use the watershed segmentation algorithm [17] to split regions into as many cells as there are maxima points (cell centers). The Watershed algorithm considers the image as a height-field and partitions the image into regions of watershed – a partition refers to a region where, when water falls it flows to the same basin. The original image is complemented so that the high intensity cell centers become low intensity basins. A minima suppression operation is applied to remove noise, make the regions which are not part of the cell as plateaus, and remove over-segmentation. Then the watershed is applied to this image to partition it into multiple regions. The corresponding regions found which contain the cells are matched against cell centers. Every region (a basin in the watershed algorithm) which contains a cell center defines the entire extent of that cell (4 in Figure 2). This segmentation enables computation of cell parameters like the area of the cell, total florescent protein (GFP) content, and other static parameters.

4.2 Tracking by Graph Matching

The goal of *tracking* is to match cells in one image to those in another (Figure 3). We build a weighted graph with nodes as cells and edges between cells from two different images. The edge weights are computed as a function of different parameters including pixel overlap between the cells when the images are overlaid one over the other, difference in the cell area, difference in total brightness and Euclidean distance between their centroids in the pixel space.

Given this weighted graph, the graph matching that matches cells in adjacent frames is done using Hungarian Bipartite Matching [12] which is enhanced to include dummy nodes that handle one-to-many matching arising due to mitosis. In the event of the mitosis only one of the daughter nodes is associated to the parent node in the previous frame, while the other daughter cell is left unassociated. The unassociated daughter cell will be very close to the parent cell in 'distance' and 'pixel-overlap' attributes. Such situations characterize the event of mitosis (cell division). The event of cell death is said to have occurred when the cell is possibly left unassocaited (because its GFP content will have decayed below threshold or it died and moved too fast as to avoid any association). Thus tracking implicitly allows for detection of important cell events. Tracking, in general, aids computation of further parameters like speed of the cell, and first order statistics.

4.3 Event Detection

The detection of interesting events that occur in the lifetime of a cell is particularly useful to biologists who frame hypothesis on these events. Questions like which substrate enhances cell division or death can be answered by observing such events.

Cell Division or Mitosis is said to have occurred when a cell in an image captured at time t is invariably mapped to two different cells in the image at time $t + \Delta t$, and each of the new cells differ greatly in area to the parent cell. Also, it can be noticed that the cell's total intensity just before mitosis equals the sum of cell intensities of the daughter cells. This is true because the brightness of the cell is proportional to GFP protein content of the cell, and when the cell undergoes mitosis the protein content is divided among its daughter cells.

Cell Death: When a cell suffers apoptosis in a culture, it loses its branches and floats away in the medium. This floating is similar to Brownian motion and has much higher velocity than the firmly rooted live cells. Thus, when a cell dies, the cell tracking algorithm usually returns no match. The best match will have a large distance from the cell in previous image. Statistically, we term this event as cell death. Thus, a cell death is also characterized by its swift mobility – when its speed is far greater than the average speed of the other cells.

4.4 Evaluation of results

We evaluated our system for its ability to segment images accurately. As Figure 2 shows, the method clearly delineates the cells (including their branch like structures) and our neurobiologist collaborators were satisfied with the quality of segmentation. On the quantitative assessment side of our method for segmentation, comparison was made with the methods based on morphological operations (as described in [1] and [15]). The low contrast in our image were not apt for segmentation (Figure 4), and were undersegmented. Similar results were found with MRF segmentation method. We tested our method by counting cells manually and tabulating them against the results found by the segmentation module. The results are summarized in the following table (Table 1). We can see that our module performs well across the image types (low, medium and high density of cells in the image). However, increase

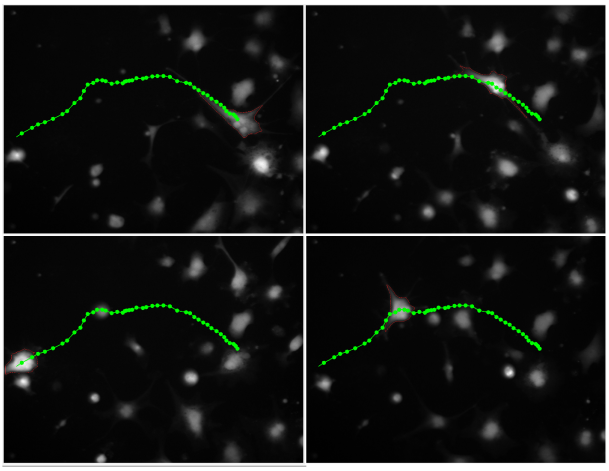

Figure 3. Tracking: Clockwise from Top-Left : Cell at time 1, Cell at time 15, Cell at time 25, Cell at time 46

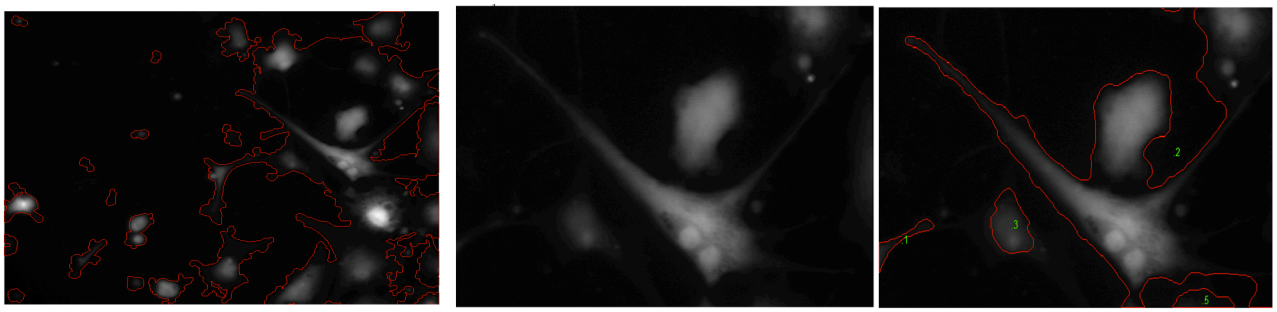

Figure 4. This gives the result of segmentation of our images using various techniques. The image on the left is result of segmentation with morphological operations described in [1]. Image in the middle is the original image. The last image is MRF segmentation with manual initialization

in density of cells cause some branches to overlap giving false positives in cell center identification, causing over-segmentation.

Density	Number of Images	Actual Number of cells	Automatic Segmentation Result
High	10	35.83	36.16
Medium	25	25	25.32
Low	10	5.5	5.5

Table 1. Segmentation results carried over different sets of images

The accuracy of tracking is directly affected by the temporal density of frames. Large time gaps between the snapshots generally lets cells change more in the unrecorded time, and this low rate of sampling of the state of cells causes the tracking to perform poorly. If a cell A in a frame at time t is associated with cell B in frame at time $t+\Delta t$, such an association is labelled a 'mismatch', and if a cell at time t goes unmatched to any cell at time $t+\Delta t$, we label it as 'missed'. Table 2 summarizes the results in tracking sub-system.

Cell's trail length in frames	Total number of mismatched cell pairs	Average missed cells per frame
0-5	(Missed all, debris)	NA
5-10	None	NA
10-15	None	NA
15-20	4	1.25
20-25	7	1.75
25-30	6	1.2
30-50	39	1.95

Table 2. Tracking results: Time delay 20 min

5 Semantic Interpretation of Space

Attaching semantic labels to syntactic data starts with classifying if a pixel in the input raw image belongs to a cell or not through the image segmentation process as described in the previous section. Once the pixels are classified, they are labelled with an integer index denoting a specific cell. All pixels that have the same label collectively represent a semantic cell. Using this semantic interpretation, parameters like area of the cell, perimeter, total brightness and other physical parameters of the cell that may have biological meaning can be computed.

The second level of semantics comes from the time domain. A cell can move between two consecutive images taken at two different time instances. All the pixels belonging to these two corresponding cells, as identified by *tracking*, in these two frames are given the same label. Using this semantic labelling through time, parameters like speed of the cell, cell division (mitosis), cell death (apostosis), rate of change of area and other first derivative statistical data that have biological meaning can be computed.

In other words, images represent the two spatial dimensions and a stack of images collected over time represents the temporal dimension (refer to Figure 5). Cells, when they move in space over time, sweep a volume in this spatio-temporal space. A point inside this volume belongs to the cell, and those outside do not. Thus, every point in this spatio-temporal space has semantic meaning representing if it belongs to a particular cell or not.

The third level of semantics is a supplementary labeling of the

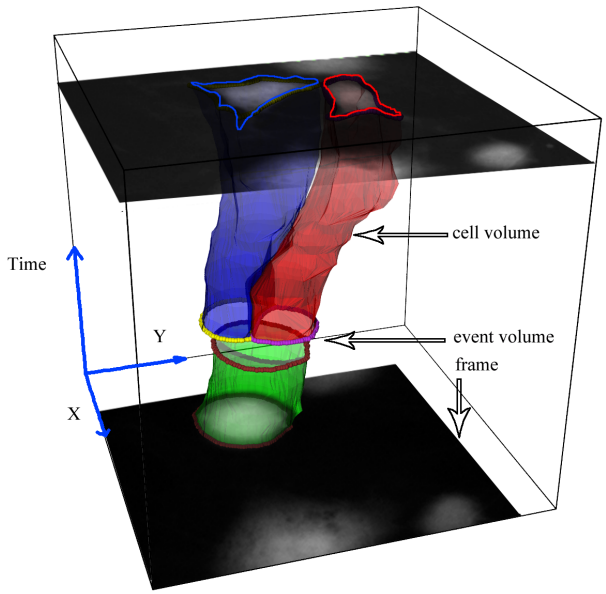

Figure 5. This image shows different semantic labels of the space using an actual data collected during an experiment. The entire 3D spatio-temporal space defines an *experiment*, a slice in the time domain defines a *frame*, the swept volume as shown in the figure defines a *cell*, the event is shown as a split in the swept volume.

points in this spatio-temporal space that defines an *event*. An event is an interesting phenomenon that occurs for a short period of time and usually is localized in space. For example, the events that we are interested in are cell divisions and cell deaths. These events are localized in the space in the neighborhood of the cells and can span a short time interval. Hence a point in this spatio-temporal space can have multiple semantic labels – those belonging to a cell and those belonging to an event.

The above formalization provides us with a framework that allows data representation, data management, and query processing to be centered around these four semantic elements: experiments, frames, cells, and events. Data management can create a hierarchy of data using these elements, all computed parameters can be associated with one of the above four semantic elements, and the query system can provide mechanisms to explore these four elements.

6 Data Management

Following the segmentation and tracking, various statistical parameters of the cell and frame are computed including area, perimeter, speed of motion, and total brightness of every cell, number of cells in every frame, etc. Data management systems in the context of visual analysis do not only try to optimize storage and access, but also have to be designed to optimize the navigation, exploration, and query processing as demanded by a typical user.

There are two kinds of data used in our system – first is the data that is used only for visualization like the raw image data that does not have semantic meaning but only has indexing, and the second kind of data like the statistical and computed data that has semantic meaning and hence will be queried upon. The former is usually the output of sensors and is stored in its native format – for example, the microscope output image is stored as image files (JPEG, TIFF, etc.). The computed data that has semantic meaning is stored in relational database management system (RDBMS) for ease in querying. But the efficiency of such RDBMS systems depends on the size of the data set. We improve the efficiency by identifying entropy reducing patterns in the computed data set using the following observation.

Semantic Statistical Data: For interactive visual analytics, fast and accurate data access is essential for unobstrusive interaction with the system. We use a relational database management system to store and query the computed, statistical data. The design of the RDBMS follows the application specific observation and needs of the data organization and query patterns. From the spatio-temporal analysis of our data set (Section 5) it is clear that the data can be classified in a hieararchical manner consisting of database tables for *experiments*, *frames*, *cells*, and *events*. The structure of the database is shown in Figure 6.

Semantic Partitions: Even within the computed data, there are special subsets of data that are all related to each other in the sense that all have the same semantic meaning and common usage– for example, after segmentation of the image, all collection of pixels belonging to the same cell will have the same semantic association, and all these pixels will be accessed and used together in any further querying or visualization. We call such data sets *partitions*. These 2D data sets, such as cell boundaries and cell-pixel association lists cannot be efficiently encoded in the form of a table in RDBMS. Further querying one data within a partition is equivalent to querying all data within the partition. So we store these individual partitions in different file formats – images or flat files, as is required by the nature of the data, and use meta links from the RDBMS systems to access the complete data.

These data in the flat files need to be formatted for use into the visual analytics system. For this, we have proposed the model of wrapper classes that read data from flat files, attach semantic meaning to the raw bytes and pass the information to the visual analytic system. This hybrid system of storing access information in RDBMS and lower (pixel) level details in flat files gives rise to an efficient data model that is also easy to implement. Further, such a system design can be queried on numerical data (statistical data) and answer visual (i.e. spatial) queries efficiently.

Semantic Images: The image data acquired from the microscopy system is a two dimensional spatial slice in the semantic space as described in Section 5. Visualization of images is achieved by first gathering the image data and attaching semantic meaning to the images from the data stored in *partition* files. The partition file consists of segmentation information in the form of pixel-cell association represented as image masks, also called as 'cell overlays'. The boundaries of the cells are stored as flat files as a sequence of (x,y) pixel coordinates. From an implementation point of view, our system has an image wrapper, which reads the image files, composes and formats them to show cell overlays and boundaries. The flat files are stored as length delimited array dumps, i.e. they contain raw bytes representing the lists of pixels prefixed with length of the following array and the identifier for the cell whose boundary is formed by that array. Caching and transfer from the hard disk is implemented using API buffers.

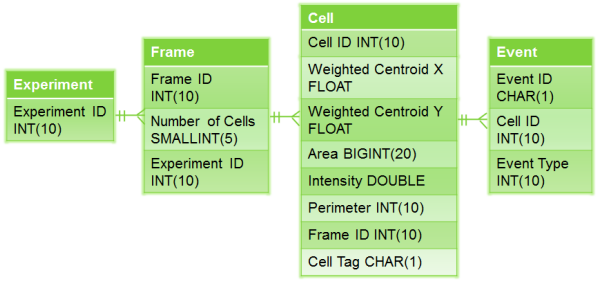

Figure 6. Entity-Relationship Diagram of the Database.

7 Visualization and Navigation

As we have seen in the earlier sections, data used in our system exists at multiple levels of hierarchy like experiments, frames and cells and in multiple formats like image and statistical data. Visual representation of this data in a way that communicates the semantic meaning is termed as *visualization*. Using one form of data visualization to access other data through possibly different levels of hierarchy or different attributes is termed *navigation*. A visual analytics system integrates these two, so that a user can recognize patterns in the correlating parameters.

7.1 Visualization Entities

The goal of a successful visualization system is to allow easy cognition. A user must be able to derive certain meaning from the visual representation of raw data. As human perception is tuned to finding relations, providing visualization methods that allow for easy projection of such inter-relation is the key to good analytics.

Data in our system can be categorized into two formats – raw data including the microscopy images that serve as the input to the system, and semantic or statistical data like attributes of a cells and events, which are derived from processing the input data. Our visualization entities should handle data of these formats. Further, since the input images are time-lapse images, our visualization system should also handle dynamic and kinematic properties of various objects.

Using the above mentioned classification, our system presents the following visualization entities:

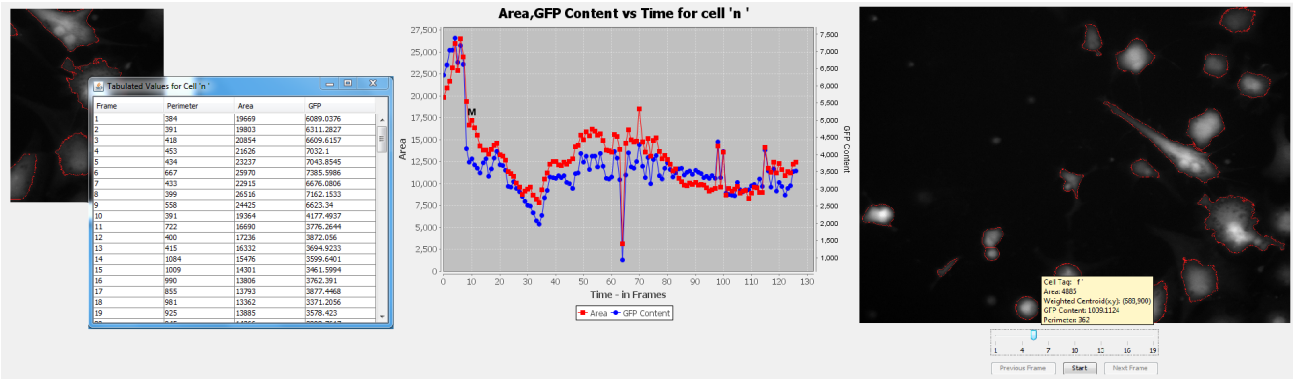

Figure 7. Visualization Entities: Tabular representation, Charts (showing how area and GFP content of a cell changes over time and a Mitosis event labelled M), Frame animation

Frame Animation: In order to provide users with the ability to visualize time-lapse data sets, for example, how cells change or move across time, we present an Image Animator (Figure 7), which is a time-delayed sequence of images that the user can interact with. This sequential display of images is essentially the *slicing* of the 3D space described in Section 5. The user can select (multiple) cells to visualize their motion in time in a new animation frame. Animations are a natural visualization mechanism to show 2D time-lapse data.

Visualization Charts: Statistical data in our system is predominantly time-varying. Line charts (as in Figure 7) are widely regarded as a good visualization technique for 1D time-varying data [6]. We have also provided the functionality by which a user can view multiple attributes at the same time using multi-line charts (as in Figure 7). Through such visualization techniques, the user can easily correlate different time-varying data. Further, we also label the charts at appropriate time instances with events in order to show further correlation between events and statistical parameters (Figure 7).

Tabular View: The tabular view is an alternative visualization method for statistical data, and is considered the best presentation technique when precise information about the data is required (as in [6]). We present the user with numerical data in a tabular form on demand (Figure 7). The tabular representation of data allows a user direct visualization of the parameters associated with one or more cells.

7.2 Navigational Entities

Navigation is described as the process of moving between one visual representation of the data to another through user interaction with the purpose of finding patterns in the data. An efficient navigation system unobtrusively and interactively fetches, composes and formats the data to present the next view of the data being analyzed.

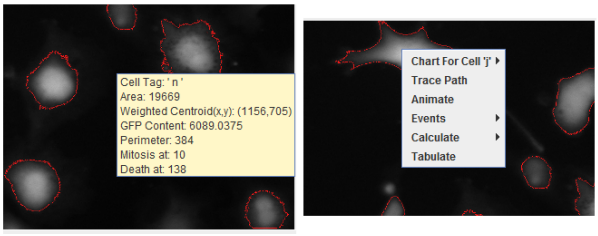

Figure 8. Navigational Entities: Tooltips and context menus.

The general principle of a visual analytics system is to present data that is essential for understanding of patterns etc., in the most uncluttered manner with the concept of 'details on demand'. This principle ensures that the user is not overwhelmed with the numerical or otherwise highly precise data when he is simply looking for an abstract view of the model. Further the navigation process requires that the visualization entities be cognizant of the semantic meaning of the entities being displayed. In our system we attach semantic meaning to the display primitives using data entities like cell masks through which every pixel of the image is aware of the cell it belongs to. This allows us to query the attached statistical parameters of the cells when the user interacts with the pixel.

There are two types of navigational data in our system – hierarchical data and attribute data. The hierarchical data, as represented in the RDBMS (Section 6), has four levels including experiments, frames, cells, and events. The attribute data defines the attributes of each of the objects in the above hierarchical levels. The system should be able to navigate to each level in the hierarchy as well as the attributes contained in each level. In order to navigate through the hierarchy we use two navigational entities, namely contextual menus and hyperlinks in charts. In order to navigate among attributes, we use ToolTips.

Contextual Menus: Contextual menus offer a set of navigation choices based on the current state of the system. Normally, the menu of choices corresponding to the selected object on which the contextual menu is invoked is generated on-the-fly. This gives a precise framework to implement navigation through the hierarchical data. One set of contextual menus has been implemented at the frame level, where a user can click on a cell or multiple cells to visualize information pertaining to the selected cells (Figure 8). The second set is at the visualization chart level where a user can select a time instant or a range and view an aggregated information related to the selected time frame. For example, a user can right click on any cell and view the average area of the cell using the context menu.

Hyperlinks in Charts: Using hyperlinked charts, a user can navigate from a chart that describes one or more time-varying attributes of a cell, to a specific frame just by choosing a point in the chart at that corresponding time instance. This then links to *frame animations* where the new image animator starts from the selected point in time. Thus hyperlinked charts provide users with the ability to navigate up the data structure hierarchy. The user can also select a time range on the chart and use the context menu to either generate cell statistics for that time range or drill down further into a new line chart which represents different cell parameters over the selected time range. This concept is also termed as *zooming* in [11]. This new chart or animation can further interactively lead to other part of data and types of visualization thus providing inter navigability

between the multiple modes of visualization. For example, a user can click on the Mitosis tag in the chart (Figure 7, center) and view the image where the cell undergoes mitosis.

Tool-Tips: A tooltip is an element in which when the user places the cursor at a particular visualized object and a small hover box displays information about the object. We extend the concept of tooltips by making them contextual in nature. In our system, when a user hovers the cursor over a cell, information about that cell such as when it divides, its area in the current frame, etc. are shown in the hover box (Figure 8), while when the cursor is over a chart, parameters like the time and the value are shown. This requires quick fetching of data for formatting and display which is made possible by our hybrid data storage described in Section 6.

8 Semantic Querying

An effective visual analytic system should not only be capable of representing and visualizing multi-modal data on request, it should also be capable of aggregating data to solve queries asked at abstract semantic level. The responses to these queries are generally required to be generated on the fly and thus elicit a conversion between the semantic queries to the low level queries that derive data from the data store.

In that direction, here we propose a model of querying language suited specifically for querying the large database of images, its semantics and the statistical data associated with the entities, all related to the cellular experimental data in neuroscience.

8.1 Query Space Formulation

We observe that the queries are typically related to the objects – frames, cells, and events, and their attributes. We assign unique integer ids to frames, cells, and events. The query space can now be thought of as an integer grid of the three dimensional space spanned by the frames, cells and events axes. A discrete point $P_{i,j,k}$ exists in this space if there is an event k associated with cell j in frame i, (Figure 9, top). A scenario with no event is tagged with a special event with id 0, Figure 9, bottom. In this formulation, the cell-frame plane (with event id 0) shows the life of the cells through time except the points where events happen. Many biological queries can be answered by finding the points in a subspace of this 3D space. The formulation of such queries starts with specifying a subset in each dimension and computing the space resulting through union or intersection of these subsets.

For example, let the frame-subset S_f be subsets of frames that are of interest in frame axis. Similarly let S_c and S_e denote the subsets of interest in the cell and event axes. Then the points in the union or intersection of these subspaces are returned respectively by the query functions

$$\mathsf{EVAL\_UNION}(S_f, S_c, S_e) = \{P_{i,j,k} | (i \in S_f) \vee (j \in S_c) \vee (k \in S_e)\}$$

$$\mathsf{EVAL\_INTER}(S_f, S_c, S_e) = \{P_{i,j,k} | (i \in S_f) \wedge (j \in S_c) \wedge (k \in S_e)\}$$

Pictorial illustrations of these operations are shown in Figure 10. We can extract the indices of just one of the axes – frames, cells, or events, by projecting the resulting points in the appropriate axes. Such a projection can be achieved by a simple *type casting* syntax. For example, the set of all cell indices T_c of a point set $\mathbf{P}$ is given by: $T_c = (\texttt{cell})\mathbf{P} = \{j | P_{i,j,k} \in \mathbf{P}\}$. Such type casting can be used, not only to get the indices of the objects, but also to get the attributes of the objects. For example, the set of areas of the cells A_c of the query point set $\mathbf{P}$ is given by $A_c = (\texttt{area})\mathbf{P} = \{\texttt{cell}[j][i].\texttt{area} | P_{i,j,k} \in \mathbf{P}\}$, where $\texttt{cell}[j][i].\texttt{area}$ gives the area of cell j at frame i. Specifically, we use multisets to allow repetition of values in the set which would enable easy computation of statistics on the values such as average and standard deviation.

The union, intersection and set difference queries along the same dimension are addressed using stand-alone set-operation functions. For example, if S_c and T_c denote two sets of cell indices, then their union, intersection and differences can respectively be computed using the function:

$$\mathsf{SET\_UNION}(S_c, T_c) = \{P_{i,j,k} | (j \in S_c) \vee (j \in T_c)\}$$

$$\mathsf{SET\_INTER}(S_c, T_c) = \{P_{i,j,k} | (j \in S_c) \wedge (j \in T_c)\}$$

$$\mathsf{SET\_DIFF}(S_c, T_c) = \{P_{i,j,k} | (j \in S_c) \wedge (j \notin T_c)\}$$

Similarly sets on frames and events will work with indices i and k. Many biologically relevant queries can be converted to these basic abstract query functions, and can be a powerful tool for hypothesis framing and hypothesis testing by the biologists.

8.2 Query Example

In this section we show how the theory of queries as formulated in the preceding sub section can be used to evaluate simple biologically relevant queries. To calculate the average area of a cell c within a range of frames f_1 to f_2, the query sequence will look as follows:

Let $F = \{f_1, \ldots, f_2\}$, and $\mathbb{U}$ be the universal set in that particular domain. Set of all query points that belongs to cell c between the range of frames is given by,

$$\mathbf{P} = \mathsf{EVAL\_INTER}(F, \{c\}, \mathbb{U})$$

The set of all areas of c in the given frame range is $A_s = (\texttt{area})\mathbf{P}$, and the average area is given by $A = \texttt{average}(A_s)$.

As a second example, let us find the set of all cells in which each cell undergoes both mitosis (cell divisions) and apostosis (cell deaths) in a given time interval (frame range). Note that `mitosis` and `apostosis` are event attributes, and can be used as a type casting, similar to `area` on a set of query points, to return a set of events that are of given type.

Let F be the set of all frames that are of interest. Let E_m and E_a be the set of events within the frame range of interest F that are labelled mitosis and apostosis respectively.

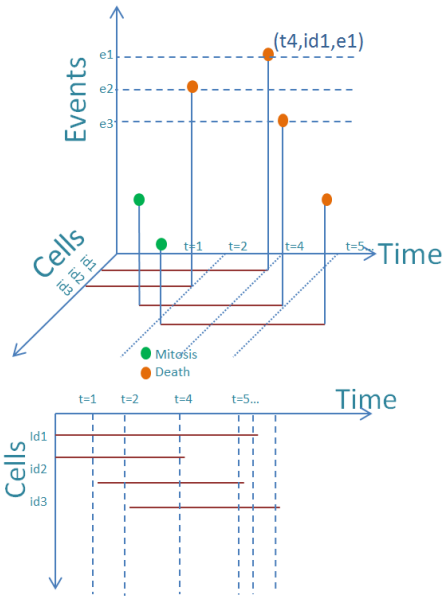

Figure 9. The *Q*uery space. The figure on top shows the three dimensional space spanned by cells, time and events. Each dot (green or orange) describes an event for cells with respective 'cell ids'. The bottom figure shows the cell-time plane of *no event*, (*event id = 0*), which represents the life line of cells.

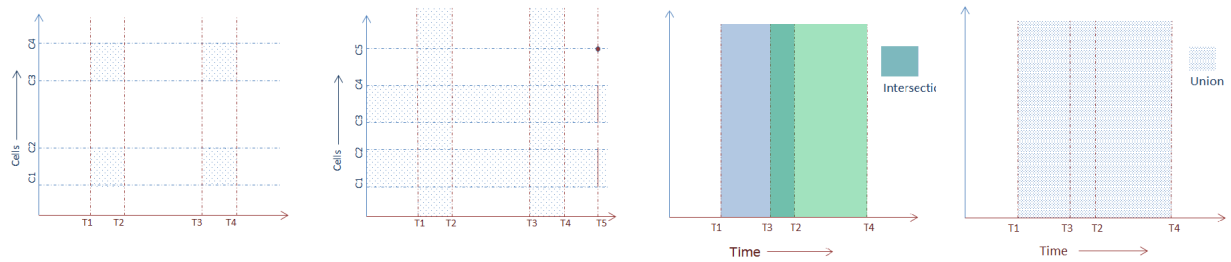

Figure 10. Left to Right, Illustration of *EVAL_INTER, EVAL_UNION, SET_INTER* and *SET_UNION*.

$$E_m = (\texttt{mitosis})(\text{EVAL\_INTER}(F, \mathbb{U}, \mathbb{U}))$$

$$E_a = (\texttt{apostosis})(\text{EVAL\_INTER}(F, \mathbb{U}, \mathbb{U}))$$

Let C_m and C_a be the set of cells that undergo mitosis and apostosis respectively within the frame range of interest. So,

$$C_m = (\texttt{cell})(\text{EVAL\_INTER}(F, \mathbb{U}, E_m))$$

$$C_a = (\texttt{cell})(\text{EVAL\_INTER}(F, \mathbb{U}, E_a))$$

Finally, set of all cells in which each cell undergoes both mitosis and apostosis is given by,

$$C_{\text{res}} = \text{SET\_INTER}(C_m, C_a)$$

Similarly many simple yet biologically significant queries can be answered by our query system. But there are complex queries that need *lists* data structure instead of *sets*. Further, nested structures like list of lists and sets of sets become important while computing functions like 'average area of each of the cells in a range of frames' that would return a list of numbers instead of a single number. We will strengthen our query system to handle such complex data structures and queryies, and implement the relevant access functions.

In our current implementation, we have provided a limited querying capability that would implement aggregate functions like `count`, `average`, `stddev` and the set function, `EVAL_INTER`, on three dimensions.

9 Evaluation of our System

The system, being an end to end pipeline, helps neurobiologists perform experiments and analyze the results. The proposed visualization and querying methods help them derive higher level biologically relevant knowledge from experiments. For example, one biologically important statistical pattern that the biologists observed using our system is that when a cell is about to undergo mitosis, it contracts, loses its branches and it's protein content aggregates. It can be seen from the chart in Figure 7, that at the time of mitosis (marked by M), the area of the cell shrinks rapidly. From our system, the biologists also noticed the decrease in cell metabolism in terms of GFP content and production just before, during and just after mitosis, as seen in the chart in Figure 7. Our collaborators could assess the time required for the the rate of GFP production to return to normal, accurately. Such discoveries of cellular behavior will be later used in other experiments to control cell divisions, mobility, metabolism and other processes.

Our neurobiologist collaborators enjoyed using the system and its fluidity in navigation through different data representations. They could visually corroborate the hypotheses and surmises that they had about the cell activities. Currently the query system is limited to what could be achieved through pull-down menus. A more powerful querying system will enable much more involved interaction of the users with the system.

10 Future Work

The modular design of our system allows for it to be modified in multiple ways to achieve scalability in functionality. In stem cell research itself, a type of analysis that biologists perform is *fate analysis*, where they analyze what the fate of a stem cell is. Stem cells change via the process of mitosis and metabolism, and finally become specific cellular tissue (like neurons or skin cells). It is useful to identify which stem cell ended up as a particular type of cell and whether the initial stem cell could actually produce multiple types of tissue. The *segmentation and tracking* module could be changed to allow identification of different types of cells based on their fluorescence under different conditions. From the data provided by the *segmentation and tracking* module, the visual analytics system can be extended to provide graphs that allow the user to see how different cells change thus allowing them to perform fate analysis.

We can also extend the existing system to allow recognition of cells other than the stem cells. The visual analytics system as it exists currently can take in such multi-cellular data and save them as different experiments. Due to our modular design, our system can be used in a different lab setup with very minor changes to only the Segmentation and Tracking parameters.

The current query system can be extended to provide higher level queries. A more formal representation of the query language can allow for a wider range of biological queries to be answered. Further, we are also working on visual query system that is as powerful as a textual query system. All the above mentioned extensions can allow for a robust Visual Analytic system that can provide immense analytical capability to cellular biologists.

11 Conclusion

In this paper, we explore the methods and implementation of a visual analytics system for biological data. We have developed methods of segmentation and tracking of human neural stem cells and built a visual analytical system of the data collected from such a tracking system while giving importance to ease of use of visualization and intuitive navigation. We have also developed an interpretation of the spatio-temporal space in order to explore the possibility of querying the system for regions of interest in such a space. We demonstrate the capability of the querying language by phrasing biologically relevant queries in abstract form. Such an interpretation also requires a fast access to layered and hierarchical data which is enabled by our hybrid data management model that satisfies the requirement of interactivity of the visualization system and the requirement of precision of a query system.

In essence, we have an end-to-end system that acquires image data, demarcates the semantic entities, provides visualization of the dynamics of those entities and allows users to navigate between many visualization entities, and finally gives a framework for solving many biologically relevant queries.

REFERENCES

[1] Anoraganingrum, D. Cell segmentation with median filter and mathematical morphology operation. In *Proceedings of International Conference on Image Analysis and Processing*, pages 1043–1046, 1999.

[2] Belongie, S. and Malik, J. and Puzicha, J. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509 –522, 2002.

[3] Bitplane Scientific Software. Imaris: 3d and 4d real-time interactive data visualization. http://www.bitplane.com/go/products/imaris, 2011.

[4] Boutros, P.C. and Okey, A.B. Unsupervised pattern recognition: an introduction to the whys and wherefores of clustering microarray data. *Briefings in bioinformatics*, 6(4):331–343, 2005.

[5] Bruckner, S. and Solteszova, V. and Groller, M.E. and Hladuvka, J. and Buhler, K. and Yu, J.Y. and Dickson, B.J. Braingazer-visual queries for neurobiology research. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1497–1504, 2009.

[6] S. Few. *Improve Your Vision and Expand Your Mind with Visual Analytics*. Perceptual Edge, 2007.

[7] Kass, M. and Witkin, A. and Terzopoulos, D. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.

[8] Keim, D. and Mansmann, F. and Schneidewind, J. and Thomas, J. and Ziegler, H. Visual analytics: Scope and challenges. *Visual Data Mining*, pages 76–90, 2008.

[9] Keim, D.A. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002.

[10] Kincaid, R. and Dejgaard, K. Massvis: Visual analysis of protein complexes using mass spectrometry. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 163–170, 2009.

[11] Kohavi, R. and Provost, F. Glossary of terms. *Machine Learning*, 30(2/3):271–274, 1998.

[12] Kuhn, H.W. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[13] Kulkarni, I. and Mukherjee, U. and Sontag, C. and Cummings, B. and Gopi, M. Robust segmentation and tracking of generic shapes of neurostem cells. In *Proceedings of the IEEE Symposium on Healthcare Informatics, Imaging, and Systems Biology*, 2011.

[14] Mange, R. and de Heras Ciechomski, P. and Swartz, M. seecell: Visualization and tracking dedicated to cell analysis. In *Proceedings of the International Conference on Innovations in Information Technology*, pages 707–711, 2009.

[15] Meyer, F. and Beucher, S. Morphological segmentation. *Journal of visual communication and image representation*, 1(1):21–46, 1990.

[16] Mukherjee, D.P. and Ray, N. and Acton, S.T. Level set analysis for leukocyte detection and tracking. *IEEE Transactions on Image Processing*, 13(4):562–572, 2004.

[17] Najman, L. and Schmitt, M. Watershed of a continuous function. *Signal Processing*, 38(1):99–112, 1994.

[18] Petushi, S. and Marker, J. and Zhang, J. and Zhu, W. and Breen, D. and Chen, C. and Lin, X. and Garcia, F.U. A visual analytics system for breast tumor evaluation. *Analytical and quantitative cytology and histology/the International Academy of Cytology [and] American Society of Cytology*, 30(5):279, 2008.

[19] Ray, N. and Acton, S.T. and Ley, K. Tracking leukocytes in vivo with shape and size constrained active contours. *IEEE Transactions on Medical Imaging*, 21(10):1222–1235, 2002.

[20] Sato, Y. and Jian Chen and Zoroofi, R.A. and Harada, N. and Tamura, S. and Shiga, T. Automatic extraction and measurement of leukocyte motion in microvessels using spatiotemporal image analysis. *IEEE Transactions on Biomedical Engineering*, 44(4):225 –236, 1997.

[21] Shneiderman, B. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the IEEE Symposium on Visual Languages*, pages 336–343, 1996.

[22] Souvenir, R. and Kraftchick, J. and Shin, M. Intuitive visualization and querying of cell motion. *Advances in Visual Computing*, pages 1061–1070, 2008.

[23] Yang, X. and Li, H. and Zhou, X. Nuclei segmentation using marker-controlled watershed, tracking using mean-shift, and kalman filter in time-lapse microscopy. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 53(11):2405–2414, 2006.