Spring 2012

# Novel computer vision algorithms for automated cell event detection and analysis

In Ae Hur
*University of Iowa*

# NOVEL COMPUTER VISION ALGORITHMS FOR AUTOMATED CELL EVENT DETECTION AND ANALYSIS

by

In Ae Hur

An Abstract

Of a thesis submitted in partial fulfillment
of the requirements for the Doctor of
Philosophy degree in Biomedical Engineering
in the Graduate College of
The University of Iowa

May 2012

Thesis Supervisor: Associate Professor Michael A. Mackey

# ABSTRACT

Live cell imaging is the study of living cells using microscope images and is used by biomedical researchers to provide a novel way to analyze biological functions through cell behavior and motion studies. Cell events are seen as morphological changes in image sequences, and their analysis has great potential for the study of normal/abnormal phenotypes and the effectiveness of drugs. While current quantitative cell analysis typically focuses on measuring whole populations of cells, we need to be able to recognize cell events at the single cell level, identify these events automatically, and analyze these events over time. For this reason, we developed and evaluated several novel automatic single cell event detection and analysis methods based on a detailed knowledge of the cell cycle and other cell event characteristics. The first method detects significant events within the temporal sequence using a machine learning method to use features derived from segmented cell images. We used a Neural Network (NN) algorithm to classify cell events to pre-defined categories. The second and third methods apply statistical and econometric techniques originally developed for time-series analysis of financial markets to facilitate the identification of cell entry into mitosis. We developed graph trend analysis and paired graph analysis methods from trend analysis and pairs trading to determine significant data points in cell feature data. The final method determines the position of cells in order to associate daughter cells with their parent cells after mitosis using Kalman filter techniques. By using the Kalman filter approach, we estimated future cell border centroid positions and successfully associated daughter cells with their parent cells after mitosis. In this study, the performance of these novel computer vision algorithms for automatic cell event detection and analysis were evaluated and verified by applying models to different image sequences from the Large Scale Digital Cell Analysis System (LSDCAS). The results show that the approaches developed can yield significant improvements over existing algorithms.

Abstract Approved: _____
Thesis Supervisor

_____
Title and Department

_____
Date

# NOVEL COMPUTER VISION ALGORITHMS FOR AUTOMATED CELL EVENT DETECTION AND ANALYSIS

by

In Ae Hur

A thesis submitted in partial fulfillment
of the requirements for the Doctor of
Philosophy degree in Biomedical Engineering
in the Graduate College of
The University of Iowa

May 2012

Thesis Supervisor: Associate Professor Michael A. Mackey

Graduate College
The University of Iowa
Iowa City, Iowa

CERTIFICATE OF APPROVAL

---

PH.D. THESIS

---

This is to certify that the Ph.D. thesis of

In Ae Hur

has been approved by the Examining Committee
for the thesis requirement for the Doctor of Philosophy
degree in Biomedical Engineering at the May 2012 graduation.

Thesis Committee: _____
Michael A. Mackey, Thesis Supervisor

_____
Fiorenza Ianzini

_____
Joseph M. Reinhardt

_____
David G. Wilder

_____
Mona K. Garvin

To my family at home and church.

ACKNOWLEDGMENTS

Looking back, I am surprised and at the same time very grateful for all I have received. All these years of PhD studies have certainly formed me as a researcher and are full of gifts.

I am deeply indebted to my advisor Dr. Michael A. Mackey whose help, stimulating suggestions and encouragement assisted me throughout the research. This work would not be complete without his guidance. He has always encouraged and enlightened me through his wide knowledge and his deep intuitions, and led me to right direction for the research. I sincerely thank Dr. Fiorenza Ianzini for her help, interest and support with regard to my research. Special thanks go to Dr. Joseph M. Reinhardt for his invaluable help on changing my research path to quantitative image analysis. I also thank Dr. David G. Wilder and Dr. Mona K. Garvin for their comments and ideas to my research. I would like to express my gratitude to Dr. Joon B. Park and Dr. HyonSook Y. Park for their endless care and support to my PhD student life in Iowa.

It was a pleasure to share doctoral studies and life with wonderful academic siblings John Kalantari and Dr. Elizabeth A. Kosmacek, and with many friends, especially NaJung, JeongYoon, HyunJung, MiJin, Liz and Aaron, in US and Korea.

Thank also to my family, my dad SungRak, my mom JaeHyun, and my brother ManKi, for their care and pray. The years spent in Iowa would not have been as wonderful and successful without my family.

Last but not least, a big thank you to my Pastor, ManHee and my church members.

ABSTRACT

Live cell imaging is the study of living cells using microscope images and is used by biomedical researchers to provide a novel way to analyze biological functions through cell behavior and motion studies. Cell events are seen as morphological changes in image sequences, and their analysis has great potential for the study of normal/abnormal phenotypes and the effectiveness of drugs. While current quantitative cell analysis typically focuses on measuring whole populations of cells, we need to be able to recognize cell events at the single cell level, identify these events automatically, and analyze these events over time. For this reason, we developed and evaluated several novel automatic single cell event detection and analysis methods based on a detailed knowledge of the cell cycle and other cell event characteristics. The first method detects significant events within the temporal sequence using a machine learning method to use features derived from segmented cell images. We used a Neural Network (NN) algorithm to classify cell events to pre-defined categories. The second and third methods apply statistical and econometric techniques originally developed for time-series analysis of financial markets to facilitate the identification of cell entry into mitosis. We developed graph trend analysis and paired graph analysis methods from trend analysis and pairs trading to determine significant data points in cell feature data. The final method determines the position of cells in order to associate daughter cells with their parent cells after mitosis using Kalman filter techniques. By using the Kalman filter approach, we estimated future cell border centroid positions and successfully associated daughter cells with their parent cells after mitosis. In this study, the performance of these novel computer vision algorithms for automatic cell event detection and analysis were evaluated and verified by applying models to different image sequences from the Large Scale Digital Cell Analysis System (LSDCAS). The results show that the approaches developed can yield significant improvements over existing algorithms.

TABLE OF CONTENTS

LIST OF TABLES

Table

LIST OF FIGURES

Figure

# CHAPTER 1

# BACKGROUND AND RESEARCH PURPOSE

## Live Cell Imaging Technology

Live cell imaging is the study of living cells using images from imaging systems such as microscopes and it has become an important research technology in most cell biology laboratories as well as in neurobiology, pharmacology, and many other related biomedical research disciplines. Live cell imaging allows one to observe continuous cell fate changes, and to examine cell viability through cell behavior and motion studies. Digital image processing and computer vision applications for live cell imaging have greatly facilitated the study of cell dynamics.

Since Schleiden and Schwann described an individual cell structure using a primitive light microscope in 1837[1], cell analysis using microscopes has become a basic research method for biologists. Although cell culture methods had developed in the 1920s to study eukaryotic cell division, research on cell division was restricted to fixed specimens until the late 1940s[2,3] and researchers could only observe the morphology of cells and macromolecules including chromosomes from static images. The development of video technology in the early 1980s led to the addition of long-term time-lapse microscopy[3–6], in which cells growing on the microscope stage are photographed at specific intervals over periods of several hours to a month[1,7,8]. With the improvement of long-term time-lapse microscopy technology, researchers could observe living cells. Further, automatic microscope stage controllers developed in the 1990s have also contributed to advance live cell imaging technology[9].

Time-lapse image sequences can reveal the dynamic behavior of cells including multiple cell division cycles better than other traditional biological methods[4]. Researchers can understand cells in culture, in particular, cell migration and division by the cell border, the position and behavior of granules, nucleus, and nucleolus of cells using image

analysis of these sequences. In addition, the length of a generation can be estimated by means of cell death and division. Mitotic onset can be observed by the nuclear envelope breakdown and cell round up[10]. Cell round up (RU) is a spherical shape change of a cell with decreased cell perimeter and increased mean intensity. In one study, a recognized model of human stem cell migration and proliferation was measured by the ability of growth factor to chemo-attract cells and/or simulate proliferation of cells. These data also can be used to analyze cell velocity and division[9].

Microscopes with a digital camera and environmental control system are basic components that must be present to acquire the image sequence. A motorized microscope with a digital camera in a live cell imaging system can select particular microscope fields and capture images at defined time intervals. Imaging cellular processes such as cell division involves the analysis of two daughter cells at a scale of about 10 μm, a range of time intervals from milliseconds to hundreds of minutes. A standard microscope equipped with 4X, 10X, and 20X objectives is adequate[9]. Researchers typically use a digital CCD camera to record live cell images. The most important feature of a live cell imaging system is its ability to maintain cells in normal physiological conditions on a microscope stage for the observation of particular cell events. This feature is accomplished with an environmental control system. To maintain cell survival for up to weeks of observation, physical parameters of the chamber such as temperature, $CO_2$ and humidity must be maintained[11]. The cell culture environment varies depending on cell line and the purpose of the experiment. The stage incubator is constructed to fit on top of the microscope stage table. Commercial stage incubators are a closed dish for the active control of temperature and gas atmosphere. In addition, the environmental control system allows for the imaged plate to change positions according to the location change of the imaging objective as time passes.

The Large Scale Digital Cell Analysis System (LSDCAS)

LSDCAS was invented at University of Iowa for the study of radiation-induced mitotic catastrophe[8]. It was later improved for the study of the dynamics and non-equilibrium properties of cells using standard culture systems[12]. Traditional methods of studying cells involving chemical fixation are conventional and provide snapshots of cell morphology and architect[13], but cannot reveal real-time cellular interactions caused by outside stimuli, abnormal cell division and other dynamic events. Furthermore, prior live cell imaging methods are very expensive and the analysis software is impossible to modify for specific goals. To overcome these limitations, LSDCAS was designed as a new research tool for live cell analysis within an open platform. LSDCAS image stream can be recorded from a few days to weeks which enables quantitative cell population studies. LSDCAS has been used in numerous studies such as cell motility, cell death, mitotic catastrophe, dendritic cell / tumor cell interactions, intracellular pro-oxidant detection using fluorescent probes, and wound healing[12,14].

Microscopes for live cell analysis systems can be divided into fluorescence and phase-contrast; LSDCAS has both microscopes. Fluorescence live cell imaging systems are especially important because they can reveal drug delivery kinetics and dynamics of intracellular structures such as kinetochores and microtubules[15,16]. But the systems using fluorescence have two major issues: photobleaching[4,17,18] and phototoxicity[17,19]. Photobleaching, also called fading, occurs when a fluorophore becomes less fluorescent due to exposure to light; Phototoxicity refers to the reduction of the lifespan and liveliness of a cell due to the toxic side effects of the fluorescent dye. Phototoxicity cannot be eliminated, but can be reduced by using UV filters, neutral density filters, and longer excitation wavelengths[12]. This is the reason why we used phase-contrast image acquisition of LSDCAS to study cell events without any side effects from the fluorescent dye.

LSDCAS is an analysis system that includes hardware and software related to live cell imaging technology. The LSDCAS hardware includes microscopes, digital cameras, environmental control systems, storage arrays and servers[20–22]. Autofocus, cell tracking[22,23], cell segmentation[20–22], mitosis detection[12,20], and motility analysis[24] programs are provided as software. LSDCAS can be divided into three components: data acquisition, data archiving and data analysis. The data acquisition component utilizes two microscopes with CCD digital cameras and stage incubators; An Olympus IX-70 inverted phase microscope (Olympus, Tokyo, Japan) equipped with a Plexiglas stage incubator (Olympus, Tokyo, Japan) and an Olympus IX-71 microscope (Olympus, Tokyo, Japan) with a LiveCell stage incubator (Pathology Devices, Inc., Westminster, USA). To acquire image sequences, a flask or multi-well dish is used on the microscope stage. To generate representative data for analyzing cell dynamics, image sequences are acquired from multiple locations on the culture dish. The microscopes with digital cameras and stage incubators are placed on air tables to minimize vibration artifacts. The data acquisition component has several features: 1) autofocus control for maintaining appropriate focus from focus changing due to thermal drift, 2) temperature controls for maintaining optimal temperature for each cell line, 3) electronic shutter controls for minimizing toxicity due to illumination, and 4) stage movement controls for maintaining appropriate imaged location. Autofocus, stage movement, and illumination shutter are controlled by Ludl MAC2000 controller (Ludl Corp., Hawthorne, NY, USA).

The second component of LSDCAS, the data archiving system, stores zlib-compressed 8-bit images in a custom file format. Captured images from the image acquisition component are transferred by campus network to the LSDCAS data center where experimental metadata is then stored in a relational database (PostgreSQL). LSDCAS users access the image data and experiment metadata using software written in Grails that is provided as several Tomcat web application. Initial early view of the experimental results can be seen as video clips which are automatically generated in

various video formats and presented to users via the web application. An automatic backup function is also provided in the data archiving component by Bacula which is an open source network backup software. In the data analysis component, cell division probability, generation length, and motility are determined from the image data by analysis programs. These programs are based on cell tracking and segmentation function from the sequence of images. Cell tracking is a method that can follow a cell from one image to the next. Level-set segmentation is used to identify cells and artifacts in an image sequence. All captured, segmented, and tracked cell images can be streamed at a steady rate or frame-by-frame, and forward or rewind stream is possible using casViewer; a graphical user interface (Fig. 1). CasViewer was designed to enable manual annotation of cell morphological changes, the events, by researchers and it stores the event information in eXtensible Markup Language (XML) files in a structured format that encapsulates the event structure. Figure 2 describes workflow of LSDCAS system with automatic event detection.

There are a number of improvements that can be made to LSDCAS; development of various applications that expand quantitative cell analysis, increase primarily the accuracy of cell segmentation and tracking, and improve cell event detection which is the primary philosophy behind the LSDCAS. The development of automatic cell events detection especially can help to reduce the cost of manual analysis.

## Cell Cycle and Events Analysis in LSDCAS

### Cell cycle phases and predefined events

The data analysis component in LSDCAS can detect both cells and artifacts in live cell image sequences using various analysis techniques, and identify cell events manually. Cell events such as cell division are seen as morphological changes in LSDCAS image sequences. The applications related to cell event analysis provide general methods that identify and analyze most adherent cell lines.

The cell cycle is the series of events leading to cell division and replication when eukaryotic cells reproduce. The cell cycle is complex and highly regulated, so the sequence of cell events corresponds to the completion of activities in each phase and the start of the next. It can be briefly divided in two periods: interphase and mitosis phase. Cells need to take in nutrients for growth before entering cell division. During interphase many biosynthetic cell activities occur to prepare for cell division, such as DNA and protein synthesis. Interphase proceeds in three stages, G1, S, and G2. During mitosis, cell growth stops and cellular energy is focused on the orderly division into two daughter cells. Mitosis has been broken down into several distinct phases, sequentially known as prophase, metaphase, anaphase, and telophase. In prophase, the nuclear envelope breaks down to allow the microtubules to reach the chromosomes, and chromosomes are captured by microtubules and separated in metaphase. Then the chromatid moves to opposite poles of the cell in anaphase and the nuclear envelopes of the daughter nuclei are formed in telophase. Cytokinesis directly follows mitosis in which cytoplasmic components are segregated to complete the formation of two identical daughter cells. The cell cycle phase in mitosis can be distinguished through DNA binding dyes and fluorescent protein (e.g. green fluorescent protein; GFP) using fluorescence microscope. But some sub-phases of mitosis and interphase can only be identified by cell morphological changes[23].

LSDCAS has twenty predefined cell event types related to morphological changes, for analysis of live cell image sequences (Table 1). These event types are related to cell cycle events and are visibly detectable. Additional events can be added to the analysis application framework, but a user must view and manually annotate the image sequences. Out of the twenty predefined events, the most significant morphological changes are round up (RU) and flatten out (FO). As mentioned earlier, mean intensity is increased and perimeter is decreased when cells enter mitosis at the RU state. Thus, the cell is in the FO state following the cell division that occurs in the RU state. If RU and

FO are defined, it is ease to detect events occurring after division. The LSDCAS analysis applications are used to describe the statistical characteristics of cell events and categorize the events.

Segmented object features and statistical characteristics

Every detected object in image sequences including cell events have ten features; timestamp, id, field number, frame number, mean intensity, perimeter, area, shape factor, x coordinate of centroid, and y coordinate of centroid. The features are measured by the analysis applications. *Timestamp* is the unix time stamp which provide a way to track time as a running total of seconds. Researchers can infer the time interval of the experiment from the timestamp. *Id* is a unique number for each cell and is assigned by tracking analysis. If two daughter cells are produced after a cell division, one daughter cell will keep the id of the parent cell and the LSDCAS system will assign a new number to the other daughter cell. *Field* and *frame number* refers to the microscope field number and an image number in ascending order, respectively. *Mean intensity* is calculated from the brightness of the pixels within an object; cell or artifact. It is depends on the resolution of the microscope and image brightness of each image. The *perimeter* of the cell is computed by a recursive distance calculation between two adjacent pixels, and *area* is calculated based on perimeter value. *Shape factor* is defined as (4.0 * Pi * area) / (perimeter * perimeter) and the value ranges from 0.0 to 1.0 (perfect circular form). Generally, RU cells have a shape factor of about 0.87. *Centroid coordinates* are obtained for the cell border using the distance data calculated when the perimeter is determined. These shape-based features can be used to distinguish between cells because they contain the majority of the information about cell morphological changes.

Based on these features of the cell border of the image, motility can be computed. Motility can be calculated by fitting distance curves to estimate cell speed in the LSDCAS image sequences. We can understand cancer metastasis, the shaping of organs

and tissues of an embryo, wound healing, and the generation of new blood vessels[24] by cell motility. Furthermore improving cell event detection using cell motility is possible because cell motility is related to cell viability. Cell motility information is stored as a text file and includes cell trajectory analyses, mean cell speed (microns/h), standard deviation of cell speed, and 95% confidence intervals on the mean. Mean cell speed time series and motility histograms are shown in Figure 3.

## Manual cell event analysis

To obtain manually detected cell event data, a user assigns one of the predefined events to particular cells in an image sequence using CasViewer. As shown in Figure 4B, the events are organized in a dialog box. User can also manually move to the next event in the cell image sequence (Fig. 4A). All of the events have a field, event name, id for cell event tree, frame number, and parent id information and it is stored in XML file format. More than one event tree file can be stored from an image stream. These manually detected events also can be modeled as temporal sequences using a graph-based representation as a directed acyclic graph (DAG)[25]. A DAG is a collection of vertices and directed edges with each edge connecting two vertices (Fig. 5A). It is used to depict cell events observed in LSDCAS image sequences. Every sequence also known as a cell event tree, has a root (i.e. Identify Cell; IC), then an event such as an RU, ND, or FO is connected as a child event and so on as shown in Figure 5B. A DAG is used in various fields such as Gene Ontology graph[26], structural RNA analysis[27], phylogenetic trees[25], and neural network model[28]. Researchers can verify the events as a DAG format.

Cell generation time distributions can be calculated using the cell event data obtained using casViewer. For cell growth analysis, the entry of mitosis (RU) and the exit from mitosis (FO) are determined using the LSDCAS event analysis main window. The user indicates these events, and the analysis application calculates the time between the RU and FO of the cells that bracket the generation time, mitosis. Then the mean cell

generation time is calculated using histogram analysis (see Figure 6). Mean cell generation time can be used to demonstrate the differences of the growth rate among cell lines and conditions.

## Summary and Research Purpose

Live cell imaging technology has become a widely accessible research tool for cell event analysis. Digital image processing and computer vision applications for live cell imaging analysis have greatly facilitated the study of cell dynamics and LSDCAS is designed to understand and determine cell motion studies using naïve cells. Detecting and analyzing cell motility and division is essential for live cell studies and automated analysis applications are crucial for research productiveness.

The goal of this research is to develop novel methodologies of automatic cell event detection and recognition at the single cell level based on a detailed knowledge of the cell cycle and other cell event characteristics. To achieve this goal, we applied our developed approaches to different image sequences to help show how our novel methodologies can yield significant improvements over existing models. We used a neural network model of machine learning methods to determine significant cell events, then applied our novel algorithms to identify cell division. The first novel method detects significant events within the temporal sequence of mitosis using graph trend analysis. The second method applies paired graph analysis to detect cell entry into mitosis. The final method determines the position of cells in order to associate daughter cells to their parent cell after mitosis. These new algorithms use the following concepts: graph based data analysis, time-series data analysis, and applied Kalman filter.

Our methods to detect cell division in image sequences involve the detection of feature changes associated with cell division. Cell division causes the cell to round up (RU) and this phase can be identified by a bright circular shape in phase-contrast microscope images. Further, the cell features (i.e. timestamp, id, field number, frame

number, mean intensity, perimeter, area, shape factor, x coordinate of centroid, and y coordinate of centroid) can be used to model feature trends associated with cell division. Especially, the four features related to the cell morphology (i.e. mean intensity, perimeter, area, and shape factor) can be used to identify events in mitosis.

### Significant cell events determination using machine learning method

Machine learning is an effective method of solving problems that involve the determination of behavior based on empirical data or past experience. The task of machine learning is to learn a mapping from input to output by optimizing parameters. The three goals of machine learning are: 1) to learn knowledge about input and output relations; 2) to make proper decisions based upon these input-output relationships; and 3) to improve the performance based on input data[29–31]. A training set is selected of a typical morphological shape of a specific cell event in the LSDCAS image streams to determine characteristics of each training cell event. After the training process, acquired characteristics of specific cell events are used to identify test sets which do not contain any data used for training. The results are presented and the performance of the machine learning method for LSDCAS dataset is demonstrated in Chapter 2.

### Cell entry into mitosis detection using time-series data analysis methods

Time-series data consists of numerical values over a time interval. The time-series data analysis is a widely used method in economics to forecast markets and make profits. In time-series data, the independent variable (x) is discrete time and the dependent variable (y) takes values dependent on time. The data in LSDCAS can be shown as time series by cell id. Each cell has ten different time-series data from ten features. These are timestamp, id, field number, frame number, mean intensity, perimeter, area, shape factor, x coordinate of centroid, and y coordinate of centroid. Each time-series data has frame

information as over time points {x} and measurements over the previous ten variables as the y values. We analyzed multivariate time-series data from each cell to detect outliers and/or uncommon trends in an attempt to identify significant cell events. In order to successfully distinguish between phenomena of interest and stabilized data, different kinds of information are normally required. For valid detection in this research, we used the concepts of trend analysis and pairs trading. Trend analysis is a method that refers to the concept of collecting information and attempting to spot a pattern, or trend, in the information. Graph trend analysis, derived from trend analysis, uses four out of the ten measured features (the mean intensity, perimeter, area and shape factor) to find outliers. Pairs trading, a popular time-series data analyses methods in econometrics, was also used to develop a novel paired graph analysis method. This method makes pairs from four features to find significant cell events. We have implemented these time-series data analysis methods for the LSDCAS. The results and the performance of graph trend analysis and paired graph analysis are presented in Chapters 3 and 4, respectively.

### Associating daughter cells with their parent cell after mitosis using applied Kalman filter

Kalman filtering is an algorithm which estimates the state of a linear system by mathematical recursive calculation. It enables the prediction of the next state of a system given its prior states. In order to detect accurate histories of individual cell fates, we need to link daughter cells to their corresponding parent cells. In this research, we used a Kalman filter as a suitable cell centroid estimator and estimated prior cell position given the current daughter cell locations. The first step of the reverse Kalman filter for associating daughter cells to their parent cell is to estimate next position of the cell centroid by adding a linear weighted average of x and y coordinates. Then we applied Kalman filter in the reverse direction (from the last to the first frame of the image sequence) to predict a parent cell from daughter cells. If the estimated centroids of the

daughter cells are in the same candidate parent cell, we associate the daughter cells to that potential parent cell. To find proper normal cell division, we used paired graph analysis to determine which cells will divide (i.e. RU cells) and used reverse Kalman filter to associate the parent cells to their daughter cells. The results of the applied Kalman filter are presented in Chapter 5.

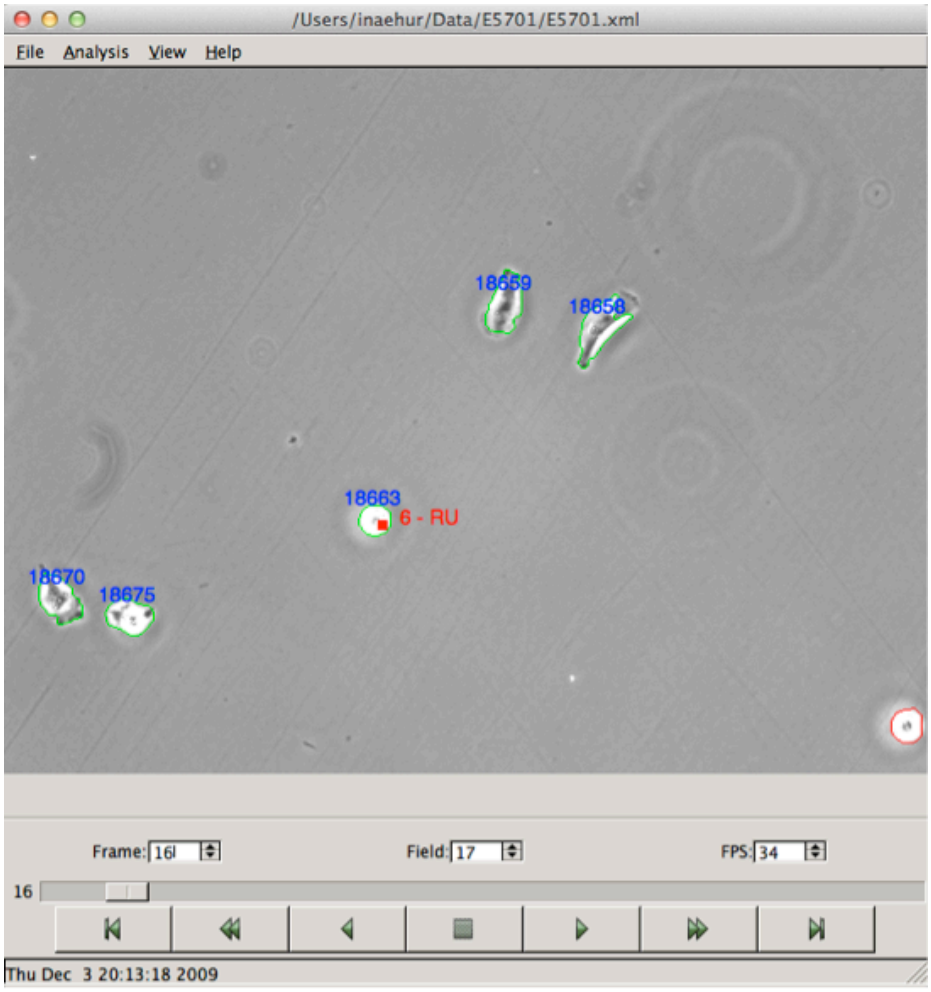



**Figure 1. casViewer.** Main window of casViewer shows segmented cell with id, event number, and designation. Cells with green border represent an actual cell; objects with a red border are partial cells or alter artifacts which can then be ignored in subsequent analysis.

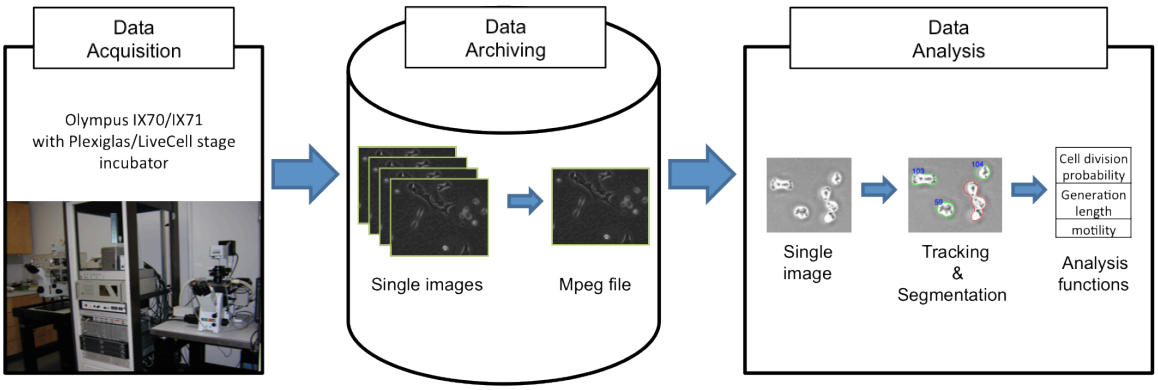



**Figure 2. A workflow of LSDCAS.** First, live cell image stream data can be collected by data acquisition component. Then, single images can be translated to mpeg file and stored in data archiving component. Lastly, user can use analysis functions using cell tracking and segmentation.

**Table 1. Predefined events in LSDCAS event analysis using casViewer.**

| Event name | Abbreviation | Description |
|---|---|---|
| Normal division | ND | A parent cell divides into two daughter cells |
| Multipolar division | MD | When a parent cell divides, the spindle has three or more poles and results in the formation of a corresponding number of daughter cells |
| Failed normal division | FND | A parent cell try to divide into two daughter cells, but only one daughter cell produces |
| Failed multipolar division | FMD | After cell division, daughter cells produces less than a number of the poles |
| Sister cell fusion | SCF | Two daughter cells are fused |
| Non sister cell fusion | NSCF | Two cells are not daughter cells are fused |
| Death at division | DD | Cell death in mitosis |
| Death at telophases | DT | Cell death after cell division |
| Interphase death | ID | Cell death in interphase (in the absence of any mitotic events) |
| Apoptosis | AP | Cell death during interphase (programmed cell death) |
| Round up | RU | Cell division |
| Flatten out – normal | FO | Cell exit from mitosis |
| Off screen | OS | Cell moves over the recording window |
| End of Movie | EOM | End of live cell image streams |
| Identify cell | IC | Start point cell for event analysis |
| Flatten out - Bi-nucleated | FOBN | Cell with two nuclei exit from mitosis |
| Flatten out - Multi-nucleated | FOMN | Cell with more than two nuclei exit from mitosis |
| Nuclear fusion | NF | Two or more nuclei are fused |
| Identify cell - Bi-nucleated | ICBN | Start point cell with two nuclei for event analysis |
| Identify cell - Multi-nucleated | ICMN | Start point cell with more than two nuclei for event analysis |

Note: Full name of event, the abbreviated name of event, and a brief description are provided.

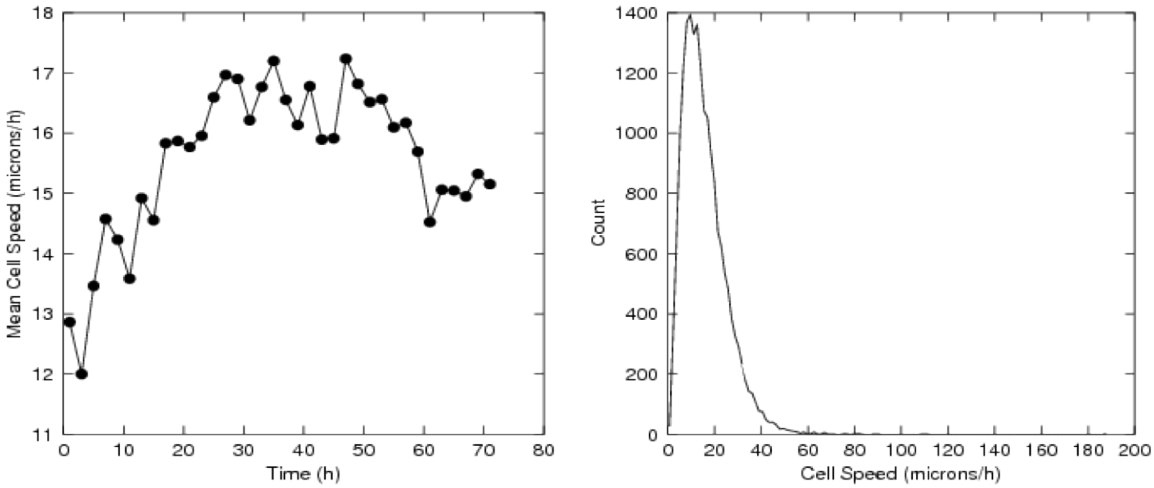
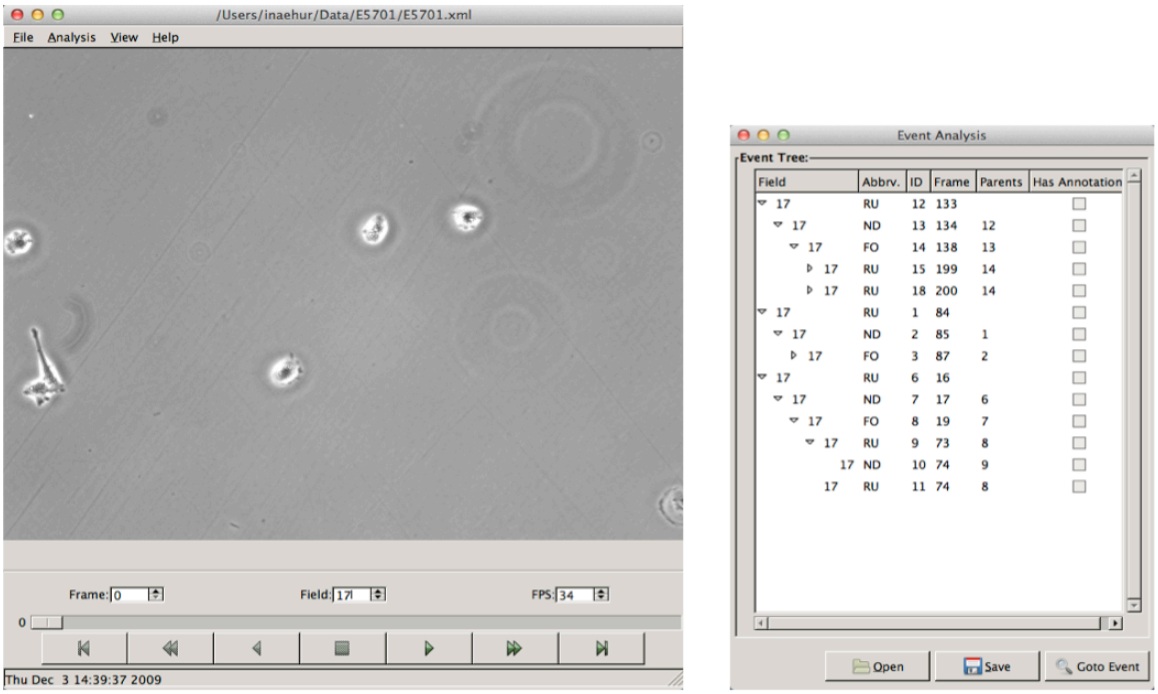



**Figure 3. Mean cell speed and motility histogram of E5701 Sample 0.** A. The mean cell speed is presented at each time point and increased until about 30 hours. Then, the mean cell speed is maintained at 15 to 17 microns/h, B. Most cells move at about 16 μm/h.




(A) (B)

**Figure 4. Manual event annotation using casViewer.** A. By using right button of mouse, a researcher can select among predefined events list. Event tree id number is assigned automatically, B. The event analysis dialog box using cell manually-identified event and indicates their logical relationship through the tree-structure shown.

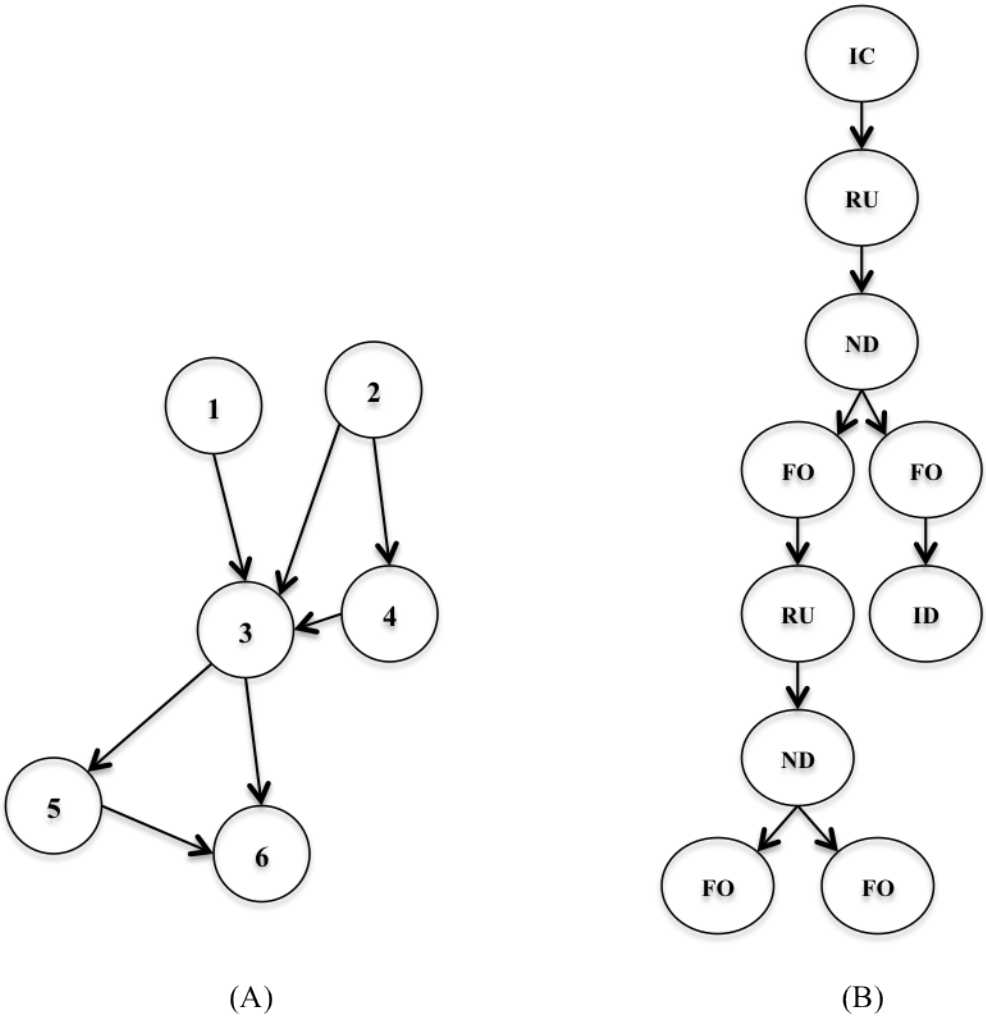



**Figure 5. A directed acyclic graph (DAG).** A. A simple example of DAG. Closed path DAG has no start and end vertex; the graph start and end at the same vertex and follow edges only in their forward direction. Unlike closed path DAG, open path DAG has a root and a forward direction, B. An example of cell event graph with two rounds of cell division from one cell. The tree begins with a identify cell (IC), then round up (RU), normal division (ND), and flatten out (FO) are follow. Cell death in interphase (ID) can be added if a cell dies after FO. If a cell dies when in RU state, the event is dead at division (DD).

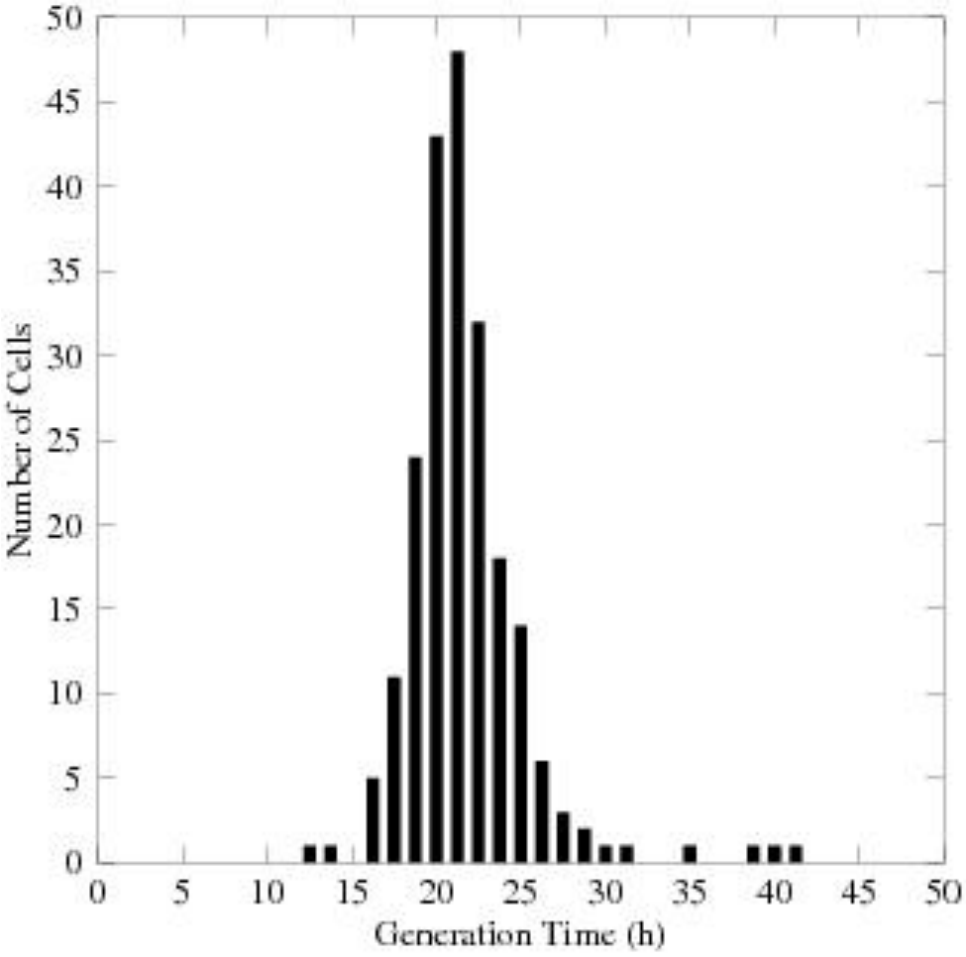



**Figure 6. Cell generation time histogram.** 214 SKOV 3 cells are used to analyze the mean cell generation time.

# CHAPTER 2

# MACHINE-BASED CELL EVENTS DETERMINATION

## Machine Learning Approaches: Unsupervised and Supervised

Machine learning is a scientific discipline that allows computers to distinguish between dissimilar objects without being explicitly programmed. It provides a cost-effective approach to automated knowledge acquisition in quantitative datasets. Machine learning algorithms can be categorized into two broad areas: unsupervised and supervised learning algorithms (Fig. 7). The difference is drawn from how the general inductive process (also called a learner) classifies data. The aim of unsupervised learning is to find regularities in input[30]. Such an algorithm should be able to discover classes based on the clustering of objects in a dataset[32]. The data have no target class which is predefined class determined by user. Thus, the input data for unsupervised learning is unlabeled and is used as a random variable set[30,33]. The self-organizing map (SOM) and adaptive resonance theory (ART) are well-known models in unsupervised learning. In contrast, supervised learning produces an inferred function, a classifier[15,32], from labeled examples in training data. A classifier should be able to define patterns in the data that relate data attributes with a target class attribute and predict a desirable output class among pre-categorized classes. These patterns are utilized to predict the values of the target attribute in future data instances. Hence, a classifier should build a generalized function from a training data set. Bayesian classification, support vector machine (SVM), decision trees, and neural networks (NN) are commonly used algorithms for supervised learning.

## Machine Learning Methods in Live Cell Imaging

In the literature, cell events are manually labeled and classified by supervised learning. Two popular analysis methods are SVM and NN. SVM is a linear maximum margin method that assigns data as points in the feature space, and then separates

categories by a maximum margin hyperplane. The goal of SVM is to find the optimum hyperplane to separate data into two categories[6,28,30,34]. Figure 8 illustrates a simple SVM classification. The geometry of the hyperplane depends on a kernel function[35]. This kernel function is possibly the best-known element of SVM. Kernel functions are used for the transformation of feature space[36] because the decision boundaries of most data sets are nonlinear and difficult to represent in closed form. When SVM is applied to image analysis, the training process is typically faster than that of other classifiers such as NN and AdaBoost[6,33]. However, heavy parameter tuning is required[37] and the complexity of SVM is independent of the dimension of the feature space[34].

NN, also called perceptron, is a mathematical model that represents the characteristics of artificially interconnected neurons. These artificial neurons, which are related to biological neurons, use simple learning processes. Although various NN models are used for image analysis, most of them utilize three layers: the input, hidden and output layers. Figure 9 shows the most commonly used feed-forward neural network structure. A feed-forward network, as shown in Figure 9, is a multilayer weighted and directed graph which consists of inputs ($x_1$ and $x_2$), output (y), nodes ($n_1$, $n_2$,…,$n_5$) and directed edges ($w_{31}$, $w_{41}$,…,$w_{54}$). Nodes are artificial neurons and directed edges are connections among input, hidden and output nodes[28–30,38–40]. When a network performs a classification task, a learning process updates a network by changing the weights for the best performance. The popularity of neural network models is increasing because of low dependency on specific data[33,41] and three advantages: the ease of optimization, the accuracy of predictive inference, and the ease of knowledge dissemination[42]. The definition of knowledge dissemination is a process to accelerate new hypotheses based on new observations and prior knowledge[43].

## Cell Event Determination in LSDCAS Image Stream using Neural Network

Classification by supervised learning can be used to determine future cell events based on attributes of predefined cell events. The first step of cell event classification would be to determine the types of training examples that would be must useful. We chose to look at single cell events, mitosis related events, and specific series of cell events. The second step would be to gather a training set that should contain representative features and attributes of these events. It is important to note that the number of features should not be too large or too small. Too few features and rates of false classification would increase to too many and we are faced with the curse of dimensionality[33,38,44]. The curse of dimensionality is a phenomenon which states computing cost increases exponentially as the number of state variables increases. The next step would be to choose an algorithm that is similar to the learned function and to apply a test data set. According to the procedure of supervised machine learning, parameters of a learning algorithm should be optimized by a validation set or adjusted by cross-validation[29,30]. The last step is to evaluate the performance of the algorithm. Performance can be measured by the accuracy in predicting a data set that was not used in training[32].

Since the neural network approach is effective in classifying objects in general, we used it to determine RU and FO cell events. A learner automatically builds a classifier for RU and FO categories by observing the characteristics of a set of cell events that were manually classified. In our studies, RU cells are predicted to be circular shaped cells that are lighter than other cell events and FO cells are determined to be number eight shape cells that are bigger than other cell events. Also four out of ten measured features (i.e. mean intensity, area, perimeter and shape factor) are valid in distinguishing RU and FO cells from the other cells.

### Training set

Manually identified cell events are collected as a training set for the neural network approach. Data for training were acquired from field 0 of E5701 and E5689 experiments. We chose to focus on the RU and FO cell events. We manually annotated cell events that represented characteristic RU and FO states. A total of 34 RU events and 29 FO events were manually annotated using casViewer. E5701 and E5689 experiments are recorded with MDA-MB-231 cells, a human breast cancer cell line.

### Test set

Field 1 from E5701 and E5689 experiments were used as test sets. The same four features extracted in the training set were also extracted in the tests set. We applied the neural network model made by the training set to test sets containing 4489 and 5964 cell events from E5689 and E5701, respectively.

### Analyze using Weka

Weka is an open source collection of machine learning algorithms and it is implemented in Java. These algorithms can be used solving various data mining problems[45]. We used Weka for applying classification and feature selection methods to our training and test data set. A multi-perceptron classifier in Weka was applied to construct a three layer, one hidden layer included, neural network classifier. Default parameter values were utilized to show the result. The ten features of the cell event data were transformed to csv (Comma-separated values) file which is the default import format for Weka. Then we selected four features and applied the multi-perceptron classifier under functions. The training results can be stored as a model file for future classification.

## Results

Manually annotated events are used as a training data set for a neural network and a test data set is used to compare against the predicted cell events for our novel methods. Typically 10-fold cross-validation was used to validate the results of the test sets. This means that the whole data set is divided into 10 parts. 10% of test set is used as an actual test set and the remaining 9 parts are the learning set for the model. But we collected a specific training set from field 0 of E5689 and E5701, and applied each model to field 1 of E5689 and E5701, respectively. Tables 2 and 3 show the true positive, false positive, true negative, and false negative cell events determined by the classifier and accuracies for the default setting of Weka. We selected 28 and 31 RU cells from E5701 and E5689 to build a training set, respectively. Then we applied the classifier trained from each experiment to test sets: E5701 and E5689 field 0. The E5701 test set has a total number of 5964 cell events and E5689 test set has a total number of 4489 cell events. Neural network successfully detected about 97% of the RU cell with 96.55% of sensitivity and 59.59% of specificity using E5701. In addition, about 94% of the RU cells are detected with 93.94% of sensitivity and 78.41% specificity using E5689. The results indicate effective classification.

## Discussion

The neural network result presented high true positive rate; about 97% for E5701 and 94% for E5689. False positive rates, however, were also high which can potentially degrade the performance of detecting true positives. In addition, the neural network is not equipped to handle a sequence of events like the RU-FO progression. This is because cell events are determined only by the four features specified. Also the neural network cannot consider the previous state of an event when they apply their classification standard. For this reason, we developed novel methods to be used to detect the cell events. A method to

detect a cell's entrance and exit from mitosis is developed from a time-series data analysis method and is described in Chapter 3.

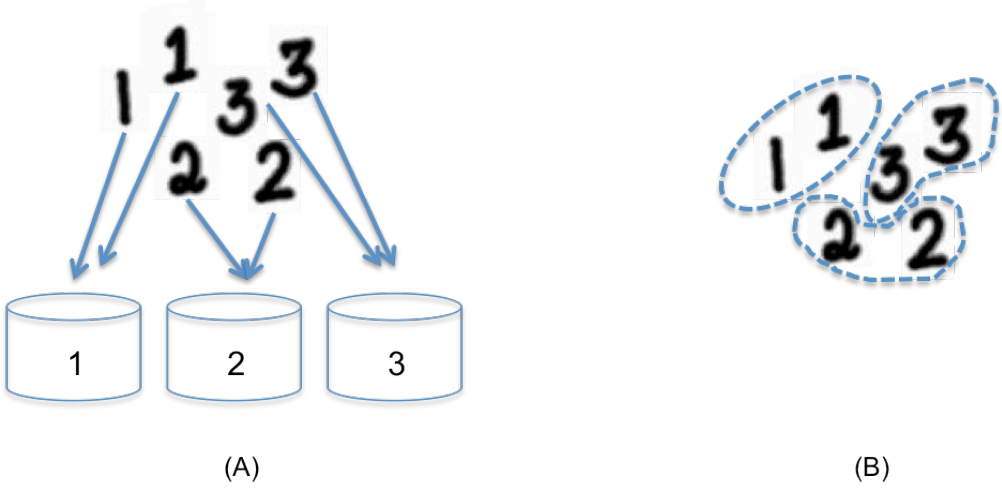



**Figure 7. Supervised and unsupervised machine learning.** A. Supervised learning predict a desirable output class among pre-categorized classes. The result group already defined by user, and the input data will categorize to the predefined group, B. Unsupervised learning divides input data into groups just with similarities. The aim of unsupervised learning is to find the regularities in the input.

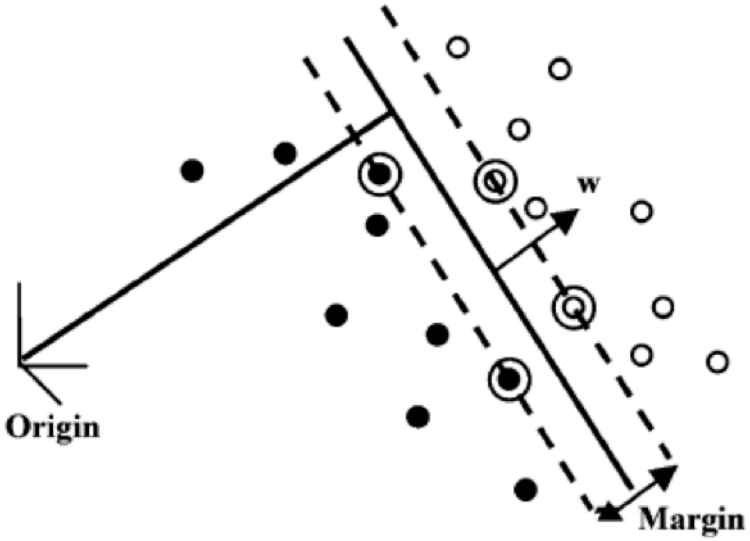



**Figure 8. A simple SVM classification.** A margin is defined as the sum of the distances of the closest points of the two classes. Samples on the margin are called support vectors and the solid line in the middle of margin is the hyperplane. The vector w is a normal vector perpendicular to the hyperplane[46].

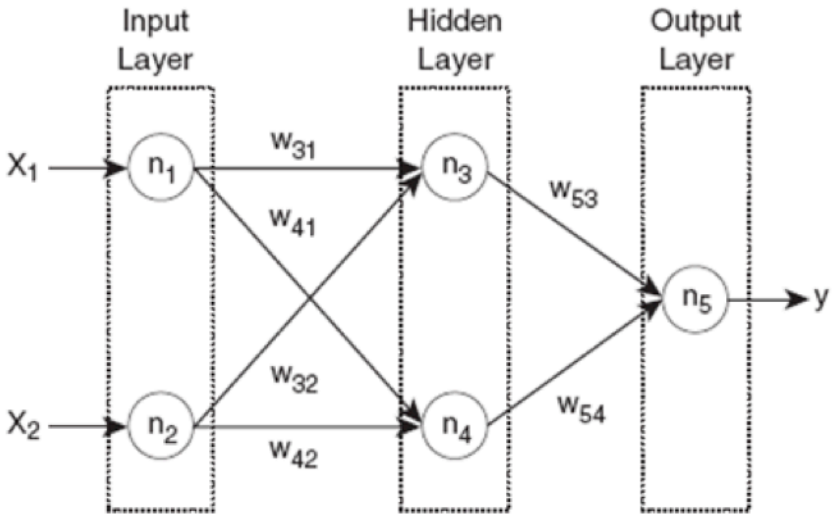



**Figure 9. A simple structure of two-layer feed-forward neural network.** A user can provide inputs (x1 and x2) and the system computes output (y) from the value of inputs, nodes (n1, n2, …, n5) and directed edges (w31, w41, …, w54)[28].

**Table 2. Neural Network results using E5701.**

| | | Detected by NN | |
|---|---|---|---|
| | | RU | Not RU |
| Manually detected | RU | 28 (TP) | 1 (FN) |
| | Not RU | 2399 (FP) | 3536 (TN) |

Note: Each training set has RU and FO events, and only RU cells are determined by NN classifier. A total number of manually detected RU cells are 29 and the accuracy of NN using E5701 is 59.76%. Also, the sensitivity is 96.55% and the specificity is 59.59%.

**Table 3. Neural network results using E5689.**

| | | Detected by NN | |
|---|---|---|---|
| | | RU | Not RU |
| Manually detected | RU | 31 (TP) | 2 (FN) |
| | Not RU | 962 (FP) | 3494 (TN) |

Note: Each training set has RU and FO events, and only RU cells are determined by NN classifier. A total number of manually detected RU cells are 33 and the accuracy of NN using E5689 is 78.53%. Also, the sensitivity is 93.94% and the specificity is 78.41%.

# CHAPTER 3

# ENTER AND EXIT MITOSIS EVENTS DETECTION USING TIME-SERIES DATA ANALYSIS METHOD

## Graph Trend Analysis

Trend analysis refers to the notion of attempting to spot a pattern in information. It can be valuable as a warning indicator of potential problems or issues by predicting future circumstances. It can also be used to estimate specific or uncertain events in the past. Trend analysis can be used to predict changes and trends in social life, technology, fashion, weather, and consumer behavior through statistical modeling of past events. Thus, past and current financial ratios are compared by trend analysis in the business field to make important decisions in formulating business strategies and making wise decisions. Different mathematical and statistical models are used to find the differences or similarities between past and current figures and situations. As in figure 10, trend lines can be fitted to collect data plotted on x/y-axes and it helps to understand the trend of data. Like the example of trend lines, we can discover the trend of feature information of a cell from an image stream and it can be used to assess the relationship among cell events.

### Features for graph trend analysis

Each single cell has ten features in a frame by frame and these features can be expressed as a line graph over time. The line graphs take four features (i.e. mean intensity, perimeter, area, and shape factor) among ten that are measured, all of which have different scales, but four features having similar fluctuation patterns during each cell cycle. We found that the graphs show a specific pattern as cells enter and leave mitosis. The mean intensity and shape factor value increase from the end of interphase to entry into mitosis, and reaches a maximum when the cell enters mitosis (i.e. RU state). Unlike mean intensity and shape factor, perimeter and area value decrease just before a cell

enters the RU state; reaching a minimum value when the cell is in the RU state (Fig. 11). On the other hand, mean intensity and shape factor values are minimal, and perimeter and area value are maximal in FO state following to RU state. For example, in Figure 12, the cell at frame 19, in the RU state in this example, has maximum intensity and shape factor value, and minimum perimeter and area value. The slopes of graphs change from negative to positive and vice-versa when the cell at frame 20, enters into the FO state. Using these four features, cell entry into the RU and FO states can be determined. Further, multiple cell divisions during image acquisition can be estimated by finding several candidates of maximum and minimum values.

Limitations, however, exist because of time interval of an image sequence. The LSDCAS image streams were recorded with time intervals (i.e. 300 seconds) and cell events can occur between two frames. If so, the actual maximum/minimum values are not recorded and the graph trend analysis does not detect the correct frame as a result. In Figure 13, for example, even though the actual RU starts between frame 4 and 5, frame 4 would be selected by graph trend analysis. In addition, FO event can occur several frames after RU and it depends on cell line and environment. If FO event does not occur in the frame immediately following the frame in which the RU event occurred, then the graph trend analysis cannot detect that the RU event occurred. To avoid these limitations, we used a moving average to create a trend line for the graph. This trend line helps to define a consistent overall graph trend, and is therefore unaffected by uncertain cell behaviors.

Graph trend analysis using moving averages

Graph trend analysis is the core technical analysis method of graphical data, and the moving average technique directly addresses the issue of how to define trends in an objective manner. A moving average is found by averaging value changes over time and can be used to analyze data. Three most commonly used moving average types are simple, weighted and exponential. The simple moving average calculates the average

with equal emphasis on all values, whereas weighted and exponential moving averages are computed with more emphasis on the most recent values. The simple moving average is computed by summing a given time period and dividing by a given time period. The equation for the simple moving average is

$$\text{Moving Average (MA)} = \frac{\sum_{i=1}^{n} xi}{n}.$$

For creating MA(5), where n = 5, the values of the last five time period are summed and divided by five. To determine the trend of a graph, the most recent value is compared to the average of the last five values. In other words, the graph is trending higher if the value is higher than average, and the graph is trending lower if the value is lower than average. These characteristics are broadly the same for all three types of moving average methods. In Figure 14, MA(5) shows the trend line of original graph. The moving average trend line of mean intensity is moving upward rapidly between frame 15 to 20. Unlike mean intensity, perimeter and area trend lines are moving downward rapidly between frame 15 and 20. With these three moving average trend lines, we identified a cells entry into RU and FO states. The relationship between cell shape and entry into these states can also be considered by examining changes in the trend line of the shape factor. Further, we overcame the limitation of the time interval problem, and reduced the effect of errors associated with errors in cell segmentation and tracking (Fig. 15).

## Results

To detect cells entering RU or FO states, a graph based time-series data analysis named as graph trend analysis is used. When cells enter the RU state, the slopes of intensity and shape factor graphs become positive, and the slopes of area and perimeter graphs become negative. In contrast, the slopes of intensity and shape factor graphs become negative, and the slopes of area and perimeter graphs become positive, when

cells enter the FO state. Further, cells can have maximum mean intensity and shape factor, and minimum perimeter and area value in the RU state if the cell divides. These graph trends provide critical signals for the RU and FO states. By using these characteristics, graph trend analysis identified the RU state when a maximum slope change of four features occurred in the same frame, and the next frame classified as a FO state frame. We used an id 2 cell from E5701 as a preliminary dataset to verify the graph trend analysis performance and found the frame of entry of cells into the RU and FO states successfully determined using graph trend analysis, as shown in figure 16.

We used the E5701 experiment to develop the graph trend analysis method, after which we applied it to E5689 and E5677. E5701 was not used for data analysis because the overall mean intensity was much brighter than other experiments which could affect the accuracy of the results. Tables 4 and 5 show the performance of the graph trend analysis in detecting RU cells. To validate our approach, we manually detected RU cells using casViewer which was used as a control. A Total of 473 and 194 RU cells were collected from E5689 and E5677, respectively. Then graph trend analysis found candidate RU cells by checking whether a maximum slope change of four features occurred in the same frame. The accuracy for E5689 is 89.03% and it is smaller than E5677 (94.79%). Although the ideal performance is for 100% accurate detection, a good alternative is to subject RUs to detect with high sensitivity and low specificity. But the graph trend analysis has low sensitivity and high specificity. Also the automatically detected RU cell events from E5689 and E5677 were significantly lower than those of the human observer; only detect less than 10% of RUs. False positive rates of both experiments (92.99% of E5689; 97.66% of E5677) are exceptionally high to use the method alone to detect significant cell events.

## Discussion

Trend analysis of time-series data analysis method is one of the advantageous ways to detect cell entry to mitosis and after mitosis cell events even though the detection rates using E5689 and E5677 were lower than a human observer. Because visually distinguishing cell states from live cell image streams is a time consuming and tedious process even for expert biologists. The reason why the human observer performed better than the automatic method is that the human observer could consider unobserved events between images. Unlike the human observer, automatic detection can only use the image information in the image sequence. As mentioned before, graph trend analysis can determine that a possible RU frame is 4 while an actual RU state is frame 5 (Fig. 13). Due to the time interval between image sequences, a maximum slope change for the four features can designate different frames as a RU state. To overcome this limitation, we expected to be able to obtain more correctly detected RU cells through extending the search criteria to consider slope changes of several frames by using a moving average. Table 6 shows the results of graph trend analysis using different moving averages. Graph trend analysis with moving average (5) gave us the best detection rate with highest sensitivity. The overall detection rate of the graph trend analysis using the moving average is better than without the moving average, but the moving average cannot detect a specific frame for the RU event, it can only detect a possible range in which the RU event can occur. This range is determined by the moving average number (i.e. MA(5) can determined 5 frame range where an RU event could possibly occur). In addition, graph trend analysis has one more limitation. Multiple cell divisions can occur in an experiment and the trend analysis is limited in that it can only detect one occurrence of the RU state for a single cell level. The performance of graph trend analysis is not satisfactory to identify the events in LSDCAS image streams and has limitations, but it provides sufficient information to find specific cell events by determining a possible range of cell events occurring.

To overcome the problems that graph trend analysis has and the necessity of general algorithm that can accept uncertain exceptions of cell movement, the paired graph analysis method in Chapter 4 is developed. Paired graph analysis is derived from time-series data analysis methods in econometrics and it also used line graph data of four features. Paired graph analysis can detect over one cell division(s) for a single cell in an experiment and the overall RU events detection rate is significantly improved over graph trend analysis.


**Figure 10. Three types of trend line[47].** Uptrend is determined when each successive peak is higher than the ones found earlier in the graph. Unlike uptrend, downtrend is specified by the movement of data when the overall direction is downward. Thus, when the horizontal data movement occurs and the forces of supply and demand are nearly equal, we called it a sideways trend.


**Figure 11. Tendency of four features.** Mean intensity and shape factor increasing when cells are entering the RU. As an opposite to mean intensity and shape factor, perimeter and area values increasing when cells are leave the RU.

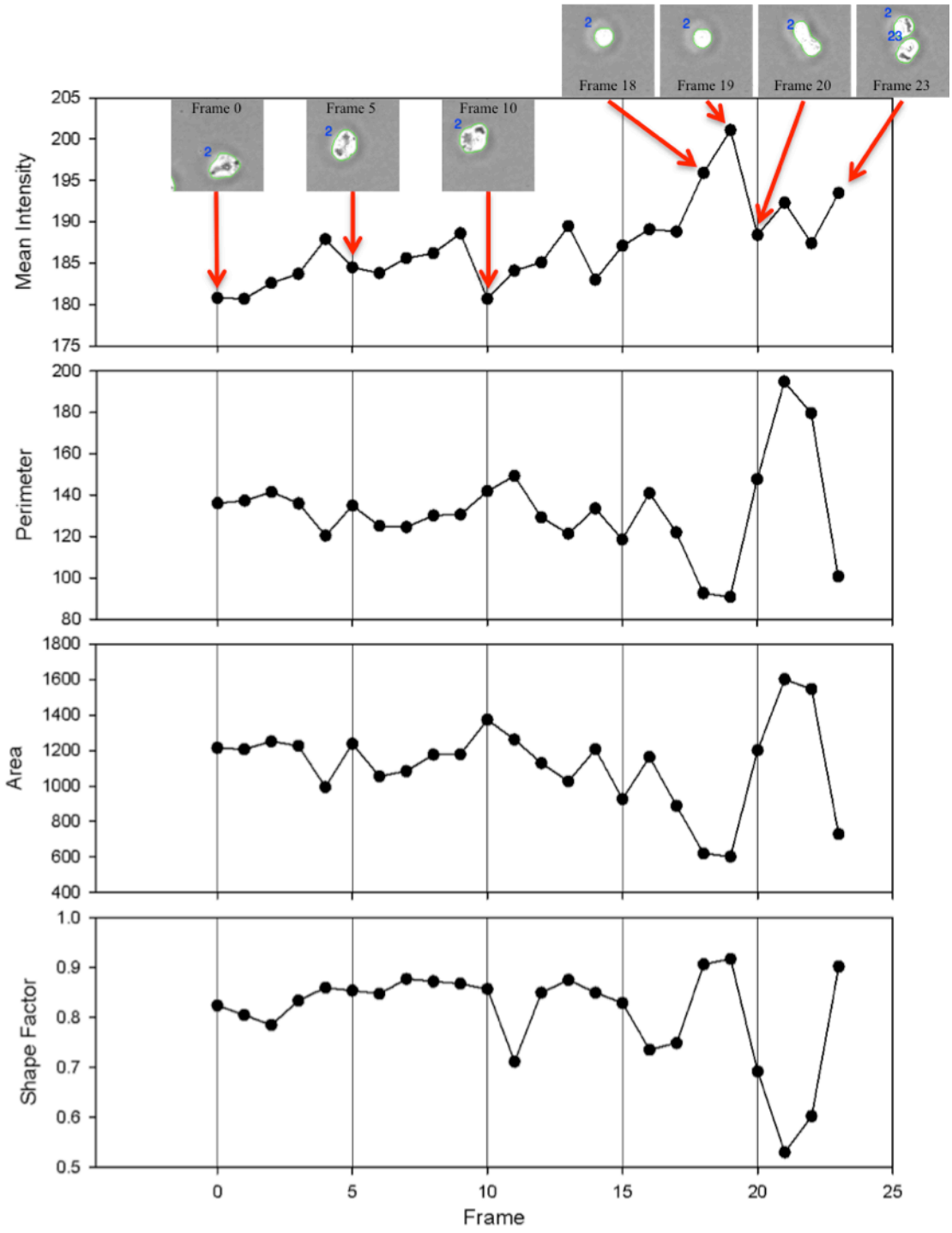



**Figure 12. Feature graphs of E5701 cell id 2 with cell event from image stream.** As the cell cycle progress, the graph of mean intensity, perimeter, area, and shape factor fluctuate.

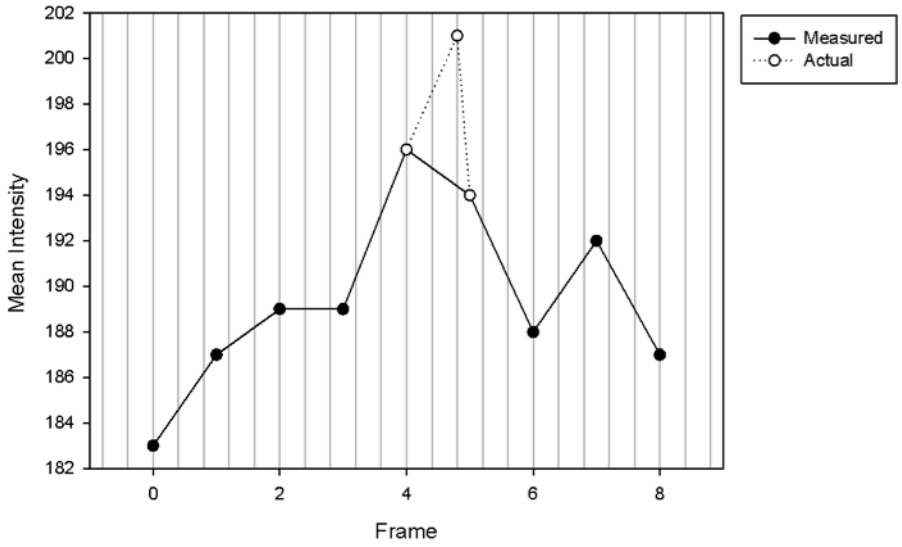



**Figure 13. A limitation of recorded image stream by time interval.** Even though RU state is entered just before frame 5, frame 4 can be selected as RU state because actual value is not observed.

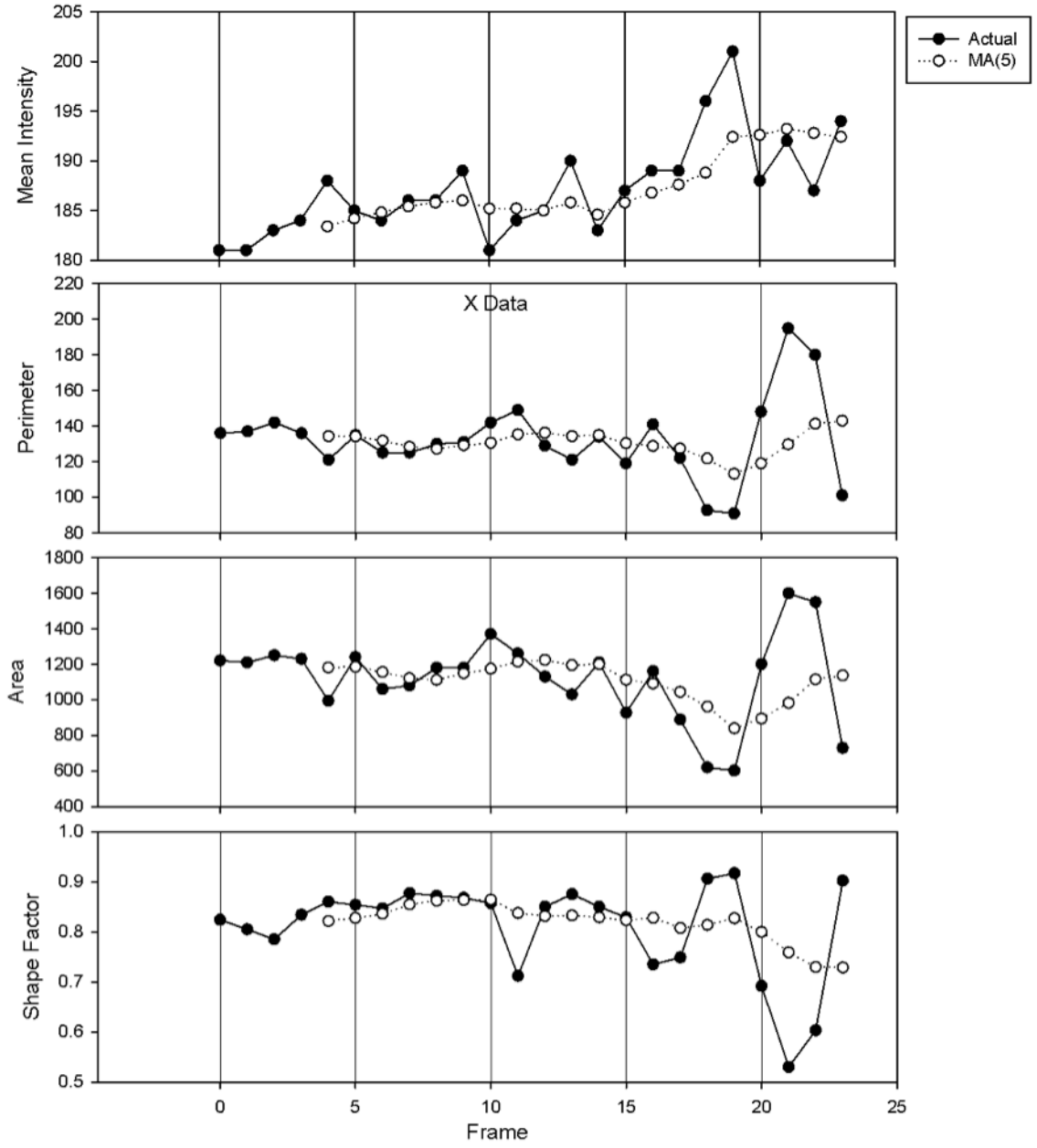



**Figure 14. Data graph with moving averages for the MA(5) trend line. The** trend line helps to understand the trend of data, and the moving average is a consistent and reliable way to define the trend. Each MA(5) line shows the overall trend of the original data.

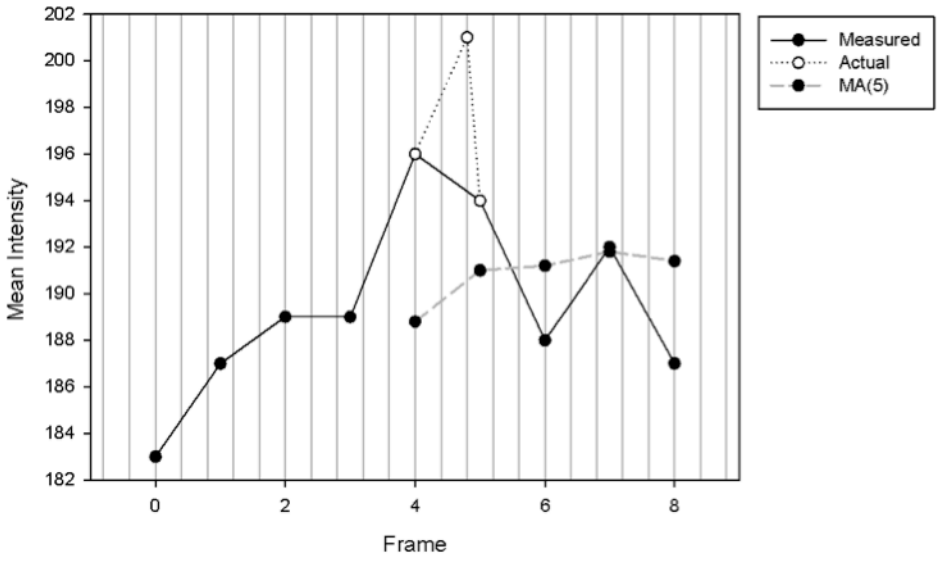



**Figure 15. MA(5) trend line to overcome a limitation of recorded image stream by time.** We can notice the slope change between frame 0 and 5 is significant through MA(5) trend line and it means the RU event can occur between frame 0 and frame 5.

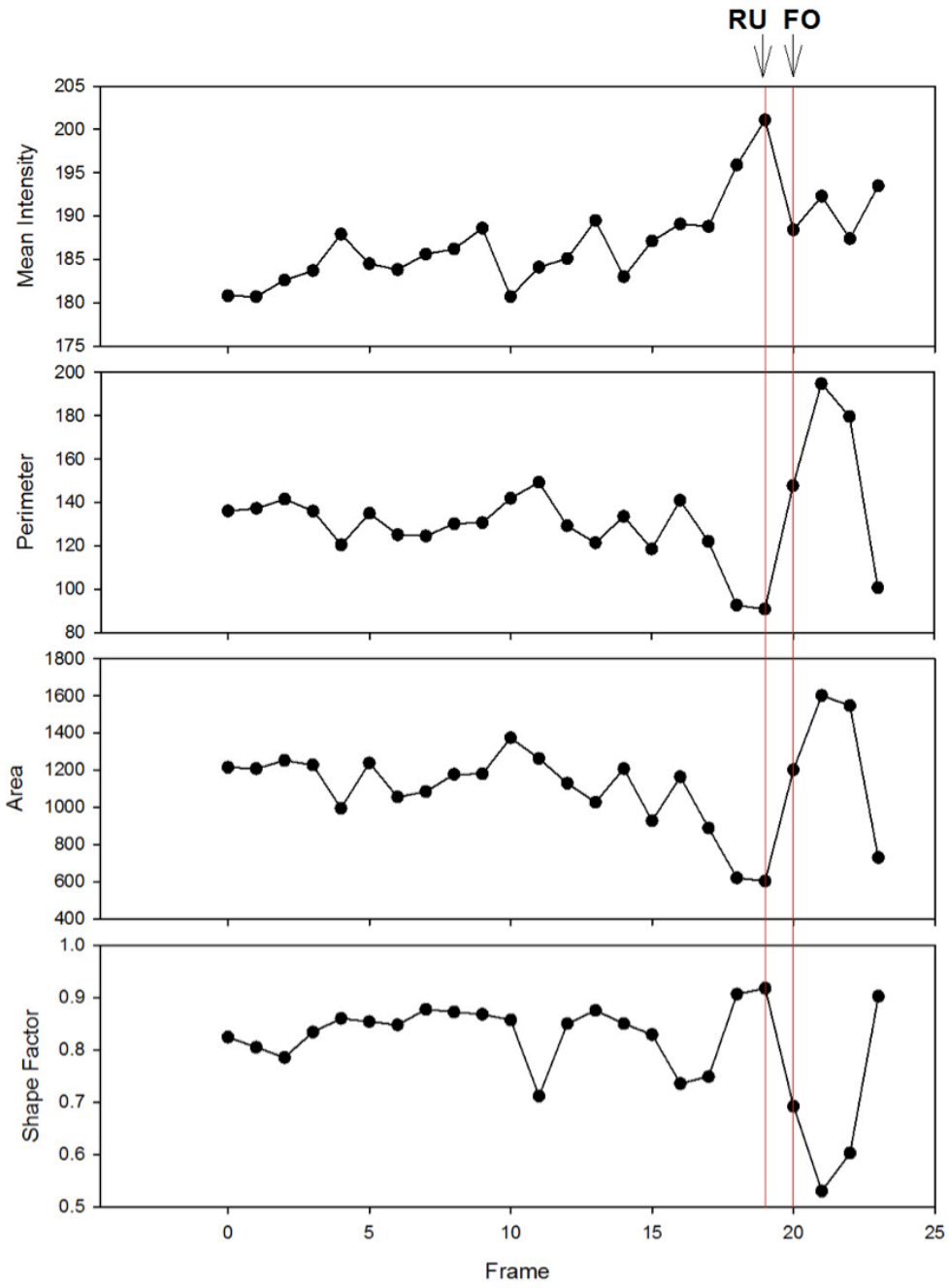



**Figure 16. Four features of E5701 Field0 cell id 2.** The cell in frame 19 and 20 were determined to be entering the RU and FO states using manual detection. Graph trend analysis also indicated entry into RU and FO states at the same frames.

**Table 4. Graph trend analysis results using E5689.**

| | | Detected by Graph trend analysis | |
|---|---|---|---|
| | | RU | Not RU |
| Manually detected | RU | 43 (TP) | 430 (FN) |
| | Not RU | 570 (FP) | 8164 (TN) |

Note: A total number of manually detected RU cells are 473 and the accuracy of graph trend analysis using E5689 is 89.14%. Also, the sensitivity is 9.09% and the specificity is 93.47%.

**Table 5. Graph trend analysis results using E5677.**

| | | Detected by Graph trend analysis | |
|---|---|---|---|
| | | RU | Not RU |
| Manually detected | RU | 12 (TP) | 182 (FN) |
| | Not RU | 500 (FP) | 12391 (TN) |

Note: A total number of manually detected RU cells are 194 and the accuracy of graph trend analysis using E5689 is 94.79%. Also, the sensitivity is 6.19% and the specificity is 96.12%.

**Table 6. Graph trend analysis results with various moving averages of E5677.**

| | True Positive | False Positive | True Negative | False Negative |
|---|---|---|---|---|
| GTA | 12 | 500 | 12391 | 182 |
| MA(2) | 17 | 282 | 12786 | 177 |
| MA(3) | 19 | 214 | 12677 | 175 |
| MA(4) | 15 | 171 | 12705 | 179 |
| MA(5) | 21 | 143 | 12748 | 173 |

Note: GTA and MA represent simple graph trend analysis and graph trend analysis with moving averages, respectively. Accuracy of GTA is 94.79%, MA(2) is 97.84%, MA(3) is 97.03%, MA(4) is 97.21% and MA(5) is 97.59%. The detection rate of GTA is 2.34%, MA(2) is 5.69%, MA(3) is 8.15%, MA(4) is 8.06% and MA(5) is 12.8%.

# CHAPTER 4

# CELL ENTRY INTO MITOSIS DETECTION USING TIME-SERIES DATA ANALYSIS METHOD

## Non-stationary and Stationary Time-series Variables

Most time-series variables are non-stationary, in that average and standard deviation can change as time passes. To test whether the variables are stationary is very important for long-term market analysis. Non-stationary variables suffer permanent effects from random, or foreign, shocks over time which will affect the forecast value. Unlike non-stationary variables, stationary variables are free from random shocks because they only cause temporary effects. Non-stationary variables have a unit root and we can determine whether a variable is non-stationary by the unit root test. Time-series variables can be formed through an autoregressive process such as

$$Y_t = \alpha Y_{t-1} + e_t$$

where $e_t$ is a random shock. A process has a unit root when $\alpha = 1$. If $\alpha = 1$, the equations become

$$Y_t = Y_{t-1} + e_t$$

$$Y_{t+1} = Y_t + e_{t+1} = Y_{t-1} + e_t + e_{t+1}$$

as time goes from t to t+1. In other words, $e_t$ will remain and the effect is not reduced over time. If $\alpha$ is $0 < \alpha < 1$, however, a unit root does not exist. The equation at t+1 becomes

$$Y_{t+1} = \alpha Y_t + e_{t+1} = \alpha^2 Y_{t-1} + \alpha^2 e_t + e_{t+1}.$$

In this situation, the effect of $e_t$ gets smaller as time goes on. A well-known unit root test is the ADF test and it uses the existence of a unit root as the null hypothesis. Further, non-stationary data can be transformed into stationary data after differencing k times, it is called integrated of order k, denoted I(k). I(0) means the variable is stationary and I(1)

means the variable is non-stationary but it can be transformed into a stationary one by first differencing.

## Paired Graph Analysis

### Pairs trading

Pairs trading is a well-known market neutral trading strategy in the stock market. It was developed in the mid-1980s by Nunzio Tartaglia[48]. It is used to enable traders to earn absolute returns in a steady manner from any market conditions such as uptrends, downtrend and sideways movement through forecasting the market. In other words, the aim of pairs trading is to exploit investment opportunities by measuring price ratios or differences of a pair of stocks and steadily earning modest returns[48,49]. A pair of stocks has to be in the same business field for example, Walmart and Target.

A quantity, called the spread, is calculated by the quoted prices of two stocks (Fig. 17). The most profitable buy or sell point is when the spread becomes wider than some confidence level. The prices are connected together by a stochastic trend, and two stocks are cointegrated if the spread is mean reverting. Mean reverting is a mathematical concept which explains the tendency to move back to the average when a value moves away from the average value. The challenge of pairs trading is to identify the stocks that tend to move together and the mean reverting in the ratio of the prices. Even if the stocks seem to be related, they might not be associated with each other, and this is the reason why verifying the cointegration test between a pair of two stocks is necessary.

### Cointegration

Cointegration is an econometric tool suggested by Engle and Granger[50] that is used to test the relationship between nonstationary time-series variables. Cointegration and correlation are two different concepts. Unlike cointegration, correlation is only applicable to stationary variables and short-memory processes. In other words,

correlation is not appropriate to analyze the long-term behavioral relationship between a pair of stocks and it refers to any of the statistical relationships between a pair of stocks in returns. Cointegration, however, refers to co-movement in raw market prices or exchange rates[51]. In fact, cointegrated series can have correlations that are quite low at times because high correlation does not necessarily imply high cointegration, and vice versa. Figure 18 shows the difference between correlation and cointegration.

A pair of variables are cointegrated if two variables having a unit root (i.e. integrated of at least order one, denoted I(1)). A cointegration test to check whether two variables are cointegrated is a two-step estimation procedure. The aim of the cointegration test is to detect any stochastic trends in stock prices and to build a general trend for a specific pair of stocks for analysis. Many methods have been developed for testing whether a cointegrating relationship exists between a pair of stocks. Of these, there are three methods that are more commonly used over the others: Engle-Granger method[50], Johansen procedure[52], and Philip-Ouliaris test[53]. General procedures of the Engle-Granger method is to run ordinary least squares (OLS) regressions on nonstationary variables and determine the relationship between the data.

## Results

### Cointegration test among features of a cell

The cointegration test is the foundation upon which pairs trading is built and the basic cointegration function can easily be found in any statistical software package. The statistical package R is free software for statistical computing. It provides time series analysis, linear regression models using ordinary least squares (OLS), spread calculation function, Augmented Dickey-Fuller (ADF) test and p-value to test pairs of stock prices for cointegration. Using R for the cointegration test is a fast and easy way to verify the result. We tested for cointegration between intensity, perimeter, area, and shape factor. Figures 19 and 20 illustrate sample R code for the cointegration test[54] using intensity and

perimeter pairs from E5701 cell id 2 by the ADF test and the Phillips-Ouliaris test, respectively. This test uses the Engle-Granger method, one of three major cointegration tests mentioned earlier, and Philip-Ouliaris method. The cointegration test using R is a preliminary test that confirms whether two feature pairs are cointegrated or not.

In figure 19, zoo and tseries library are loaded for handling time-series data. Then, the csv files are read and two columns are chosen (the mean intensity and the perimeter) into one t.zoo object using the merge function. After creating a data frame for applying statistical functions, we tested whether time-series variables were cointegrated. The unit root test is the first step in general pairs trading. However, we constructed the spread, then tested the spread for a unit root in R. The *lm* function in R is a formula that specifies the linear model. A simple linear equation with no y intercept is given as follows

$$y_i = \beta x_i + e_i$$

where $e_i$ is random error. We used first regression coefficient (β) as a hedge ratio for calculating the spread. A hedge ratio is a ratio comparing the value of a position protected via a hedge with the size of the entire position itself and the spread is a gap between two stock prices. In paired graph analysis, the spread is a gap between two features such as mean intensity and perimeter. The equation for calculating spread is

$$\text{spread} = \text{intensity} - (\beta * \text{perimeter}).$$

Then ADF test will shows the p-value of the mean intensity and perimeter pair through spread value. Similar to the ADF test, R provides the Phillips-Ouliaris test named *po.test* and it also uses the zoo and tseries library in the R package (Fig. 20).

If a p-value is smaller than a given statistical significance level, generally a 0.05 or 95% confidence interval, the relation is statistically meaningful. Tables 7 and 8 show p-values among intensity, perimeter, area and shape factor features.

Table 7 summarizes the p-values obtained using the ADF test of spread (i.e. column A – (β * row B)) and the Phillips-Ouliaris cointegration test using id 2 cell. We used two different cointegration tests to determine whether the data is cointegrated or not,

because each cointegration test provides a different result. By the ADF test, only the perimeter and area pair are cointegrated ($p < .05$). In other words, either the tests all failed or the perimeter and area pair worked. The reason why the p-values of id 2 cell are below the statistical significance level, is a because of small sample size. Id 2 cell has only 24 data points and that is not enough to determine whether pairs are cointegrated or not. Even though many economists argue about optimal data points for the cointegration test, they agree that larger data sets are better for the stability of predictions. In addition, although the 95% confidence level is the most common, it is not always the most reasonable. Choosing a significance level can depend on sample size. To confirm the p-value issue, id 6 cell was also tested with the cointegration test. Id 6 cell is proper to test because it divided twice during cell growth and imaged 59 frames in LSDCAS system. Table 8 gives the p-values for each of the cointegration tests and confirms that the variables are cointegrated.

In table 8A, we can see that all the pairs of id 6 cell are cointegrated according to the p-values which are less than 0.05. The most significant parings are those with area, as indicated by the p-values which are listed only as being less than 0.01. Spread value from [intensity – (β * perimeter)] was been used for paired graph analysis because it has the minimum measurable p-value among the ratios.

Paired graph analysis with divergence threshold

The (Stock A – Stock B) and (Stock A / Stock B) graphs are the two simplest methods to do pairs trading when two stocks have the same trading patterns (i.e. constant price ratio). Each has its own merit and some traders seem to prefer one method over the other. For example, if one stock is trading at $1000 and another at $500, the difference is 500 but the ratio is only 2. This is the reason why using the ratio may yield a better comparison of the actual value of each stock. Traders can sell, or buy, one stock when the

ratio of a pair of stocks reach a specific standard deviation generally two standard deviations (S.D. or std), and stop trading stock when the ratio returns to the mean (Fig. 21).

As for pairs trading, a ratio graph of two features can be used to analyze trading pairs. We chose the intensity and perimeter pair to apply pairs trading rules to because this pair has the minimum measurable p-value from the unit root test (Table 8). Figure 22 shows the fluctuation of the (intensity / perimeter) ratio graph of id 2 and 6 cell from E5701. Two cells were collected and used as preliminary data to verify the paired graph analysis method. When the (intensity / perimeter) ratio values go down after hitting the divergence threshold, it can indicate entry into the RU state, and pinpoint the relevant frame. Using this rule, it is possible that frame 19 and frame 58 can be in the RU state because the graph is going down after hitting the divergence threshold between frame 19 and 20 of Figure 22A, and between frame 58 and 59 of Figure 22B.

A critical step in paired graph analysis and pairs trading is selecting appropriate divergence thresholds. Just as the divergence threshold determines the critical point to earn maximum profit in pairs trading, a proper divergence threshold for paired graph analysis can identify specific cell events. 2 S.D. is generally used for the divergence threshold in the pairs trading method[55,56] and we applied the same threshold to validate the paired graph analysis. Unlike stock data, recorded cell image data has dynamic movement because cells can attach or fuse with neighboring cells when they grow. To find the best threshold for detecting the most true positives (i.e. RU events) we tested various divergence thresholds (Table 9). We first utilized a 2 S.D. threshold in our paired graph analysis. This yielded a low true positive rate, and so we lowered this threshold to 1.5 S.D., which is used by some traders[56]. The 1.5 S.D. threshold yielded a high false positive rate. Therefore, in our final analysis, we tuned the threshold parameters to the data. The optimum threshold for these data is 1.65 S.D., and we used that value in our analysis. Tables 10 and 11 show the RUs detection results using the 1.65 S.D. divergence

threshold when applied to E5689 and E5677, respectively. The results show that our method has a good sensitivity; 75.69% for E5689 and 72.16% for E5677. E5689 has 594 RU cells from a total of 9207 cell events and E5677 has 194 RU cells from a total of 13085 cell events. Only 6.45% and 1.48% of RU cells exist in E5689 and E5677. As such, high false positives are always to be expected. Based on the results, we determined the paired graph analysis method is well-suited to these data and potentially to other similar data.

## Discussion

Paired graph analysis is derived from pairs trading which is commonly used to detect a maximum earning point in the stock market. We developed paired graph analysis method which is inspired by pairs trading. Our pairs trading inspired method yielded an improvement over graph trend analysis.

The major goal of automatic cell events detection and analysis system development is identifying cell division at the single cell level. If the automatic system can associate daughter cells to their parent cells, then we can calculate cell doubling time which is useful in understanding cell dynamics under various environmental conditions such as radiation and pre-clinical studies for chemicals. To achieve this goal, we developed an algorithm that can detect cell division the details of which are in Chapter 5.

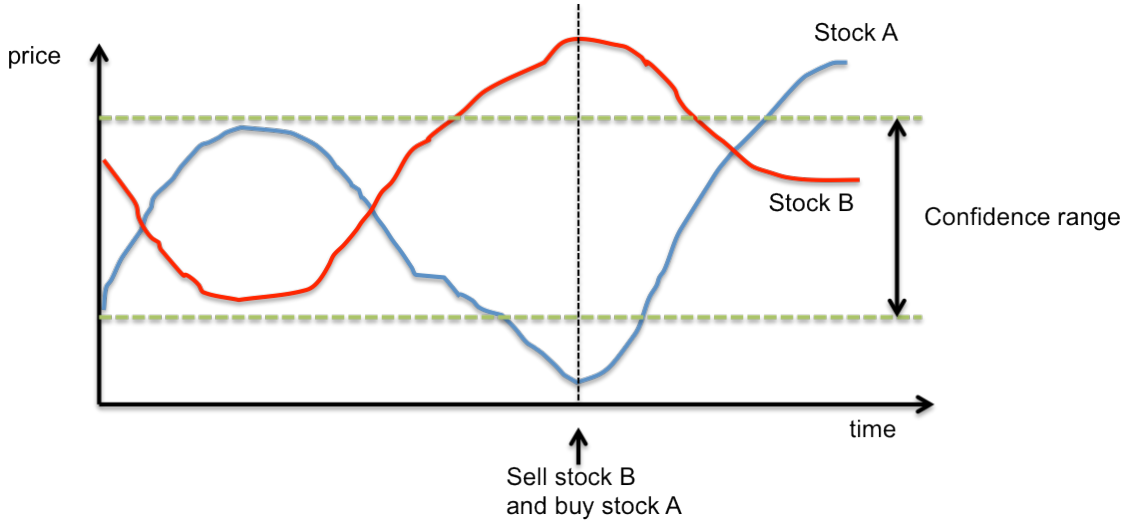
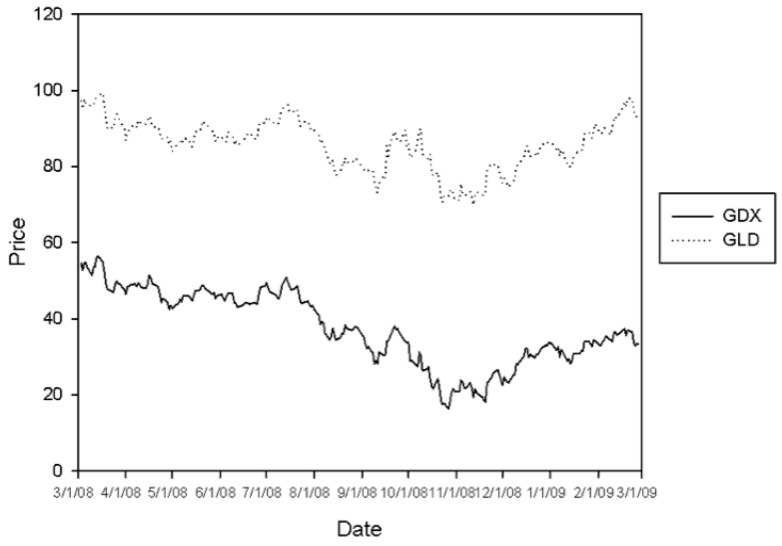
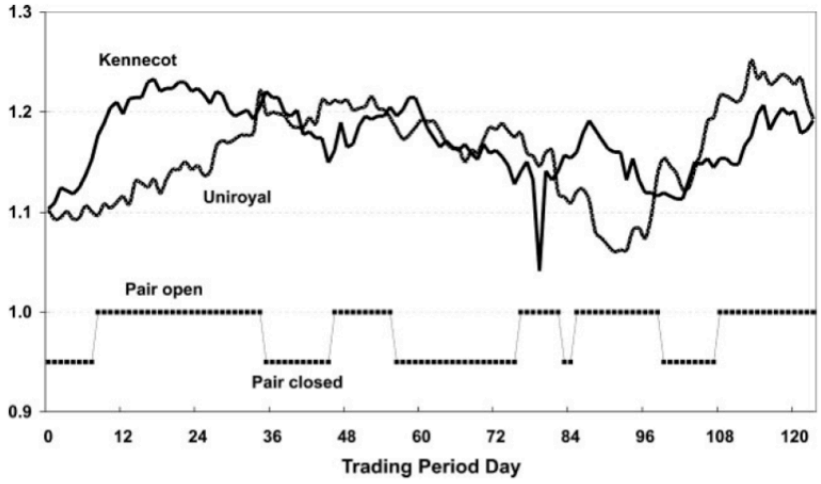



**Figure 17. Pair trading.** When the price difference between stock A and B is greater than the confidence range (i.e., two standard deviations), it is recommended to sell stock A and buy stock B. The price difference of cointegrated stocks, A and B, will go back to the confidence range because of the mean reverting property.




(A)




(B)

**Figure 18. Difference between correlation and cointegration.** A. Correlation graph between SPDR gold shares (GLD) and Market vectors gold miner ETF (GDX) from March, 2008 to February, 2009. The prices move together but are not mean revertent, B. Cointegration graph between Kennetcott and Uniroyal from August, 1963 to January, 1964. The prices move together and have a mean reverting property[48].

```
# Load the zoo and tseries packages.
#
:> library(zoo)
:> library(tseries)
# Read the CSV file into data frames.
#
:> input <- read.csv("./E5701_id_2.csv", stringsAsFactors=F)
# The 5th and 6th column each contains intensity and perimeter values. The zoo
# function can create zoo object which contain several columns from input data.
#
:> intensity <- zoo(input[,5])
:> perimeter <- zoo(input[,6])
# t.zoo is a zoo object with two columns by the merge function.
#
:> t.zoo <- merge(intensity, perimeter, all=FALSE)
# Create a data frame for applying statistical functions
#
:> t <- as.data.frame(t.zoo)
# The lm function builds linear regression models using ordinary least squares
# (OLS). This linear model have zero intercept and first regression coefficient
of
# model is extracted for hedge ratio.
#
:> m <- lm(intensity~perimeter+0, data=t)
:> beta <- coef(m)[1]
# Compute the spread
#
:> sprd <- t$intensity - beta*t$perimeter
# Augmented Dickey-Fuller test is for a unit root test.
#
:> ht <- adf.test(sprd, alternative="stationary", k=0)
# Show the p-value. A small p-value means that the spread is mean-reverting.
#
:> ht$p.value
```

Figure 19. **Cointegration test using R.** R code to test whether the intensity and perimeter pair of id 2 cell from E5701 is cointegrated using the ADF test.

```
# Load the zoo and tseries packages.
#
:> library(zoo)
:> library(tseries)
# Read the CSV file into data frames.
#
:> input <- read.csv("./E5701_id_2.csv", stringsAsFactors=F)
# The 5th and 6th column each contains intensity and perimeter values. The zoo
# function can create zoo object which contain several columns from input data.
#
:> intensity <- zoo(input[,5])
:> perimeter <- zoo(input[,6])
# t.zoo is a zoo object with two columns by the merge function.
#
:> t.zoo <- merge(intensity, perimeter, all=FALSE)
# Phillips-Ouliaris cointegration test
#
:> po.test(t.zoo, demean=FALSE)
```

**Figure 20. Cointegration test using R.** R code to test whether the intensity and perimeter pair of id 2 cell from E5701 is cointegrated using the Phillips-Ouliaris test.

**Table 7. P-values of intensity, perimeter, area, and shape factor pairs by the ADF test and the Phillips-Ouliaris cointegration test using E5701 cell id 2.**

| A \ B | intensity | perimeter | area | shape |
|---|---|---|---|---|
| intensity | | 0.2063011 | 0.1731417 | 0.1803626 |
| perimeter | 0.2069329 | | **0.04938569** | 0.2080008 |
| area | 0.1740278 | 0.05025257 | | 0.1979882 |
| shape | 0.1804134 | 0.2070812 | 0.1980294 | |

(A)

| A \ B | intensity | perimeter | area | shape |
|---|---|---|---|---|
| intensity | | 0.06075 | 0.07075 | 0.1257 |
| perimeter | 0.05514 | | 0.1492 | 0.06398 |
| area | 0.06078 | 0.1398 | | 0.05316 |
| shape | 0.1198 | 0.06605 | 0.05887 | |

(B)

Note: The id 2 cell from E5701 was used to test statistical significance ($p < 0.05$). A. All the p-values by ADF test are larger than 0.05 except perimeter and area pair. Only perimeter and area pair has statistical significance, B. All the p-values by Phillips-Ouliaris test are larger than 0.05.

**Table 8. P-values of intensity, perimeter, area, and shape factor pairs by the ADF test and the Phillips-Ouliaris cointegration test using E5701 cell id 6.**

| A \ B | intensity | perimeter | area | shape |
|---|---|---|---|---|
| intensity | | 0.01456453 | < 0.01 | 0.01484952 |
| perimeter | **0.01425749** | | < 0.01 | 0.01629093 |
| area | < 0.01 | < 0.01 | | < 0.01 |
| shape | 0.01507427 | 0.01691893 | < 0.01 | |

(A)

| A \ B | intensity | perimeter | area | shape |
|---|---|---|---|---|
| intensity | | **0.0652** | 0.0359 | 0.0325 |
| perimeter | 0.04761 | | 0.01 | 0.04817 |
| area | 0.02343 | 0.01 | | 0.03275 |
| shape | 0.02837 | **0.05434** | 0.04281 | |

(B)

Note: The id6 cell from E5701 was used to test the statistical significance. A. All the p-values by ADF test are less than 0.05 showing that all the relations among four features are statistically significant at the 95% confidence level, B. All the p-values by the Phillips-Ouliaris test. Only two pairs have larger than 0.05 p-values.

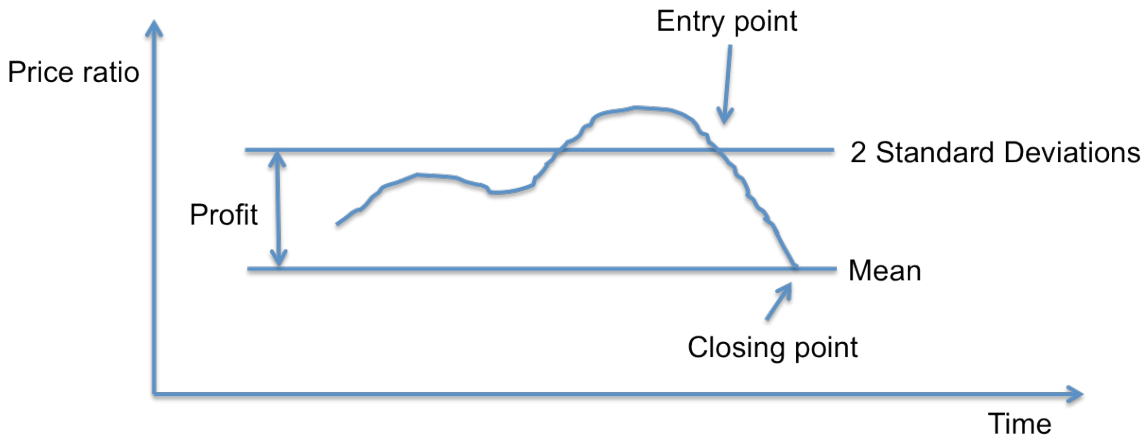



**Figure 21. Pairs trading rules.** The price ratio value between the entry point and the closing point will be the expected profit amount and ideal stock trading interval.

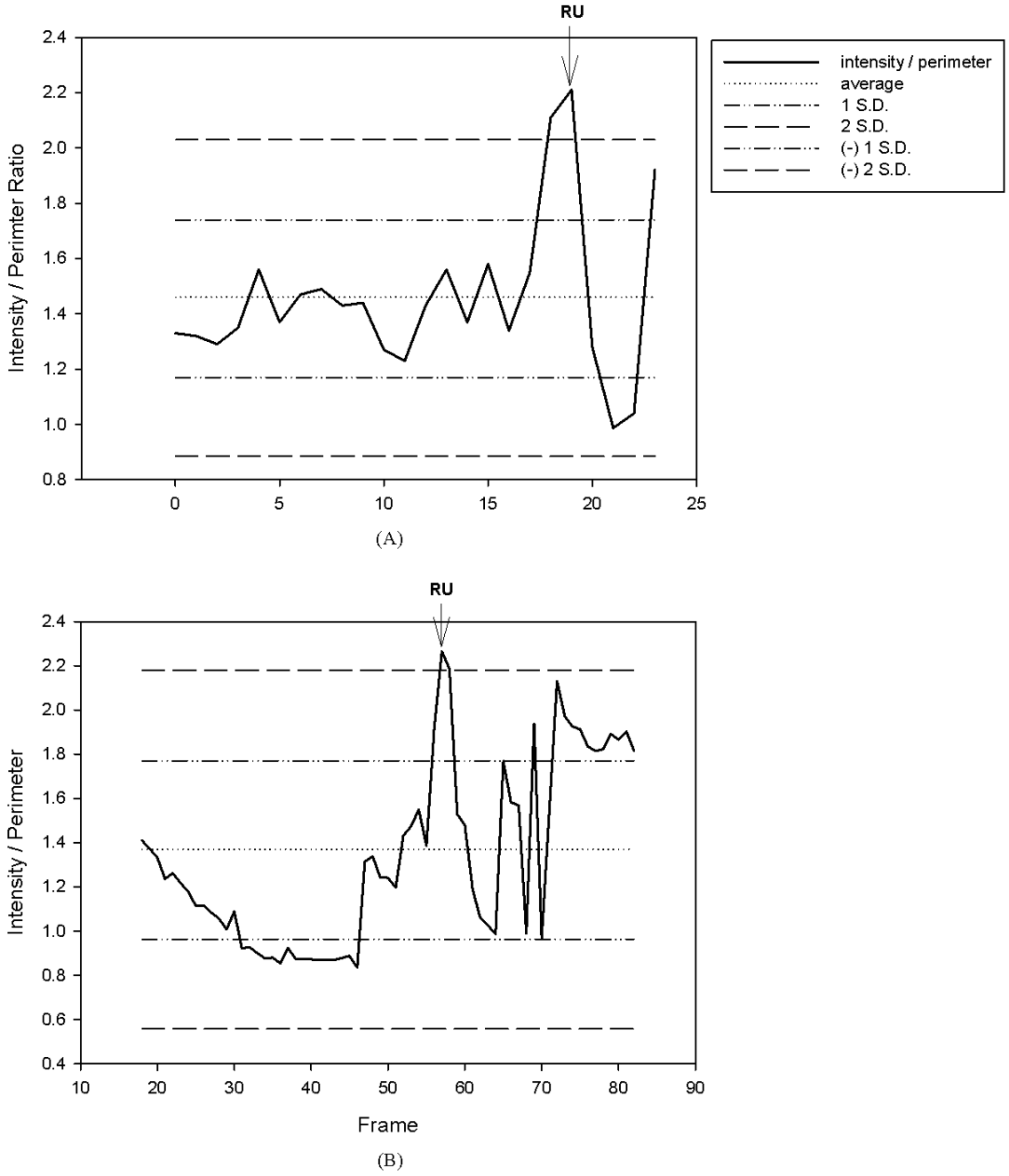



**Figure 22. Intensity / perimeter ratio graph with 1 S.D. and 2 S.D. of id 2 and 6 cell from E5701.** A. By the pairs trading rules, frame 19 is the most candidate RU frame of this cell cycle in id 2 cell ratio graph, B. Frame 58 is the most suspicious RU frame of id 6 cell.

**Table 9. Correctly detected RU events rate with Divergence threshold test using E5677.**

| | True Positive | False Positive | True Negative | False Negative |
|---|---|---|---|---|
| 1.5 S.D | 145 | 992 | 11899 | 49 |
| 1.65 S.D. | 141 | 585 | 12306 | 53 |
| 1.75 S.D. | 139 | 475 | 12416 | 55 |
| 2.0 S.D | 124 | 313 | 12578 | 70 |

Note: S.D. represents standard deviation and it used as a divergence threshold to detect candidate RU cells. Accuracy of 1.5 S.D. is 92.04%, 1.65 S.D. is 95.12%, 1.75 S.D. is 95.95% and 2.0 S.D. is 97.07%. The detection rate of 1.5 S.D. is 12.75%, 1.65 S.D. is 19.42%, 1.75 S.D. is 22.64% and 2.0 S.D. is 28.38%.

**Table 10. Paired graph analysis with 1.65 S.D. divergence threshold results using E5689.**

| | | Detected by Graph trend analysis | |
|---|---|---|---|
| | | RU | Not RU |
| Manually detected | RU | 358 (TP) | 115 (FN) |
| | Not RU | 865 (FP) | 7869 (TN) |

Note: A total number of manually detected RU cells are 473 and the accuracy of paired graph analysis using E5689 is 89.36%. Also, the sensitivity is 75.69% and the specificity is 90.1%.

**Table 11. Paired graph analysis with 1.65 S.D. divergence threshold results using E5677.**

| | | Detected by Graph trend analysis | |
|---|---|---|---|
| | | RU | Not RU |
| Manually detected | RU | 140 (TP) | 54 (FN) |
| | Not RU | 620 (FP) | 12271 (TN) |

Note: A total number of manually detected RU cells are 194 and the accuracy of paired graph analysis using E5677 is 94.85%. Also, the sensitivity is 72.16% and the specificity is 95.19%.

# CHAPTER 5

# DAUGHTER CELLS TO THEIR PARENT CELL ASSOCIATE AFTER MITOSIS

## Object Tracking and Position Estimation using Kalman Filter

Object tracking is a process to develop a trajectory by detecting objects and establishing a correlation between these objects across frames of the image stream. Object tracking can be divided into six main classes: deterministic methods, statistical methods, template and density based appearance models, multi-view appearance models, contour evolution, and matching shapes[57]. In live cell imaging, Kalman filtering is the most commonly used method in statistical object tracking.

R.E. Kalman invented a Kalman filter in 1960[58]. A Kalman filter is mathematical recursive process to estimate the state of a linear system with minimum squared error[59]. The aim of a Kalman filter is to use measurements observed over time to determine a linear system that approximates the measurement values in a least squared error optimal sense. It is widely used in object tracking and predicting, and has even been used in vehicle navigation systems[60]. Simple and recursive algorithms provide current estimates of the position coordinates using statistical models to properly update each new measurement relative to past information. Commonly used are past position, speed, acceleration, and direction. The state equation and measurement equation of a Kalman filter for general linear system are

$$\text{State equation: } x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}$$

$$\text{Measurement equation: } y_k = Hx_k + v_k$$

where $u_{k-1}$ is the control input, $w_k$ is process noise and $v_k$ is measurement noise. The noise variables are independent, normally distributed random variables.

$$P(w) \sim N(0, Q),$$

$$P(v) \sim N(0, R).$$

The process noise covariance Q and measurement noise covariance R matrices can be time-varying. The terms are typically not known in practical applications. The state equation indicates an overall signal wave of the system and the measurement equation demonstrates measurable values among signals of the system. The state of a system, denoted by the vector x, describes information about current state of the system, including spatial coordinates, angles, and acceleration bias. Because it is impossible to estimate x directly, x is estimated from repeated observation. In other words, the state of system can only be estimated by measurements of the system[60,61]. Tuning the error covariances is a challenging task for obtaining more accurate estimated x values, since the vector y is corrupted by measurement noise[59,62]. A Kalman filter is composed of an initial value and two main steps, prediction and correction (Fig. 23). $\hat{x}_k^-$ is a prior state that must be estimated forward from time step $k-1$ to step $k$. $\hat{x}_k$ denotes a posterior state that is estimated from given measurement $y_k$ at step $k$. $P_k^-$ and $P_k$ indicate a prior and a posterior estimation error covariance matrix, respectively. The optimal value can be estimated by recursive data processing using $\hat{x}_k^-$ and $y_k$.

## Results

For detecting accurate histories of individual cell fates, we need to link daughter cells with their corresponding parent cell. Accurate cell growth analysis of each image sequence will be possible only after a single cell pedigree is correctly identified. We applied reverse Kalman filter to estimate two centroids of two daughter cells and determine their parent cell by determining whether two estimated centroids are ever located in a single cell. We also combined paired graph analysis to find entry into the RU state and reverse Kalman filter to estimate centroids for finding accurate daughter cells after entry into the FO state.

Cell centroid estimation using Kalman filter

As mentioned, a Kalman filter is a recursive process to estimate the state of a linear system and often used to track objects in space over time. Cell motility in image sequences is nonlinear, but we can estimate this motion using a recursive set of linear system equations. Thus, cell motility can be estimated by previous motility and future cell centroid position can be also estimated by previous position. Figure 24 illustrates the process of identifying daughter cells after cell division. We used a Kalman filter for this process.

At first, a Kalman filter in the LSDCAS system needs initialized values of the state ($\hat{x}_{k-1}$) and the error covariance ($P_{k-1}$). We use 4x1 matrix for the state and 4x4 identity matrix for the error covariance. Then, we project the state ($\hat{x}_k^-$) and the error covariance ($P_k^-$) ahead from the initial estimated values. Predicted centroid position ($y_k$) can be generated after $\hat{x}_k^-$ calculation and is a 4x1 matrix. Kalman gain ($K_k$) is computed, $\hat{x}_k$ is updated with the actual centroid value ($z_k$). $P_k$ is also updated. The system continues to predict the cell centroid via optimal recursive data processing; repeating calculations until convergence. Results using test data to verify Kalman filter using LSDCAS data are shown in Figure 25. The solid line represents the true position, or actual, and dotted the line is the estimated position, or predicted position. The average difference between actual and predicted is about 0.086 pixels for the x-coordinate and 0.73 pixels for the y-coordinate, and the predicted centroid position was within the cell border. If the estimated centroid of a cell is within the border of a cell in a subsequent image, the LSDCAS system identifies these two cells as being the same cell. In other words, a Kalman filter can be used as a secondary verification tool for cell tracking. Further, we can separate single cells from the attached cell.

Reverse tracking and estimation using Kalman filter

Normal cell division produces two daughter cells from a parent cell. This event can be detected through a reverse application of a Kalman filter. In normal cell division, a cell splits into two daughter cells, each of which moves in different directions. The reverse Kalman filter can identify the parent cell for a pair of daughter cells by estimating the centroid of a parent cell from two centroids of daughter cells. We determine a normal division if two estimated centroids are within one cell border at any time point when we track backward (Fig. 26). An example of normal cell division shows the RU, FO and two daughter cells formations as time progresses in Figure 26A. In Figure 26B, the cells corresponding to id is 4 and 19 have an estimated (green dot) and actual (red dot) centroid at frame 21 and 22 by using reverse Kalman filter. Then two estimated centroids of id 4 and 19 at frame 20 are centered within one cell (i.e. id 4). In this case, we can determine id 4 cell normally divided into id 4 and 19. Thus, cell centroid position prediction using reverse Kalman filter is able to determine whether a cell has undergone normal cell division estimate via reverse Kalman filtering.

Table 12 summarizes the result of applying the reverse Kalman filter with candidate RU states obtained from paired graph analysis. If two estimated centroids share a candidate parent cell and the candidate parent cell had been detected as a candidate RU by paired graph analysis, we can assume the cell divided into two daughter cells through mitosis. Also we add the additional conditions to improve accuracy: 1) shape factor of candidate RU cells should be over 0.6, 2) candidate RU cells should be segmented at least 3 frames in a row, and 3) area of candidate RU cells should be larger than half of median cell area and smaller than 1.5 times of median cell area in frame 0 of each experiment. Candidate RU cells should recognized by the segmentation application because we need at least 3 steps for detecting cell division; Round Up (RU), Normal Division (ND) and Flatten Out (FO). Thus, we setup the area value range in order to eliminate false positive RU cells. RU cells tend to have the smallest area values and

frame 0 has the smallest number of cells in image streams. We set the broad range in the previous conditions to improve the accuracy rate of our analyses by using these known characteristics. The results of reverse Kalman filter with paired graph analysis shows the detection rate of the RU-FO states with daughter cells (mitosis). An accuracy rate of E5689 is 88.57% and 94.86% of E5677.

## Discussion

We expected the accurate measurement of cell pedigree is possible if we can detect appropriate RU state using paired graph analysis and/or graph trend analysis. By using reverse Kalman filter with paired graph analysis, we achieved overall 91.72% accuracy of our novel algorithm to detect cell division automatically. Total number of manually detected normal division related cell events is 1700 (425 single cell division * 4 (RU, ND, and 2 FO)) and this corresponds to 7.63% of all cell events in E5689 and E5677. Further, total number of false positives is similar with true positives.

To validate our algorithm we used a sub-tree detection and comparison method based on graph theory. Unlike other trees, all the cell events in LSDCAS image streams can be shown as a directed acyclic graph (DAG) and over time. By using the time variable and the type of event, we can determine whether the cell events detected by manual and automatic detection are identical. Details about the quantitative analysis of cell event detection using sub-tree detection and comparison methods is in Chapter 7.

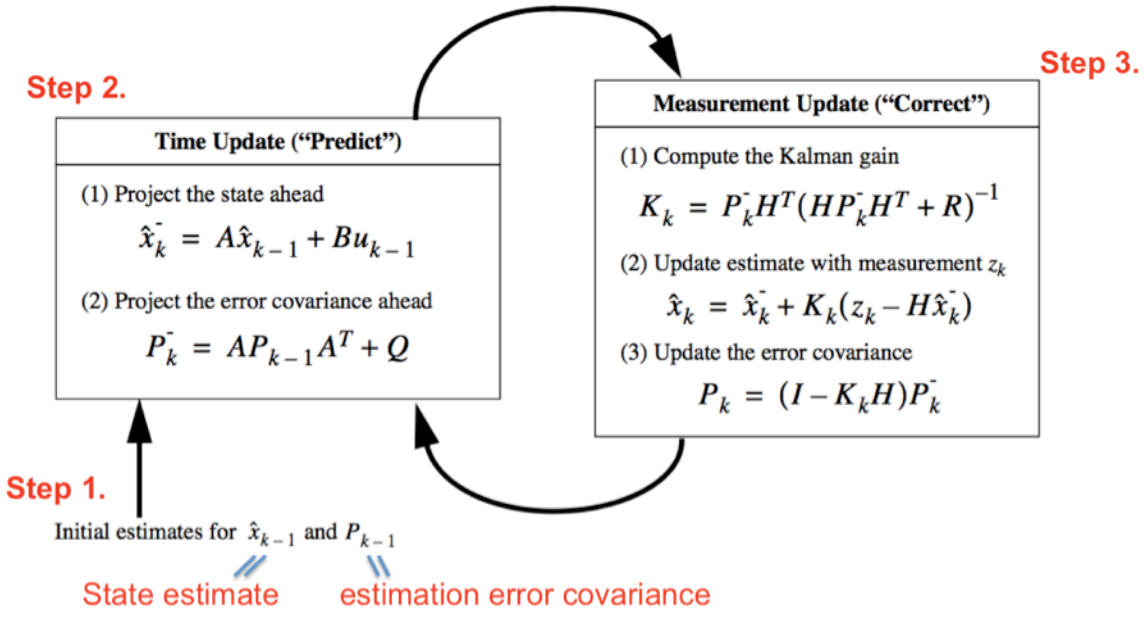



**Figure 23. Three steps of Kalman filter estimation.** The time update projects the current state estimation and error covariance forward in time. Then, the measurement update modifies the estimate by an actual measurement.

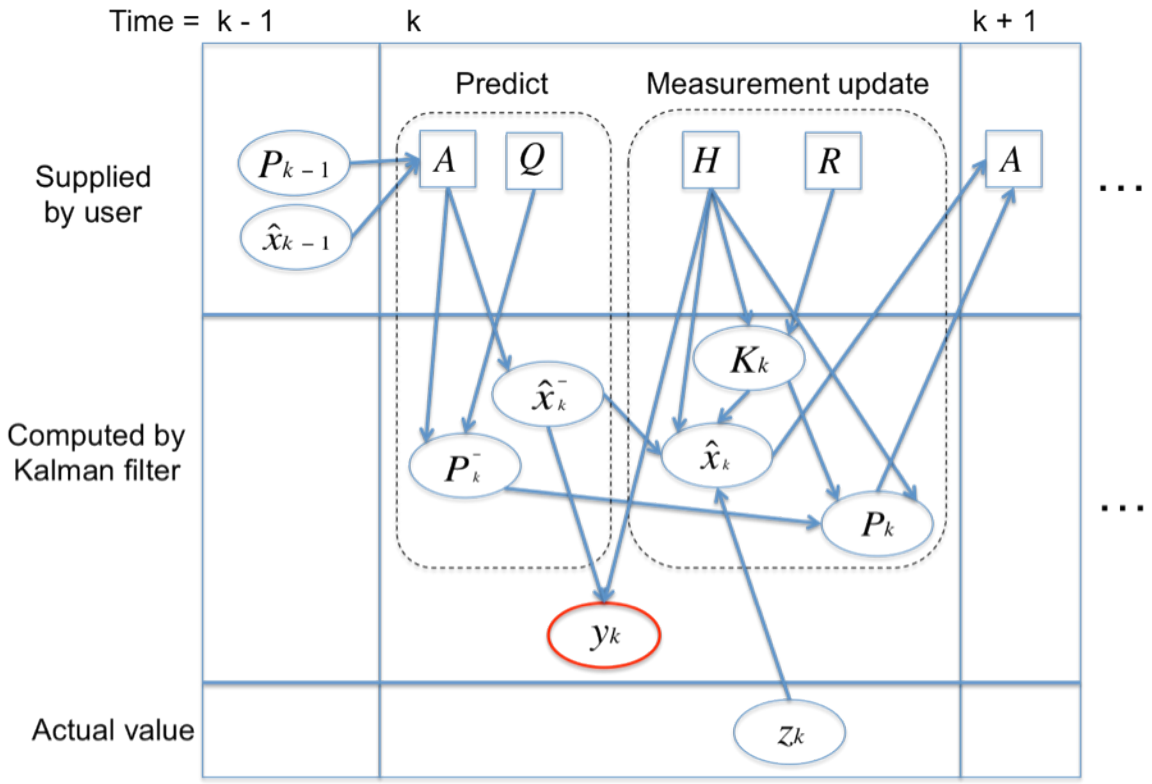



**Figure 24 .Workflow of Kalman filter for LSDCAS.** User determines and initializes $A$, $Q$, $H$, $R$ before the estimation begins, and $P$ and $\hat{x}$ at time step $k-1$. Then, $\hat{x}_k^-$ and $P_k^-$ estimates forward from time step $k-1$ to step $k$. Estimated values $y_k$ at time step $k$ can be calculated by $H$ and $\hat{x}_k^-$. $K_k$, $\hat{x}_k$, and $P_k$ are computed using $H$, $R$ and $z_k$. The recursive process of Kalman filter repeats these calculations till the estimation is complete.

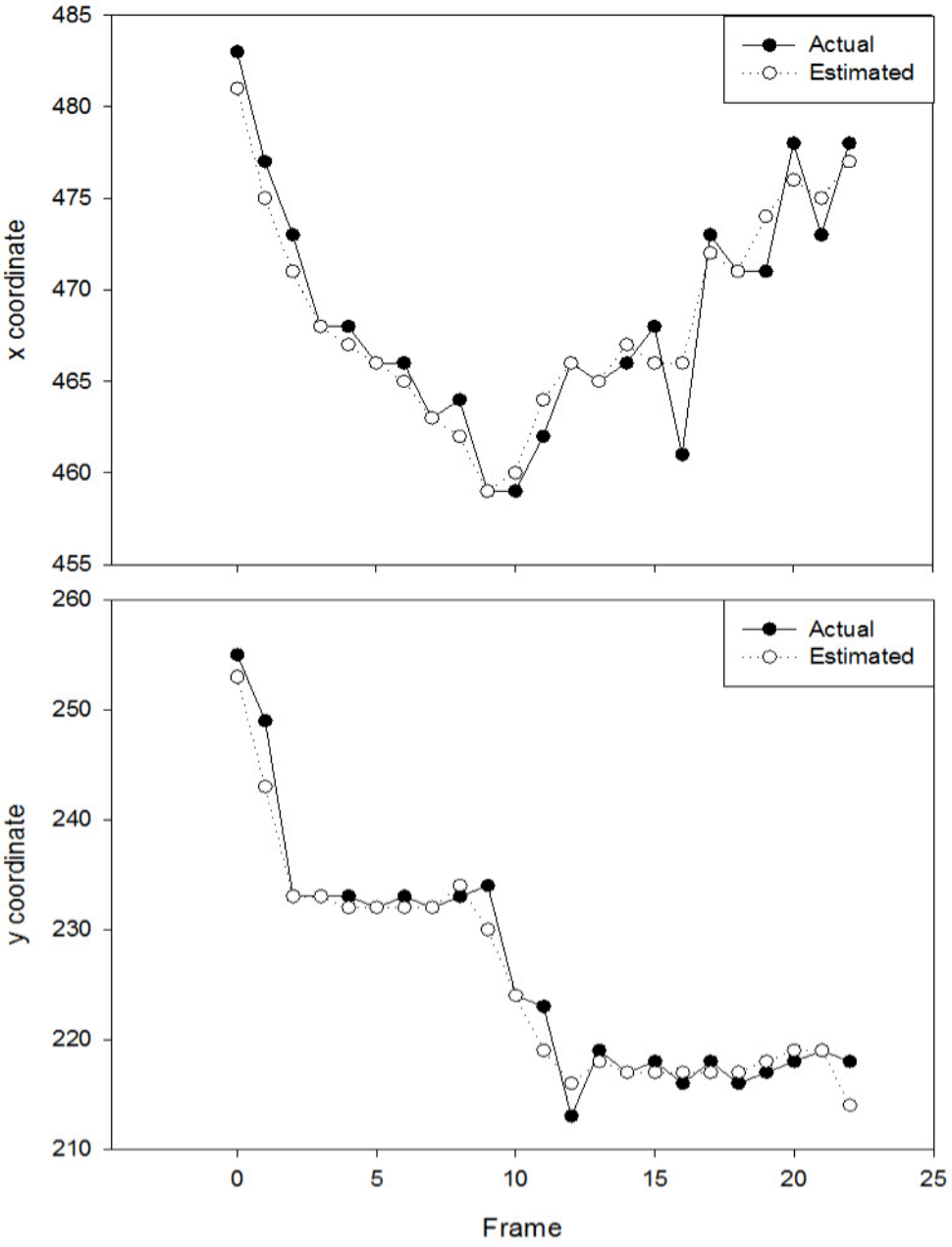



**Figure 25. Actual and estimated cell position of E5701 cell id 2 using Kalman filter.** (x, y) coordinates of actual and estimated. The maximum difference between actual and estimated x coordinate is 5 pixels in frame 16, and 6 pixels for the y coordinate in frame 1.

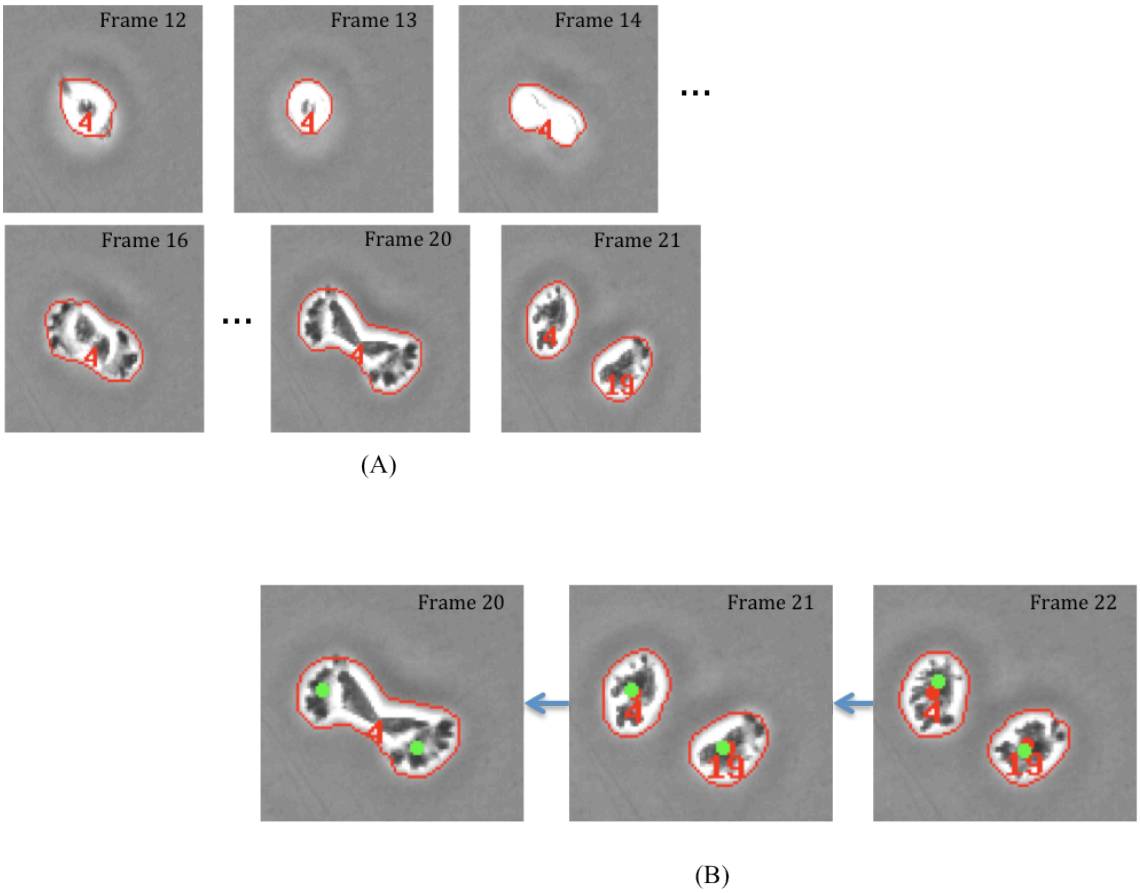



**Figure 26. Identify the parent cell for a pair of daughter cells using reverse Kalman filter.** A. An example of cell tracking for normal cell division from E5689 experiment. The cell is the RU at frame 13, the FO between frame 14 and 20, and divided two daughter cells at frame 21, B. Green and red dot represent estimated and actual cell centroid position, respectively. Id 4 cell divided normally to id 4 and 19. Two daughter cells, id 4 and 19, are used to estimate the centroid by reverse Kalman filter.

**Table 12. Reverse Kalman filter with paired graph analysis results.**

| | Total number of normal division (Manually detected) | Detected normal division | False positives | False positive rate |
|---|---|---|---|---|
| E5689 | 317 | 207 | 153 | 42.50% |
| E5677 | 108 | 73 | 133 | 64.56% |

Note: Manually detected cell divisions are only count a cell division which cell normally divided and forms two daughter cells. The detection rate of E5689 is 57.5% and E5677 is 35.44%.

# CHAPTER 6
# QUANTITATIVE ANALYSIS OF CELL EVENTS DETECTION METHODS

## Sub-tree Detection and Comparison

An essential component to our research is to extract the set of cell division events that includes the RU and FO cell events. Such a set of cell divisions is termed a sub-tree in this research. The detection of sub-trees requires the detection of RU events by paired graph analysis and the association of daughter cells to their parent cells. These two results can be combined and represented as a sub-tree. The performance of our automatic cell division detection algorithm can be verified by comparing its results with manually annotated cell division trees. Manually detected cell events are modeled as temporal sequences and represented using directed acyclic graphs (DAGs). Our automatically detected cell division events can be compared to this graph. The sequential order of the RU and FO are defined by the timestamp (frame). The general procedures for sub-tree detection and comparison method as follows: 1) find and match the graphs by identifying nodes that represent the same events in both graphs. Since the event name acts as a key, we can use it as the primary matching constraint. 2) Compare graph topology with respect to minimum time differences (these being defined by the distance between two nodes). Some pairs of nodes corresponding to different events are similar with respect to their structure and may represent the same events tree. 3) Detect and consider missed nodes. Normal cell division is assigned as a unit, or sub-tree, of the whole cell pedigree and we can verify our novel methodologies by the detection rate of these sub-trees. Furthermore, two adjacent sub-trees in different time frames can be considered to be connected by a single edge and we can build a whole pedigree by utilizing connections. Our methods focus on finding RU and FO cells, therefore we have ignored other cell events in the cell pedigree. Time differences between RU and FO between manual and

automatic detection can also be used to measure the magnitude of a cell division. Figure 27 is the example of RU events detection and the comparison procedure of manually and automatically detected cell events.

Root Mean Square (RMS)

The RMS is an error calculation method. We used it to quantify the performance of the reverse Kalman filter with the paired graph analysis. The RMS value is used to determine the absolute error of the system and it can be calculated for a series of discrete values. We have applied it to the frame differences between RU and FO events in order to compare the manual annotation and automatic detection methods. Our sub-tree cell division information can be distinguished by a combination of the event name and the frame that the event occurs. Accordingly, we calculated the absolute error of the frame differences between manually and automatically detected cell events, in what is a useful way to determine the performance of our method. The RMS squares each frame difference, and sums them, guaranteeing that differences in frame in either direction (either earlier or later for either procedure) are accumulated. Dividing by n-1 gives a standardized value, akin to calculating a variance. Taking the square root of this standardized value yields a typical frame difference between events detected between the two methods. The formulation of RMS is defined as

$$\text{RMS} = \sqrt{\frac{\Sigma(frame\ difference\ between\ RU\ and\ FO)^2}{n-1}}$$

where n is the number of sub-trees detected by both manual and automatic detection. Table 13 shows the RMS result of E5689 and E5677. The maximum frame difference in E5689 and E5677 are 4 and 6 frames, respectively. We were able to identify correctly 66.39% of cell events using the reverse Kalman filter with the paired graph analysis, with an overall false positive rate of 33.61%. For detected cell events, our RMS was 0.0532

for E5689 and 0.15 for E5677. The time differences for E5689 and E5677 are only 15.9 and 45 seconds, respectively (Table 13). E5689 has a total number of 3400 frames, 170 frames in each field, and E5677 has a total number of 4340 frames, 217 frames in each field, with a 300 second frame interval. The RMS values are only 15.9 seconds over 14 hours of E5689 and 45 seconds over 18 hours of E5677. Small RMS values demonstrate that our reverse Kalman filter with paired graph analysis method can detect cell division with 0.0313% of E5689 and 0.069% of E5677 error rates.

## Median Analysis

The RMS method is informative in determining the accuracy of the timing of our algorithm when a cell event is detected by our algorithm, but does not evaluate the rate at which cell events are detected. To determine further the efficacy of our reverse Kalman filter with paired graph analysis algorithm, a one-number summary of this quantity for a typical run is desirable. Statisticians often use measures of central tendency to demonstrate a typical value of a distribution. The median is one of the commonly used measures. Due to its robustness against outliers, we have chosen the median as the preferred measure of central tendency for our experiment. We ran our algorithm on E5689 and E5677 experiments. Because manual detection yields 100% accuracy in detecting cell events, we consider the number of cell events detected manually as the full set of cell events in the experiment. We then compute the proportion of cell events that our method was able to detect. The median of these proportions will then present the typical percent of cell events our algorithm can correctly identify. This yields a typical accuracy of the algorithm.

The estimated proportion of cell divisions detected automatically compared to manually is 0.653 for E5689 and 0.676 for E5677. In other words, reverse Kalman filter with paired graph analysis can detect approximately 66% of all cell divisions in these two samples.

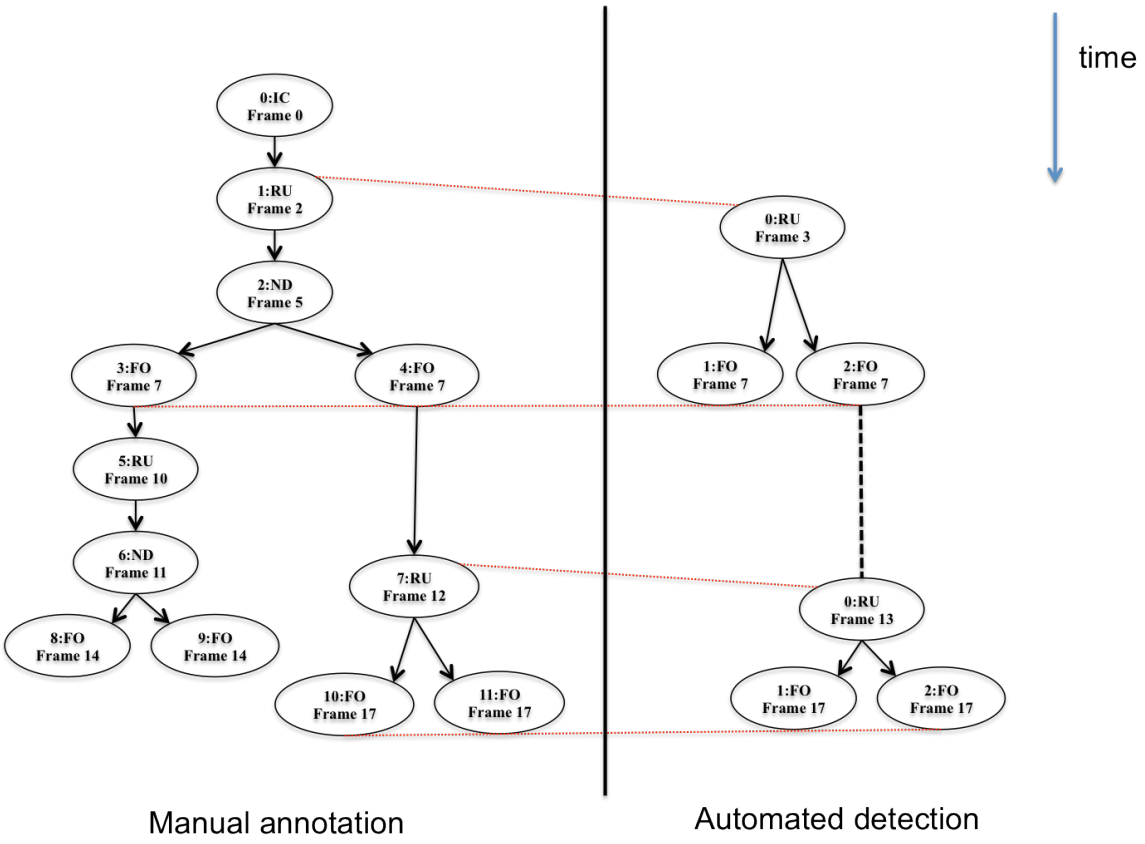



**Figure 27. Comparing manually annotated and automatically detected tree.** Automated cell event analysis methods can detect RU and FO states with frame information when it happens. By matching events name and minimum frame difference value, we can compare the result between manual annotated and automatically detected events, and evaluate the performance of analysis methods.

**Table 13. The RMS results of E5689 and E5677.**

| | RMS | |
|---|---|---|
| | frame | time |
| E5689 | 0.0531576 | 15.9 |
| E5677 | 0.150480 | 45 |

Note: 59 sub-trees out of 207 sub-trees in E5689 and 38 sub-trees out of 73 sub-trees have frame differences between manual annotation and automatic detection.

# CHAPTER 7

# DISCUSSION AND CONCLUSIONS

In this dissertation, three novel methodologies have been developed to detect cell morphological change automatically. These three methods use the graph based times-series data analysis and the applied Kalman filter. We have validated the paired graph analysis method for RU detection, the reverse Kalman filter method for FO-RU detection, and the combined paired graph analysis and reverse Kalman filter for cell division detection. The results of three novel methodologies are shown in Table 14. The novel methods are developed from graph trend analysis to reverse Kalman filter with paired graph analysis. For the data analyzed here, these methods yield improvement over the current methods in the literature with respect to detection rate and false positive rate. Because there are no unique characteristics in the data sets we analyzed that would seem to favor our methods, it is expected that these trends will hold true in other data sets as well.

## Machine Learning Based Cell Events Determination

Manually annotated RU events from E5701 and E5689 field 0 were used as a training data set for a neural network and a test data set (E5701 and E5689 field 1) is used to compare against the predicted cell events. We collected a total number of 59 RU cells from E5701 and E5689 to build a training set, and then we applied the classifier trained from each experiment to the test set. The neural network successfully detected 28 RU cells out of 29 actual RU cells from E5701 test set. This set has a total number of 5964 cell events. Further, E5689 test set has a total number of 4489 cell events and the neural network classifier detected 31 RU cells from 33 actual RU cells. In this research, we determined that a neural network is suitable model to detect cell events by true positive rates; about 97% of E5701 and about 94% of E5689.

The neural network, however, has two limitations. First, false positive rates were also high, a characteristic which can potentially degrade the performance of detecting true positives. Second, the neural network is not equipped to handle a sequence of events (i.e. the RU-FO progression). These facts indicate that additional methods for detecting cell events are still desirable, especially those that can detect a sequence of events or lower false positives rates. We have developed a method to detect a cell's entrance and exit from mitosis using time-series data analysis techniques in order to address one of the concerns of the neural network approach.

## Enter and Exit Mitosis Events Detection using Time-Series Data Analysis Method

To detect cells entering RU or FO states, we have developed a graph based technique based on the time-series data analysis known as graph trend analysis. We utilize this technique based on four features: mean intensity, area, perimeter and shape factor. When cells enter the RU state, the slopes of intensity and shape factor graphs become positive, and the slopes of area and perimeter graphs become negative. In contrast, the slopes of intensity and shape factor graphs become negative, and the slopes of area and perimeter graphs become positive when cells enter the FO state. Further, cells can have maximum mean intensity and shape factor, and minimum perimeter and area value in the RU state if the cell divides.

We collected a total of 667 manually annotated RU cells from E5689 and E5677 as a control to validate our approach. The rate of detection is high (E5689: 89.03% and E5677: 94.79%), but the graph trend analysis also has high false positive rates; less than 10% of events labeled as RU by our algorithm are truly RU events. Because of exceptionally high false positive rates (92.99% of E5689; 97.66% of E5677), we extended the search criteria to reduce false positives by utilizing a moving average which can is capable of considering slope changes over several frames.

We tested four different moving average criteria and the overall detection rates are better than without the moving average criteria. In addition, we determined graph trend analysis with a moving average (5) is the most suitable criteria for our dataset. But the moving average has one limitation: it cannot detect a specific frame for the RU event and can only detect a possible range in which the RU event occured. This range is determined by the moving average number (i.e. MA(5) determined a 5-frame range where an RU event could possibly have occurred).

To overcome these issues and to additionally account for uncertain exceptions in cell movement, we developed the paired graph analysis method.

## Cell Entry into Mitosis Detection using Time-series Data Analysis Method

Paired graph analysis is derived from time-series data analysis methods in econometrics and we have applied it to the line graph data of four features. Paired graph analysis can detect more than one cell division for a single cell in an experiment and we have demonstrated that the overall RU events detection rate is significantly improved as compared to graph trend analysis.

The mean intensity and perimeter are selected by a cointegration test and their ratio graph was used to find significant outliers. These outliers indicate RU events. We tested various threshold values to find the optimum threshold for our dataset. The 1.65 S.D. threshold was selected for our analyses. The results using E5689 and E5677 show that our method has sensitivity: 75.69% of E5689 and 72.16% of E5677. E5689 has 594 RU cells from a total of 9207 cell events and E5677 has 194 RU cells from a total of 13085 cell events. Only 6.45% and 1.48% of cells undergo an RU event in E5689 and E5677. As such, high false positives are always to be expected. Based on the results, we determined the paired graph analysis method is well-suited to these data and potentially to other similar data.

In addition, a major goal of our automatic cell events detection and analysis system development is the identification of cell division at the single cell level. If the automatic system can associate daughter cells to their parent cells, then we can calculate cell doubling time. This is useful in understanding cell dynamics under various environmental conditions, such as radiation and pre-clinical studies for chemicals. To achieve this goal, we have developed an algorithm that can detect cell division using paired graph analysis and applied Kalman filter.

Daughter Cells to Their Parent Cell Associate After Mitosis

A Kalman filter is a recursive process used to estimate the state of a linear system and is often used to track objects in space over time. Using a Kalman filter we can estimate future cell positions in LSDCAS image streams. Additionally, normal cell division can be detected through a reverse application of the Kalman filter. In normal cell division, a cell splits into two daughter cells, each of which moves in different directions. A reverse Kalman filter can identify the parent cell for a pair of daughter cells by estimating the centroid of the parent cell from the two centroids of the daughter cells.

We applied a reverse Kalman filter with paired graph analysis because all cell divisions start from RU cells. The result of the reverse Kalman filter utilized in conjunction with the candidate RU states obtained from paired graph analysis show that this combined algorithm is suitable to detect cell division automatically in our dataset. The overall accuracy rate of the reverse Kalman filter with paired graph analysis technique is 62%. The total number of manually detected normal division related cell events is 3644 and this corresponds 16.3% of all cell events in E5689 and E5677. Furthermore, total number of false positives is smaller than true positives. We consider this to be satisfactory performance for our data set.

In addition, to build and analyze whole cell pedigrees by connecting single cell division, we used a sub-tree detection and comparison method based on graph theory as a

quantitative analysis of cell events detection. Unlike other trees, all the cell events in LSDCAS image streams can be shown as directed acyclic graphs (DAGs). By using the time variable and the type of event, we can determine whether the cell events detected by manual and automatic detection are identical.

## Quantitative Analysis of Cell Events Detection Methods

We performed a quantitative comparison between our automatic cell detection algorithm and manual cell division sub-trees. We found our approach yields overall 50% of detection rate with respect to manual effort. To validate our approaches we also used a set of 7740 images from 20 image streams each from two different experiments; E5689 and E5677. The desired interval between frames in the LSDCAS image stream was 300 seconds. The dataset contained a total of 22,292 objects and 2045 cells were annotated as undergoing division during the time-lapse capture. This corresponds to 309 trees and 425 sub-trees.

Using this method, we were able to automatically detect 52% the true cell division sub-trees with our automatic detection method.

**Table 14. The detection rate and false positive rate results of three novel methodologies which developed in this research.**

| | Graph Trend Analysis | | Paired Graph Analysis | | Reverse Kalman Filter with PGA | |
|---|---|---|---|---|---|---|
| | Detection Rate | False Positive Rate | Detection Rate | False Positive Rate | Detection Rate | False Positive Rate |
| E5689 | 7.01% | 92.99% | 29.27% | 70.73% | 57.5% | 42.50% |
| E5677 | 2.34% | 97.66% | 18.42% | 81.58% | 35.44% | 64.56% |

Note: graph trend analysis and paired graph analysis methods only detect single cell events. But reverse Kalman filter with paired graph analysis method detects cell divisions.

# APPENDIX

## DETAILED RESULTS BY FIELD IN EXPERIMENTS

### Graph Trend Analysis

1. Graph trend analysis results using E5689.

| Field | A total number of RU events (Manually detected) | Detected RU events | False positives |
|---|---|---|---|
| 0 | 15 | 0 | 20 |
| 1 | 47 | 6 | 39 |
| 2 | 23 | 1 | 28 |
| 3 | 11 | 0 | 13 |
| 4 | 25 | 1 | 41 |
| 5 | 23 | 3 | 58 |
| 6 | 22 | 1 | 28 |
| 7 | 29 | 3 | 28 |
| 8 | 13 | 4 | 21 |
| 9 | 22 | 2 | 32 |
| 10 | 23 | 1 | 20 |
| 11 | 27 | 4 | 26 |
| 12 | 17 | 0 | 20 |
| 13 | 30 | 3 | 37 |
| 14 | 16 | 1 | 13 |
| 15 | 25 | 3 | 26 |
| 16 | 22 | 2 | 25 |
| 17 | 19 | 0 | 19 |
| 18 | 31 | 6 | 39 |
| 19 | 33 | 2 | 37 |
| Total | 473 | 43 | 570 |

Note: Overall detection rate is 43 / 613 = 7.01% and false positive rate is 570 / 613 = 92.99%.

2. Graph Trend analysis result using E5677.

| Field | A total number of RU events (Manually detected) | Detected RU events | False positives |
|---|---|---|---|
| 0 | 5 | 0 | 17 |
| 1 | 13 | 1 | 14 |
| 2 | 17 | 2 | 31 |
| 3 | 11 | 1 | 19 |
| 4 | 15 | 1 | 24 |
| 5 | 4 | 0 | 5 |
| 6 | 3 | 0 | 22 |
| 7 | 7 | 0 | 34 |
| 8 | 12 | 1 | 27 |
| 9 | 12 | 1 | 34 |
| 10 | 8 | 2 | 16 |
| 11 | 15 | 1 | 38 |
| 12 | 11 | 0 | 20 |
| 13 | 15 | 1 | 32 |
| 14 | 3 | 0 | 30 |
| 15 | 9 | 1 | 29 |
| 16 | 8 | 0 | 29 |
| 17 | 8 | 0 | 23 |
| 18 | 8 | 0 | 29 |
| 19 | 10 | 0 | 37 |
| Total | 194 | 12 | 500 |

Note: Overall detection rate is 12 / 512 = 2.34% and false positive rate is 500 / 512 = 97.66%.

3. Graph trend analysis results with various moving averages of E5677

| Field | A total number of RU events (Manually detected) | Detected RU by GTA | Detected RU events by GTA with MA(2) | Detected RU events by GTA with MA(3) | Detected RU events by GTA with MA(4) | Detected RU events by GTA with MA(5) |
|---|---|---|---|---|---|---|
| 0 | 5 | 0 | 1 | 1 | 0 | 1 |
| 1 | 13 | 1 | 1 | 1 | 1 | 2 |
| 2 | 17 | 2 | 2 | 3 | 2 | 0 |
| 3 | 11 | 1 | 2 | 1 | 1 | 0 |
| 4 | 15 | 1 | 0 | 2 | 0 | 1 |
| 5 | 4 | 0 | 0 | 0 | 1 | 1 |
| 6 | 3 | 0 | 0 | 0 | 0 | 0 |
| 7 | 7 | 0 | 1 | 0 | 1 | 0 |
| 8 | 12 | 1 | 0 | 1 | 1 | 1 |
| 9 | 12 | 1 | 1 | 1 | 3 | 4 |
| 10 | 8 | 2 | 1 | 1 | 1 | 1 |
| 11 | 15 | 1 | 3 | 2 | 0 | 2 |
| 12 | 11 | 0 | 1 | 1 | 0 | 0 |
| 13 | 15 | 1 | 3 | 0 | 1 | 1 |
| 14 | 3 | 0 | 0 | 0 | 0 | 0 |
| 15 | 9 | 1 | 0 | 0 | 2 | 2 |
| 16 | 8 | 0 | 0 | 0 | 0 | 1 |
| 17 | 8 | 0 | 0 | 1 | 1 | 2 |
| 18 | 8 | 0 | 0 | 2 | 0 | 1 |
| 19 | 10 | 0 | 1 | 2 | 0 | 1 |
| Total | 194 | 12 | 17 | 19 | 15 | 21 |

Paired graph analysis

1. Paired graph analysis with various divergence threshold results using E5677.

| Field | Correctly detected RU events / total detected RU events (1.5 S.D) | Correctly detected RU events / total detected RU events (1.65 S.D) | Correctly detected RU events / total detected RU events (1.75 S.D) | Correctly detected RU events / total detected RU events (2.0 S.D) |
|---|---|---|---|---|
| 0 | 5 / 35 | 4 / 28 | 4 / 25 | 4 / 17 |
| 1 | 10/ 31 | 9 / 27 | 9 / 24 | 9 / 21 |
| 2 | 12/ 58 | 12 / 50 | 12 / 42 | 12 / 32 |
| 3 | 7 / 24 | 7 / 19 | 7 / 17 | 6 / 12 |
| 4 | 10 / 57 | 10 / 51 | 10 / 43 | 9 / 33 |
| 5 | 4 / 30 | 4 / 28 | 4 / 23 | 4 / 19 |
| 6 | 3 / 31 | 3 / 28 | 3 / 24 | 3 / 14 |
| 7 | 4 / 38 | 4 / 34 | 4 / 27 | 2 / 17 |
| 8 | 11 / 42 | 10 / 39 | 9 / 32 | 8 / 25 |
| 9 | 10 / 49 | 9 / 43 | 9 / 39 | 9 / 32 |
| 10 | 5 / 41 | 5 / 35 | 5 / 27 | 3 / 20 |
| 11 | 10 / 54 | 10 / 50 | 10 / 39 | 8 / 26 |
| 12 | 7 / 41 | 7 / 39 | 7 / 34 | 6 / 23 |
| 13 | 9 / 55 | 9 / 47 | 9 / 40 | 8 / 29 |
| 14 | 2 / 29 | 1 / 21 | 1 / 19 | 1 / 11 |
| 15 | 7 / 37 | 6 / 25 | 6 / 23 | 6 / 16 |
| 16 | 5 / 33 | 5 / 23 | 5 / 21 | 5 / 14 |
| 17 | 8 / 49 | 8 / 45 | 8 / 39 | 8 / 28 |
| 18 | 8 / 68 | 8 / 61 | 8 / 50 | 7 / 31 |
| 19 | 9 / 35 | 9 / 33 | 8 / 26 | 7 / 17 |
| Total | 146 / 1137 | 140 / 726 | 139 / 614 | 121 / 437 |

2. Paired graph analysis with 1.65 S.D. divergence threshold using E5689.

| Field | Total number of RU events (Manually detected) | Detected RU events | False positives |
|---|---|---|---|
| 0 | 15 | 12 | 25 |
| 1 | 47 | 39 | 68 |
| 2 | 23 | 14 | 34 |
| 3 | 11 | 9 | 26 |
| 4 | 25 | 21 | 49 |
| 5 | 23 | 14 | 58 |
| 6 | 22 | 17 | 40 |
| 7 | 29 | 20 | 38 |
| 8 | 13 | 10 | 25 |
| 9 | 22 | 17 | 58 |
| 10 | 23 | 13 | 32 |
| 11 | 27 | 22 | 26 |
| 12 | 17 | 12 | 37 |
| 13 | 30 | 24 | 59 |
| 14 | 16 | 13 | 51 |
| 15 | 25 | 15 | 58 |
| 16 | 22 | 18 | 40 |
| 17 | 19 | 19 | 33 |
| 18 | 31 | 21 | 58 |
| 19 | 33 | 28 | 50 |
| Total | 473 | 358 | 865 |

Note: Overall detection rate is 358 / 1223 = 29.27% and false positive rate is 865 / 1223 = 70.73%.

3. Paired graph analysis with 1.65 S.D. divergence threshold using E5677.

| Field | Total number of RU events (Manually detected) | Detected RU events | False positives |
|---|---|---|---|
| 0 | 5 | 4 | 24 |
| 1 | 13 | 9 | 18 |
| 2 | 17 | 12 | 38 |
| 3 | 11 | 7 | 12 |
| 4 | 15 | 10 | 41 |
| 5 | 4 | 4 | 25 |
| 6 | 3 | 3 | 25 |
| 7 | 7 | 4 | 30 |
| 8 | 12 | 10 | 29 |
| 9 | 12 | 9 | 34 |
| 10 | 8 | 5 | 30 |
| 11 | 15 | 10 | 40 |
| 12 | 11 | 7 | 36 |
| 13 | 15 | 9 | 45 |
| 14 | 3 | 1 | 20 |
| 15 | 9 | 6 | 19 |
| 16 | 8 | 5 | 18 |
| 17 | 8 | 8 | 46 |
| 18 | 8 | 8 | 64 |
| 19 | 10 | 9 | 26 |
| Total | 194 | 140 | 620 |

Note: Overall detection rate is 140 / 760 = 18.42% and false positive rate is 620 / 760 = 81.58%.

Reverse Kalman filter with paired graph analysis

1. Reverse Kalman filter with paired graph analysis results using E5689.

| Field | Total number of normal division (Manually detected) | Detected normal division | False positives |
|---|---|---|---|
| 0 | 13 | 10 | 10 |
| 1 | 24 | 13 | 18 |
| 2 | 16 | 8 | 8 |
| 3 | 5 | 5 | 5 |
| 4 | 15 | 13 | 10 |
| 5 | 13 | 9 | 5 |
| 6 | 19 | 14 | 7 |
| 7 | 18 | 10 | 10 |
| 8 | 8 | 7 | 6 |
| 9 | 20 | 11 | 9 |
| 10 | 14 | 6 | 5 |
| 11 | 20 | 12 | 6 |
| 12 | 14 | 8 | 2 |
| 13 | 19 | 12 | 9 |
| 14 | 17 | 16 | 10 |
| 15 | 14 | 9 | 7 |
| 16 | 17 | 9 | 6 |
| 17 | 18 | 13 | 4 |
| 18 | 16 | 12 | 11 |
| 19 | 17 | 10 | 5 |
| Total | 317 | 207 | 153 |

Note: Overall detection rate is 207 / 360 = 57.5% and false positive rate is 153 / 360 = 42.5%.

2. Reverse Kalman filter with paired graph analysis results using E5677.

| Field | Total number of normal division (Manually detected) | Detected normal division | False positives |
|---|---|---|---|
| 0 | 5 | 3 | 7 |
| 1 | 7 | 4 | 3 |
| 2 | 6 | 4 | 10 |
| 3 | 4 | 2 | 2 |
| 4 | 8 | 5 | 10 |
| 5 | 3 | 3 | 4 |
| 6 | 3 | 3 | 6 |
| 7 | 4 | 3 | 7 |
| 8 | 8 | 8 | 6 |
| 9 | 9 | 8 | 10 |
| 10 | 4 | 1 | 8 |
| 11 | 6 | 5 | 6 |
| 12 | 5 | 3 | 5 |
| 13 | 4 | 3 | 10 |
| 14 | 1 | 0 | 4 |
| 15 | 7 | 4 | 6 |
| 16 | 4 | 3 | 3 |
| 17 | 6 | 3 | 9 |
| 18 | 6 | 2 | 11 |
| 19 | 8 | 6 | 6 |
| Total | 108 | 73 | 133 |

Note: Overall detection rate is 73 / 206 = 35.44% and false positive rate is 133 / 206 = 64.56%.

REFERENCES

1. Lodish H. *Molecular cell biology*. 6th ed. New York: W.H. Freeman; 2008.

2. Dunn GA, Jones GE. Cell motility under the microscope: Vorsprung durch Technik. *Nature Reviews Molecular Cell Biology*. 2004;5(8):667–672.

3. Rieder CL. Mitosis Through the Microscope: Advances in Seeing Inside Live Dividing Cells. *Science*. 2003;300(5616):91–96.

4. Stephens DJ. Light Microscopy Techniques for Live Cell Imaging. *Science*. 2003;300(5616):82–86.

5. Ahmed WM, Leavesley SJ, Rajwa B, et al. State of the Art in Information Extraction and Quantitative Analysis for Multimodality Biomolecular Imaging. *Proc. IEEE*. 2008;96(3):512–531.

6. Huang K, Murphy RF. From quantitative microscopy to automated image understanding. *Journal of biomedical optics*. 2004;9:893.

7. Dormann D, Weijer CJ. Imaging of cell migration. *The EMBO journal*. 2006;25(15):3480–3493.

8. Ianzini F, Mackey MA. Development of the large scale digital cell analysis system. *Radiation protection dosimetry*. 2002;99(1-4):289–293.

9. Deasy BM, Chirieleison SM, Witt AM, Peyton MJ, Bissell TA. Tracking stem cell function with computers via live cell imaging: identifying donor variability in human stem cells. *Operative Techniques in Orthopaedics*. 2010;20(2):127–135.

10. Hinchcliffe EH. Using long-term time-lapse imaging of mammalian cell cycle progression for laboratory instruction and analysis. *CBE—Life Sciences Education*. 2005;4:284–290.

11. Alberts B. *Molecular biology of the cell*. 4th ed. New York: Garland Science; 2002.

12. Davis PJ, Kosmacek EA, Sun Y, Ianzini F, Mackey MA. The large-scale digital cell analysis system: an open system for nonperturbing live cell imaging. *Journal of Microscopy*. 2007;228(3):296–308.

13. Joshi SD, Davidson LA. Live-cell Imaging and Quantitative Analysis of Embryonic Epithelial Cells in *Xenopus laevis*. *JoVE*. 2010;(39).

14. Ianzini F, Bresnahan L, Wang L, Anderson K, Mackey MA. The Large Scale Digital Cell Analysis System and its use in the quantitative analysis of cell populations. In: *Microtechnologies in Medicine & Biology 2nd Annual International IEEE-EMB Special Topic Conference on*.; 2002:470–475.

15. Conrad C. Automatic Identification of Subcellular Phenotypes on Human Cell Arrays. *Genome Research*. 2004;14(6):1130–1136.

16. DeLuca JG, Gall WE, Ciferri C, et al. Kinetochore Microtubule Dynamics and Attachment Stability Are Regulated by Hec1. *Cell*. 2006;127(5):969–982.

17. Obaid AL, Loew LM, Wuskell JP, Salzberg BM. Novel naphthylstyryl-pyridinium potentiometric dyes offer advantages for neural network analysis. *Journal of neuroscience methods*. 2004;134(2):179–190.

18. White J, Stelzer E. Photobleaching GFP reveals protein dynamics inside live cells. *Trends in Cell Biology*. 1999;9(2):61–65.

19. Pattison D, Davies M. Actions of ultraviolet light on cellular structures. *Cancer: Cell structures, carcinogens and genomic instability*. 2006:131–157.

20. Yang F, Mackey MA, Ianzini F, Gallardo G, Sonka M. Cell segmentation, tracking, and mitosis detection using temporal context. *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2005*. 2005:302–309.

21. Yang F, Mackey MA, Ianzini F, Gallardo GM, Sonka M. Segmentation and quantitative analysis of the living tumor cells using large-scale digital cell analysis system. In: *Proceedings of SPIE*. San Diego, CA, USA; 2004:1755–1763. Available at: http://link.aip.org/link/PSISDG/v5370/i1/p1755/s1&Agg=doi.

22. Ersoy I, Bunyak F, Mackey MA, Palaniappan K. Cell segmentation using Hessian-based detection and contour evolution with directional derivatives. In: *15th IEEE International Conference on Image Processing, 2008. ICIP 2008*.; 2008:1804–1807.

23. Padfield D, Rittscher J, Thomas N, Roysam B. Spatio-temporal cell cycle phase analysis using level sets and fast marching methods. *Medical Image Analysis*. 2009;13(1):143–155.

24. Coskun H, Li Y, Mackey MA. Ameboid cell motility: A model and inverse problem, with an application to live cell imaging data. *Journal of theoretical biology*. 2007;244(2):169–179.

25. Kosmacek EA. Live cell imaging technology development for cancer research. *Theses and Dissertations*. 2009:388.

26. Goeman JJ, Mansmann U. Multiple testing on the directed acyclic graph of gene ontology. *Bioinformatics*. 2008;24(4):537.

27. Sato K, Mituyama T, Asai K, Sakakibara Y. Directed acyclic graph kernels for structural RNA analysis. *BMC bioinformatics*. 2008;9(1):318.

28. Tan P-N. *Introduction to data mining*. London: Addison Wesley; 2006.

29. Mitchell T. *Machine Learning*. New York: McGraw-Hill; 1997.

30. Alpaydin E. *Introduction to machine learning*. 2nd ed. Cambridge Mass.: MIT Press; 2010.

31. Shariff A, Kangas J, Coelho LP, Quinn S, Murphy RF. Automated Image Analysis for High-Content Screening and Analysis. *Journal of Biomolecular Screening*. 2010;15(7):726–734.

32. Glory E, Murphy RF. Automated Subcellular Location Determination and High-Throughput Microscopy. *Developmental Cell*. 2007;12(1):7–16.

33. Jain AK, Mao RPW. Statistical pattern recognition: A review. *IEEE Transactions on pattern analysis and machine intelligence*. 2000;22(1):4–37.

34. Harder N, Mora-Bermúdez F, Godinez W, et al. Automated analysis of the mitotic phases of human cells in 3D fluorescence microscopy image sequences. *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2006*. 2006:840–848.

35. Winters-Hilt S, Merat S. SVM clustering. *BMC bioinformatics*. 2007;8(Suppl 7):S18.

36. Dong M, Wu jing. Localized support vector machines for classification. *2006 International Joint Conference on Neural Networks*. 2006;1:799–805.

37. Huang K, Murphy RF. Automated classification of subcellular patterns in multicell images without segmentation into single cells. *Proc IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI'04)*. 2004:1139–1142.

38. Haykin S. *Neural networks and learning machines*. 3rd ed. New York: Prentice Hall/Pearson; 2009.

39. Scott JA. Artificial neural networks and image interpretation: A ghost in the machine. In: *Seminars in Ultrasound, CT, and MRI*.Vol 25.; 2004:396–403.

40. Bishop C. *Natural networks for pattern recognition*. Repr. Oxford [u.a.]: Oxford Univ. Press; 2007.

41. Basu JK, Bhattacharyya D, Kim T, Kolkata I, Daejeon K. Use of Artificial Neural Network in Pattern Recognition. *International J International Journal of Software Engineering and Its Applications ournal of Software Engineering and Its Applications ournal of Software Engineering and Its Applications*. 2010;4(2).

42. Lisboa PJ, Taktak AF. The use of artificial neural networks in decision support in cancer: a systematic review. *Neural Networks*. 2006;19(4):408–415.

43. Dougherty E. *Genomic signal processing and statistics*. New York: Hindawi Pub. Corp. 2005.

44. Orlov N, Johnston J, Macura T, Shamir L, Goldberg I. Computer vision for microscopy applications. *Vision Systems: Segmentation and Pattern Recognition. Vienna, Austria: I-Tech Education and Publishing*. 2007.

45. Weka 3 - Data Mining with Open Source Machine Learning Software in Java (http://www.cs.waikato.ac.nz/ml/weka/).

46. Long X, Cleveland WL, Yao YL. Automatic detection of unstained viable cells in bright field images using a support vector machine with an improved training procedure. *Computers in Biology and Medicine*. 2006;36:339–362.

47. http://www.babypips.com/school/images/grade1/trendlines-example.png.

48. Gatev E, Goetzmann WN, Rouwenhorst KG. Pairs trading: Performance of a relative-value arbitrage rule. *Review of Financial Studies*. 2006;19(3):797–827.

49. Vidyamurthy G. *Pairs trading : quantitative methods and analysis*. Hoboken N.J.: J. Wiley; 2004.

50. Engle RF, Granger CWJ. Co-integration and error correction: representation, estimation and testing. *Econometrica*. 1987;55(2):251–76.

51. Jameson R. *Managing energy price risk*. 2nd ed. London: Risk; 1999.

52. Johansen S. Estimation and hypothesis testing of cointegration vectors in Gaussian vector autoregressive models. *Econometrica: Journal of the Econometric Society*. 1991:1551–1580.

53. Phillips PCB, Ouliaris S. Testing for cointegration using principal components methods. *Journal of Economic Dynamics and Control*. 1988:205–230.

54. Pfaff B. *Analysis of integrated and cointegrated time series with R*. 2nd ed. New York: Springer; 2008.

55. Herlemont D. Pairs Trading, Convergence Trading, Cointegration. *YATS Finances and Technology*. 2003.

56. Ehrman D. *The handbook of pairs trading : strategies using equities, options, and futures*. Hoboken N.J.: John Wiley & Sons; 2006.

57. Yilmaz A, Javed O, Shah M. Object Tracking: A Survey. *ACM Comput. Surv.* 2006;38(4):Article 13.

58. Kalman RE, others. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*. 1960;82(1):35–45.

59. Welch G, Bishop G. An Introduction to the Kalman Filter. 2000.

60. Simon D. Kalman filtering. *Embedded Systems Programming*. 2001;14(6):72–79.

61. Hartikainen E, Ekelin S. Tuning the Temporal Characteristics of a Kalman-Filter Method for End-to-End Bandwidth Estimation. *IEEE E2EMON*. 2006.

62. Akesson BM, Jorgensen JB, Poulsen NK, Jorgensen SB. A tool for kalman filter tuning. *Computer Aided Chemical Engineering*. 2007;24:859–864.