

UC Riverside

UC Riverside Electronic Theses and Dissertations

Title

Computational Video Bioinformatics for Understanding the Dynamics of Living Cells

Permalink

<https://escholarship.org/uc/item/49x4g4mk>

Author

On, Vincent Wong

Publication Date

2018

Supplemental Material

<https://escholarship.org/uc/item/49x4g4mk#supplemental>

Copyright Information

This work is made available under the terms of a Creative Commons Attribution-ShareAlike License, available at <https://creativecommons.org/licenses/by-sa/4.0/>

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
RIVERSIDE

Computational Video Bioinformatics for Understanding the Dynamics of Living Cells

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Electrical Engineering

by

Vincent Wong On

September 2018

Dissertation Committee:

Dr. Bir Bhanu, Chairperson

Dr. Prue Talbot

Dr. Hyle Park

Copyright by
Vincent Wong On
2018

The Dissertation of Vincent Wong On is approved:

Committee Chairperson

University of California, Riverside

Acknowledgements

I thank my mentor and advisor Dr. Bir Bhanu for his guidance and for the qualities he has instilled in me to help me succeed. I would like to thank my committee members, Dr. Prue Talbot and Dr. Hyle Park for being on my defense committee and for providing guidance through my academic career. I would like also thank Dr. Yingbo Hua and Dr. Ilya Dumer for being on my oral examination committee. I am grateful to my colleagues in VISLAB for their advice and support: Dr. Anthony Bianchi, Dr. Alberto Cruz, Dr. Ninad Thakoor, Dr. Asong Tambo, and all my other peers in VISLAB. I would like to thank the IGERT Video bioinformatics faculty and fellows for their knowledge and support. I would also like to give my gratitude to all the members of the Talbot Lab and Stem Cell Core who have provided me with their time, support, and knowledge. Finally, I would like to give special thanks to Dr. Atena Zahedi for her wonderful help and support on our many collaborations. Support for this work was provided in part by the NSF IGERT: Video Bioinformatics (# DGE 0903667), the National Institute of Mental Health (# MH67121), the California Institute for Regenerative Medicine (# NE-A0005A-1E), and the Tobacco-Related Disease Research Program of CA (#22FT-0127 and #20PT-0184). Portions of this dissertation appear in “Evaluating Cell Processes, Quality, and Biomarkers in Pluripotent Stem Cells Using Video Bioinformatics” by Zahedi et al. 2016 PLOS ONE, “Automated Spatio-Temporal Analysis of Dendritic Spines and Related Protein Dynamics” by On et al. 2017 PLOS ONE, and “PhaseR4D: Time-Lapse 3D Phase Contrast Imaging Using Supervised Regression” by On et al. 2018 PLOS ONE.

I dedicate this to my mother Tina On, my sister Chi Tran, and my close friends and family for their love and support which helped me to succeed in life.

ABSTRACT OF THE DISSERTATION

Computational Video Bioinformatics for Understanding the Dynamics of Living Cells

by

Vincent Wong On

Doctor of Philosophy, Graduate Program in Electrical Engineering

University of California, Riverside, September 2018

Dr. Bir Bhanu, Chairperson

Recent advances in microscopy technologies such as high-throughput imaging, super-resolution, and 3D microscopy have revolutionized our ability to study cells and their underlying biological processes. However, many research groups are overwhelmed by the quantity and complexity of this new data. Traditional methods are time consuming, subject to bias, and difficult to reproduce. Because of this, it is highly advantageous to develop convenient software and tools to help cell researchers perform and analyze experiments. Video bioinformatics is an interdisciplinary field that automatically processes, analyzes, and visualizes biological spatiotemporal data using biology, computer science, and engineering methods. Here we present three video bioinformatics projects and software toolkits that automatically analyze, classify, and visualize biological processes and structures in multidimensional image sets. All three software packages were developed using novel machine learning, image processing, and computer vision algorithms. Unique microscopy datasets were collected for each experiment and were used to test and validate each developed software package. (1) StemCellQC, a bioinformatics toolkit that can automatically extract features from phase contrast videos of human embryonic stem cells, produce analyses, and classify cell

health. (2) PhaserR4D, a software that can produce live 3D phase contrast videos by fusing phase and fluorescent image stacks captured on commercially available microscopes. (3) DendritePA, a pattern recognition software that can analyze subpixel protein trafficking events in neurons by using spatiotemporal information present in multichannel fluorescence videos.

TABLE OF CONTENTS

Chapter 1 Introduction.....	1
Chapter 2 StemCellQC: Feature Analysis and Classification.....	7
Introduction.....	7
Materials and Methods.....	9
Results.....	28
Discussion.....	36
Chapter 3 PhaseR4D: Time-Lapse 3D Phase Contrast Reconstruction.....	40
Introduction.....	40
Related Work.....	43
Materials and Methods.....	45
Results.....	64
Chapter 4 DendritePA: Fluorescent Protein Dynamics.....	73
Introduction.....	73
Related Work.....	77
Materials and Methods.....	81
Results.....	97
Chapter 5 Conclusions	108
References.....	113

List of Figures

Figure 2.1. Features related to hESC colony growth.....	15
Figure 2.2. Surface protrusions on colonies.....	17
Figure 2.3. Features related to hESC colony motility.....	18
Figure 2.4. Features related to cell death.....	20
Figure 2.5. Two-Feature Plot Analysis.	23
Figure 2.6. hESC Health Timeline and Biomarkers.....	39
Figure 3.1. PhaseR4D Overview Diagram.....	48
Figure 3.2. Corresponding Phase Contrast and Fluorescent Images.....	50
Figure 3.3. Variance Plots for Various Fluorescent Labels.....	53
Figure 3.4. Cell Base Estimation from 2nd Derivative.....	55
Figure 3.5. Cell Base for 3D Phase Contrast.....	58
Figure 3.6. Live 3D Reconstruction of Phase Contrast over time.....	62
Figure 3.7. Regressed Depth Maps and 3D Reconstructions.....	64
Figure 3.8. Local Surface Patches at Specific Cell Locations.....	67
Figure 4.1. Cofilin dynamics in dendritic spines.....	74
Figure 4.2. System Overview Diagram.....	83
Figure 4.3. Effect of preprocessing video frames.....	86
Figure 4.4. Dendrite Segmentation.....	88
Figure 4.5. Spine Segmentation.....	91
Figure 4.6. Spine energy image.....	96
Figure 4.7. Ground-truth decision tree.....	100

Figure 4.8. ROC Plots.....	103
Figure 4.9. SEI classification examples.....	104
Figure 4.10. Cofilin-Spine Graphs.....	107

List of Tables

Tables 2.1. Classification Results Using 48 Hours of Video.....	25
Tables 2.2. Classification Results Using 36 Hours of Video.....	25
Tables 2.3. Classification Results Using 24 Hours of Video.....	26
Tables 2.4. Classification Results Using 12 Hours of Video.....	26
Tables 3.1. Local Surface Patch Surface Types and χ^2 -Divergence.....	67
Tables 3.2. Classification Results for Single Cell Control Samples.....	69
Tables 3.3. Classification Results for Single Cell Treated Samples.....	69
Tables 3.4. Classification Results for Control Colony Samples.....	70
Tables 3.5. Classification Results for Treated Colony Samples.....	70
Tables 4.1. Dendrite segmentation results.....	98
Tables 4.2. Spine segmentation results.....	100
Tables 4.3. Classification results.....	102
Tables 4.4. Confusion Matrices.....	102

Chapter 1 Introduction

Bioinformatics is an interdisciplinary field of science that combines computer science, engineering, mathematical, and statistical methods to address complex biological questions. Bioinformatics integrates the use of computers, software tools, and databases to analyze physical specimens and understand their complex systems. In addition, bioinformatics also focuses on the development of new technology that aid in the study of biology. These technologies include new software, algorithms, databases, and specialized hardware. Quantitative tools such as these are indispensable to modern biologists due to their ability to efficiently streamline experiments and provide alternative analyses [1]–[3].

While bioinformatics can be applied to many forms of biological data, image sets and videos of biological specimens collected using microscopy is particularly informative. Due to the complexity and size of video data, extracting and managing information from videos is extremely difficult to do without the aid of computers. Video bioinformatics is defined as the automated processing, analysis, and visualization of biological spatiotemporal data. This field integrates expertise from life sciences, computer science, and engineering while aiming to extract biological knowledge and develop tools using dynamic image sets and microscopy videos. Like traditional bioinformatics, video bioinformatics is an interdisciplinary field that develops methods and software tools for the understanding biological data. However, video bioinformatics focuses specifically on multidimensional data and more than a single image or data point. In an individual image, analysis is limited to a snapshot of biological processes that are

naturally dynamic and continuous. By using live video imaging, video bioinformatics can provide powerful insight that would not be possible from traditional bioinformatics [4], [5].

Recent advances in microscopy technologies such as high-throughput imaging, super-resolution microscopy, and 3D microscopy have revolutionized our understanding of dynamic biological processes. These modern technologies can be used for time-lapse imaging of cellular and molecular systems across multiple spatial dimensions. However, modern biologists are overwhelmed by the amount of data these devices can produce. To manage this data, proper tools for their organization, analysis, and interpretation must be developed. Because of this, it is highly advantageous to create video bioinformatics software and toolkits that can aid researchers rapidly, automatically, and accurately analyze their data. Such tools will enable the study of biological processes that were previously impossible or too time consuming to analyze. These technologies also allow for multiscale studies both spatially and temporally. These experiments will also benefit from greater objectivity and repeatability.

Video bioinformatics has particularly useful applications in cell biology. The complexity of biological videos is more challenging than structured medical data and requires interdisciplinary research. As there are many forms of microscopy such as bright field, fluorescence, and phase contrast, there are many ways to visualize and study cells. Each modality provides unique information that the others do not. For the case of phase contrast, internal substructures of a cell become visible. To analyze these substructures, a region-of-interest (ROI) must be detected on the image. Traditionally

this required an expert in the field to manually outline the objects by hand. This can be very time consuming and susceptible to human bias. Video bioinformatics can automatically outline substructures using segmentation algorithms which will improve efficiency and reduce errors. In addition, software may be developed to be used on high-throughput data by individuals who have no experience with bioinformatics or computer science. In fluorescence microscopy, the expression of biological processes may be labelled and visualized in an image. In this case, bioinformatics may be used to extract features from the image that were invisible to the naked eye. In some cases, it may be insufficient to use only one microscopy modality to analyze or visualize a biological process. Video bioinformatics can extract information from multiple modalities and fuse them for the desired application.

In chapter 2, we present our work on the use of computer vision and machine learning algorithms to study human pluripotent stem cells in culture. There is a foundational need for quality control tools in stem laboratories engaged in basic research, regenerative therapies, and toxicological studies. These tools require automated methods for evaluating cell processes and quality during in vitro passaging, expansion, maintenance, and differentiation. By performing image segmentation and feature extraction on cell microscopy images, researchers can obtain useful metrics to help them analyze biological events and processes. Previously, biologists were forced to perform such studies manually by drawing ROIs by hand. Such tasks were very time consuming and prone to human error or bias. With video bioinformatics, the process can be reduced to a matter of minutes and reproducibility is greatly improved. Once various features

have been extracted, they may be used to automatically classify images or videos using machine learning classifiers. By providing a supervised learning algorithm with sufficient ground truth data, a model is created to automatically classify new specimens.

Chapter 2 introduces Stem Cell Quality Control (StemCellQC), our open source software which can automatically segment phase contrast videos of stem cell colonies using various segmentation methods. Each colony in a frame is tracked across subsequent frames. StemCellQC can merge tracks to account for merging cell colonies and can create new tracks for daughter colonies that split from a larger colony. After segmentation, an array of morphological and dynamic features is extracted. Each feature or a combination features can be plotted against time to provide researchers additional tools to analyze their data. Lastly, StemCellQC apply the extracted features to supervised learning classifiers to automatically label microscopy videos. While StemCellQC is able to classify many kinds of classes, chapter 2 specifically uses StemCellQC to study the health of stem cells. In addition, StemCellQC can be used to predict the class at 48 hours using only features from the first 12 hours.

In chapter 3, we use image processing and regression algorithms to produce live 3D phase contrast videos. 3D reconstruction methods can be very useful for visualizing dynamic cell morphology and biological processes related to their internal structures. 3D microscopy is traditionally performed with fluorescent imaging which may alter the underlying biological processes and requires cells to be fixed and killed. Development of 3D phase contrast methods is important to bypass these deficiencies as phase contrast microscopy is noninvasive. Previously, there were no methods to reconstruct phase

contrast images in 3D without specialized equipment. Chapter 3 presents Phase Contrast Regression 4D (PhaseR4D), a novel automated 3D reconstruction software for rendering phase contrast microscopy images in 3D over time. Using variance information from z-stack phase contrast and fluorescent images, PhaseR4D trains supervised regression algorithms to produce a depth map. The regressed depth map and original images are then used to reconstruct a 3D phase contrast volume. Once trained, PhaseR4D no longer needs fluorescence images and can perform 3D reconstructions with only phase contrast images. Another major significance of PhaseR4D is that once trained, live time-lapse 3D reconstructions can be used to study 3D features over time, which has never been done before. Also, extracted 3D phase contrast features are used to train supervised machine learning classifiers to automatically and accurately classify biological conditions.

In chapter 4, we study the interaction of biological proteins and cellular structures using video bioinformatics. Cofilin and other Actin-regulating proteins are essential in regulating the shape of dendritic spines, which are sites of neuronal communications in the brain. The analysis of cofilin motility in dendritic spines using fluorescence video-microscopy may allow for the discovery of its effects on synaptic functions. To date, the flow of cofilin has not been analyzed by automatic means. Chapter presents Dendrite Protein Analysis (DendritePA), a novel automated pattern recognition software to analyze protein trafficking in neurons. Using spatiotemporal information present in multichannel fluorescence videos, DendritePA generates a temporal maximum intensity project that enhances the signal-to-noise ratio of important biological structures, segments and tracks dendritic spines, estimates the density of proteins in spines, and

analyzes the flux of proteins through the dendrite/spine boundary. The motion of a dendritic spine is used to generate spine energy images, which are used to automatically classify the shape of common dendritic spines such as stubby, mushroom, or thin. By tracking dendritic spines over time and using their intensity profiles, the system can analyze the flux patterns of cofilin and other fluorescently stained proteins.

Chapter 2 StemCellQC: Feature Analysis and Classification

A. Introduction

Human pluripotent stem cells (hPSC) have enormous potential for enhancing our understanding of human prenatal development, modeling diseases-in-a-dish, treating patients with degenerative diseases, and evaluating the effects of drugs and environmental chemicals on cells that model human embryos and fetuses [6]–[8]. In each of these applications, there is a foundational unmet need for technology to non-invasively monitor the quality of hPSC during passaging, expansion, growth, experimentation, and differentiation [9], [10]. Ideally such tools should be rapid, non-invasive, resource saving, and non-biased. Video bioinformatics, which involves mining data from video images using algorithms that speed analysis and eliminate human bias, offers a solution to this problem and can be used to produce high quality software for stem cell applications [5], [11]–[17].

Prior applications of video bioinformatics tools have successfully identified pluripotent stem cell colonies based on colony morphology [18], thereby speeding induced pluripotent stem cell (iPSC) derivation and reducing cost. Another study applied image processing software to fluorescent videos to identify iPSC after reprogramming [19], and a video bioinformatics method was developed to identify in vitro fertilized human embryos that will progress to blastocysts by 2 days after fertilization [20]. A recent report used phase-contrast video segmentation to generate lineage trees of neural stem cells using cell location, shape, movement, and size [21]. We previously developed

in-house video segmentation tools to analyze single hESC and small colonies [12]–[15]. In a toxicological application of video bioinformatics using CL-Quant software [22], cigarette smoke treatment altered hESC colony growth (area) and health [11]. While the above studies looked at single endpoints, multiple features related to cellular processes and health can be extracted from video data thereby enhancing the depth of analysis and providing data on the kinetics of each endpoint. However, no software currently exists for automatic detection of pluripotent stem cell processes and quality in culture.

It is highly desirable to be able to multiplex multiple endpoints from a single experiment. The purpose of this study was to develop a high-content profiling software platform, StemCellQC, to automatically identify cell processes affected by culture/treatment and to classify the health of individual hESC colonies based on features extracted from phase contrast microscope video data. The method automatically segments the input colonies (non-labeled phase contrast images), extracts relevant novel features for each colony, utilizes the changes in features over time to identify cell processes that are affected by treatment, and statistically classifies healthy and unhealthy/dying colonies. StemCellQC's feature analysis and classification system provide an effective method to evaluate pluripotent stem cell colony processes and quality before use in experiments or clinical applications. Because hESC model the epiblast cells of embryos [23], which if harmed can lead to embryonic death or development of congenital defect(s), StemCellQC has the potential to be used as a novel technology to identify toxicants or drugs that could affect cellular processes in young embryos.

B. Materials and Methods

1. Overall Design of the System

The overall design of the system will be discussed in Results (S1 Fig). Each component is presented in the following Methods section.

2. Culture and Collection of hESC Videos

H9 hESC, purchased from WiCell Stem Cell Institute (Madison, Wisconsin), were maintained on Matrigel coated 6-well plates in mTeSR complete medium (Stem Cell Technologies, Vancouver, Canada) in a 37°C incubator with 90% humidity and 5% CO₂ [11], [24]. When cultures reached 80–85% confluency, they were detached using Accutase (eBiosciences, San Diego, CA) for 1 minute and used in experiments.

For live cell imaging in the BioStation CT (Nikon Instruments, Melville NY), hESC colonies were plated at 25–30% confluency and allowed to attach for 24 hours. To create groups of colonies that were healthy, unhealthy and dying, hESC were treated with Marlboro Red cigarette smoke solutions as described previously [11], [25]. Sidestream smoke (SS) was used at a dose of 0.1 puff equivalent (PE), where 1 PE is the amount of smoke that dissolves in 1 ml of medium in 60 sec. The 0.1PE concentration of SS smoke has an estimated concentration of nicotine of 0.2 µg/ml [26], [27], which is within the estimated tissue range of passive smokers [25].

All imaging was done using a 10x phase contrast objective in the BioStation CT using automatic Z-focus; cells were not stained, labeled, or genetically modified thereby

permitting non-invasive analysis of cells. StemCellQC was tested on different magnifications (4x and 20x) and performs well. The dataset for feature analysis was made up of 34 videos of individual hESC colonies. 23 colonies were treated for 48 hours with sidestream cigarette smoke, while 11 control colonies were incubated in culture medium only.

3. Development and Use of StemCellQC Software

StemCellQC was written and developed with MATLAB 2015a programming environment. The MATLAB source code, a stand-alone executable version of this algorithm, and supplied test data are available online at <http://vislab.ucr.edu/SOFTWARE/software.php>. Scqc_multi.m is the main program of the code and requires the following MATLAB toolboxes: Statistics and Machine Learning, Bioinformatics, System Identification, Image Processing, and Model-Based Calibration. The standalone alone executable requires the installation of the 64-bit version of MATLAB Runtime R2015a (8.5) available at <http://www.mathworks.com/products/compiler/mcr/>.

4. Categorization of hESC as Healthy, Unhealthy or Dying

Before the StemCellQC software is run, a training dataset was collected. Categorization of colonies as healthy, unhealthy, or dying by the end of 48 hours was provided by experts in culturing hESC and was based on biological observations listed in a decision tree (S2 Fig). Categorization was used to validate the program's predictions of colony health.

5. Video Segmentation

An edge-based method and a region-based method were used to segment colonies in video images (frames). The edge-based method convolves the Sobel edge operator with the image to produce a gradient magnitude image and a gradient direction image. All gradient magnitudes below a calculated threshold were ignored and the rest were used as edges. This calculated threshold was automatically computed by the “edge” function in the Image Processing Toolbox from MATLAB. The edges in the image were dilated using two line structuring elements (vertical and horizontal) of three pixel lengths to merge connected regions of the colony. Connected components were then filled and smoothed with image erosion using a diamond structuring element of one pixel radius to produce the segmentations. Segmented objects that are smaller than a user specified threshold, 3000 pixels in our case, were removed [28].

The Otsu’s region-based method for segmentation [29], [30], which was used to compute the solidity feature, is the ratio of the colony area divided by the area of the convex hull. The convex hull can be visualized as the shape enclosed by a rubber band stretched around a region-of-interest (ROI) [31]. Otsu’s method produces a slightly larger segmented boundary which is smoother than the edge-based method. The main purpose of using solidity was to detect dead cells that were extruded from the stem cell colonies. Edge-based segmentation provides tight edge boundaries, which does not include the dead cells that are in the process of being expelled from the colony; whereas, Otsu’s “larger” segmentation includes the dead cells. The concave regions of the segmentation that are produced by the dead cells affect the sensitivity of solidity. Therefore, solidity

changes (primarily due to dead cells) are more pronounced when using Otsu's method. This larger boundary results in a larger convex hull which increases the sensitivity of the solidity feature. This allowed for better distinction between the peaks and valleys in the solidity plot.

For the region-based method, initially frames were smoothed to remove a small amount of noise using a 3x3 Gaussian filter. Next, Otsu's thresholding-based method was used to separate the pixels into the background and foreground by finding the optimal threshold for segmenting an image [29], [30]. Connected components were then found in the binary image. A morphological open operation (used to open gaps between loosely connected objects) was performed on the binary image to disconnect loosely connected pixels in the foreground. This is carried out by first eroding an object of interest (a connected component) and then dilating the output with a structuring element. To disconnect the objects, a circle with a radius of 12 pixels worked the best, and it was kept fixed for all the experiments. For both erosion and dilation, every pixel in the object was individually probed by the structuring element. The end result was a set of filtered connected components. Any holes in this region were filled, and the final ROIs were used to extract features.

6. Validation of Segmentation

The accuracy of segmentation was determined by manually segmenting hESC colonies using ImageJ and comparing the area and perimeter values to those obtained from the automatic segmentation (S3 Fig).

7. Feature Extraction

Features based on appearance, morphology, and dynamics were extracted from segmented colonies. Dynamic features were obtained from morphological features by computing the rate of change of a feature over time. For example, to segment the protrusions extending from hESC, the main body of the colony (obtained using a morphological open algorithmic operation) was subtracted from the total colony segmentation. Also, the bright-to-total area ratio is the number of bright pixels divided by the area. Bright pixels were found by first computing a histogram of the intensities in the segmented colony to acquire the mean (μ) and standard deviation (σ). Then, a range of intensities (lower bound threshold = $\mu + 3 \sigma$, upper bound threshold = $\mu + 6 \sigma$) that best described dead cells in colonies were taken as the bright pixels. A complete list of features and their definitions is provided in S9 Fig.

8. Identification of Key Features

Both biologically-based feature selection and statistical-based feature selection were used to identify those features that provided information on affected cellular processes and to distinguish healthy, unhealthy, and dying colonies. In addition, all the features were exhaustively applied to the classifier.

For biologically-based feature selection, plot observations over time can be interpreted by the user. A subset of features are shown as graphical plots in Figs 2.1–2.4. From the feature plots, the user can observe the non-overlapping standard error of the means (SEMs) to get a good indication of groups that are significantly different from each other. For a more rigorous statistical analysis, two-way ANOVAs with the

Bonferroni post-test were performed to identify those features that were significantly different in the plotted data. For the solidity feature, a one-tailed independent samples t-test was used to determine if significant differences existed between means of healthy/unhealthy versus /dying colonies at 12 hours.

Features Related to Colony Growth

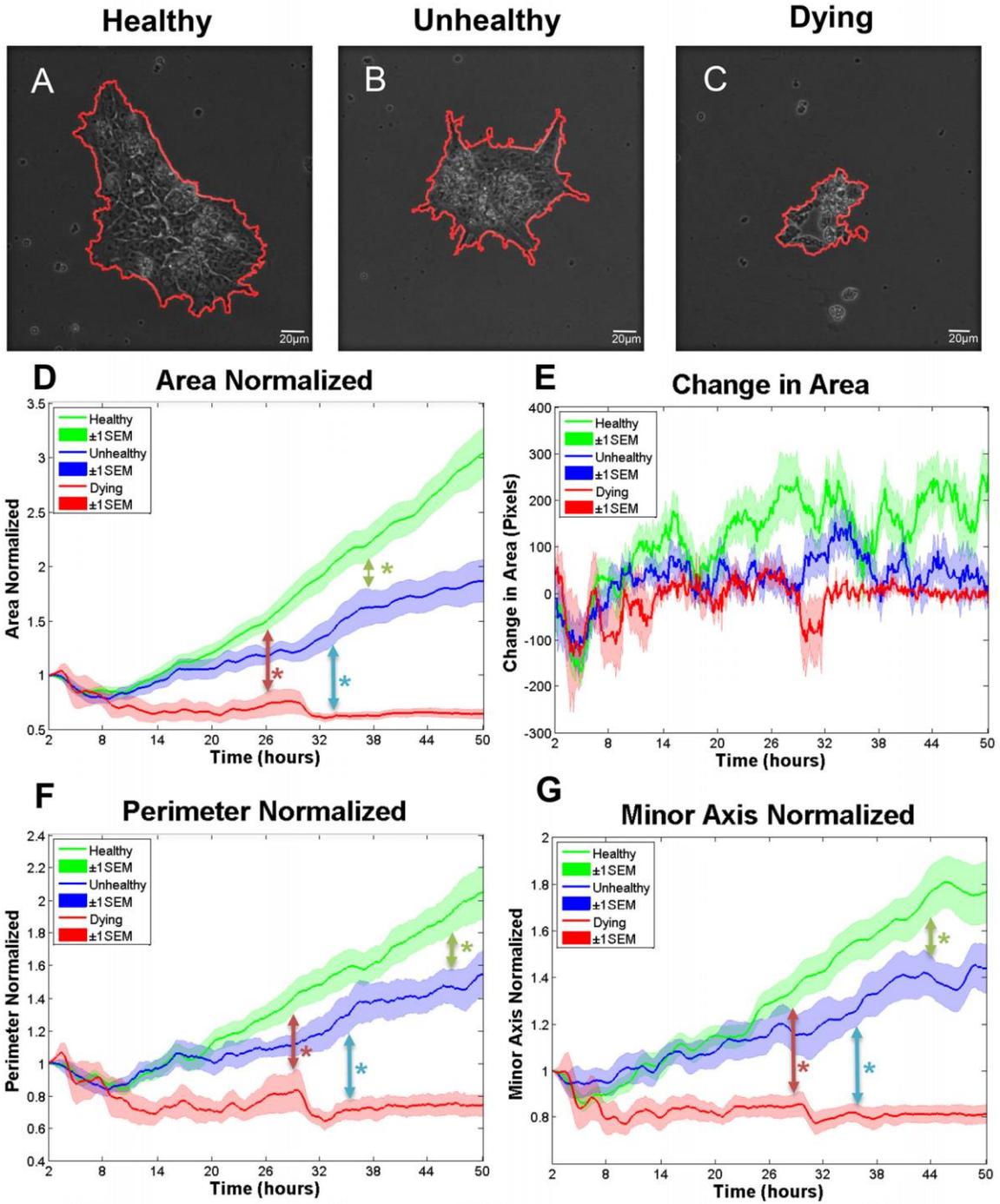


Fig 2.1. Features related to hESC colony growth. The outline of segmentation for a healthy (A), unhealthy (B), and a dying colony (C) at the last recorded frame. (D) Area normalized to the first time point for colonies that were healthy, unhealthy, and dying. Colonies first became significantly different by 2-way ANOVA at 37.6 hours for healthy vs unhealthy (green arrow), at 33.5 hours for unhealthy versus dying (blue arrow), and at 26.2 hours for healthy versus dying colonies (red arrow). (E) Change in area over time showing second contraction of dying colonies at 30–32 hours. (F) Perimeter over time normalized to the first time point for colonies that were healthy, unhealthy, and dying. Colonies first became significantly different by 2-way ANOVA at 46.6 hours for healthy versus unhealthy groups (green arrow), at 35.4 hours for unhealthy versus dying groups (blue arrow), and at 28.9 hours for healthy versus dying groups (red arrow). (G) Minor axis normalized to the first time point for colonies that were healthy, unhealthy, and dying. Colonies first became significantly different by 2-way ANOVA at 44.3 hour for healthy versus unhealthy groups (green arrow), at 36.5 hour for unhealthy versus dying groups (blue arrow), and at 28.9 hour for healthy versus dying group (red arrow). Number of colonies per group = 16 healthy, 12 unhealthy, 6 dying. Data are plotted as means \pm SEM for each group. Arrows indicate first values that differed significantly from the control by 2-way ANOVA (* = $P < 0.05$).

Protrusion-Related Features

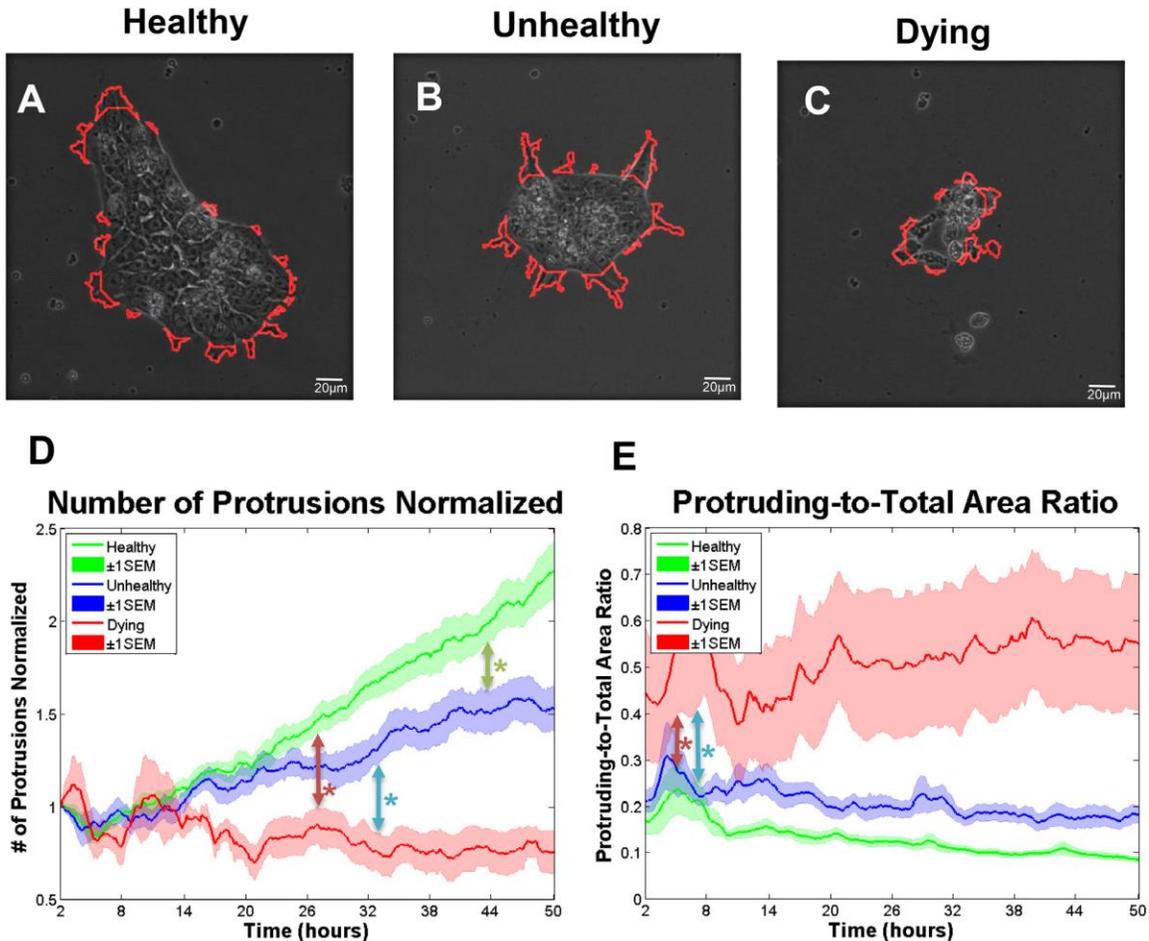


Fig 2.2. Surface protrusions on colonies can be used to study cell morphology and growth. (A) Segmentation of protrusions (red outline) for a healthy colony (A), unhealthy colony (B), and dying colony (C) at the last recorded time frame. (D) Number of protrusions over time normalized to the initial time point for healthy, unhealthy, and dying colonies. Colonies first became significantly different by 2-way ANOVA at 43 hours for healthy versus unhealthy groups (green arrow), at 33.3 hours for unhealthy versus dying groups (blue arrow), and at 27.1 hours for healthy versus dying group (red arrow). (E) Protruding-to-total area ratio for healthy, unhealthy, and dying colonies. Colonies first became significantly different by 2-way ANOVA at 6.8 hours for unhealthy versus dying groups (blue arrow), and at 4.8 hours for healthy versus dying groups (red arrow). Number of colonies per group = 16 healthy, 12 unhealthy, 6 dying. Data are plotted as means \pm SEM for each group. Arrows indicate first values that differed significant from the control by 2-way ANOVA (* = $P < 0.05$).

Motility-Related Features

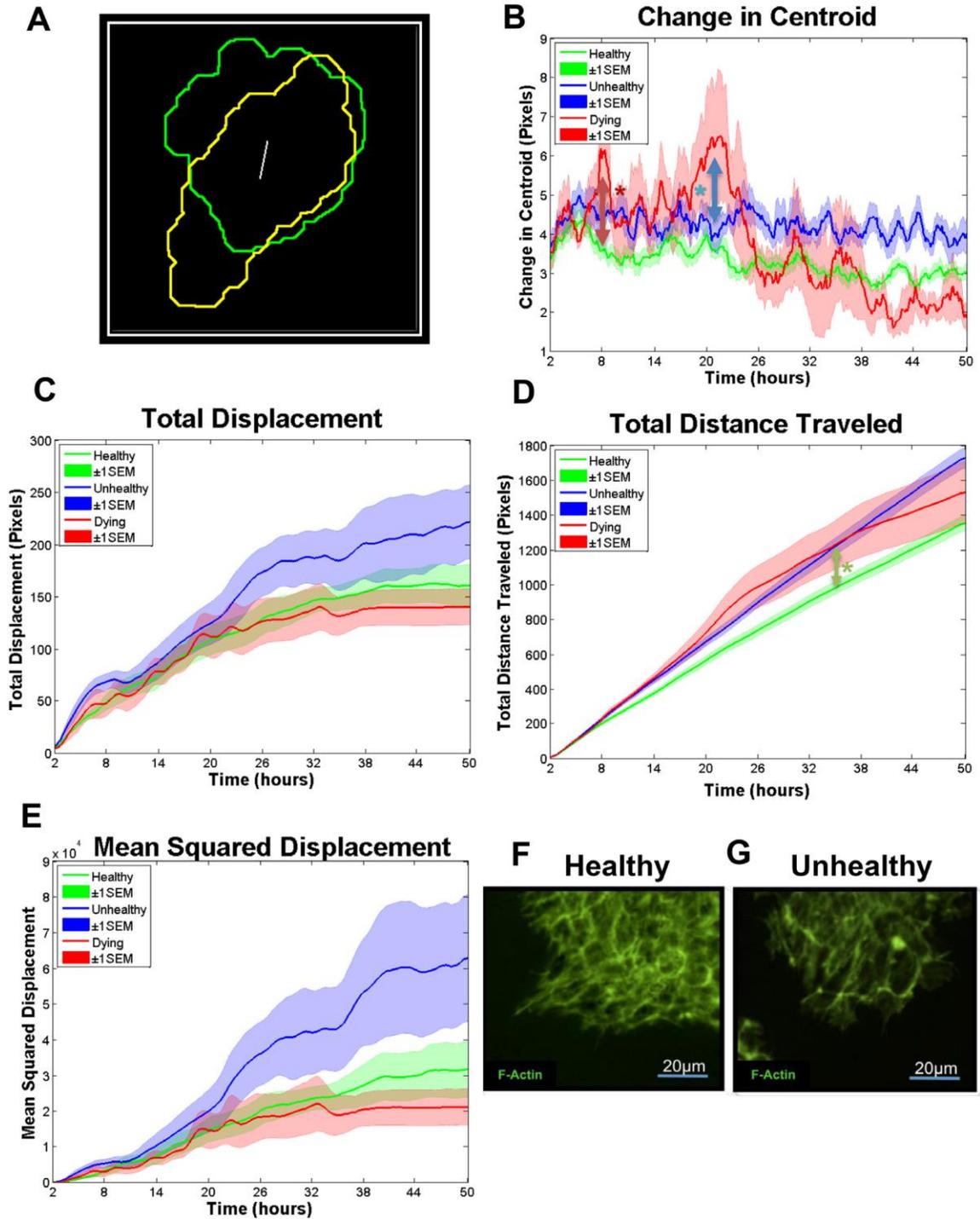


Fig 2.3. Features related to hESC colony motility. Extracted contour of a healthy colony at 16 hours (yellow line) and 24 (green line) hrs. The distance between the centroids is indicated by the white line. (B) Change in centroid over time for healthy, unhealthy, and dying colonies. Colonies first became significantly different by 2-way ANOVA at 7.7 hours for healthy versus dying groups (red arrow), and at 20.6 hours for unhealthy versus dying groups (blue arrow). (C) The total displacement for healthy, unhealthy, and dying colonies. (D) The total distance traveled for healthy, unhealthy, and dying colonies. Colonies first became significantly different by 2-way ANOVA at 34.7 hours for healthy versus unhealthy groups (green arrow). (E) The mean squared displacement (MSD) for healthy, unhealthy, and dying colonies. (F-G) Localization of actin microfilaments in a healthy (F) and an unhealthy (G) colony which had fewer microfilaments than the untreated control. Number of colonies per group = 16 healthy, 12 unhealthy, 6 dying. Data are plotted as means \pm SEM for each group. Arrows indicate first values that differed significant from the control by 2-way ANOVA (* = $P < 0.05$).

Apoptosis-Related Features

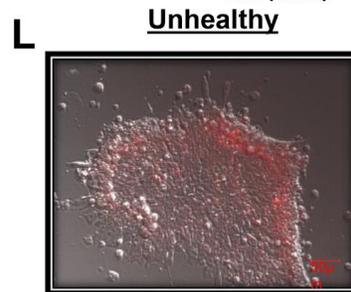
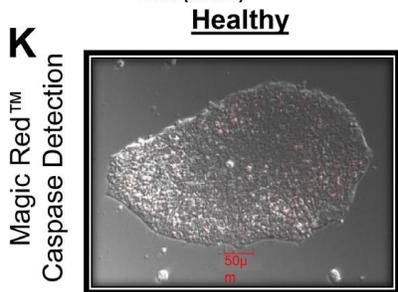
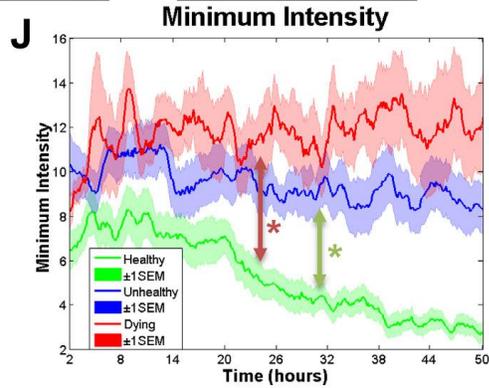
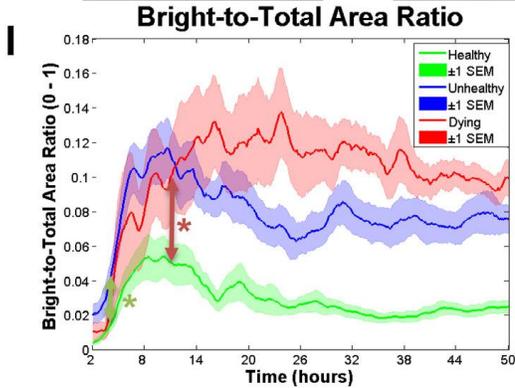
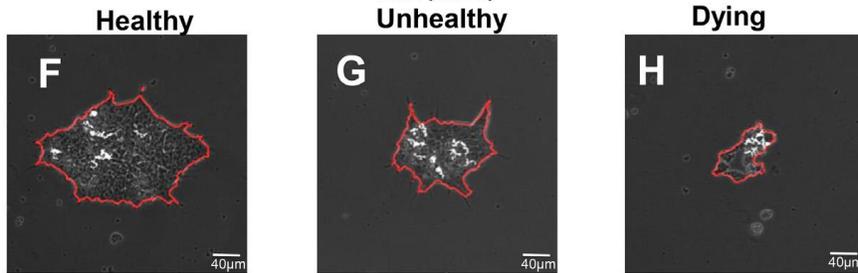
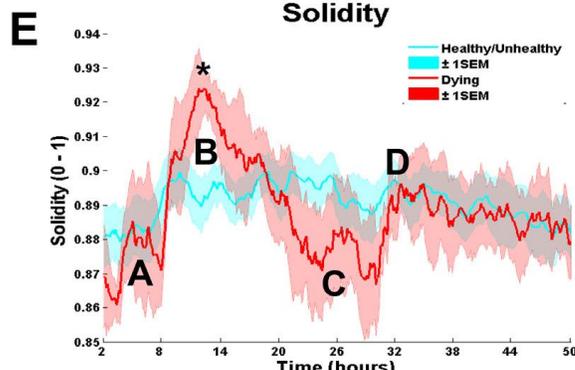
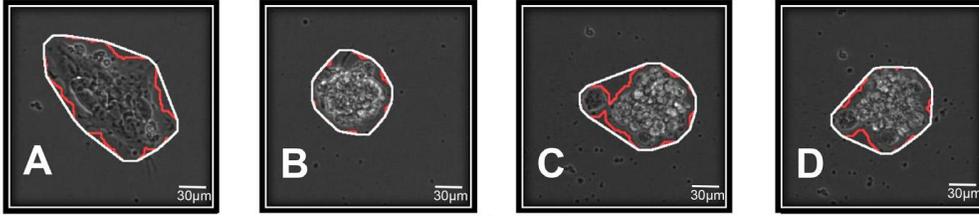


Fig 2.4. Features related to cell death. (A-D) Frames representing the beginning of the video (A), the highest and lowest solidity values respectively (B and C), and the time of death of dying colonies (D). E) Solidity values over time for healthy/unhealthy (blue) versus dying colonies (red). Colonies that eventually died are distinguished by a large peak in solidity between 8–24 hours. A one-tailed independent sample t-test at 12 hours revealed that the two groups were significantly different ($P = 0.0285$). (F-H) White regions on top of hESC colonies (outlined in red) represent dead cells, shown at the end of recording for a healthy colony (F), unhealthy colony (G), and dying colony (H). (I) Bright-to-total area ratio over time for healthy, unhealthy, and dying colonies. Colonies first became significantly different by 2-way ANOVA at 4 hours for healthy versus unhealthy groups (green arrow), and at 11.5 hour for the healthy versus dying groups (red arrow). (J) Minimum intensity values for healthy, unhealthy, and dying colonies. Colonies first became significantly different by 2-way ANOVA at 31.5 hour healthy versus unhealthy groups (green arrow), and at 24 hours for healthy versus dying groups (red arrow). (K-L) A healthy (K) and an unhealthy (L) colony incubated with Magic Red to identify activated caspases 3&7. Number of colonies per group = 16 healthy, 12 unhealthy, 6 dying. Data are plotted as means \pm SEM for each group. Arrows indicate first values that differed significant from the control by 2-way ANOVA (* = $P < 0.05$).

Statistical-based methods are useful in cases where the graphs for features may not reveal obvious effects, and they are good starting points to identify combinations or subsets of useful features. Filter methods which select variables regardless of the classification model are preferable for StemCellQC because of the use of multiple classifiers. 11 feature selection algorithms (10 methods from the Feature Selection @ Arizona State University toolbox [32]) and quadratic programming feature selection [33] were run on our dataset. These methods include Correlation-based Feature Selection (CFS) [34], Chi Square (Chi2) [35], Fast Correlation-based Filter (FCBF) [36], Fisher 9 [37], Gini Index 16 [38], InfoGain 6 [39], Sparse Multinomial Logistic Regression (SBMLR 3) [40], t-test [41], Kruskal Wallis [42], and Minimal-Redundancy-Maximal-Relevance [43]. The inputs for the feature selection algorithms are the average slope of each feature. The slope (incremental difference) is computed for each pair of adjacent frames for individual features. Next, the mean of these slopes is calculated for all 24 features. These features individually or in groups can be used to train the classifier.

9. Correlation of Key Features to Cellular Processes

Key features were interpreted to identify cellular processes such as growth, motility, and apoptosis, which differed in the healthy, unhealthy and dying groups. Also, two-feature plots were constructed to detect correlations between features and identify temporal patterns over time (Fig 2.5, S1 and S2 Videos).

Multi-Feature Analysis and User-Derived Features

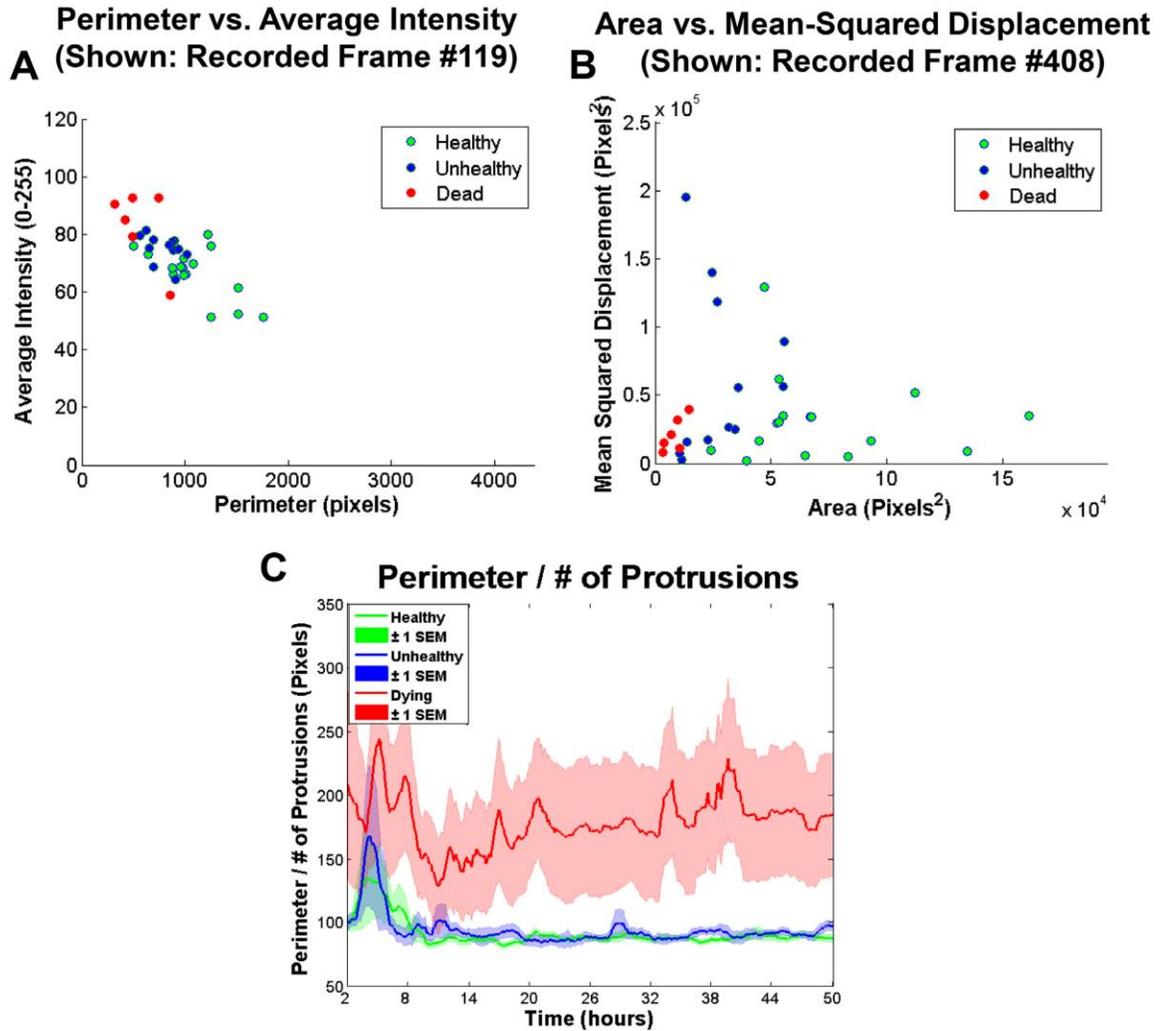


Fig 2.5. Two-Feature Plot Analysis. (A) Average intensity compared to perimeter running plot shown at approximately at 16 hours of incubation for all individual healthy (green), unhealthy (blue), and dying (red) hESC colonies. (B) Mean-squared displacement compared to area running plot shown at approximately 16 hours for all individual healthy (green), unhealthy (blue), and dying (red) hESC colonies. (C) User derived equation (perimeter divided by number of protrusions) plotted for healthy, unhealthy, and dying colonies.

10. Classification as Healthy or Unhealthy/Dying

To automatically classify the dataset, all features measurements were first normalized with maximum-minimum normalization. 410 frames (collected over a 48 hour period) multiplied by 24 features results in a 9840 feature space. Therefore, in order to reduce the number of dimensions, the input value to the classifier was the mean slope of the data. Classes used by the classifier were healthy and unhealthy (the unhealthy group included dying colonies which were not classified separately since this group contained only 6 colonies).

To test the ability of the features to distinguish healthy versus unhealthy/dying colonies, several biologically selected features and additional features selected by 11 statistical methods were used to train three classifiers: (1) support vector machines (SVM), (2) K-nearest neighbor (KNN), and (3) naïve Bayes [37]. SVM uses the training data to create a boundary in multi-dimensional space, which can be used to classify future data samples. KNN takes a test sample and compares it to the K-nearest training samples in a multi-dimensional space. The KNN algorithm was used with $k = 3$ (the 3 closest neighbors to the sample). A majority vote is taken by these neighbors and used as a label for the test sample. Naïve Bayes is a probabilistic classifier based on Bayes' theorem that uses strong assumptions that features are independent from one another. An exhaustive test using all possible combinations of features was performed to determine the best classification results using combination of features.

A summary of the classification results using single features, combinations of features, and statistically determined features are shown in Tables 2.1–2.4 in the Results section.

48 Hours	*Classification Techniques		
	SVM	K-NN, k = 3	Naive Bayes
Single Features			
1) Area	94.12 ± 0.00	94.12 ± 0.00	94.00 ± 0.91
2) Number of Protrusions	90.71 ± 1.35	96.06 ± 1.32	91.24 ± 0.65
3) Total Distance Travelled	84.24 ± 1.20	74.06 ± 1.48	84.88 ± 1.01
Combination of Features			
1) Area, Orientation, Num. of Protrusions	94.12 ± 0.00	94.71 ± 1.15	94.12 ± 0.00
2) Num. of Protrusions, Min. Intensity	97.06 ± 0.00	97.06 ± 0.00	96.47 ± 1.15
3) Major Axis, Minor Axis, Change in Centroid	93.53 ± 1.57	92.94 ± 1.62	90.00 ± 1.27
Feature Selection Methods			
**CFS	91.76 ± 1.32	96.47 ± 1.32	91.76 ± 1.32
***Chi Square	91.76 ± 0.00	91.76 ± 0.00	95.29 ± 1.61
****QPFS	91.76 ± 1.32	94.12 ± 3.60	91.76 ± 2.46

*Classification of colonies as healthy or unhealthy using three different classification techniques: SVM, KNN, and Naive Bayes.
**CFS selected the following features: Area, Number of Protrusions, and Change in Area.
***ChiSquare selected the following features: Area, Number of Protrusions, and Major Axis Length
****Quadratic Programming Feature Selection selected the following features: Total Distance Travelled, Major Axis Length, Minimum Radius.

doi:10.1371/journal.pone.0148642.t001

Table 2.1. Classification Results Using 48 Hours of Video.

36 Hours	*Classification Techniques		
	SVM	K-NN, k = 3	Naive Bayes
Single Features			
1) Area	85.18 ± 1.25	88.00 ± 1.49	85.18 ± 1.49
2) Number of Protrusions	81.94 ± 1.10	73.65 ± 1.72	81.00 ± 1.34
3) Bright Area Ratio	81.76 ± 3.22	85.29 ± 0.00	76.47 ± 2.08
Combination of Features			
1) Area, Min. Radius, Num. of Protrusions, Change in Area, Change in Perimeter	96.47 ± 1.32	94.71 ± 1.32	94.71 ± 1.32
2) Area, Min. Radius, Change in Area, Change in Perimeter	95.88 ± 1.61	92.94 ± 1.61	95.88 ± 1.61
3) Area, Avg. Radius, Change in Area, Change in Perimeter	95.29 ± 1.61	92.94 ± 1.61	95.88 ± 2.63
Feature Selection Methods			
**CFS	91.18 ± 2.08	88.24 ± 2.08	90.00 ± 1.61
***SMBLR	85.88 ± 1.32	91.76 ± 1.32	90.59 ± 1.32
****FCBF	86.47 ± 1.61	87.65 ± 1.32	92.94 ± 1.61

*Classification of colonies as healthy or unhealthy using three different classification techniques: SVM, KNN, and Naive Bayes.
**CFS selected the following features: Area, Perimeter, Minor Axis Length, Minimum Intensity, Bright Area Ratio, Number of Protrusions, Change in Perimeter, Total Distance Travelled.
***SMBLR selected the following features: Number of Protrusions, Change in Perimeter, Minor Axis Length, Bright Area Ratio, Total Distance Travelled.
****FCBF selected the following features: Area, Perimeter, Minimum Intensity, Bright Area Ratio, Change in Perimeter, total Distance Travelled.

doi:10.1371/journal.pone.0148642.t002

Table 2.2. Classification Results Using 36 Hours of Video.

24 Hours	*Classification Techniques		
	SVM	K-NN, k = 3	Naïve Bayes
Single Features			
1) Area	83.35 ± 1.49	74.94 ± 2.00	82.47 ± 1.37
2) Number of Protrusions	75.29 ± 1.85	67.41 ± 1.95	71.35 ± 1.51
3) Perimeter	75.29 ± 4.36	73.53 ± 2.08	75.29 ± 1.61
Combination of Features			
1) Area, Orientation, Num. of Protrusions, Change in Perimeter, Change in Centroid	83.53 ± 1.61	91.17 ± 0.00	88.82 ± 1.31
2) Area, Avg. Radius, Change in Perimeter, Change in Centroid	88.24 ± 2.08	83.53 ± 1.61	90.59 ± 1.32
3) Orientation, Bright Area Ratio, Change in Perimeter, Change in Centroid	87.65 ± 1.32	88.24 ± 2.94	84.12 ± 1.61
Feature Selection Methods			
*CFS	84.71 ± 1.32	72.35 ± 4.46	78.24 ± 1.61
**SBMLR	78.24 ± 5.73	69.42 ± 3.35	83.53 ± 3.35
***FCBF	84.71 ± 1.32	72.35 ± 4.46	78.24 ± 1.61

*Classification of colonies as healthy or unhealthy using three different classification techniques: SVM, KNN, and Naive Bayes.
**CFS selected the following features: Area, Minor Axis Length, Bright Area Ratio, Change in Perimeter, Total Distance Travelled.
***SBMLR selected the following features: Area, Orientation, Number of Protrusions, Total Distance Travelled.
****FCBF selected the following features: Area, Minor Axis Length, Bright Area Ratio, Change in Perimeter, Total Distance Travelled.

doi:10.1371/journal.pone.0148642.t003

Table 2.3. Classification Results Using 24 Hours of Video.

12 Hours	*Classification Techniques		
	SVM	K-NN, k = 3	Naïve Bayes
Single Features			
1) Bright Area Ratio	67.06 ± 1.32	54.12 ± 1.61	64.70 ± 2.94
2) Total Distance Travelled	71.76 ± 2.63	65.29 ± 2.46	61.18 ± 3.83
3) Change in Centroid	64.71 ± 0.00	57.06 ± 3.35	67.06 ± 2.46
Combination of Features			
1) Max. Radius, Ratio of Protrusion Area, Change in Area, Change in Centroid	74.12 ± 2.46	75.88 ± 1.31	79.41 ± 3.60
2) Eccentricity, Min. Radius, Ratio of Protrusion Area, Change in Perimeter, Change in Centroid	71.18 ± 3.83	78.24 ± 3.35	78.24 ± 1.61
3) Min. Radius, Max Intensity, Change in Centroid	71.18 ± 2.46	80.59 ± 2.63	73.53 ± 0.00
Feature Selection Methods			
*CFS	71.76 ± 2.63	65.29 ± 2.46	61.18 ± 3.83
**SBMLR	67.65 ± 2.94	77.06 ± 2.46	64.12 ± 3.22
***FCBF	71.76 ± 2.63	65.29 ± 2.46	61.18 ± 3.83

*Classification of colonies as healthy or unhealthy using three different classification techniques: SVM, KNN, and Naive Bayes.
**CFS selected the following features: Total Distance Travelled.
***SBMLR selected the following features: Change in Area, Total Distance Travelled.
****FCBF selected the following features: Total Distance Travelled.

doi:10.1371/journal.pone.0148642.t004

Table 2.4. Classification Results Using 12 Hours of Video.

11. Classification Validation

The classification experiments were run with 10-fold cross validation where the dataset was partitioned into 10 parts. The 34 videos gave six partitions containing three videos each and four partitions containing four videos each. A 10-fold partition of the

training data was used, which allowed for computation of a standard deviation of the results. One part was used as the test data once, while the other nine parts were used as training data. The partitions were randomized, and this process was repeated with 5 random permutations of the data. A percentage of correctly classified samples was calculated for each permutation by comparison to the manual labels. The classification results were then used to find the mean and standard deviation.

12. Molecular Validation

Apoptotic activity was detected using the Magic Red Caspases 3&7 Detection FLICA Kit (Immunochemistry Technologies, LLC, Bloomington, MN) as described previously [11]. Fluorescent staining of F-actin was performed using a phalloidin-Alexa Fluor 488 conjugate (Invitrogen, Carlsbad, CA) diluted 1:200 in 1% goat serum in phosphate buffered saline. hESC colonies in chamber slides were fixed using 4% paraformaldehyde for 10 minutes, incubated in blocking solution (3% goat serum in PBS) at room temperature for 1 hour, washed 5 times, and incubated in phalloidin-Alexa Fluor 488 for 1 hour at room temperature. Samples were mounted using Vectashield with DAPI (Vector Laboratories, Burlingame, CA) and imaged with a Nikon Eclipse Ti fluorescent microscope (Nikon, Melville, NY).

C. Results

1. Feature Analysis

Features were analyzed graphically to identify those that differed in the healthy, unhealthy, and dying groups. Sets of affected features were then grouped according to the biological processes they represented (morphology, growth, motility, death) (Figs 2.1–2.4, S4 Fig). The classifiers were run with 48, 36, 24, and 12 hours of video to show their effectiveness at different time points. For all durations, all 24 features were run singularly through the classifiers and the ones with the highest accuracy are shown in Tables 2.1–2.4 (Single Features). Additionally, exhaustive searches for combinations of up to 5 features were run to identify the most accurate results (Table 2.1 Combination of Features). Lastly, the best results from the 11 existing feature selection algorithms are also shown in Tables 2.1–2.4 (Feature Selection Methods).

2. Features Related to Growth as Biomarkers of hESC Health

Extracted features related to colony growth (area, perimeter, minor axis, protrusions) were evaluated in healthy, unhealthy, and dying colonies. Area (the total number of pixels inside a segmented colony; S5Fig) differed in healthy, unhealthy and dying colonies (Fig 2.1A-2.1E). When area was normalized to the initial time point to account for variability in the starting size, all groups displayed an initial contraction which decreased area and lasted about 8 hours (Fig 2.1D). Contraction was likely caused

by changes in temperature/CO₂ during transfer to the BioStation. After contraction, healthy colonies displayed a steady increase in area until the end of recording, while unhealthy colonies grew at a slower and variable rate. Both groups followed a similar trend up to 16 hours, after which healthy and unhealthy growth rates deviated and become distinguishable by about 24 hours. Dying colonies could be distinguished from the healthy and unhealthy groups by about 10 hours when growth rates for the dying group clearly diverged. At 30 hours, dying colonies underwent a second contraction leading to a sharp decrease in area (Fig 2.1D and 2.1E). These colonies were interpreted to be dead based on this pronounced decrease in size and shedding of dead cells.

Perimeter, defined as the number of pixels constituting the colony periphery (red outline in Fig 2.1A-2.1C and 2.1F; S5 Fig), provided additional information about colony growth. There was divergence in the perimeters of healthy and unhealthy colonies at approximately 25 hours (Fig 2.1F), after which the rate of change in perimeter for unhealthy colonies slowed until about 33 hours when it underwent a growth spurt that lasted 3 hours. The dying colonies diverged from the other two groups at 8 hours and had an abrupt decrease in size at 30 hours, as was seen with area.

Minor axis (smaller axis of an ellipse fitted to a colony) (S6 Fig) was affected in unhealthy/dying colonies (Fig 2.1G). After 23 hours, healthy colonies showed a steep increase in minor axis, suggesting that once healthy colonies reach a critical size, they have a less-elongated morphology.

Protrusions are dynamic cell processes that extend off colonies and take a variety of shapes (Fig 2.2A-2.2C; S6 Fig). They allow colonies to attach, spread, and migrate

[44], [45]. The number of protrusions increased on healthy and unhealthy colonies and decreased on dying colonies during incubation (Fig 2.2D). The protruding-to-total area ratio, which is defined as the ratio of protrusion area divided by total colony area, had an inverse relationship with colony growth. Protrusion area decreased gradually in healthy and unhealthy groups but increased slightly in the dying group (Fig 2.2E).

3. Colony Motility

The change in centroid feature allowed tracking of stem cell colony movement. This feature is determined by finding the centroid of each colony and calculating the distance between two successive frames (S7 Fig). Outlines of a hESC colony at two times and the change in centroids are shown in Fig2.4A. Change in centroid oscillations were smaller in the healthy and unhealthy groups than in the dying group (Fig 2.3B). The unhealthy and healthy groups were similar in the magnitude of their oscillations, but overall motility was less in the healthy colonies, probably because the larger sized healthy colonies required more energy and coordination for directed movement. After a certain area was reached, the center of the healthy colonies moved very little as the colony continued to expand. It is also possible that smoke stimulated motility in unhealthy hESC to facilitate escape from exposure. The dying colonies displayed erratic motility and showed a significant decrease in movement after 20 hours as they were approaching death. Movements detected after death (30 hours) are due to slight segmentation differences between frames.

Total displacement detected how far a colony moved from its original starting point (Fig 2.3C; S7 Fig), while total distance traveled is the sum of the entire trajectory of movement (Fig 2.3D; S7 Fig). These features revealed information on the pattern of travel. Dying colonies traveled more up to 30 hours (when they died) than the other two groups, but their displacement was low indicating that they moved erratically near their original starting point. Unhealthy colonies moved further from their point of origin and travelled a longer total distance than healthy colonies. Both the healthy and unhealthy colonies displayed remarkably little variance in total distance travelled (Fig 2.3D).

The mean squared displacement (MSD) feature measures Brownian motion [46] and can be used to study cellular migration [47]. MSD is defined by the equation: $MSD(t) = ([x(t+t0) - x(t0)]^2 + [y(t+t0) - y(t0)]^2)$, where MSD (t) can be approximated as $\sim t^{\beta(t)}$. The logarithmic derivative exponent β can be used to determine the particular mode of motility, with $\beta > 1$ indicating super-diffusive movement, a form of diffusion where the colonies occasionally undergo very long steps. $\beta < 1$ indicates sub-diffusive movement, defined as a tendency for the colonies not to diffuse due to trapping (inability to move). For Brownian motion, or a random walk, β is approximately 1. The MSD feature is robust because it uses the squared value of displacement, making it less sensitive to small fluctuations. The MSD plot shows a similar trajectory for all three groups up until about 11 hours (Fig 2.3E), after which the healthy colonies display Brownian motion ($\beta = 1.04$). For dying colonies, sub-diffusive motility ($\beta = 0.21$) was observed from 23–50 hours, consistent with their death after 30 hours. The unhealthy group demonstrated sub-

diffusive motility from 22–35 hours and 35–50 hours (β values = 0.76 and 0.52, respectively).

To investigate the molecular basis of the aforementioned effects on motility, F-actin was labeled with phalloidin-Alexa 488. Healthy colonies (Fig 2.3F) had a more robust actin cytoskeleton than unhealthy colonies (Fig 2.3G). Although F-actin was partially depolymerized by smoke treatment, there was sufficient functional F-actin in the treated colonies to allow colony movement. A decrease in F-actin may be linked to a decrease in the number of focal adhesions, which may facilitate motility in the unhealthy group [48], [49]. Other studies have reported the inverse correlation between cell motility and polymerization state of the actin cytoskeleton [50].

4. Solidity as Predictor of Apoptosis

As colonies became rounder or more convex, their solidity increased and approached 1. Fig 2.4A-2.4D show hESC colonies at different times with outlines of their segmentations (red lines) and convex hulls (white lines). Solidity, which measured convexity (Fig 2.4E; S8 Fig), identified colonies that were destined to die by 48 hours. Solidity for the healthy/unhealthy groups combined changed little for 48 hours (Fig 2.4E). These two groups were combined since solidity was a predictor of colony death, not health. In contrast, dying colonies had a significant spike in solidity at about 12 hours due to contraction and rounding of the colonies (B label in Fig 2.4E). This was followed by a drop that reached a minimum at 30 hours (C label in Fig 2.4E), when death occurred

and extrusion of dead cells caused the convex hull to be less circular. Graphs of solidity can be used to identify at 12 hours, colonies that will die by 48 hours.

5. Colony Brightness Identifies Dying Cells

As cells within a colony die, they are extruded to the top of the colony where their brightness increases. The white areas in Fig 2.4F are dead cells on a healthy colony at the end of incubation. Significantly more dead cells were present on the unhealthy (Fig 2.4G) and dying colonies (Fig 2.4H). To quantify dead cells on top of colonies, a bright-to-total area ratio feature was used. This feature measured the number of bright pixels in the colony as a ratio to the total area and is an indicator of cell death. All groups exhibited an increase in bright-to-total area ratio during the first 6 hours when the colonies contracted (Fig 2.4I), after which the bright-to-total area ratios of healthy and unhealthy colonies decreased and the ratio for the dead colonies increased up to 16 hours and stayed elevated.

To compare the progression of colony brightness over time, a minimum intensity feature (lowest pixel intensity in the colony) was monitored (Fig 2.4J). Throughout incubation, healthy colonies displayed a lower minimum intensity than the unhealthy and dying colonies. These data support the idea that the unhealthy and dying colonies failed to spread as well on Matrigel as healthy colonies.

To confirm cell death, colonies were labeled with Magic Red which detects activated caspases 3&7, biomarkers for apoptosis. As shown by the red staining in Fig

2.4K and 2.4L, unhealthy colonies exhibited more caspase 3&7 activity than the healthy colonies.

6. 3D Visualization of Features and Custom Features

To mine additional biological information such as correlation of features, StemCellQC can plot features against each other and play the plot as a video over time (S1 and S2 Videos). In Fig 2.5A, perimeter and average intensity, when plotted against each other, showed an inverse relationship (indicative of dead cells). In Fig 2.5B, area and the mean-squared displacement features were plotted against each other to highlight individual colonies with elevated MSD values (mainly colonies from the unhealthy group). This type of analysis can also reveal outlier colonies within a group. In addition, StemCellQC is able to plot mathematical equations using the original 24 features. In Fig 2.5C, a user derived equation, ratio of perimeter to the number of protrusions, is plotted. This plot displays an estimate for the average length of a protrusion for each class and shows that the protrusions on dying colonies are about twice as long as those on healthy/unhealthy colonies (Fig 2.5C).

7. Classification Results

The input values used by the classifiers were the mean slopes of each feature. The nine individual features found by user-interpreted feature selection were tested separately giving each feature a classification rate (Table 2.1). Area was the best individual feature

at predicting health with a 94% accuracy when using any classifier. By combining features that are not related to the same process, accuracy increased. When the number of protrusions and minimum intensity were combined, the system's ability to distinguish hESC colony health improved to 97% accuracy when using any classifier. Results for feature selection algorithms (CFS, ChiSquare and QPFS) were also shown. All three had at least 91% accuracy and CFS was 96.47% accurate with KNN.

The classifiers were also run with the first 36, 24, and 12 hours, which are shown in Tables 2.2–2.4. For 36 hours (Table 2.2), area was the best feature with 88% accuracy, and combination of features improved results to 96.47%. For 24 hours (Table 2.3), area was still the strongest feature with 83.35% accuracy and a combination improved results to 91.17% accuracy. For 12 hours (Table 2.4), however, total distance travelled is the strongest feature with 71.76% accuracy. It should be noted that for 12 hours most individual features performed at about 50–60% accuracy which is slightly better than chance, however when we combine features, we are able to improve classification to 80.59% accuracy. Judging colony health by eye after a mere 12 hours of time is biased and difficult, making an 80.59% classification rate very useful. These tables show that while a certain combination of features work best using the full 48 hours of time, another combination may produce a more accurate classification if less time is used. With shorter video duration, there is an increase in accuracy using a combination of features; whereas, with longer durations, a single strong feature (such as area) can be sufficient to get accurate results.

D. Discussion

StemCellQC is an innovative, cost effective, non-invasive software tool that utilizes bioinformatics to automatically monitor dynamic cell processes, cell morphology, and cell health during passaging, culture, expansion, maintenance, or experimental treatment of pluripotent stem cells. StemCellQC eliminates the need for labeling with dyes or fluorescent probes and eliminates tedious manual classification, which significantly decreases analysis time and classification errors due to observer bias.

Graphical plots of features provide quantifiable, real-time data on living hESC and are excellent analytical tools for comparing features across treatments and cell types. The plots can help users visualize trends or features that are not easily detectable by manual inspection. Cell process analysis is especially valuable in toxicological or drug studies as it provides insight into the mode of action of the treatment. For example, smoke treatment inhibited growth (area, perimeter, minor axis and protrusions), increased motility (change in centroid, total displacement, total distance traveled and MSD), and increased apoptosis (solidity and intensity features). Chemical treatments other than cigarette smoke may affect other features, and in such cases, other cell processes could be revealed by feature analysis. Multiplexing cellular process information (colony growth rate, motility, and apoptosis) increases the power of analysis, and in toxicological studies, this greatly increases the probability of detecting an effect if one exists.

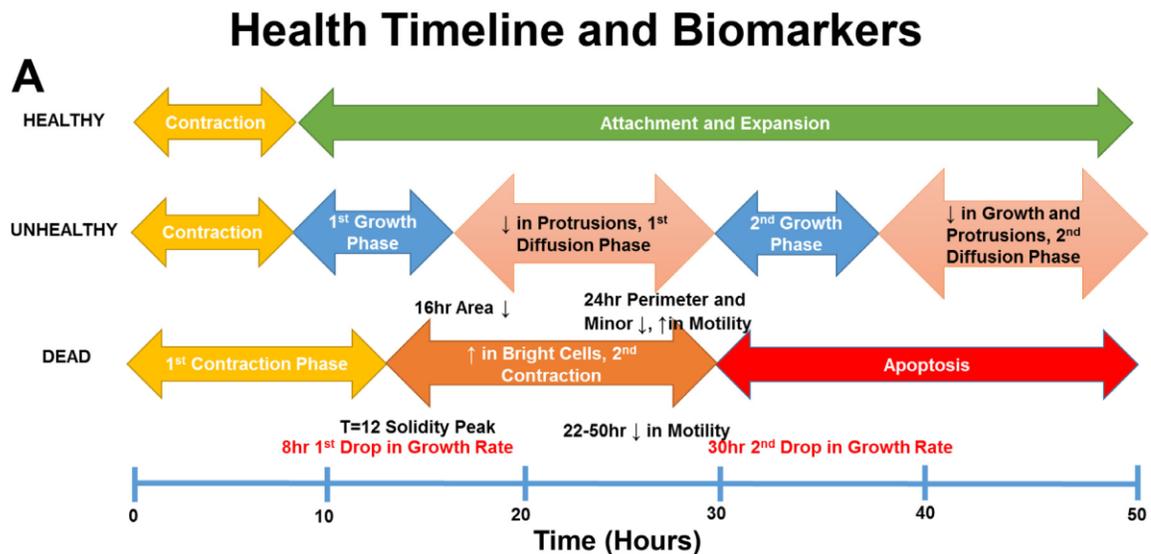
StemCellQC can plot user-derived equations of features (for example: perimeter/number of protrusions) for customized types of analysis. The software can also create videos of various features plotted against each other. These plots enable correlation

between features and help determine how biological processes are related over time. For example, an inverse relationship was found between colony size and colony brightness (dead cells on a colonies' surface).

Feature analysis, when combined with a classifier, enabled identification of healthy, unhealthy, and dying colonies. Area, which classified with 94% accuracy, was the strongest feature for predicting colony health. Changes in area are not always detected by human observation, especially when colonies do not die but experience stunted growth. In clinics or research laboratories, a decrease in growth rate may signal a problem with the culture or cell quality, and this would be rapidly detected in cultures monitored using StemCellQC. While smoke treatment slowed colony growth, factors that increase growth rate may be equally important and detectable by StemCellQC. For example, when chromosomal translocations occur in hESC, growth can be accelerated [51], and this would not be desirable in clinical or research labs. Combinations of features successfully increased the accuracy of classification of unhealthy/dying colonies to 97%. Depending on the rigor needed, change in area by itself will usually be sufficient to distinguish healthy from unhealthy/dying colonies.

By comparing changes in features over 48 hours, biomarkers that predict biological outcomes were found at early time points (Fig 2.6). For example, growth rate separated dying from healthy/unhealthy colonies by 16 hours and further separated healthy from unhealthy colonies by 26 hours (Fig 2.1D; Fig 2.6A and 2.6B). Similar distinctions can be made from graphs for other growth features (perimeter, minor axis, and number of protrusions) (Figs 2.1F and 2.1G and 2.2D and 2.2E). Change in centroid

was the strongest motility biomarker which cleanly separated healthy from dying colonies as early as 8 hours (Fig 2.3D). Solidity successfully separated dying colonies from healthy/unhealthy by 12 hours when used with Otsu's segmentation (Fig 2.4E), and bright-to-total area ratio separated all three groups from each other by 14 hours (Fig 2.4I). The biomarkers for dying colonies are powerful tools for monitoring apoptosis in living cultures without use of labels or probes, which themselves often produce unwanted effects.



B

Extracted Biological Information	Time (hr)	Biomarker
Bright-to-total area ratio elevated as early as 6 hour	6-11	Apoptosis
Protruding-to-total area ratio elevated as early as 7 hour	7	Retraction/Apoptosis
Change in centroid significantly peaks as early as 8 hour	8	Motility/Apoptosis
Mean-squared displacement β values of $\gg 1$	11	Super-diffusive Motility
Peak in solidity at 12 hour	12	Apoptosis
Elevated total distance traveled as early as 14 hour	14	Motility/Apoptosis
Area separates from healthy group at 16 hour	16	Growth
Minimum intensity value declines as early as 20 hour	20	Health
Number of protrusions separates from healthy group at 22 hour	22	Growth/Morphology
Minor axis shows a sudden increase at 23 hour	23	Growth
Minimum intensity value elevated as early as 24 hour	24	Apoptosis
Change in centroid drops to consistent near-zero values	26	Motility/Apoptosis
Sudden drop in change in Area at 30 hour	30	Apoptosis
Mean-squared displacement β value of $\ll 1$	30	Sub-diffusive Motility

Fig 2.6. hESC Health Timeline and Biomarkers. (A) Changes in several feature values and biological events during 48 hours of incubation for healthy, unhealthy and control groups. This type of plot can be used to compare events in different groups. (B) Biomarkers that can be used to identify healthy, unhealthy, and dying colonies and their earliest detection times.

Chapter 3 PhaseR4D: Time-Lapse 3D Phase Contrast Reconstruction

A. Introduction

Phase contrast microscopy is an optical imaging modality that converts phase shift differences in light as it passes through a transparent object to differences in brightness. Phase shifts are invisible to the naked eye but become visible when they are converted to variations in brightness. This type of microscopy is regularly used to capture high-contrast images of various transparent samples which include biological cells, microorganisms, internal structures and other microscopic objects. Phase contrast imaging is especially important in biological sciences and it has many unique advantages over other microscopy techniques. Specimens imaged with phase contrast capture more detail than other noninvasive methods like bright field microscopy. The converted phase shift differences, as the transmitted light passes through a sample, provide additional contrast which exposes internal structures that were previously invisible. Before the development of phase contrast imaging techniques, researchers required fluorescent labeling to visualize these substructures. However, this required special preparation of samples may alter them biologically, potentially introducing artifacts. Fluorescent labeling also required the cells to be killed and fixed which prevented time lapse studies of the same specimen. Another advantage of phase contrast is that it is not orientation dependent and can be imaged at angle without producing artifacts. Phase contrast microscopy has traditionally been constrained to 2-dimensional (2D) visualization.

However, this is insufficient for thicker objects that are larger than the microscope's depth-of-field [52]. As a result, advancements in 3-dimensional (3D) phase contrast microscopy are significant, in that they allow researchers to fully analyze any objects of interest.

In biology, 3D reconstruction methods can be useful for visualizing the cell morphology and biological processes related to their internal structures. A single image may contain various types of microscopic objects such as living stem cells, debris, culture media and differentiated cells. The height and surface curvature of these objects are not immediately apparent. For example, tightly-packed monolayers of cells are common in certain cell types such as epithelial cells. A 3D rendering may reveal membrane ridges that were previously invisible, which may aid in distinguishing the boundaries in individual cells. Additionally, 3D phase contrast reconstruction can be used to identify individual cells in tightly packed colonies such as pluripotent stem cells, and cells which grow as tightly-packed monolayers such as cancer cells. 3D features may also be used to examine biological processes and structures that are not apparent in 2D. Such features can be used in machine learning algorithms for classifications of biological conditions.

In this paper, we introduce PhaseR4D (Phase Contrast Regression 4D), a novel automated 3D reconstruction software for rendering phase contrast images in 3D over time. Unlike other existing software, PhaseR4D can be used with any microscopy system that can image at set focal distances. Once trained to a dataset, PhaseR4D can render time-lapse phase contrast images in 3D over time using only phase contrast images.

PhaseR4D has many applications in basic biology, drugs discovery, and medicine. Live cell imaging using phase contrast can provide information on cell shape, organelles, and motility, as well as cell processes such as apoptosis and cell division [53]. We apply PhaseR4D to visualize human embryonic stem cell (hESC) colonies as well as human adenocarcinoma alveolar basal lung epithelial (A549) cells, which typically grow as tightly-packed colonies or monolayer respectively. Previously, we tracked the behavior and health of hESCs in culture using non-invasive 2D phase contrast imaging [54]. In this paper, we expand to 3D time-lapse phase contrast imaging, which allows us to visualize both the cellular dynamics and volumetric changes over time. It is important to be able to develop non-invasive methods to track the health and quality of stem cells during culturing, expansion and manipulation, since often these cells are used for cellular therapies and implanted into patients. The introduction of exogenous markers could present critical health hazards and unwanted modifications. In a different application, it is equally important to be able to study the state of cancer cells with respect to progression of the disease. Imaging has the powerful capability to reveal morphological and functional information at all stages of cancer care [55]. Cancer cells can undergo various morphological and migratory changes such as an epithelial-to-mesenchymal transition (EMT), a process that enables their metastasis in order to migrate and spread to neighboring tissues [56]. Furthermore, 3D imaging is advantageous in high-throughput screening, drug testing and discovery [57]. In this study, A549 lung cancer cells were treated with a tobacco product and PhaseR4D was used to detect changes in 3D.

B. Related Work

Some preliminary work reported in this paper was originally presented at the International Conference on Image Processing 2018 [58]. While our main goal is to reconstruct a 3D volume with phase contrast images, we acquire fluorescent images for training PhaseR4D and for validating the results. Confocal or super-resolution microscopy are traditionally used for 3D microscopy due to their ability to remove out-of-focus light rays [59]. However, most of the existing microscopes with these capabilities do not have the ability to image both fluorescence and phase contrast images at the same time. Due to this, our method is designed to work with non-confocal imaging systems which are far more common in cell research laboratories and can collect both phase contrast and fluorescent images. To perform fluorescent imaging, markers are first introduced to the sample using various immunolabeling or transfection techniques. By labeling the membrane of a cell or cellular structure, we can obtain an accurate depth ground-truth.

The method used in PhaseR4D assumes that different sections of a sample will be more in focus at different heights. This is especially true for thick cells which will have some regions in focus for an image but other regions in the same image will be out of focus. Due to this, PhaseR4D uses a method inspired by existing all-in-focus algorithms. Many all-in-focus algorithms begin by acquiring images that are focused at different distances separated by a specified step size [60]. Such methods will either estimate a defocus or in-focus parameter to compute the focal plane in which a pixel location is

most in focus [61]. The estimated focal distance for each pixel can be collected in to a depth map which summarizes the depth information of an object. From this depth map, a 3D render of the object can be generated in a multidimensional matrix. In addition to 3D reconstructions, the generated depth maps may be used to produce all-in-focus microscopy images or to automatically focus microscopy systems during imaging.

To the best of our knowledge, PhaseR4D is the first and only open source software to perform 3D phase contrast reconstruction without specialized equipment. Chen et al. [62] have produced a 3D reconstruction of polystyrene beads using phase contrast, however they constructed a custom LED array and axial motion stage to image the sample at different angles. Their method would, therefore, be unavailable for use to the majority of cell research groups. Our proposed method used in PhaseR4D can be used with any existing microscopy system that has both phase contrast and fluorescence. Once PhaseR4D is trained, fluorescence will no longer be necessary and can be used with only phase contrast images.

As compared to previous work, the key contributions of PhaseR4D are: a) creating 3D reconstructions from z-stack images without specialized equipment, b) rendering of 3D reconstruction only using phase contrast microscopy, c) generating 3D phase contrast videos of live cells over time, d) identifying 3D cell surfaces using local surface patches, and e) using 3D features to automatically classify biological conditions and quality.

C. Materials and Methods

1. Cell Culture, Immunolabeling and Imaging

H9 pluripotent human embryonic stem cells (hESCs), purchased from WiCell Stem Cell Institute (Madison, Wisconsin, USA), were maintained on Matrigel coated 6-well plates in mTeSR complete medium (Stem Cell Technologies, Vancouver, Canada). When cultures reached 80–85% confluency, they cultures were passaged using ReLeSR (Stem Cell Technologies, Vancouver, Canada) and plated onto 35mm dishes to be used in experiments. A549 lung epithelial cells (human type II pulmonary alveolar adenocarcinoma cells) were obtained from (ATCC CCL-185, Manassas, VA, USA) and cultured in Ham's F-12 media supplemented with 10% fetal bovine serum (ATCC, Rockville, MD). Cells were grown until 80% confluency, at which point they were detached using 0.25% trypsin EDTA/DPBS. A549 cells were passaged every 2-3 days and medium was replenished every other day. Cells were treated with an electronic cigarette e-liquid for 3 days to induce distinct morphological changes. Cells were grown in a 37°C incubator with 90% humidity and 5% CO².

The hESC colonies were plated at 25–30% confluency and allowed to attach for 24 hours. A549 cells were treated for 4 days with 1% E-liquid from a popular electronic cigarette brand. Both the hESC colonies and A549 cells were fixed with 4% PFA for 15min, followed by blocking in 10% donkey normal serum (Sigma-Aldrich, St. Louis, MO, USA) in 0.1% Triton X (Bio-Rad, Hercules, CA, USA). Primary antibodies used were: rabbit monoclonal non-phospho β -catenin Ser45 (clone D2U8Y) (1:1000 dilution, Catalog #19807, Cell Signaling, Danvers, MA, USA), mouse monoclonal CD-44 (clone 156-3C11)

(1:400 dilution, #3570, Cell Signaling), mouse monoclonal Acetylated α -Tubulin (clone 6-11B-1)(1:500 dilution, Catalog #T7451, Sigma-Aldrich, St. Louis, MO, USA), and rabbit monoclonal Cyclin D1 (clone EPR2241) (1:50 dilution, Catalog #ab134175, Abcam, (1:500 dilutions, ambridge, MA, USA). Secondary antibodies used were Alexa-Fluor 488 Donkey anti Mouse and Alexa-Fluor 594 Donkey anti Rabbit (Life-Technologies, Carlsbad, CA, USA). Lastly cells were mounted in Vectasheild anti-fade mounting medium (Vector Laboratories, Burlingame, CA, USA).

In the A549s, Cyclin D1 is a marker important in cell-cycle and is localized to the nucleus. CD44 is a cancer stem cell marker and was localized to the plasma membrane. There was an increase in the Cyclin D1 and CD44 expression on the treated A549 lung cancer cells. Acetylated α -tubulin is a stable form of tubulin, which is localized to the cytoplasm. In the hESCs, the β -catenin is a dual function protein involved in cell-cell adhesion and gene transcription, and its labeling in normal cells is localized to the cell junctions. The data set consists of 4 conditions: single cell control (6 samples), single cell treated (5 samples), control colonies (6 samples), and treated colonies (13 samples).

All images in our datasets have a resolution of 800x600 pixels. To reduce processing time, all fluorescent and phase contrast images were resized to 400x300 by subsampling every other pixel in the image. Phase contrast and fluorescence 3D images were acquired using the BioStation IM (Nikon Instruments, Melville NY) which is a compact live cell incubator and imaging system. 99 slices were collected at 0.3 μ m spacing. A 20x objective was used to gather individual cells and small cell clusters. All images were of A549 lung epithelial cells.

2. Development and use of PhaseR4D Software.

The PhaseR4D is designed in a modular manner with three parts: fluorescent depth map generation, variance measure extraction, and phase contrast 3D reconstruction using supervised regression. A diagram of our workflow is shown in Fig 3.1. PhaseR4D was written and developed with MATLAB 2016a programming environment. The MATLAB source code, a stand-alone executable version of this algorithm, and supplied test data are available online at <http://vislab.ucr.edu/SOFTWARE/software.php>. PhaseR4D.m is the main program of the code and requires the following MATLAB toolboxes: Statistics and Machine Learning, Bioinformatics, Image Processing, and System Identification. The standalone executable requires the installation of the 64-bit version of the MATLAB Runtime R2016a (9.0.1) which is available at <http://www.mathworks.com/products/compiler/mcr/>.

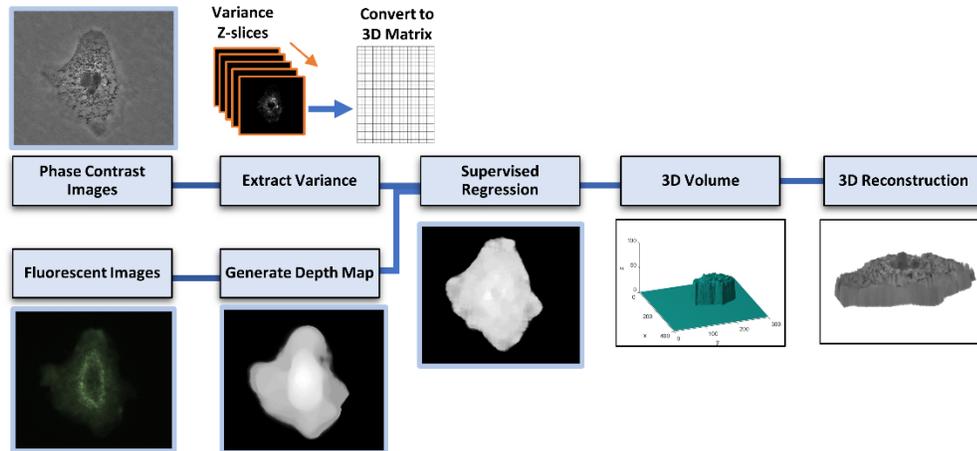


Fig 3.1. PhaseR4D Overview Diagram. PhaserR3D extracts variance values from phase contrast z-stacks which are equally spaced apart. The extracted variances are used as input features into supervised regression algorithms. A depth map generated from a corresponding set of fluorescent images is used as the labels for training the regression subsystem. Once trained, PhaseR4D computes a depth map containing the estimated regressed height values of the phase contrast images. A 3D rendering is generated from the depth map by projecting it into a 3D volume and it can be visualized over time. A base height is estimated for the volume and intensity values from the original phase contrast image stacks are reapplied to volume to finalize the 3D reconstruction.

3. 3D Reconstruction of Fluorescent Ground-Truth

PhaseR4D extracts variance features from phase contrast image slices collected at set intervals and estimates an optimal depth value using supervised regression. Fluorescent images are used to generate a ground truth for the height of each surface and to provide labels to the phase contrast data to be used in training. The regressed height values are then reshaped into a depth map which is used for 3D phase contrast image reconstruction.

4. Ground-Truth Generation

During data collection for each sample, the user is required to focus the objective at the center plane of the object. From this focused plane, h' planes above and below are imaged across the z -axis of the microscope. Each of these planes are equally separated by a step size of s micrometers. In total, $h = 2h' + 1$ planes are imaged for each object of interest in each imaging experiment. Each image represents a different focal plane in the volume. To validate and train PhaseR4D, both phase contrast and fluorescent images of size $r \times c$ must be captured at the same magnification and location. Phase contrast and fluorescent images collected at various planes for the same cell are shown in Fig 3.2. $z = 1$ is the bottom of the image stack and $z = 99$ is the top of the stack. As z increases various sections of the cell will come into focus and then come out of focus as z approaches 99. Near the border of the cell, pixels for both phase contrast and fluorescence become focused at about $z = 29$. However, the regions in the center of the cell where the nucleus is located does not become focused until about $z = 36$. The borders also come out of focus sooner than the center of the cell. This indicates that the height of a cell varies over its cytoplasm. Videos generated from all 99 stacks are included in the supplemental material to help visualize the height variations in a cell.

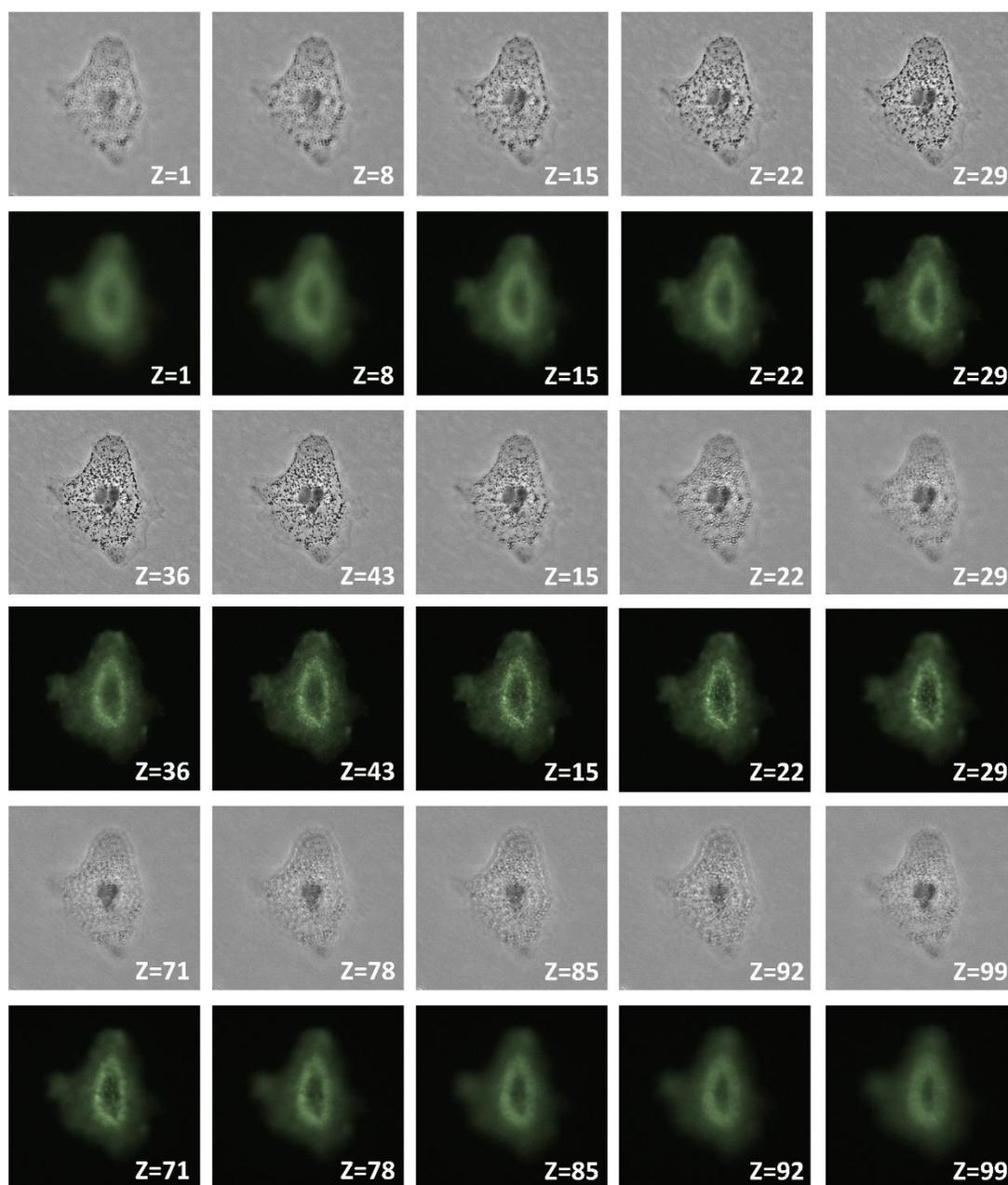


Fig 3.2. Corresponding Phase Contrast and Fluorescent Images. All images are from the same cell and were collected at 20x magnification. 15 of the 99 stacks are shown for both phase contrast (odd rows) and fluorescence (even rows). Each stack is separated by 0.3 μm .

To generate a ground-truth for the supervised regression algorithms used during phase contrast 3D reconstruction, a fluorescent depth map is generated at the same location and magnification. Traditionally, these images are collected with confocal or structured illumination microscopy (SIM) because of their ability to remove out-of-focused light [63]. However, since there are not many microscopy systems that can collect these types of images with phase, we must compute a depth map while taking blur into account. To generate our fluorescent depth map, an in-focus parameter is extracted from the set of fluorescent images collected at different focal planes. Several in-focus measures have previously been used to estimate the height of a surface at different points. Yao et al. [64] have shown that variance is a particularly strong metric to compute an all-in-focus image. PhaseR4D begins by computing a variance volume V by convolving a variance filter through each voxel. Because only the most in-focused plane is of interest, the convolving filter is an $m \times m$ 2D filter. This assumes that the most in focused plane for a pixel in the image stack is also the plane with the highest detail or variance at that neighborhood. From our testing, we have found $m=5$ pixels worked the best empirically. During image collection, cells, culture media, and other objects may sway slightly even when the cell is fixed. This is common when the mechanical instruments in the microscopy system are refocusing, changing objectives, or cycling though capture points. Because this will affect the variance calculation, it is important to register the images beforehand if sway is significant [65]. All other image stacks should be registered to the most focused image ($z \sim 50$) as reference.

A fluorescent depth map D is generated from the variance volume by the following algorithm.

1. Initialize D as a zeros matrix with the same size as the input images.
2. Loop across i and j for all pixels of D .
3. Compute the maximum value of V at (i,j) across the z -axis.
4. The value of $D(i,j)$ is the index of the maximum value across the z -axis.

The choice of fluorescent label and imaging step size are important for our method. A small step size will give more resolution in the z -axis, however more z -stacks may be needed to capture the entire volume. Because PhaseR4D computes a depth map based on the location of the surface of the cell, a fluorescent label that is localized only to the exterior cell surface is preferable. A label that fills the entire cell volume will work as well but may produce more out-of-focus light. Fig 3.3 shows contour plots representing the normalized variance through the z -axis. Each contour shows the changing variance of a pixel in the ROI of the depth map. Fig 3.3A shows the variance plot for a hESC labeled with Acetylated α -Tubulin. Fig 3.3B and 3C shows the corresponding depth map and fluorescent image at $z = 50$. Acetylated α -Tubulin is a volume filler and Fig 3.3D shows the variance plot for the same hESC but strained with cell junction protein, β -Catenin. Fig 3.3G shows the variance for an A549 cell labeled with CD44 which is localized to the exterior surface of the cell. Here the variance peak is sharp, and the contours increase exponentially, making it easier to distinguish the optimal focus plane for each pixel.

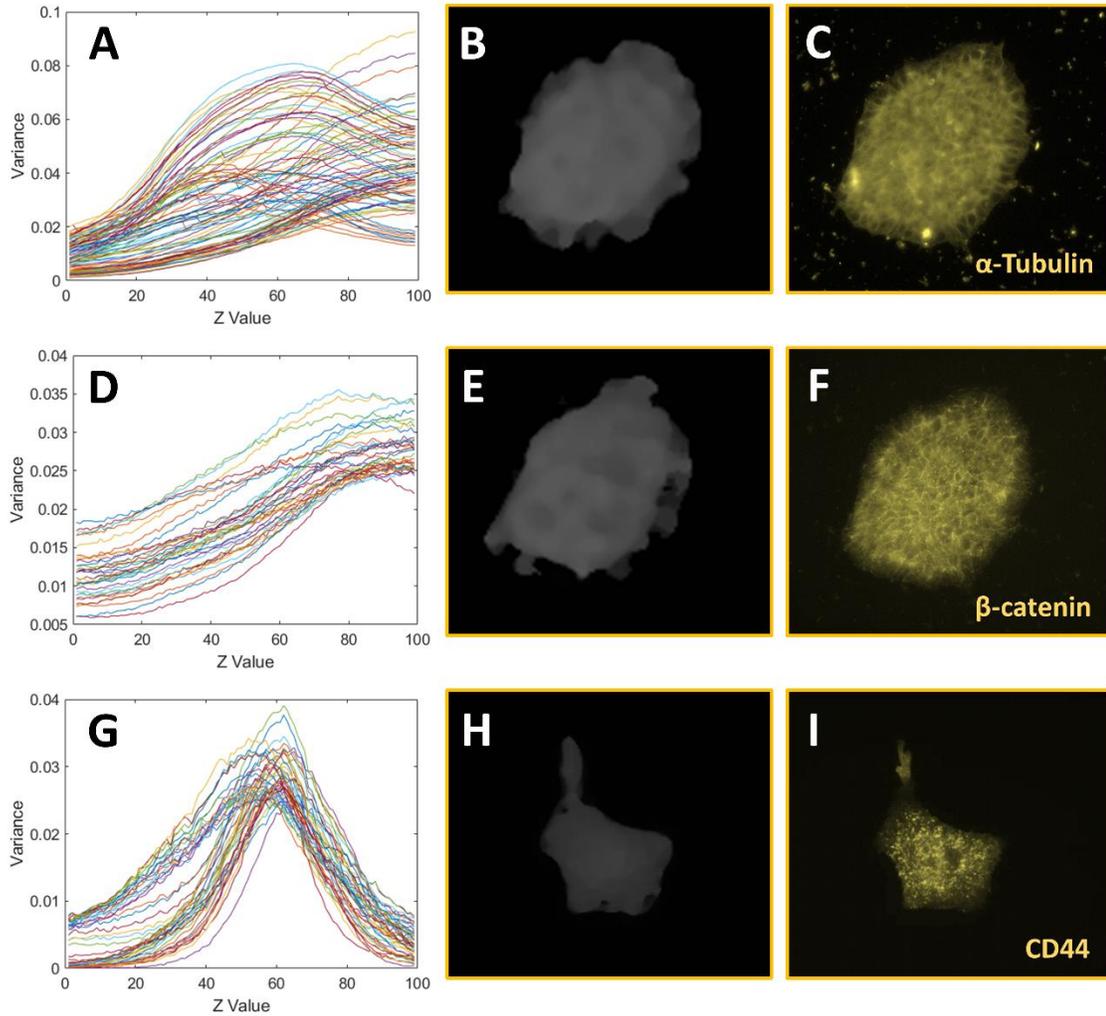


Fig 3.3. Variance Plots for Various Fluorescent Labels. A) Variance plot of pixels with large variances across z axis. B) Depth map generated from variance plots shown in A. C) Fluorescent image for hESC colony labeled with Acetylated α -Tubulin at stack $z=50$. It corresponds to the contour plot shown in A. D-F) Variance plot, depth map, and fluorescent image for hESC colony labeled with β -Catenin at stack $z=50$. G-I) Variance plot, depth map, and fluorescent image of A549 cells labeled with CD44 at stack $z=50$.

Since there may be some errors in the depth map at each pixel, a median filter may be used to smooth the values. The depth map can be refined by segmenting a region-of-interest (ROI) around the objects. This is done by using Otsu's method [29] on every image in the set, which separates the images into a foreground and background. A single ROI mask is generated by combining the foregrounds from every image. Any pixel of D that is not in the ROI is set to zero.

From Fig 3.3G, we can see that most pixels begin to suddenly increase in variance around the same point. This seems to indicate that the fluorescence is beginning to come into focus. This can also be seen in Fig 3.3D and Fig 3.3G. While fluorescence reaches its peaks at different locations for each marker, both Acetylated α -Tubulin and β -Catenin begin to increase around the same region showing the start of detail at same height. We assume that the height in which the variance begins to suddenly increase is the base of the cell. To find this point, we first compute the derivative of each contour and it can be seen in Fig 3.4. The location of the maximum value for the first derivative is a point of inflection in the original curve (Fig 3.4B). While this indicates the location of greatest increase of variance, variance has been increasing before this point. Therefore, the 2nd derivative is computed to find the location of greatest increase in the change of variance. In Fig 3.4C, we see two local maxima for the 2nd derivative. However, we are only concerned with the first local maxima which is towards the bottom of the image volume. The location of the first maxima of each contour is the base index b_i . Since there are multiple contours, the base index b of the entire volume is the median of all b_i .

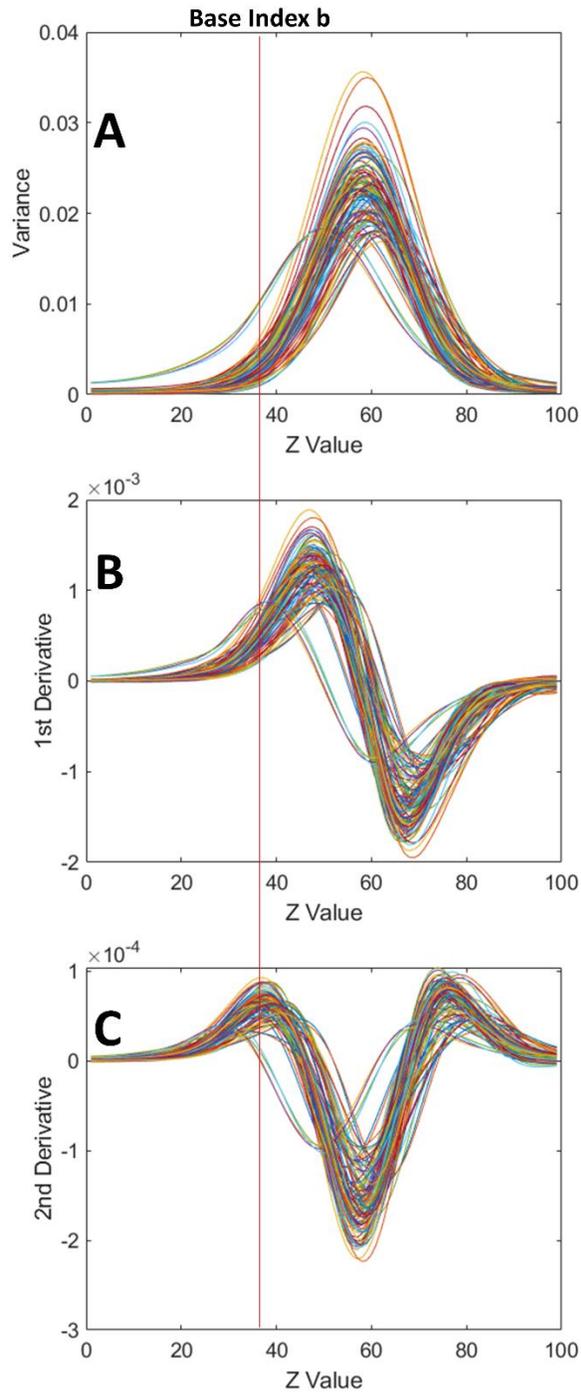


Fig 3.4. Cell Base Estimation from 2nd Derivative. A) Contour plot of variance across z-axis for pixel locations on cell. Contours have been smoothed with gaussian filters. B) Contour plot of 1st derivative of variances. C) Contour plot of 2nd derivative of variances. Red line represents the computed base index.

Once PhaseR4D has computed the depth map and base height, a 3D rendering can be built. Because the microscopy system is imaging only the top of cells, the PhaseR4D assumes that every voxel below the top voxel of the object and above the base height is a part of the object. This is a fair assumption for our data which comprise of in vitro cells on a dish. Once these cells attach to the substrate, the cells lay flat on the surface and there cannot be any parts of the cells that are passed the bottom of the dish. Using this constraint, a labelled volume L is created from the depth map as follows:

$$L(i, j, k) = \begin{cases} 1 & \text{if } b \leq k \leq D(i, j) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

L will be of the size $r \times c \times h$ which is the same as the original image stacks. Color information can be brought back from the original images and put into the matrix L to produce a color 3D reconstruction.

5. 3D Reconstruction of Phase Contrast Images

The method used by PhaseR4D to perform phase contrast 3D reconstruction is to generate a depth map using supervised regression. The input features used by the supervised regression algorithms are the variance values at each plane in the phase contrast image stack. A variance volume V_p is generated for phase contrast using the same method that was used to generate the fluorescent variance volume. The only difference is that instead of using the fluorescence images, the phase contrast images are used. To reduce the effect of background information in our regression, we remove the background from every z plane in V_p . This is done by using the edge based segmentation method used in [54] for each z plane to create a set of foregrounds. Connected components are removed

from the foreground if they are smaller than a specified number of pixels. These foregrounds are combined to create a single ROI mask which is applied to every z plane in V_p .

While the maximum variance location is sufficient for estimating the height in fluorescence images, it is not feasible to reconstruct phase contrast with it alone. Fig 3.5A shows the variance plots of all pixels in the cell. All contours in the ROI are displayed to show that the variance patterns in phase contrast are not as consistent as in fluorescence (as seen in Fig 3.4). For many pixels, the variance seems to increase sooner than usual and has a 2nd local maximum (black arrows in Fig 3.5A). This is due to the effect of blurring in phase contrast images, where a ripple or haloing effect occurs when out of focus. This is especially apparent at the bottom of the phase contrast images in Fig 3.5B where a small piece of debris is located (red arrow). You can see that variance is at a local minimum when the debris is most in focus. Because of this an alternate method is required to estimate the depth map.

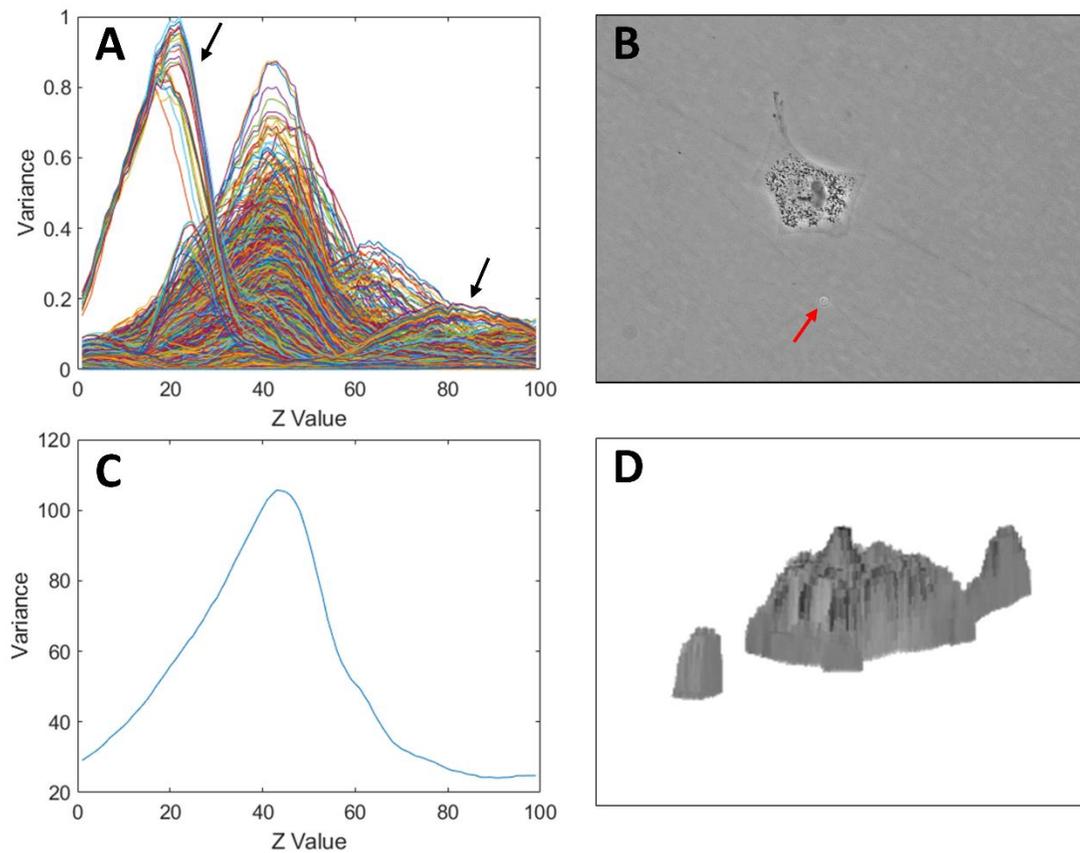


Fig 3.5. Cell Base for 3D Phase Contrast. A) Contour plot of variance across z-axis for all phase contrast pixels in ROI. Black arrows represent 2 peaks for a debris shown in Fig 3.5B. B) Phase contrast image at z=50. Red arrow indicates debris which is not visible in fluorescence image. C) Single variance contour generated from variance of background at each slice. D) Final 3D reconstruction after rendering 3D volume with base index.

Our proposed method used by PhaseR4D is to use the variance as inputs into existing supervised regression algorithms. After acquiring V_p , variance vectors are extracted by examining variance values along the z-axis and will be used as an input sample into the supervised regression algorithms. For a volume of size $r \times c \times h$, there will be $r \times c$ variance vectors of size $h \times 1$. A label $H(i,j)$ is also assigned to each sample vector based on corresponding value on the fluorescent depth map $D(i,j)$.

6. Phase Contrast Depth Map from Supervised Regression

Once the variance vectors are extracted, they will be used to train a regression model. We have tested three different regression models: regression trees [66], linear support vector regression (LSVM) [67], least squares boosting regression (LSBoost) [68]. Each model will take a vector of $h \times 1$ inputs and estimate a height value. After estimating a height value for the $r \times c$ samples of an image volume, these values are reshaped into a depth map D_p of size $r \times c$.

Phase contrast requires a different method for estimation of the base height than fluorescence. For phase contrast variances, there is no specific height in which all the variance contours begin increasing together. This is apparent in Fig 3.5A where there are multiple increasing slopes for the same contour. To deal with this, we make use of the background information which is not visible fluorescence. Using the ROI, we extract the background variance from every z stack. Plotting this gives the contour plot in Fig 3.5C. The height with the maximum variance where the base of the dish becomes focused. This height is taken as the base index for phase contrast, b_p .

7. 3D Reconstruction from Depth Map

Because the microscopy system is imaging only the top of cells, the method assumes that every voxel below the top voxel of the object is a part of the object. This is a fair assumption for our data which comprise of in vitro cells on a dish. Once these cells attach to the substrate, the cell lay flat on the surface. Using this constraint, a labelled volume L_p is created from the depth map as follows:

$$L_p(i, j, k) = \begin{cases} 1 & \text{if } b_p \leq k \leq D_p(i, j) \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Because the height values are regressed for each pixel in a depth map and they are independent of their neighbors, the 3D reconstruction may not be smooth. To create a smoother 3D reconstruction, a 2D median filter is convolved with the depth map before being used to create the labelled volume. The median filter will remove isolated outliers and replace them with the median value of the neighborhood. Once L_p is computed, intensity information from the phase contrast images are brought back to create the final phase contrast reconstruction which is shown in Fig 3.5D.

8. Phase Contrast 3D Reconstruction on Live Cells over Time

After a PhaseR4D has constructed a regression model to generate depth maps of specific data conditions, the model can be used to generate depth maps for new image sets of the same conditions. Because of this, the fluorescent depth maps are no longer necessary as they were only used to label the voxels in the phase contrast image sets. Additionally, the cell samples do not have to be fixed for imaging. Capturing phase contrast images (approximately 1/32-1/8 sec) is orders of magnitude faster than capturing fluorescent

images (approximately 0.3-1 secs), thereby making it possible to render 3D in live images without producing artifacts. This allows PhaseR4D to perform 3D reconstruction of the same sample over time. This type of visualization would be beneficial for analyzing biological processes that may not be visible in 2D images alone, such as height variations over time. Fig 3.6 displays the 3D reconstruction and most focused image slice for a sample imaged over time. This sample was imaged for 99 z-planes, 145 timepoints about 20 minutes apart for 48 hours. The videos related to this sample are included in the supplemental materials. 4 cells are imaged over time, each displaying different morphologies. All cells were stressed to detect changes in morphology and started dying at about 24 hours. PhaseR4D was able to create a 3D reconstruction of each 4 cell for 48 hours. Errors did appear after 24 hours as PhaseR4D was not trained on dead cells. These videos show that it is possible to use only phase contrast images captured over time and from a single vertical angle to generate 3D reconstructions of live dynamic cells. Since the cells are live and unfixed, all 99 image slices must be captured at a sufficient rate. For consistency, the median base index is of all time points is used as the base for all frames in the video.

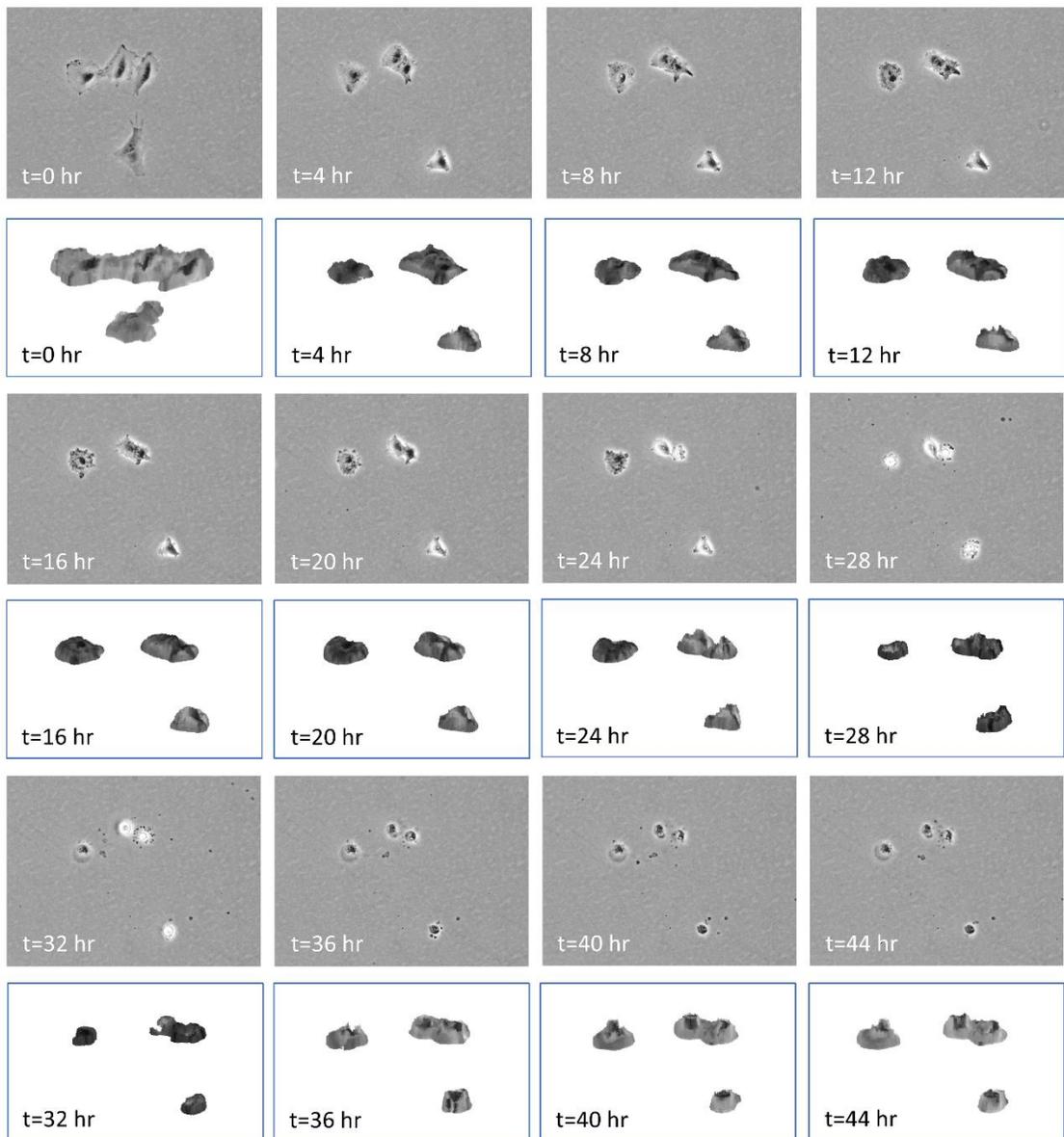


Fig 3.6. Live 3D Reconstruction of Phase Contrast over time. Odd rows: Phase contrast images of live control A549 cells captured at the same location over 48 hours (4-hour time steps). All images are at $z=50$. Even rows: Final 3D reconstruction at the corresponding time point.

9. Classification of Biological Processes using 3D Features

Once a 3D reconstruction is completed, 3D features may be extracted from the reconstructed surfaces. PhaseR4D extracts the volume, average volume intensity, and average shape index in each reconstructed phase contrast volume. The shape index, S_i , a value which can be used to identify the surface type of a point on a surface, can be computed as follows:

$$S_i(P) = \frac{1}{2} - \frac{1}{\pi} \tan^{-1} \frac{k_1(P) + k_2(P)}{k_1(P) - k_2(P)} \quad (3)$$

where P is the point of interest on the 3D surface, k_1 is the maximum principal curvature, and k_2 is the minimum principal curvature [69]. S_i will have a value between 0 and 1 and can be used to classify the surface based on the following ratios: spherical cup [0, 1/16), trough [1/16, 3/16), rut [3/16, 5/16), saddle rut [5/16, 7/16), saddle [7/16, 9/16), saddle ridge [9/16, 11/16), ridge [11/16, 13/16), dome [13/16, 15/16), and spherical cap [15/16, 1] [70]. These shapes can be further summarized into convex [0, 5/16), saddle [5/16, 11/16), and concave [11/16, 1].

After the 3D features are extracted from a phase contrast dataset, they can be used as inputs into supervised classifiers. These features were used to train three classifiers for classification of biological process: (1) discriminant analysis (DA) classifier, (2) k-nearest neighbor (KNN), and (3) error-correcting output codes (ECOC) multiclass model. The discriminant analysis classifier is trained by fitting a Gaussian distribution to each class. Test samples are compared to each class distribution and assigned to the class with the lowest misclassification cost [71]. k-nearest neighbor classifiers label new observations by

comparing them to the k -nearest training samples in a multi-dimensional space [37]. $k = 5$ was used for all experiments. ECOC is a classifier that reduces a multiclass classification problem into a set of L binary classifiers. PhaseR4D uses a support vector machines (SVM) classifier for every pair of classes, $L = 3$. A new test observation is assigned to the class that minimizes the losses of the L binary learners [72].

D. Results

Our datasets consisted of cell volumes with $400 \times 300 \times 99$ data points each. A leave one out cross-validation was performed for each regression model by training on all cell volumes but one and testing on the last volume. Fig 3.7 displays the depth maps of ground truth and the 3 regression models. The figure also shows the 3D reconstruction generated by the depth maps. All sub images are of the same cell.

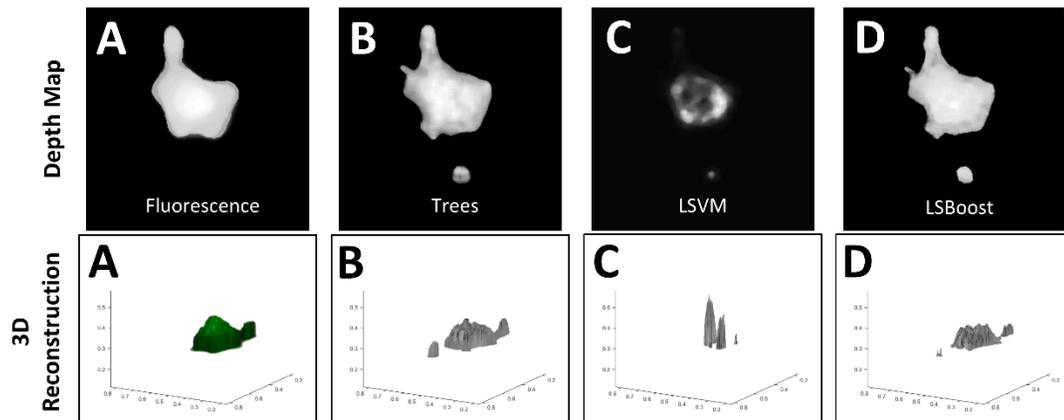


Fig 3.7. Regressed Depth Maps and 3D Reconstructions. First row: Depth maps generated from A) fluorescence images of A549 cells, B) supervised regression trees, C) linear support vector regression, and D) least-squares boosting. Second row: Final 3D reconstructions generated from corresponding depth maps.

In order to validate the quality of the curvature of our surfaces, we compared sections of our phase contrast depth map with our fluorescent ground-truth using local surface patches (LSP) [73], [74]. Chen and Bhanu have been shown LSPs to be very useful in the study of 3D ear recognition. LSP is a local surface descriptor that is characterized by a centroid, a local surface type, and a 2D histogram. To compute the local surface type and 2D histogram, the shape indices around a neighborhood must be computed [75]. In addition to the shape indices, a dot product of surface normal vectors at P and each of its neighbors are needed for the 2D histogram. This dot product has a value between -1 and 1 and is computed for every neighbor in a specified range. The shape index and dot products for every neighbor are collected into the histogram.

Two LSP histogram can be compared using the χ^2 -divergence which is one of the most used divergence measures used for assessing the dissimilarity between two probability density functions. Once normalized, a histogram can represent an approximation of PDF. The dissimilarity between two histograms, Q and V, is shown in the following equation [76]:

$$\chi^2(Q, V) = \sum_i \frac{(q_i - v_i)^2}{q_i + v_i} \quad (4)$$

The dissimilarity value will be between 0 and 2. Two histograms with a higher value will be more dissimilar, while a lower value means they are similar. Fig 3.8 shows the multiple LSP histograms and their corresponding location in the 3D surface. The size of our LSP histograms are 34x17, the same bin sizes used by Chen and Bhanu [74]. Chen and Bhanu showed that a dissimilarity value of ~0.5 was extremely similar (same object with slightly

different view), a value of ~ 1.0 was similar (same type of surface), and ~ 2.0 was extremely dissimilar (completely different surface type). The figure shows 4 LSPs and their corresponding locations. Table 3.1 contains the χ^2 -divergence between LSP 1 and the other 3 LSPs. LSP 1 and 2 are on the same cell and both are on nuclei. From the measured shape indices at these positions, both locations have a concave surface. The dissimilarity value for these two histograms is 0.428 which indicate that they are similar. LSP 3 is also on the same cell but has a concave surface. The dissimilarity to LSP 1 one is much higher in this case. LSP 4 is a local surface patch that has been extracted from the fluorescent 3D reconstruction of the same cell. The location of this patch is very close in proximity to LSP 1 and has the same surface type. The dissimilarity between these two histograms is low at 0.374 indicating that they are very similar.

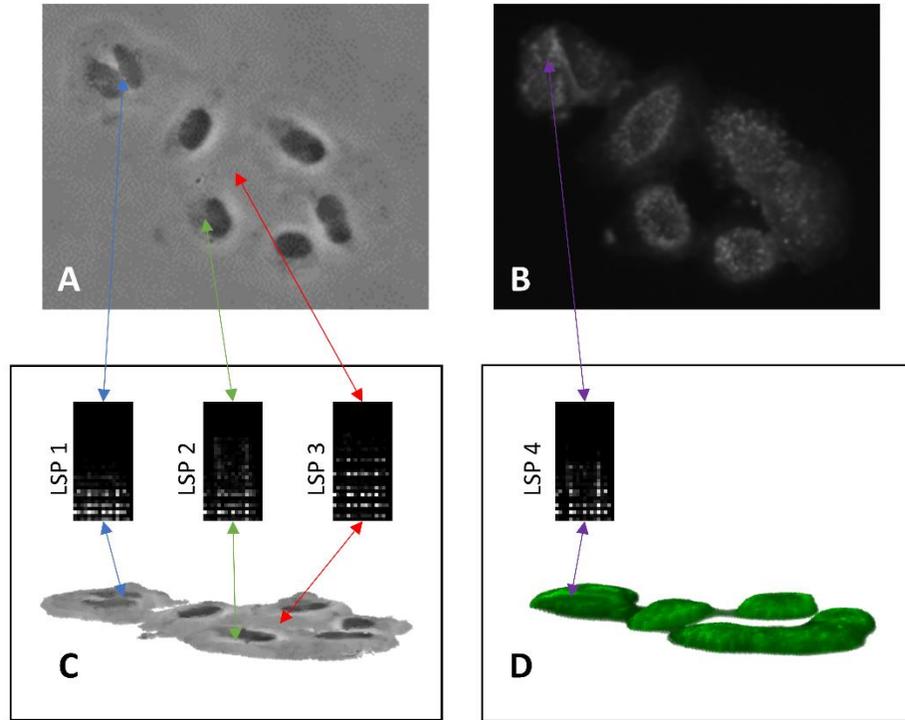


Fig 3.8. Local Surface Patches at Specific Cell Locations. Four LSP histograms and their corresponding locations on the 3D reconstruction and z=50 image slice. LSP 1 and 2 are on the same cell and both are located on a nucleus with a concave surface. LSP 3 is on the same cell but in between nuclei with a convex surface. LSP 4 on the corresponding fluorescent location to LSP 1.

Local Surface Patches	LSP 1	LSP 2	LSP 3	LSP 4
Surface Type	Concave	Concave	Convex	Concave
χ^2 -divergence compared to LSP 1	N/A	0.428	1.312	0.374

Table 3.1. Local Surface Patch Surface Types and χ^2 -Divergence Local surface patch surface types of the 4 LSPs displayed in Fig 3.8. Displays distance measure computed using χ^2 -Divergence between LSP1 and the other 3 LSPs.

Because there are 400*300 points in our depth map, we do not generate an LSP for each. LSPs are only generated for feature points on a surface. A point is considered feature point if it is a regional maximum on the depth map and has a shape index greater than 11/16. This limits the possible shapes at the local maximum to convex shapes: ridge, dome, and spherical cap. In addition, a point may also be considered a feature point if it is a regional minimum and has a shape index less than 5/16. These points represent concave regions and can be classified as spherical cups, trough, or ruts. Concave feature points will be very rare in single cells as most minima will be at the edge of cells. However, concave feature points are far more common in colonies as pits will form between cell boundaries. A feature point in the test case will only be compared with a feature point in the ground-truth if they are both the same shape type (concave or convex) and their centroids are within 25 pixels.

Because fluorescent markers target specific cellular structures, other objects such as debris are not detected. However, they are still visible in phase contrast. One such object can be seen in the bottom of the regressed depth maps (Fig 3.7B-3.7D). These objects are then detected in the phase 3D reconstructions but not in the fluorescence. While this will negatively affect our 3D reconstructions results because it will produce false positives when compared to our ground-truth, it opens the possibility of 3D reconstructing objects that fluorescence cannot.

Table 3.2 shows the average dissimilarity, precision, and recall results on the single cell control conditions. The average dissimilarity is computed from all matched LSP histograms between the regressed and ground truth 3D reconstructions. If no match to the

ground-truth is found, N/A is displayed. The table displays the results for all three supervised regression algorithms. Tables 3.3, 3.4, and 3.5 display the corresponding information for the single cell treated, control colony, and treated colony conditions, respectively.

Method	Trees			LSVM			LSBoost		
	Mean Dissim.	Precision	Recall	Mean Dissim.	Precision	Recall	Mean Dissim.	Precision	Recall
Control 1	0.97	74.82%	57.70%	0.59	36.15%	24.84%	0.95	72.42%	57.34%
Control 2	0.83	54.21%	86.23%	0.72	19.59%	33.97%	0.73	54.86%	81.28%
Control 3	0.89	83.15%	56.43%	1.34	28.40%	18.74%	0.85	80.92%	62.90%
Control 4	1.13	87.95%	77.06%	1.19	32.13%	24.57%	0.96	84.23%	69.52%
Control 5	0.63	85.48%	82.70%	0.85	28.05%	30.70%	1.01	83.61%	74.03%
Control 6	1.46	42.53%	94.55%	1.10	15.93%	37.29%	0.59	75.99%	90.59%
Average	0.99	71.36%	75.78%	0.97	26.71%	28.35%	0.85	75.34%	72.61%
SD	0.28	18.71%	15.57%	0.29	7.62%	6.87%	0.16	11.02%	12.14%

Table 3.2. Classification and Dissimilarity Results for Single Cell Control Samples. Precision, recall, and average dissimilarity results for Trees, LSVM, and LSBoost on 6 single cell control samples.

Method	Trees			LSVM			LSBoost		
	Mean Dissim.	Precision	Recall	Mean Dissim.	Precision	Recall	Mean Dissim.	Precision	Recall
Treated 1	0.29	88.31%	92.12%	N/A	26.67%	36.84%	0.75	88.06%	86.82%
Treated 2	0.83	89.90%	89.69%	1.18	37.86%	49.05%	0.73	93.57%	84.63%
Treated 3	1.23	70.83%	91.24%	1.07	22.88%	47.20%	0.74	70.42%	89.08%
Treated 4	0.75	99.40%	66.97%	N/A	34.74%	14.55%	1.02	99.31%	60.33%
Treated 5	0.95	77.10%	85.37%	0.85	44.64%	52.63%	0.83	77.98%	83.40%
Average	0.81	85.11%	85.08%	1.03	33.36%	40.05%	0.82	85.87%	80.85%
SD	0.34	11.24%	10.45%	0.17	8.72%	15.42%	0.12	11.68%	11.67%

Table 3.3. Classification and Dissimilarity Results for Single Cell Treated Samples. Precision, recall, and average dissimilarity results for Trees, LSVM, and LSBoost on 5 single cell treated samples.

Method	Trees			LSVM			LSBoost		
Samples	Mean Dissim.	Precision	Recall	Mean Dissim.	Precision	Recall	Mean Dissim.	Precision	Recall
Colony 1	0.89	58.15%	88.34%	0.95	15.33%	26.91%	0.98	56.51%	83.45%
Colony 2	0.57	47.32%	93.07%	1.34	29.68%	62.41%	0.86	46.02%	87.88%
Colony 3	0.81	83.81%	83.10%	N/A	20.29%	19.00%	0.92	83.00%	75.52%
Colony 4	0.80	81.53%	90.81%	0.81	36.75%	23.75%	1.07	81.34%	85.34%
Colony 5	0.78	47.26%	90.43%	N/A	15.20%	35.07%	0.76	46.38%	85.16%
Colony 6	0.93	93.20%	82.38%	1.45	28.95%	30.51%	1.06	92.96%	73.10%
Average	0.80	68.54%	88.02%	1.14	24.37%	32.94%	0.94	67.70%	81.74%
SD	0.13	20.11%	4.37%	0.31	8.77%	15.45%	0.12	20.53%	5.98%

Table 3.4. Classification and Dissimilarity Results for Control Colony Samples. Precision, recall, and average dissimilarity results for Trees, LSVM, and LSBoost on 6 control colony samples.

Method	Trees			LSVM			LSBoost		
Samples	Mean Dissim.	Precision	Recall	Mean Dissim.	Precision	Recall	Mean Dissim.	Precision	Recall
Treated Colony 1	0.92	82.79%	43.45%	N/A	39.04%	14.10%	0.97	80.09%	42.91%
Treated Colony 2	1.17	67.83%	67.04%	N/A	23.87%	21.83%	0.77	67.00%	64.23%
Treated Colony 3	0.72	73.63%	61.51%	0.60	28.55%	20.11%	0.96	72.46%	57.89%
Treated Colony 4	1.11	71.07%	61.30%	1.39	23.89%	19.75%	1.40	69.78%	58.79%
Treated Colony 5	0.47	55.41%	88.47%	N/A	15.94%	23.90%	0.90	54.00%	81.22%
Treated Colony 6	1.21	63.34%	86.49%	N/A	24.32%	28.35%	0.84	63.61%	80.72%
Treated Colony 7	0.88	68.31%	78.73%	0.88	39.33%	22.16%	0.85	68.22%	74.34%
Treated Colony 8	1.00	46.57%	80.81%	N/A	15.70%	26.75%	0.82	44.19%	73.98%
Treated Colony 9	1.08	52.87%	74.77%	0.98	24.41%	40.75%	0.51	51.44%	71.79%
Treated Colony 10	0.97	52.44%	84.05%	0.59	18.64%	40.29%	0.89	50.83%	79.24%
Treated Colony 11	0.74	40.20%	95.66%	1.01	17.53%	33.63%	0.68	40.08%	90.65%
Treated Colony 12	1.06	55.96%	69.17%	N/A	17.60%	26.56%	0.87	51.77%	63.35%
Treated Colony 13	1.19	68.27%	54.72%	1.17	31.89%	30.43%	1.03	67.10%	53.44%
Average	0.96	61.44%	72.78%	0.94	24.67%	26.82%	0.88	60.04%	68.66%
SD	0.22	12.00%	14.94%	0.29	8.08%	7.90%	0.21	12.05%	13.31%

Table 3.5. Classification and Dissimilarity Results for Treated Colony Samples. Precision, recall, and average dissimilarity results for Trees, LSVM, and LSBoost on 13 treated colony samples.

For all conditions, the Trees and LSBoost outperformed LSVM. For single cell control results, LSBoost had slightly better precision and was able to produce surfaces that were slightly more similar. As seen in table 3.2, all algorithms performed better for the treated samples. This is likely due the dark vesicles noted in the cytoplasm that are visible on phase contrast images (Fig 3.2). These vesicles produce more contrast (variance) for regions inside the cells. Both control and treated colonies performed worse compared to single cells. While the precision results are worse due to more false positives, recall results were the same. Also, PhaseR4D was able to find many similar surfaces when compared to the ground truth for colonies. This shows that while PhaseR4D works on colonies, it does perform better for single cell conditions by producing fewer false positives. Precision results for all conditions are also lower because of unlabeled objects in fluorescent such as debris. Since these objects are not invisible in phase contrast, they will appear as false positive voxels. Additionally, because of the haloing and blurring effects of phase contrast, as objects become defocus, they will appear larger at the edges producing more false positives. Improving the edge detection to account for this will produce a better ROI and reduce the number of false positives.

After completing the 3D reconstruction of phase contrast images, 3D features were extracted and used to classify images as control or treated cells. Tobacco products have been shown to induce signs of EMT in cancer cells, which include increased motility, elongated morphology, and loosely formed colonies [56]. The treated cells in the A549 datasets showed changes in cell morphology and less cell-to-cell contracts visually, and

their 3D features could be used to automatically distinguish from the control A549s. The extracted features, the number of cells in the volume, and a condition label of treated or control were used as training inputs to our machine learning classifiers.

Our total dataset for A549 cells consisted of 30 samples, of which 12 were control and 18 were treated. Classification was done with 10-fold cross validation by partitioning the data into 10 equal sets. Each set is used as the test set once while the other 9 are used for training. The experiments were repeated 10 times by randomly shuffling the dataset so that the sets would be randomly sorted. In total each classifier was trained and test 100 times. The DA classifier had a classification accuracy of $91.00 \pm 2.25\%$, the KNN had an accuracy of $88.00 \pm 2.33\%$, and the ECOC had an accuracy of $90.33 \pm 1.89\%$. This shows that is possible to classify a biological process such as EMT using 3D features.

Chapter 4 DendritePA: Fluorescent Protein Dynamics

A. Introduction

Dendritic spines are small protrusions located on the surface of neurons, which receive inputs from other neurons and are the active sites for neuronal communications called synapses. These synapses are often remodeled by the rapid turnover of the actin cytoskeleton, which is regulated by various actin-binding proteins [77], [78]. Cofilin is an actin-severing protein and its activity is regulated by phosphorylation at Ser3 [79], [80]. Cofilin mediated remodeling of the actin cytoskeleton is critical in regulating the shape and functionality of dendritic spines. Therefore, the localization and phosphorylation state of cofilin within dendritic spines can affect the synaptic functions. Cofilin-S3A is a constitutively active mutant form of cofilin, where the Ser3 is substituted to alanine, which can constitutively bind and remodel actin filaments, whereas in inactive form of Cofilin-S3D the serine is substituted with aspartate [77], [78]. The S3A mutant cannot be inactivated by phosphorylation, hence it is always in its active “severing” state, which leads to filamentous actin (F-actin) depolymerization, whereas the S3D substitution prevents the cofilin binding to F-actin, which reduces F-actin depolymerization. These two mutants are often used to study the mechanism of cofilin phospho-regulation in neurons [81], [82] and are shown in Fig 4.1. Although various studies have reported on the functional down-stream effects of the cofilin mutants, the changes in their dynamics have been relatively unexplored to date.

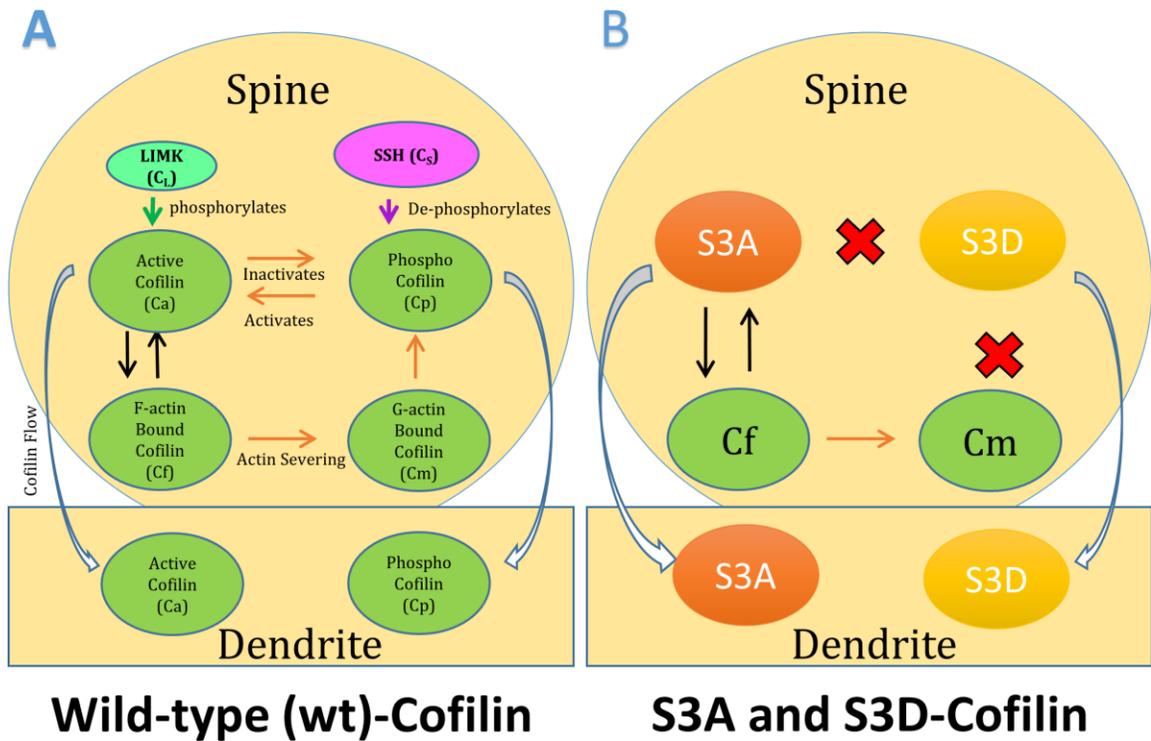


Fig 4.1. Cofilin dynamics in dendritic spines. A) When wild-type cofilin is active (Active Cofilin Ca) it binds to F-actin (F-actin Bound Cofilin Cf) and severs it into G-actin (G-actin Bound Cofilin Cm). When cofilin is phosphorylated (Phospho Cofilin Cp) on Ser3, it cannot bind the actin. This phospho-regulation is mediated by two upstream players, LIMK which phosphorylates cofilin, and SSH which dephosphorylates cofilin. Both Active and Phospho Cofilin can move from the spines into the dendrites. B) Cofilin-S3A, which cannot be phosphorylated, can bind to actin (Cf) and sever it into G-actin (Cm). However, Cofilin-S3D cannot bind to actin. S3A and S3D are not able to convert between each other, since they are different mutants. However, both may flow in and out of dendritic spine regions.

In this paper, we investigate the dynamics of the actin-severing protein, cofilin, and its effects on remodeling of dendritic spines. Dendritic spines contain the post-synaptic sites of excitatory synapses in the central nervous system (CNS) [83]–[87]. Dysregulation of dendritic spines can have a strong impact on brain functions and underlie cognitive decline associated with neurological diseases. Cofilin can regulate the remodeling of dendritic spines through the disassembly and reorganization of F-actin cytoskeleton, which provides the structure to dendritic spines. Elevated levels of cofilin have previously been shown to contribute to loss of synapses and spines in neurodegenerative disorders, such as Alzheimer’s disease (AD) [88], [89]. However, the precise mechanism underlying cofilin-mediated loss of synapses is unclear. Therefore, it is important to quantify the motility of cofilin and examine how the localization of cofilin affects dendritic spine shape.

Most previous studies involving the effects of proteins on neurons have primarily used manual examination, segmentation, and classification [90]. Most of these biological studies have used popular user-operated software such as ImageJ [91] to manually segment dendritic spines, other studies have used visualization systems such as Imaris or NeuroLucida [92]. However, both Imaris and NeuroLucida require z-stack information and are sensitive to parameter selection. These manual methods are prone to human bias and are extremely tedious and time-consuming processes when performed on multiple images. Because of this, it is advantageous to develop an image analysis software such as DendritePA to automatically segment dendritic spines and extract features for the analysis of live fluorescence videos.

To automatically relate cofilin motility with dendritic spine shapes, dendritic spines must be automatically segmented. Because of the small size of dendritic spines, it is very hard to acquire images with sufficient resolution and contrast to properly analyze the dynamic entities and structures. To compensate, many experiments use the maximum intensity projection of a z-stack instead of data from a single image. For our work, capturing these z-stacks would be disadvantageous, as we are examining two separate fluorescence channels; a green channel, which detects wild-type (wt)-Cofilin-GFP to assess cofilin motility and a red channel to detect tdTomato providing spine structural information. While it may be possible that overexpression of proteins cause artificial effects, our previous studies with the cofilin mutants have showed no adverse effects of GFP-tagged cofilin or TdTomato on neurons [81]. In addition, the genetic cofilin mutants cofilin-S3A and cofilin-S3D were used to study the effects of cofilin activity on its dynamics and spine shape. To analyze the spatiotemporal relationship between dendritic spines and cofilin, a time series of sufficient temporal resolution must be captured. Here we present a method that uses the spatiotemporal information of the video to improve the signal-to-noise ratio of each frame without having to acquire z-stack data.

In this paper, we present DendritePA, a novel automated pattern recognition system that analyzes protein localization in neurons using multi-channel fluorescence microscopy and relates it to dendritic spine shape and protein activity state. Unlike previous work, our DendritePA uses video bioinformatics algorithms to automatically obtain spatiotemporal pattern information on protein dynamics. The system is used specifically to examine the effects of cofilin motility on the shape and evolution of

dendritic spines. Fluorescence microscopy is used because the pixel intensity is assumed to be proportional to the amount of stained proteins in the region. Due to the small size of cofilin, which is at the subpixel level or smaller than a pixel at 40x magnification, individual molecules of cofilin cannot be tracked. However, our DendritePA can estimate the changes in cofilin density within the dendritic spines by measuring their intensity levels. DendritePA also uses a spine energy representation derived from an existing motion pattern representation called gait energy image (GEI) to summarize spine motion into a single image. Doing so allows for the extraction of useful features that can be used to classify segmented spine shapes. By relating the spine shapes with the observed cofilin trafficking dynamics, it is possible to examine the underlying biological processes.

B. Related Work

Some preliminary work reported in this paper were originally presented at the International Conference on Pattern Recognition 2016 [93]. To the best of our knowledge, before our previous conference paper, cofilin has never been automatically quantified. Another actin-regulating protein paxillin, has previously been automatically analyzed in non-neuronal fluorescence images [94]. However, only paxillin dense regions that are clearly visible are examined. These paxillin dense regions are also sparse and appear much brighter than the rest of the cell allowing for simple segmentation. However, this is not the case for cofilin located in neurons as it is more uniformly distributed. Also, cofilin dense regions are not static, forming and dispersing over time,

making tracking these clusters challenging. Bosch et al. [82] have manually studied the effect of cofilin localization on the remodeling of dendritic spines. They classified cofilin transport in dendritic spines into four patterns: persistent increase in concentration, transient increase, transient decrease, and persistent decrease. In their work, they found that cofilin transport patterns correlated with the remodeling of dendritic spine shapes.

There are some current methods that automatically inspect the flow of proteins in cells. Many of these techniques evaluate individual particle trajectories over time by using frame by frame object detection [95] and associating the objects across time. An issue with these techniques is that they do not perform well with high particle density and background noise. Another method is to separate the cell into regions and measure particle flux by the intensity level or protein density in the regions [96], [97]. Pecot et al. [97] designed an approach that involves partitioning the cell into predefined sections of set sizes and shape. By checking the quantity of particles or tracking changes in the intensity levels, they could estimate the flux of these particles through the boundaries between sections. A drawback of this approach is that regions must be rigid and the choice of region size affects the performance. Experiments on live samples utilized micro-fabricated patterns [98] to constrain the cell shape so that the partitioned regions remained consistent throughout the experiment.

To efficiently correlate cofilin motility with dendritic spine shape, spines must be automatically segmented and classified. Spine segmentation methods can be separated into two groups, classification-based [99] and centerline extraction based [100]–[104].

The classification-based methods classify individual pixels into various groups such as spine, dendrite, or background [99]. The software NeuronStudio by Rodriguez et al. [99] utilizes the pixel distance to the nearest surface point as a feature in classification, however, this can generate spurious spine detection and it is sensitive to noise. NeuronStudio also requires manual input by the user before the segmentation process can begin. Centerline extraction based methods involve detecting the backbone or central region of dendrites and segmenting spines by their relationship to the central region. Traditional methods may experience issues when the dendrite width varies along its orientation. In our previous work, the method [105] was used to detect a center region using gradient vector flow [106] instead of a thin backbone. However, this method does not completely capture the segmentation of a dendrite. Instead, in this paper, we choose to build upon the method used by Basu et al. [104] called 2dSpAn. This method uses a set of convolution kernels at varying angles to accurately segment a dendrite. This method can compensate for varying dendritic spine widths. However, the software requires user input of seed points for the kernels, does not consider dendrites with large curvatures, and may overestimate the size of the dendrite at the base of large spines. In our work, we address each of these issues.

Using segmented spine information, automated classification of shape type is critical in analyzing biological conditions. Basu et al. [104] utilized a decision tree method to classify spines by using neck length, spine height, and head width. An issue with these features arises when the resolution is low because they are measured in only a few pixels. This prompts an increased likelihood of measurement error and resulting in

misclassifications. The primary feature utilized by DendritePA is an adaption of gait energy image (GEI) [107]. GEI is a spatiotemporal gait representation that has been widely used to characterize human walking patterns and has previously been shown to be highly effective for recognition of different individuals. DendritePA uses a spine energy image (SEI) along with other features in the classification of dendritic spine shapes such as mushroom, thin, and stubby. Unlike previous spine classification methods, SEI allows for the use of spatiotemporal information in classification.

As compared to previous work, the key contributions of our work are: a) Present DendritePA (protein analysis) software which is an automated, unbiased program that can be used to segment and track dendrite spines, analyze cofilin patterns in fluorescence live videos, and classify dendrite spine shape. b) Develop for the first time an automated algorithm suite to quantify the movement of cofilin in dendrites and spines, and correlate it to spine shape using multi-channel fluorescence live videos. c) Use spatiotemporal information to enhance the signal-to-noise ratio in videos and perform automated analysis of multiple fluorescent probes in time-lapse videos for tracking the local distribution of cofilin while simultaneously analyzing the effects on spine shape. d) Segment spines and dendrites using convolution kernels that can adapt to changing angles automatically. e) Automatically classify individual dendritic spine shapes using SEI and other features. f) Examine the effect of cofilin activation using wild-type cofilin, cofilin-S3A, and cofilin-S3D. Understanding the dynamics of cofilin motility and activation within sub-neuronal compartments is critical to understanding its function in regulating the morphological structure and functionality of synapses.

C. Materials and methods

1. Animal protocol

All animal care protocols and procedures were approved by the UC Riverside Animal Care & Use Program, which is accredited by AAALAC International, and animal welfare assurance number A3439-01 is on file with the Office of Laboratory Animal Welfare (OLAW).

2. Mice

Mice were obtained from Jackson laboratories, housed in an AAALAC-accredited facility under 12-h light/dark cycles and fed standard mouse chow. Food and water were provided *ad libitum*. All procedures were approved by the Institutional Animal Care and Use Committee at the University of California, Riverside.

3. Hippocampal neuron cultures

Cultures of hippocampal neurons were prepared from embryonic day 15 (E15) or E16 pups. Briefly, hippocampal cells were treated with papain (0.5 mg/ml) and DNase (0.6 µg/ml) for 20 min at 37°C, mechanically dissociated, and then plated on glass coverslips that had been pre-coated with poly-DL ornithine (0.5 mg/ml in borate buffer) and laminin (5 µg/ml in PBS). The hippocampal cells were cultured in Neurobasal medium with 25 µM glutamine, 1% penicillin-streptomycin, and B27 supplement (Invitrogen, Carlsbad, CA), under 5% CO₂/10% O₂ atmosphere at 37°C. Hippocampal neurons were transfected with ptdTomato and pcDNA3-EGFP-cofilin, pcDNA3-EGFP-

cofilinS3A or pcDNA3- EGFP-cofilinS3D plasmids to express tdTomato and wt-Cofilin-GFP, cofilin-S3A-GFP or cofilin-S3D-GFP at 10 days in vitro (DIV) using a calcium phosphate method, as previously described [81].

4. Live imaging

Time-lapse imaging was performed on 14 DIV hippocampal cultures under an inverted fluorescent microscope (model TE2000; Nikon) with a 40x air Fluor objective and monitored by a 12-bit CCD camera (model ORCA-AG; Hamamatsu) using Image-Pro software (Media Cybernetics). During imaging, the cultures were maintained in Hank's solution supplemented with 1.8 mM CaCl₂, 0.45% glucose, and 0.1% BSA at 37°C and 5% CO₂, and images were captured at 3 min intervals for 1 h. Cofilin was visualized by GFP fluorescence and dendritic spines were identified with tdTomato. Briefly, samples were encoded for blind analysis. In each experiment, 2-3 coverslips were analyzed for each condition. At least ten spiny pyramidal neurons were randomly imaged in each group.

5. Development and use of DendritePA software.

The DendritePA is designed in a modular manner with three parts: Dendritic spine segmentation, protein motility extraction, and cofilin-spine shape analysis. A diagram of our workflow is shown in Fig 4.2. DendritePA was written and developed with MATLAB 2016a programming environment. The MATLAB source code, a stand-alone executable version of this algorithm, and supplied test data are available online at <http://vislab.ucr.edu/SOFTWARE/software.php>. DendritePA.m is the main program of the code and requires the following MATLAB toolboxes: Statistics and Machine Learning,

Bioinformatics, Computer Vision System, Image Processing, Mapping, and System Identification. The standalone executable requires the installation of the 64-bit version of MATLAB Runtime R2016a (9.0.1) available at <http://www.mathworks.com/products/compiler/mcr/>.

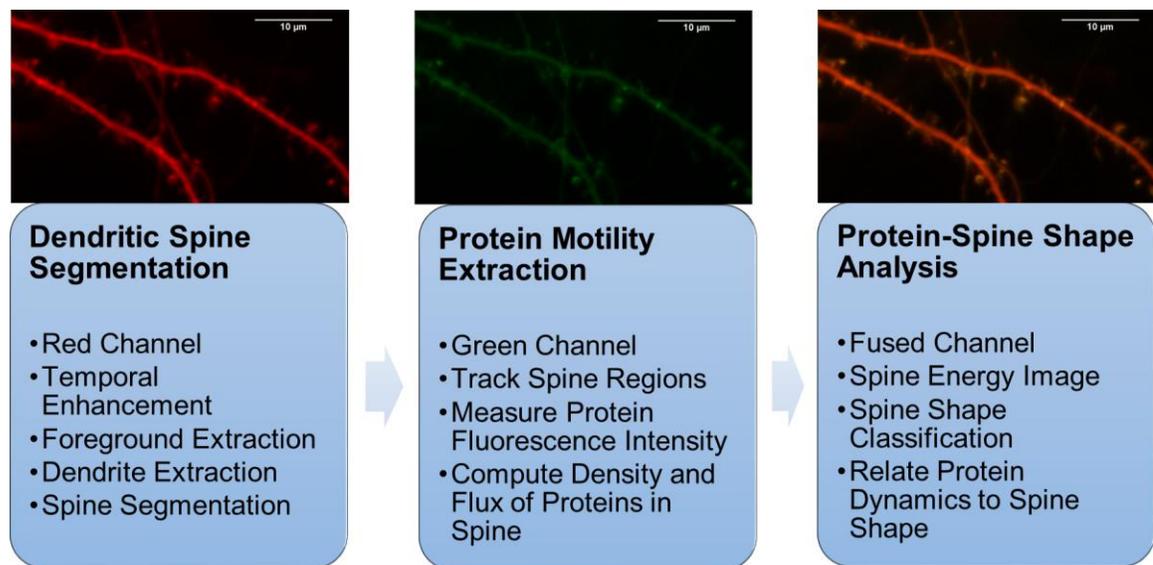


Fig 4.2. System Overview Diagram. DendritePA is designed with three subsystems: Dendritic spine segmentation, Protein Motility Extraction, and Protein-Spine Shape Analysis. Dendritic spine segmentation subsystem uses the red fluorescence channel to extract the foreground, central region, and spines. Protein motility extraction subsystem uses green fluorescence channel to measure cofilin levels and transport in spines. Protein-spine shape analysis subsystem uses both channels and temporal information to relate spine shape with cofilin flow.

6. Dendritic spine segmentation

Dendritic spine segmentation is performed on the red color channel of our fluorescence data. This channel is stained with TdTomato, which fills the cell, providing structural information used for examining the cell morphology. Common issues with fluorescence microscopy include hazy background noise, lack of contrast, and bleaching of intensity over time [108]. The background may auto-fluoresce and structures such as

other dendrites and axons that are out of focus may affect the structures of interest. To account for these issues, it is useful to preprocess a video. Top-hat filtering has been used to reduce background fluorescence [109]. For this work, a top-hat filter using a disk with a radius of 50 pixels was used on each frame. After completing top-hat filtering, a 3 by 3 median filter was also used to reduce noise. Previous methods have used histogram matching to correct for photo-bleaching. This step is important for segmentation of dendritic spines in later frames as well as getting the correct intensity of fluoresced proteins. Every frame after the first was histogram matched using the first frame as a reference [110].

7. Temporal maximum intensity projection

Dendritic spine segmentation begins by estimating the foreground in each frame. The foreground in our case is any pixel brightly illuminated by fluorescent proteins in a dendritic structure. DendritePA starts by computing the maximum intensity projection using all frames in a video. For a video consisting of N_v frames, this temporal maximum intensity projection (TMIP) is defined as follows:

$$T(x, y) = \max_t I(x, y, t), \quad (1)$$

where $I(x, y, t)$ is the image or frame at time t . Since TMIP uses the maximum intensity of a pixel along the time dimension, a TMIP pixel will have a larger intensity if a structure strongly fluoresced at that location for any time in the video. The pixel value will be minimal for any background structures such as dendrites or axons outside of the focal distance. The TMIP is then max-min normalized producing filter whose values will

be used as weights for enhancing the signal-to-noise ratio in each frame. For every frame, the TMIP is multiplied to the image as weights. This image enhance procedure is summarized in the following equation:

$$I'(x, y, t) = \frac{T(x, y) - T_{\min}}{T_{\max} - T_{\min}} * I(x, y, t). \quad (2)$$

By preprocessing with a TMIP, structures that are brighter in the TMIP will be enhanced in each frame, while background structures such as axons or dendrites outside the depth of focus will be suppressed. Since the data in the present frame is utilized, no artifacts will be generated from the bright areas in the TMIP. The TMIP, an original image, and a temporally enhanced image are shown in Fig 4.3. TMIP is only applied for the segmentation step and not utilized in the cofilin analysis step. This is because it is undesirable to modify the intensity in such a way that would change the relative intensity differences between pixels. An initial segmentation of the dendrites and spines can now be computed by using the Otsu's method [29]. This initial segmentation is the foreground which can be used to extract the central region of the dendrite. With the foreground computed, the contours are then acquired by removing all interior pixels of the foreground leaving only the outline.

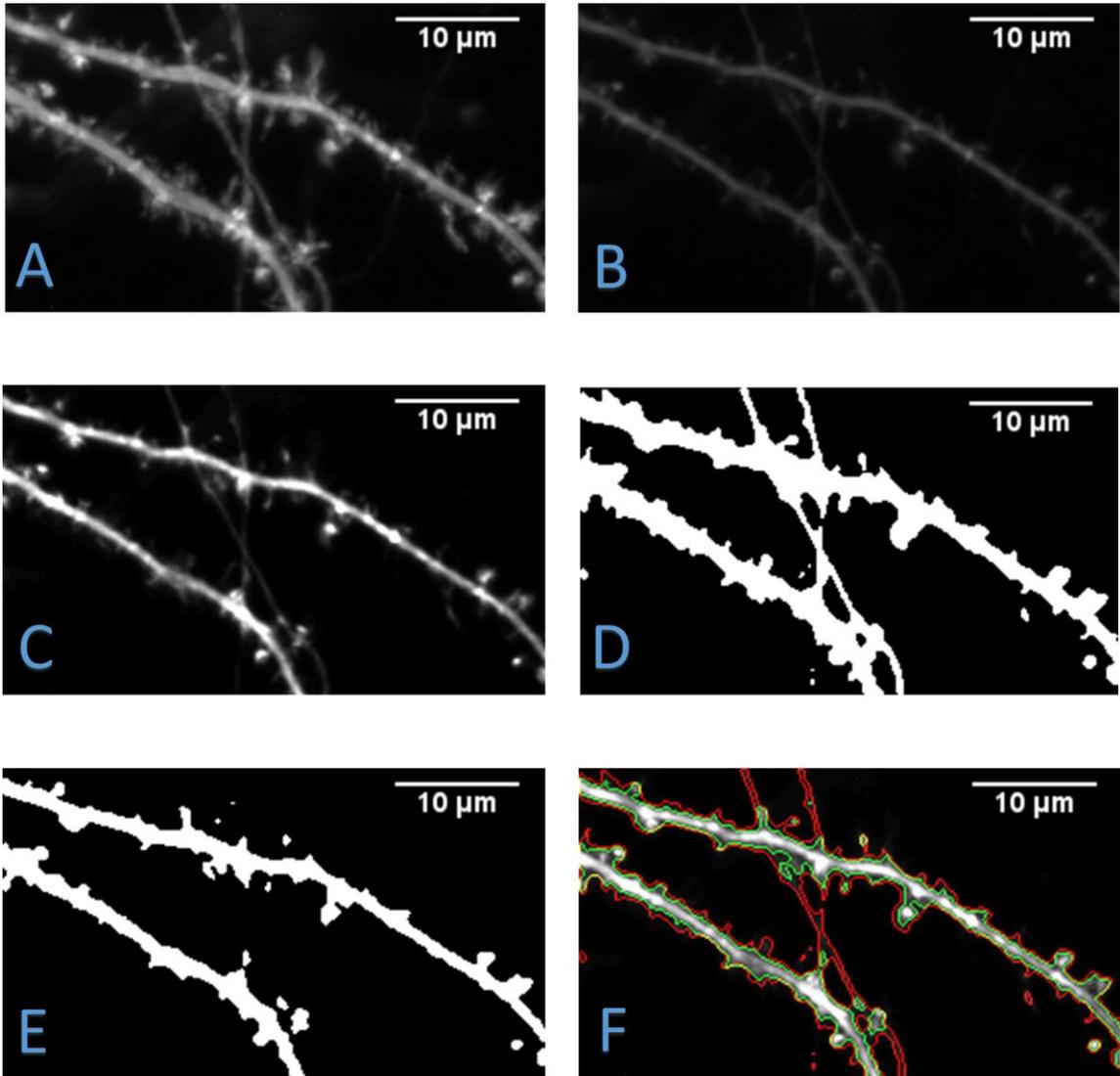


Fig 4.3. Effect of preprocessing video frames. A) Temporal maximum intensity projection (TMIP) computed from an image sequence of dendrites. B) An original frame from the image sequence. C) An enhanced image generated by combining the original frame with the TMIP. D) Extracted foreground segmentation done on the original frame without TMIP. E) Foreground extracted after enhancement with TMIP. F) An overlay of foreground contours with (green outline) and without (red outline) TMIP on enhanced frame. This reveals that TMIP significantly improves segmentation of foreground.

8. Foreground and dendrite segmentation

Upon computing the foreground, we segment the central regions or backbone of dendrites. Past strategies have relied on basic skeletonization of the initial segmentation until only a thin backbone remain. The skeleton of a foreground image (Fig 4.4A) is shown in Fig 4.4B. However, this skeleton does not give the best representation of a dendrite and does not give data regarding the changing width of the dendrite. Precise dendrite information is critical because numerous spine segmentation algorithms are dependent on accurately segmented dendrites. In our previous work [93], we used the method outlined in [105] to find the central region. This method used a modified gradient vector flow (GVF), in which vectors are orientated towards the center of a structure. Starting from every edge pixel, the algorithm follows the path of vectors until it encounters a vector greater than 90 degrees from the current vector. Both these pixels will now be marked as a central dendrite pixel. While this method is an improvement over skeletonization, it does not fully capture the outline of a dendritic spine as it will not detect the outer edges of a dendrite. Another issue is that the algorithm will not detect central regions near large dendrites or crossing structures. This is because the GVF vectors are oriented in a spiral for these regions and will not be 90 degrees from each other, which causes an infinite loop as it traces the vectors.

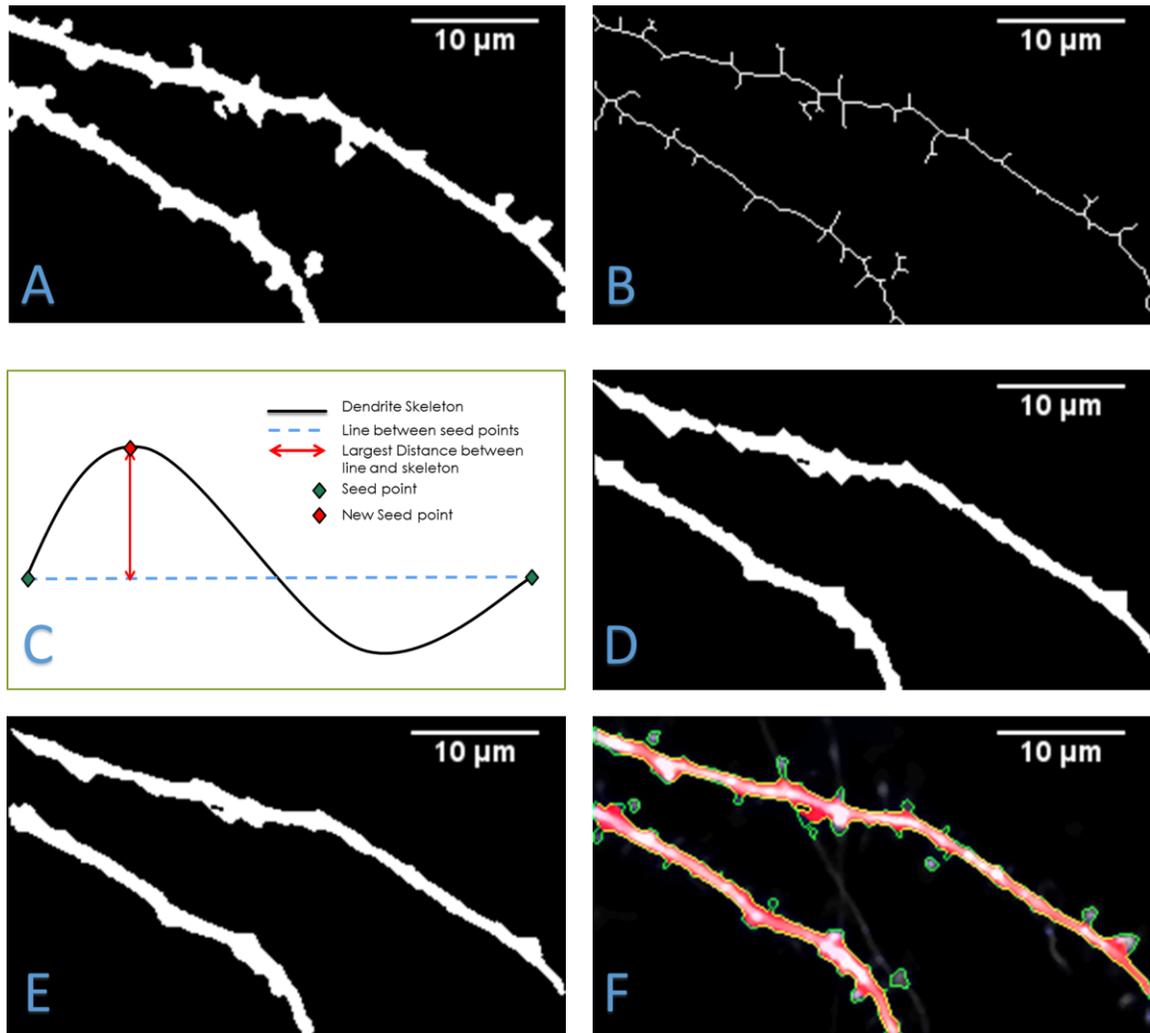


Fig 4.4. Dendrite Segmentation. A) Foreground segmentation of one image. B) Skeletonization of the segmentation done by removing pixels from the peripheries until only an individual pixel remains. C) Diagram of piecewise linear approximation method used to automate dendrite segmentation. D) Dendrite segmentation using convolution kernels. E) Low pass filter output performed after the convolution step. F) Segmented foreground contours (green outline) and final dendrite segmentation (red) overlaid on to the original frame.

For this work, we adapted the convolution kernel method of [104] called 2dSpAn. In this method, a foreground segmentation containing spines and dendrites is computed either with manual thresholding or automatic Otsu's method [29]. The user specifies two points on the dendrite. The angle of flow is computed based on the two points selected.

Based on this angle, a set of two 3 by 3 convolution kernels is selected. Starting from one point, a convolution kernel is continuously applied until it reaches the other point. Next, starting from the opposite point, the complementary convolution kernel is applied until it reaches the first. The intersection of the two regions generated by the kernels is taken as the segmentation of the dendrite.

Because the kernels are chosen by the original angle of flow, this method fails for dendrites with large curvature. To automate this method and to account for large curvatures, we propose the following steps. The foreground segmentation is skeletonized and trimmed. The two endpoints of the dendrite skeleton that are furthest from each other are used as the first two seed points for the kernel method. Additional seed points can now be computed using piecewise linear approximation as shown in Fig 4.4C. A line connecting the two current seed points is computed and the dendrite skeleton pixel that is furthest from this line is evaluated. If the distance between the dendrite skeleton pixel and the closest point on the line is greater than distance d (which is set to 15 pixels for our experiments), then this skeleton pixel will be used as a new seed point. The seed point generation step is repeated for all pairs of seed points until no more seed points can be generated. This allows the dendrite to be computed in a piecewise process, which accounts for large curvature in the dendrite. Once all seed points are found, the convolution kernel method of 2dSpAn is used to generate a dendrite segmentation as shown in Fig 4.4D.

Another issue with 2dSpAn is that for spines with large bases, a portion of the spine will be included in the dendrite segmentation. Also, by breaking the dendrite into

piecewise segments, regions at the seed points will not be segmented perpendicular to the angle of flow. By applying a low pass filter to the dendrite segmentation contour, the method smooths contour of segmentation. This repairs both types of regions by reducing the high-frequency changes in the dendrite segmented contour as seen in Fig 4.4E. Fig 4.4F shows the contour of the foreground (green) and the segmentation of the dendrite (red) overlaid onto the original image.

9. Spine segmentation and declumping

Once a segmentation of the dendrite is obtained, DendritePA begins the spine segmentation process. Using the generated backbone, a distance map is computed [111]. This distance map measures the distance of each foreground pixel to the closest dendrite pixel. Only pixels in the computed foreground contour are considered dendritic spine candidates. Using the inner distance map and the foreground contour, all regional maxima are used as possible dendritic spine detections. These regional maxima correlate with the furthest spine pixels from the central regions of the dendrite and are considered seed points for spine detection. All contour pixels that are closer to the seed point than the closest backbone pixel are considered a part of that spine as shown in Fig 4.5A. The end points of this contour are found and a line is drawn to connect them. This line represents the interface or boundary between the spine and its dendrite. The contour is now filled and will be used as the segmentation of the spine as displayed in Fig 4.5B.

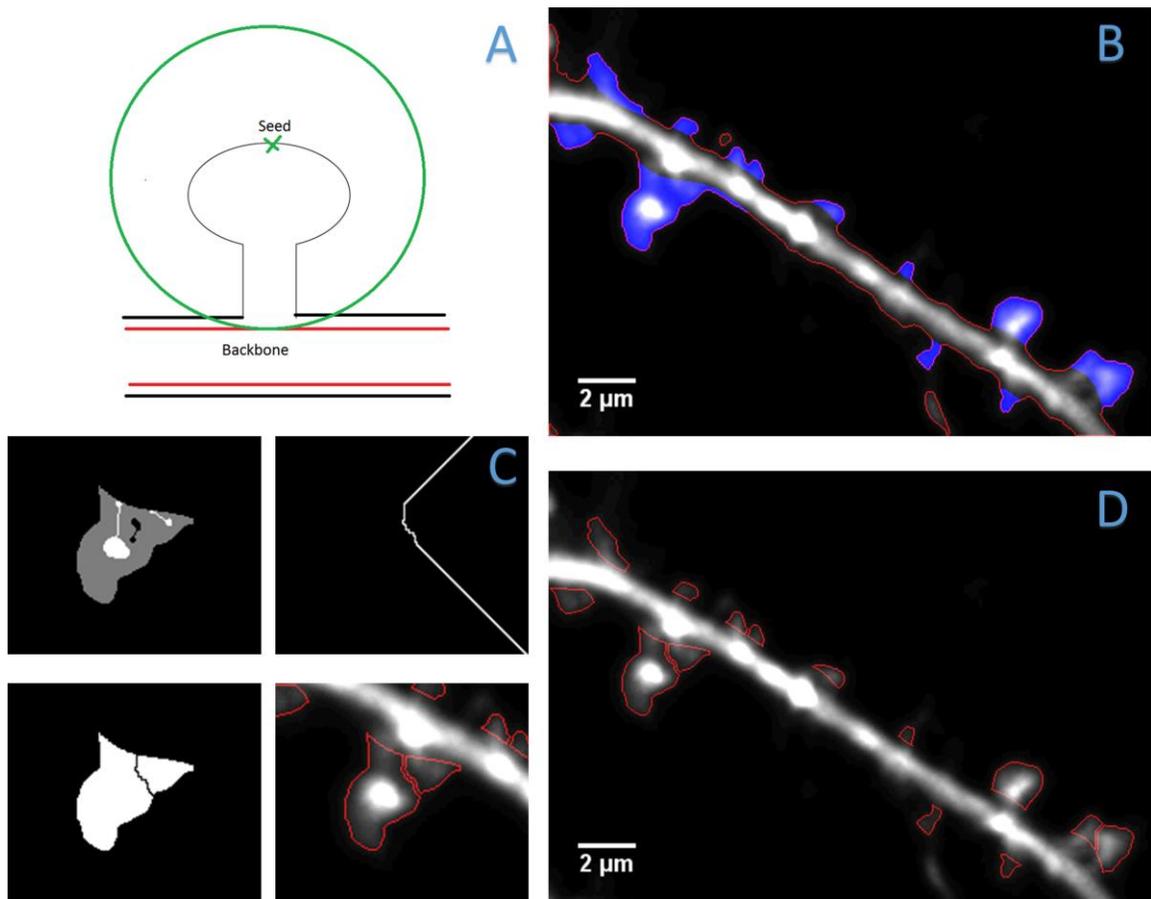


Fig 4.5. Spine Segmentation. A) Diagram of dendritic spine contour extraction using a seed point. This contour is used as the initial segmentation of a dendritic spine. B) Spine segmentation (blue) and foreground contour (red) overlaid onto the image. C) Images of procedure to declump a segmented dendritic spine. Marker-controlled watershed composite image (top left), boundary generated by watershed algorithm (top right), declumped segmentation (bottom left), and contour of declumped segmentations overlaid onto image (bottom right). D) Final spine segmentation displaying split and declumped spines.

While many spines will be properly segmented, some spines may be merged due to their close proximity to each other. Spines that are slightly touching at the base can be split by computing the distance to dendrite of every contour pixel. We trace these values and detect if a regional minimal distance is between two regional maximal distances. If the minimal distance is less than half of either maxima, draw a dividing line from the

minima to closest dendrite pixel. For spines with severe overlap, we propose a marker control watershed method to declump the dendritic spines. For each spine region, examine the intensity and detect regional minima and maxima. If the centroid of two maxima are not parallel to the dendrite, connect the maxima with a line. This line must not cross any minimal regions. The same is done for any pair of regional minima as long as the line does not cross any maximal regions. Using the spine segmentation, connected maxima, and connected minima, a composite image is generated to serve as the markers for the watershed method. Background and regional minima are set to a low value of zero, regional maxima are set to a high value of 255 and the rest of the spine segmentation is set to a middle value. The composite image is inverted and the watershed will begin filling at the regional maxima. As the algorithm continues, the middle values will begin to form the watershed boundaries while taking the regional minima into account. The regional maxima were connected to reduce the number of watershed boundaries, while connected regional minima were used to aid in the shape of the boundary. The generated watershed boundary is applied to the spine mask, separating the segmentation in two. The declumping process is illustrated in Fig 4.5C and the final segmentations after splitting and declumping are shown in Fig 4.5D.

10. Cofilin motility extraction

Cofilin motility extraction is performed exclusively on the green channel of our fluorescence microscopy videos. The channel shows GFP-labeled cofilin, (wt)-Cofilin-GFP, which allows for cofilin density to be visually analyzed. In order to analyze the

motility of cofilin, at least two adjacent frames are needed. Because individual cofilin molecules exist at the subpixel level and cannot be resolved at our (40x) magnification, they cannot be tracked individually. Since the green intensity channel tracks the GFP-tagged cofilin proteins, the intensity is directly proportional to the amount of cofilin in the pixel. While the visual changes in cofilin density are difficult to examine by eye, the DendritePA is able to analyze this data by using spatiotemporal information in the red structural channel. Utilizing the previous dendritic spine segmentations, DendritePA can estimate the amount of cofilin contained in these structures. The framework starts by performing data association of segmented spines in neighboring frames to produce dendritic spine tracks. Association is performed by choosing the segmentation with the largest percentage of overlap with an existing track. The ratio of overlap is sufficient as the spines are attached to a fixed location on the dendrite. Most of the movement in dendritic spines is attributed to sway and shape change, whereas the sway of the dendrite itself is negligible. A new track may be created if the segmentation has little or no overlap with a track. Dendritic spines may not be detected in every frame as they may shrink into the dendrite or sway in and out of the z-axis, thereby going out of view. Because of this, a spine segmentation may be associated with any existing track if there is overlap with the last known location.

11. Cofilin flux

Since we are concerned with the flow of cofilin through sections of a cell, it is advantageous to relate this concept to fluid dynamics. The differential form of the continuity equation in fluid mechanics is written as:

$$\frac{\delta\rho}{\delta t} + \text{div}(f) = s, \text{ such that } f = \rho v, \quad (3)$$

where ρ is the density of the fluid particles, f is the flux of the fluid through a boundary, v is the velocity, and s is a source term. To find the change in the amount of cofilin in the spine, we want to solve for the $\text{div}(f)$ which is the “flux density” and represents the amount of flux entering or leaving a point. In fluorescence microscopy, fluorescence intensity levels are proportional to the amount of tagged proteins in a region. This leads to the follow proportionality formula:

$$\rho \propto p_i(t) = \sum_{(x,y) \in S(t)} i(x,y,t), \quad (4)$$

where $p_i(t)$ is the integrated density in spine $S(t)$ at time t , and $i(x,y,t)$ is the intensity of pixel (x,y) . Using integrated density accounts for changing spine area and is commonly used for analyzing fluorescence microscopy images [112]. It can be assumed that cofilin neither produced nor consumed in the spine. This allows the source term s to be set to zero for all calculations of flux. Also as there is only one boundary between the spine and the dendrite, cofilin flux must be either in or out of this boundary. Solving for $\text{div}(f)$, the continuity equation becomes:

$$\text{div}(f) \propto -\frac{\delta\rho_i(t)}{\delta t}. \quad (5)$$

Cofilin flux and density can now be compared by examining the intensity levels in the green cofilin channel. An increase in the intensity in a spine represents an increase in the cofilin density at the spine and a flux of cofilin into the spine. Conversely, a decrease in the intensity represents a decrease in cofilin density as well as a flux of cofilin out of the spine.

12. Cofilin-spine shape analysis

After estimating the motility of cofilin and dendritic spines morphology, we correlate their effect on one another. The initial step is to automatically classify the shape of the dendritic spine using machine learning. To do this, we obtain the spine energy image representation of a spine. Generating the SEI starts by cropping the binary segmentation of each spine in a track. All binary spines images are then transformed so that the spine-dendrite boundary is aligned with the x-axis. These aligned cropped images are resized into 10x10 images. Given the registered binary images $B_t(x,y)$ at time t for a spine track of N frames, the spine energy image can be computed as follows:

$$S(x, y) = \frac{1}{N} \sum_{t=1}^N B_t(x, y). \quad (6)$$

An example of the aligned binary images and SEI are shown in Fig 4.6. Since the SEI images are 10x10 pixels, the dimension of the feature vector is 100, which leads to a problem with the curse of dimensionality. This feature vector needs to be reduced while

minimizing the loss of information. Recent studies [113], [114] have shown that local binary pattern applied to Gait Energy Image can reduce dimensionality. By applying uniform LBP [115] to SEI, the feature vector is reduced to 59 dimensions.

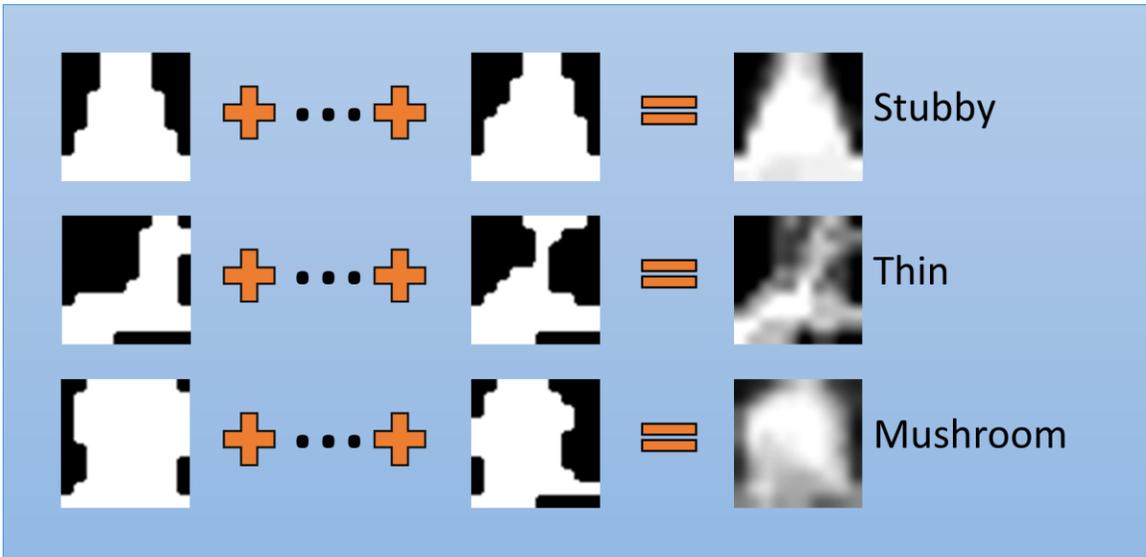


Fig 4.6. Spine energy image. Examples of aligned binary images and spine energy image for each class stubby, thin, and mushroom shaped dendritic spines. The aligned and resized spine segmentations of a single spine track are combined to produce a single spine energy image (images in the furthest right column). This image is a representation of the motion of the spine for the image sequence.

While SEI has been shown in our previous work [93] to classify spine tracks well, it cannot classify individual spine segmentations by itself. To classify individual spine, additional features must be used. Before being passed to a classifier the SEI feature vector is combined with the area, height, width, and average intensity of each spine. These features were used to train three classifiers: (1) discriminant analysis (DA) classifier, (2) k-nearest neighbor (KNN), and (3) error-correcting output codes (ECOC) multiclass model. The discriminant analysis classifier trains by fitting a Gaussian distribution to each class. New data is compared to each class distribution and assigned

to the class with the lowest misclassification cost [71]. A k-nearest neighbor classifier labels a new observation by comparing it to the k-nearest training samples in a multi-dimensional space [37]. $k = 5$ was used for all experiments. ECOC is a classifier reduces a multiclass classification problem into a set of L binary classifiers. DendritePA uses a support vector machines (SVM) classifier for every pair of classes, $L = 3$. A new observation is assigned to the class that minimizes the losses of the L binary learners [72].

D. Results

Our data set for segmentation consists of seven live fluorescence videos, which contain 3428 spines across all analyzed frames. TdTomato was used to label the entire structure of the cell (dendritic structural information) and wild type (wt)-Cofilin-GFP was used to label cofilin (cofilin distribution information). Videos varied in length from 39 frames and were collected at intervals varying from every 30 seconds to 60 seconds over approximately 20 minutes. All videos were collected at 40x magnification and were 128 by 128 in image resolution. To improve image quality, all frames were resized to 512 by 512 with bicubic interpolation [116]. Most of the segmentation algorithm parameters were kept constant for each video, except a multiplier for Otsu's computed threshold which ranged from 0.2 to 0.4.

Dendrite segmentation was validated using the first frame of each of the seven videos. An expert in the field labeled each frame by selecting only regions that are a part of the dendrite and not the spines. Tests were performed using a GVF generated

backbone, 2dSpAn’s convolution algorithm with piecewise linear approximation, and DendritePA after using a low pass filter. Results are obtained on a pixel detection basis and they are shown in Table 4.1. While DendritePA has a slightly worse precision, it has a much better recall.

Method	Precision	Recall	True Positive	False Positive	False Negative
GVF Dendrite	75.22%	4.21%	4.21%	1.39%	95.79%
Piecewise 2dSpAn	75.97%	84.33%	84.33%	26.66%	15.67%
DendritePA	73.23%	90.32%	90.33%	33.01%	9.67%

Segmentation results for dendrites using GVF, 2dSpAn, and DendritePA.

<https://doi.org/10.1371/journal.pone.0182958.t001>

Table 4.1. Dendrite segmentation results. Segmentation results for dendrites using GVF, 2dSpAn, and DendritePA.

Ground-truth for dendritic spine was created by labeling every spine in every frame manually by two experts in the field. In order to reduce user bias, the decision tree shown in Fig 4.7 was used as a guideline during ground-truth labeling. For comparison, our spine detection method is tested against the NeuronIQ software [117]. NeuronIQ was chosen because it is a fully automated dendritic spine segmentation software. 2dSpAn and NeuronStudio both required series of manual inputs for each frame, making them inappropriate for comparisons. While NeuronIQ was able to segment without preprocessing, using a TMIP to preprocess the videos improved segmentation results. A spine is considered detected if the segmentation has at least 50% overlap with the ground-truth data. Table 4.2 shows the results of segmentation for DendritePA, NeuronIQ with TMIP preprocessing, and NeuronIQ alone. NeuronIQ alone seems to produce many false positives. By using TMIP to preprocess the videos, the SNR is improved allowing NeuronIQ to perform better. DendritePA was able to achieve comparable recall, but could

greatly improve precision as it has nearly half as many false positives. Because the classification of spines is dependent on the quality of their segmentations, only true positives were passed to the tracking and classification steps.

DendritePA was also run without bicubic interpolation and compared to ground-truth sub-sampled at 128 by 128 resolution. Using the original resolution (128 by 128), DendritePA with TMIP had a precision of 59.00% and recall of 16.09%. These results are noticeably worse when compared to DendritePA with bicubic interpolation. To show that bicubic interpolation does not have a major effect on cofilin analysis, we computed the average intensity of cofilin in dendritic spines (green channel) for both 128 by 128 and 512 by 512. The average values for seven wild-type videos were 0.78 ± 0.13 for the original resolution and 0.76 ± 0.13 for the bicubic interpolation. The values were very similar between the two scales of resolution, and the standard deviation was identical. Also, because we are only interested in relative change of intensity compared to the previous frame, we do not require exact intensity values if the relationship between frames is maintained.

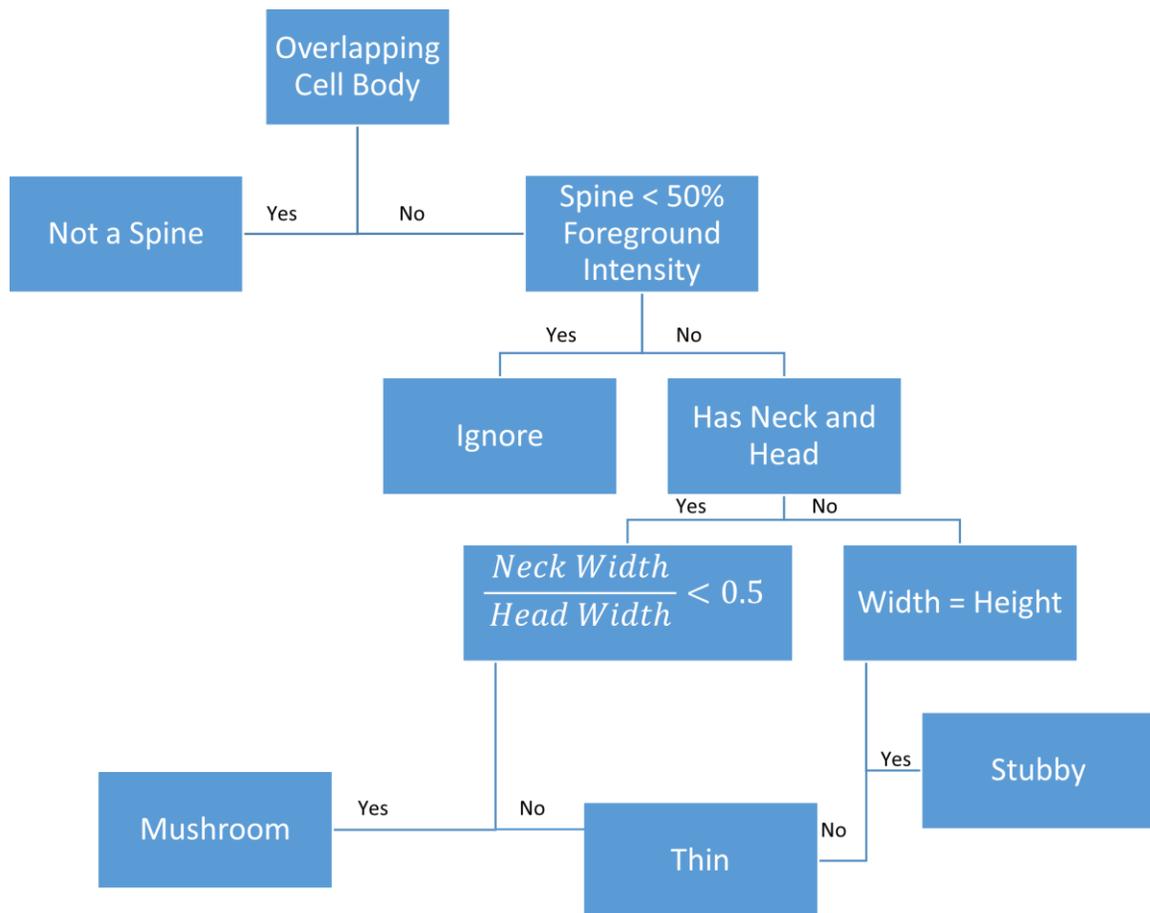


Fig 4.7 Ground-truth decision tree. Decision tree for generating ground truth of dendritic spine segmentation and classification. Spines in images were manually labelled (to be used as ground truth), using this decision tree to reduce bias and improve reproducibility. Spines were marked if they satisfied the conditions of the stubby, thin, or mushroom shapes.

Method	Precision	Recall	True Positive	False Positive	False Negative
NeuronIQ	28.13%	61.26%	61.26%	156.45%	38.74%
NeuronIQ with TMIP	53.69%	71.17%	71.18%	61.38%	28.82%
DendritePA	60.28%	61.93%	61.93%	40.64%	38.07%
DendritePA with TMIP	63.48%	67.24%	67.24%	38.68%	32.76%

Segmentation results for DendritePA, NeuronIQ with TMIP, and NeuronIQ alone.

<https://doi.org/10.1371/journal.pone.0182958.t002>

Table 4.2. Spine segmentation results. Segmentation results for DendritePA, NeuronIQ with TMIP, and NeuronIQ alone.

Classification of spines was done with 10-fold cross validation by partitioning the data into 10 equal sets. Each set is used as the test set once while the other 9 are used for training. The experiments were repeated 20 times by randomly shuffling the dataset so that the sets are randomly generated. Each spine was classified as stubby, thin or mushroom. While DendritePA is able to classify other phenotypes such as branched dendritic spines, it requires a training library with sufficient examples of the class. Because our data did not have enough branched spines, we focused our study on the three most common phenotypes [88], [118], [119]. Results are compared to a decision tree method that uses spine height and width features. Table 4.3 displays the classification results. The three classifiers using the proposed features outperform the traditional decision tree method by over 20%. The decision tree method was sensitive to small measurement errors at this resolution, while our proposed method is robust due to the spatiotemporal information in the spine energy images. While the classification rate of the classifiers performs well with all videos, a subset of videos was used to build the classifier used in later sections. Table 4.4 displays a confusion matrix of the three classifiers for videos 1-7 and Fig 4.8 shows the receiver operating characteristic curve (ROC) of the three classifiers for the same videos. ROC curves are useful in assessing the performance of an algorithm's ability to detect an object. The larger the area under the curve (AUC), the better the classifier is at classifying the specific spine shape. From the Table 4.3 and the ROC plots, KNN slightly out performs the ECOC and DA classifiers.

Classifier Results	DA	KNN	ECOC	Tree
Video 1	75.71 ± 0.47	81.35 ± 0.72	79.79 ± 0.90	52.19 ± 0.00
Video 2	59.90 ± 2.94	59.4 ± 3.32	54.10 ± 3.14	50.00 ± 0.00
Video 3	77.22 ± 0.82	75.36 ± 0.80	76.93 ± 0.89	63.79 ± 0.00
Video 4	81.97 ± 0.74	84.05 ± 0.69	81.33 ± 0.80	55.81 ± 0.00
Video 5	93.00 ± 0.72	98.26 ± 0.00	98.26 ± 0.00	79.52 ± 0.00
Video 6	76.34 ± 0.80	75.84 ± 0.75	74.55 ± 0.47	69.18 ± 0.00
Video 7	83.68 ± 1.07	90.44 ± 0.69	84.03 ± 0.43	58.49 ± 0.00
Videos 1–7	76.40 ± 0.29	80.61 ± 0.41	76.79 ± 0.33	62.10 ± 0.00

Classification results on ROIs 1–7 using DA, KNN, ECOC, and manual decision tree.

<https://doi.org/10.1371/journal.pone.0182958.t003>

Table 4.3. Classification results. Classification results on ROIs 1-7 using DA, KNN, ECOC, and manual decision tree.

Real Class	Predicted Class			
	DA Classifier	<i>Stubby</i>	<i>Thin</i>	<i>Mushroom</i>
<i>Stubby</i>		499	49	50
<i>Thin</i>		62	192	57
<i>Mushroom</i>		73	52	507
KNN Classifier	<i>Stubby</i>	508	40	50
	<i>Thin</i>	62	192	57
	<i>Mushroom</i>	55	44	533
	ECOC Classifier	<i>Stubby</i>	<i>Thin</i>	<i>Mushroom</i>
<i>Stubby</i>	526	27	45	
<i>Thin</i>	84	174	53	
<i>Mushroom</i>	98	50	484	

Confusion Matrices for discriminant analysis, k-nearest neighbor, and error-correcting output codes.

<https://doi.org/10.1371/journal.pone.0182958.t004>

Table 4.4. Confusion Matrices. Confusion Matrices for A) Discriminant analysis, B) K-nearest neighbor, and C) Error-correcting output codes.

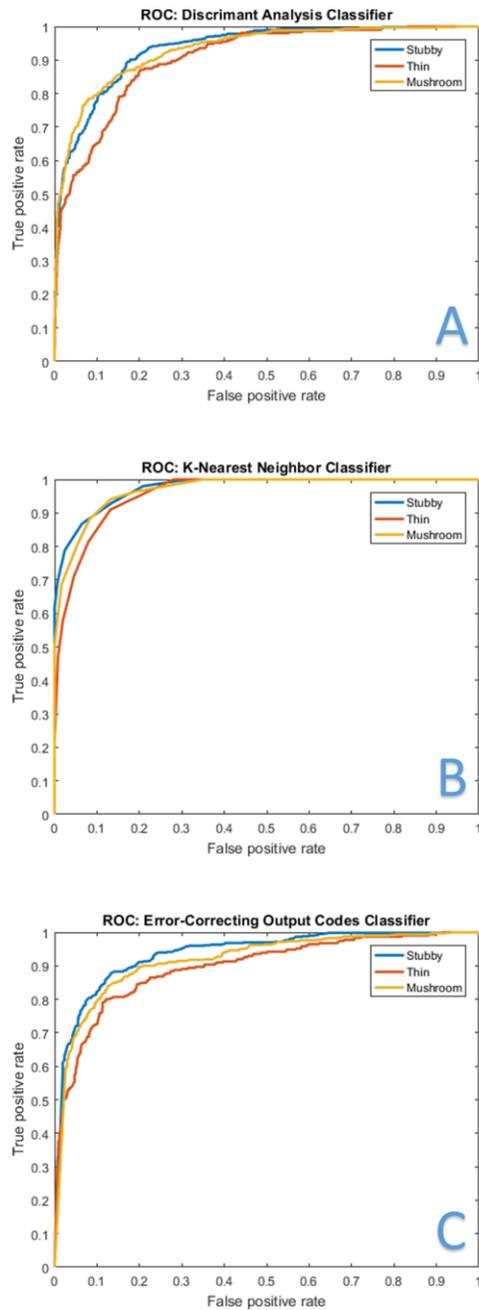


Fig 4.8. ROC Plots. Receiver operating characteristic (ROC) curves for stubby (blue curves), thin (red curves), and mushroom (yellow curves) shaped spines. The ROC plots illustrate the ability of each classifier to distinguish the specific class when varying a discrimination threshold. A larger area under the curves (AUC) represents a better classifier. A) Discriminant analysis (DA) classifier. B) K-nearest neighbor (KNN) classifier. C) Error-correcting output codes (ECOC) classifier. K-nearest neighbor classifier had the largest AUC, proving to be the most accurate.

Upon closer examination of the SEI as shown in Fig 4.9, the SEI can be sorted visually into their mode spine class. Stubby spines tend to have a flat uniform base and a sharp apex. Thins have smaller middle regions and are blurry do to swaying over time. Mushrooms are round and have smaller bases. Fig 4.9 also displays SEIs that were misclassified. SEI seem to be commonly misclassified as stubby if they are triangular, as thin if they blurry or have an unusual base, and as mushroom if they are more circular.

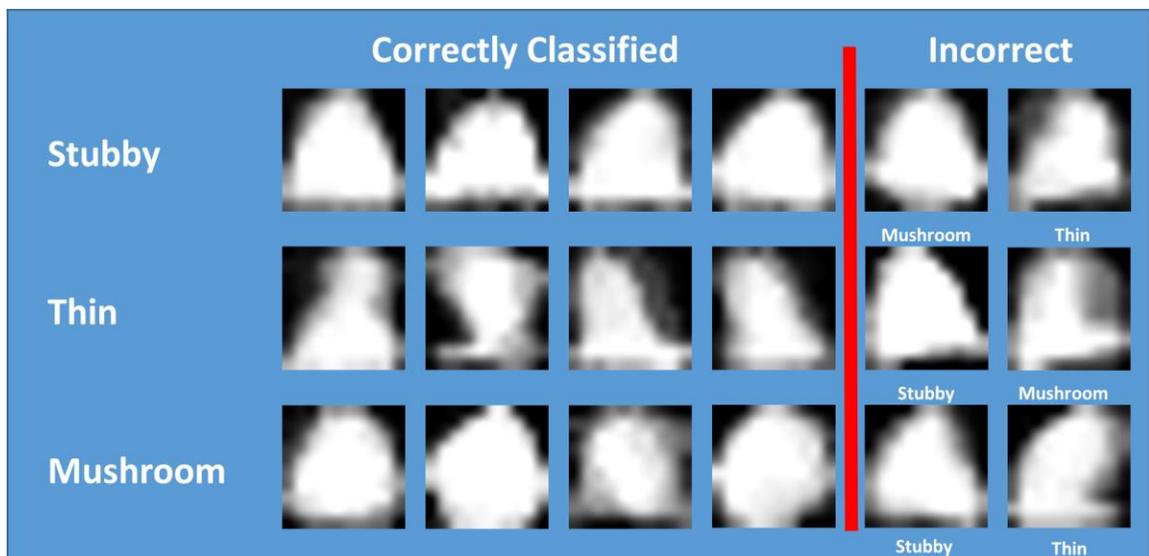


Fig 4.9. SEI classification examples. Each row displays spine energy images (SEI) that were classified by DendritePA as stubby, mushroom, or thin. The last two columns represent misclassified SEI, where the actual class of the spine track is shown below each SEI.

1. Cofilin conditions and spine shape

Three videos of S3A and three videos of S3D were collected using fluorescence microscopy. Each video has 40 frames and were captured at 40x magnification with 128 by 128 pixels. Like the wild-type videos, all S3A and S3D frames were resized to 512 by 512 using bicubic interpolation. TdTomato was used to label the actin cytoskeleton that

make up the structure of the spine and CofilinS3A or CofilinS3D was used to label cofilin proteins. Visual observations of S3A show a relatively even distribution of stubby, thin and mushroom shaped spines. The density of spines were uneven across the video with some regions very dense and others less dense. For S3D, most spines were mushroom or stubby and had mild spine density.

DendritePA was used to segment and analyze videos showing the dendrite spines containing wt-cofilin, cofilinS3A, and cofilinS3D. A discriminant analysis classifier built with the ground truth was used to classify the data. Fig 4.10A shows the percentage of each spine class for the wild-type cofilin, cofilinS3A, and cofilinS3D videos. These percentages were significantly different ($P < 0.0001$ using Chi-squared test) and show that dendritic spine shape can be altered by introducing exogenous cofilin. For the case of the wild-type condition, most spines are stubby and mushroom, which indicate that spines are mostly mature for this condition. With the activation of cofilin in the cofilinS3A condition, there is a noticeable increase of thin spines and a decrease in mushrooms. This is consistent with the expectation that cofilin breaks down the F-actin in spines. With more cofilin being active the spines become less mature. Interestingly, we see that in the cofilinS3D videos there is a significant decrease in the number thin spines, while the ratio of stubby and mushroom spines remains the same. This indicates that by suppressing cofilin activation, it may be possible to induce more mature spines. Fig 4.10B-4.10D shows the percent of cofilin flux for each spine class across the three cofilin mutants. For all three spine classes, the ratio of cofilin flux moving in (cofilin density increases in spine, shown in green) or out of spines (cofilin density decreases in spine, shown in red) is

consistent for all conditions and did not show significant difference (thin p-value = 0.84, stubby p-value = 0.60, mushroom p-value = 0.89). This indicates that the type of exogenous cofilin has little effect on the flow of cofilin. Also, because the ratio of in and out flux is constant within a spine class regardless of cofilin condition, cofilin flux has less importance in spine type than cofilin activation. In the time frame we examined (20 minutes), we did not notice a difference in the rate in which spine shape changed based on cofilin type. Although this was not our focus, it may be of interest to study in the future with longer videos.

Although endogenous cofilin is present in all three groups, the use of different exogenous cofilin is the only variable that is changed between the conditions. These exogenously expressed mutant forms of cofilin will affect spine morphology by competing with endogenous cofilin for binding to several actin proteins and enzymes. This is done to ensure that changes are due to only one alteration in the experimental setup. Because plasmid transfection efficiency is a concern, we measured the mean expression levels in all videos. The mean fluorescence intensity of cofilin S3A-GFP was 20.66 ± 1.20 , cofilinS3D-GFP was 38.06 ± 12.54 , and wt-cofilin was 23.29 ± 7.94 . The groups were shown to be not significantly different when compared using one-way ANOVA followed by Tukey's multiple-comparison post-tests.

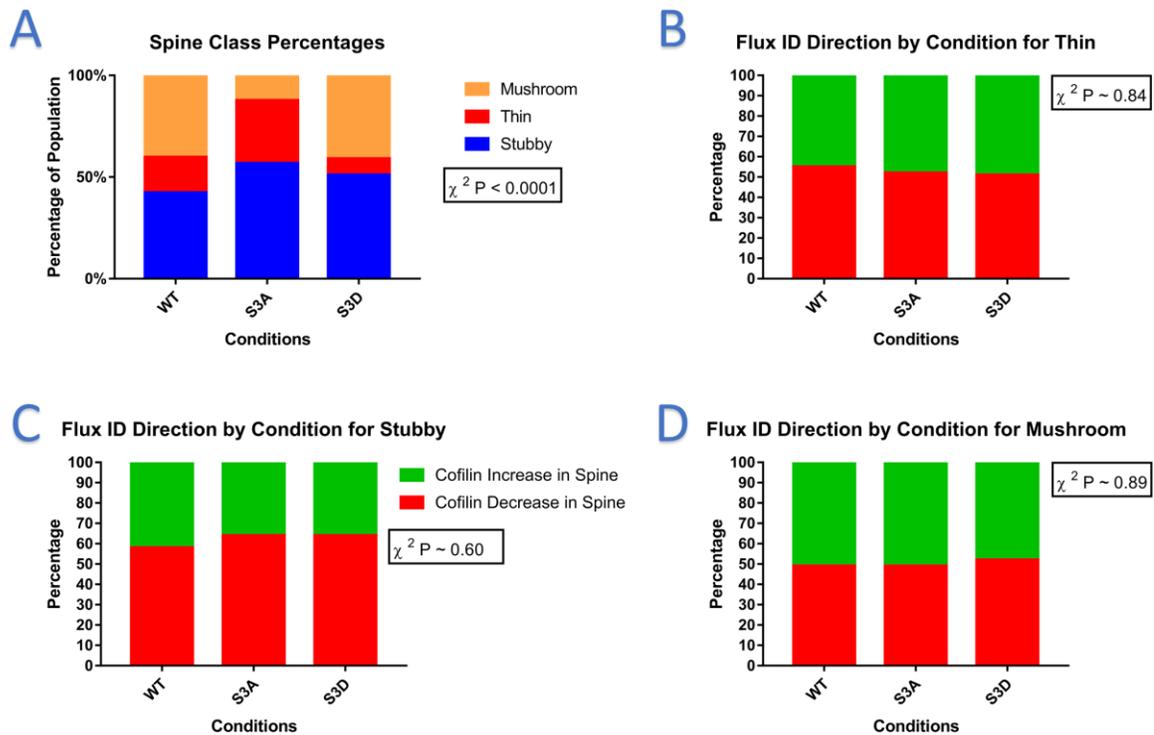


Fig 4.10. Cofilin-Spine Graphs. A) Bar graph of the percentage of detected spines classified as mushroom (orange), thin (red), stubby (blue) for each condition: WT, S3A, and S3D. The percentages for each condition were significantly different ($P < 0.0001$ using Chi-squared test). B-D) Bar graphs of the flux direction computed with integrated density for B) Thin, C) stubby, and D) mushroom spines. For all cases, the percentages were not significantly different using Chi-squared analysis (thin p-value = 0.84, stubby p-value = 0.60, mushroom p-value = 0.89), indicating that cofilin flux has less importance on spine type.

Chapter 5 Conclusions

Due to developing technologies, cell biology has become a rapidly advancing and complex field of research. Managing the amount of data available to researcher can be daunting. While the introduction of bioinformatics tools and software to cell biology applications have greatly reduced the difficulty, there is still an urgent need to develop advance techniques, algorithms, and technology to make full use of the data. In this work we have presented three video bioinformatics software packages that examine various types of live microscopy data.

In chapter 2, we introduced StemCellQC, video bioinformatics toolkit for analyzing cell processes, evaluating cell quality, and discovering biomarkers. It is designed for use with pluripotent stem cell colonies in culture and is adaptable to other cell types. It can be used retrospectively or on-the-fly to solve numerous problems. There are at least four applications for StemCellQC. *First*, core facilities that culture pluripotent cells for distribution to research labs could monitor cell quality using non-invasive morphological tools to guarantee that distributed cells meet an acceptable uniform standard from day-to-day. This is especially important when the results of a research study may ultimately affect a patient's health. *Second*, StemCellQC can serve as a quality control tool in future clinics that deliver therapies based on pluripotent stem cells. Such clinics will need to maintain and differentiate cells that meet future FDA criteria for transplantation to patients. A record of cell quality produced by StemCellQC would be an important part of a patient's medical record and could be mined after cell transfer to

patients to better understand those features that work best for patient treatment. *Third*, hPSC can differentiate into specific cell types that can be used for studying genetic disorders, such as Huntington’s disease [120]. StemCellQC can monitor the behavior of cells/colonies in disease-in-a-dish models to determine how cells respond to drug treatments [121]. *Fourth*, StemCellQC could be used in laboratories that perform drug testing or that monitor chemical toxicity. Multiplexing data enhances the discovery of toxicants and biomarkers. hESC provide an excellent model for prenatal development, a process that cannot be studied experimentally in humans [24] and which is generally sensitive to environmental chemicals [122].

We are currently using StemCellQC with other pluripotent cell types and experimental conditions and found that it performed very well. We have found clear cut effects on processes such as growth, motility, death and morphology using StemCellQC with cells grown in optimal and suboptimal media, indicating StemCellQC will be useful for recognizing culture conditions that are not satisfactory. As more treatments are used, we anticipate that other processes or effects may be observed. In the future, StemCellQC software can be enhanced by adapting it to single cells and including features that correlate to cell processes such as stress, differentiation, and pluripotency. More classifiers can be added, and additional biomarkers will likely be discovered with new applications of the software.

In chapter 3, we present a machine learning software for the 3D reconstruction of phase contrast images called PhaseR4D. By using fluorescence data as the ground truth,

the system was able to train various regression models to compute a depth map. From this depth map, a 3D reconstruction can be generated which can detect objects that are unlabeled in fluorescence. Unlike other 3D phase contrast methods, the proposed system does not require custom hardware. Also, unlike other methods, our system only requires one angle to be imaged and it can be used by any research group with a microscopy system capable of imaging in both phase and fluorescence. Once trained, PhaseR4D may be used for 3D reconstruction phase contrast volume without the corresponding fluorescent volume. This is highly beneficial into showing 3D phase contrast images over time, as cells do not need to be fixed for fluorescent labeling. In addition, 3D features are shown to be capable of automatically classifying biological conditions such as toxicological treatments inducing alterations in cell morphology and behavior.

Toxicological screening using animal systems are considered unethical and highly regulated in some countries. Therefore, non-invasive evaluation of cancer-causing agents (environmental toxicants such as tobacco, etc.) or anticancer drugs on cells in vitro is beneficial. Expanding feature profiling to 3D features can improve pattern recognition algorithms. Also, it can aid in the detection of morphological changes not visible in 2D, such as height, volume, and shape index, and functional features such as degree of cell-to-cell contacts. PhaseR4D has the ability to monitor the progression of 3D cultures over time, and can be applied to automatically classify experimental conditions, or ensure culture quality. In conclusion, our open-access software can advance science on many fronts, facilitating the paradigm shift from 2D to 4D (3D over time) imaging and analytical methods.

In chapter 4, we present a pattern recognition software called DendritePA to analyze protein trafficking in neuronal fluorescence microscopy videos. Using spatiotemporal information, the DendritePA is able to enhance low contrast/low resolution images by computing a temporal maximum intensity projection which is used to improve the signal-to-noise ratio in every frame. Dendrite spines were automatically segmented using an improved kernel convolution method. Temporal dynamics of spines were used to generate a spine energy image, which is useful in classifying different spine shapes. Multiple classifiers were used to classify individual spine segmentations as stubby, thin, and mushroom. Lastly, we were able to estimate cofilin flux patterns and correlate them with the changing spine morphology over time. Mushroom/stubby spine shapes are recognized as mature/stable spines, whereas thin spines are classified as immature/unstable. By examining S3A and S3D conditions of cofilin, our data suggests that the level of activation of cofilin greatly affects the shape of spines. Highly activated cofilin seems to lead to structural instability of the spines. This is consistent with the actin-severing/remodeling function of cofilin.

Future advancements of video bioinformatics of living cells would include improved segmentation and visualization of cellular structures and processes. Better segmentation and feature extraction would greatly improve confidence in classification, regression, and prediction results. This is applicable at different scales including colonies, individual cells, and proteins. Another avenue of research could focus on developing algorithms related to data of high spatial and/or temporal dimensions. Such data would include 3D, time-lapse, and microscopy fusion. Each provides unique information not

present in individual microscopy images. 3D can provide additional structural information, time-lapse can provide details of dynamic biological events, and fusing multiple microscopy datasets can help analyze the correlation between separate biological processes. Lastly, combining all three data-types could provide a more complete picture of cellular dynamics and would be highly advantageous to modern researchers.

References

- [1] N. M. Luscombe, D. Greenbaum, and M. Gerstein, “What is bioinformatics? An introduction and overview,” *Yearb. Med. Inform.*, vol. 10, no. 1, pp. 83–100, Aug. 2001.
- [2] J. Fox, “What is Bioinformatics?,” *The Science Creative Quarterly*, 2006. [Online]. Available: bioinformatics.ubc.ca/about/what_is_bioinformatics/.
- [3] J. Xiong, *Essential bioinformatics*. 2006.
- [4] B. Bhanu and P. Talbot, *Video Bioinformatics*, vol. 22. Cham: Springer International Publishing, 2015.
- [5] S. Lin, S. Fonteno, S. Satish, B. Bhanu, and P. Talbot, “Video Bioinformatics Analysis of Human Embryonic Stem Cell Colony Growth,” *J. Vis. Exp.*, no. 39, May 2010.
- [6] V. Tabar and L. Studer, “Pluripotent stem cells in regenerative medicine: challenges and recent progress,” *Nat. Rev. Genet.*, vol. 15, no. 2, pp. 82–92, Feb. 2014.
- [7] S. Lin and P. Talbot, “Mouse and Human Embryonic Stem Cells: Can They Improve Human Health by Preventing Disease?,” *Curr. Top. Med. Chem.*, vol. 11, no. 13, pp. 1638–1652, Jul. 2011.
- [8] Z. Zhu and D. Huangfu, “Human pluripotent stem cells: an emerging model in developmental biology,” *Development*, vol. 140, no. 4, pp. 705–717, Feb. 2013.
- [9] K. G. Chen, B. S. Mallon, R. D. G. McKay, and P. G. Robey, “Human Pluripotent Stem Cell Culture: Considerations for Maintenance, Expansion, and Therapeutics,” *Cell Stem Cell*, vol. 14, no. 1, pp. 13–26, Jan. 2014.
- [10] S. P. Sheehy, F. Pasqualini, A. Grosberg, S. J. Park, Y. Aratyn-Schaus, and K. K. Parker, “Quality Metrics for Stem Cell-Derived Cardiac Myocytes,” *Stem Cell Reports*, vol. 2, no. 3, pp. 282–294, Mar. 2014.
- [11] S. Lin, S. Fonteno, J.-H. Weng, and P. Talbot, “Comparison of the Toxicity of Smoke from Conventional and Harm Reduction Cigarettes Using Human Embryonic Stem Cells,” *Toxicol. Sci.*, vol. 118, no. 1, pp. 202–212, Nov. 2010.
- [12] B. X. Guan, B. Bhanu, P. Talbot, and S. Lin, “Automated Human Embryonic Stem Cell Detection,” in *2012 IEEE Second International Conference on Healthcare Informatics, Imaging and Systems Biology*, 2012, pp. 75–82.

- [13] B. X. Guan, B. Bhanu, N. Thakoor, P. Talbot, and S. Lin, "Human Embryonic Stem Cell Detection by Spatial Information and Mixture of Gaussians," in *2011 IEEE First International Conference on Healthcare Informatics, Imaging and Systems Biology*, 2011, pp. 307–314.
- [14] B. X. Guan, B. Bhanu, N. S. Thakoor, P. Talbot, and S. Lin, "Automatic cell region detection by k-means with weighted entropy," in *2013 IEEE 10th International Symposium on Biomedical Imaging*, 2013, pp. 418–421.
- [15] B. X. Guan, B. Bhanu, P. Talbot, and S. Lin, "Bio-Driven Cell Region Detection in Human Embryonic Stem Cell Assay," *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, vol. 11, no. 3, pp. 604–611, May 2014.
- [16] P. Talbot, N. I. zur Nieden, S. Lin, I. Martinez, B. Guan, and B. Bhanu, "Use of Video Bioinformatics Tools in Stem Cell Toxicology," in *Handbook of Nanotoxicology, Nanomedicine and Stem Cell Use in Toxicology*, Chichester, UK: John Wiley & Sons, Ltd, 2014, pp. 379–402.
- [17] "The UC Riverside NSF Integrated Graduate Education Research and Training (IGERT) Program on Video Bioinformatics." [Online]. Available: <http://www.cris.ucr.edu/IGERT/index.php>.
- [18] K. Tokunaga *et al.*, "Computational image analysis of colony and nuclear morphology to evaluate human induced pluripotent stem cells," *Sci. Rep.*, vol. 4, no. 1, p. 6996, May 2015.
- [19] C. M. Megyola *et al.*, "Dynamic Migration and Cell-Cell Interactions of Early Reprogramming Revealed by High-Resolution Time-Lapse Imaging," *Stem Cells*, vol. 31, no. 5, pp. 895–905, May 2013.
- [20] C. C. Wong *et al.*, "Non-invasive imaging of human embryos before embryonic genome activation predicts development to the blastocyst stage," *Nat. Biotechnol.*, vol. 28, no. 10, pp. 1115–1121, Oct. 2010.
- [21] M. Winter *et al.*, "Vertebrate neural stem cell segmentation, tracking and lineaging with validation and editing," *Nat. Protoc.*, vol. 6, no. 12, pp. 1942–1952, Nov. 2011.
- [22] S. V. Alworth, H. Watanabe, and J. S. J. Lee, "Teachable, High-Content Analytics for Live-Cell, Phase Contrast Movies," *J. Biomol. Screen.*, vol. 15, no. 8, pp. 968–977, Sep. 2010.
- [23] J. Nichols and A. Smith, "Pluripotency in the Embryo and in Culture," *Cold Spring Harb. Perspect. Biol.*, vol. 4, no. 8, pp. a008128–a008128, Aug. 2012.

- [24] S. Lin and P. Talbot, "Methods for Culturing Mouse and Human Embryonic Stem Cells," 2011, pp. 31–56.
- [25] S. Lin, V. Tran, and P. Talbot, "Comparison of toxicity of smoke from traditional and harm-reduction cigarettes using mouse embryonic stem cells as a novel model for preimplantation development," *Hum. Reprod.*, vol. 24, no. 2, pp. 386–397, Feb. 2009.
- [26] M. Knoll, R. Shaoulian, T. Magers, and P. Talbot, "Ciliary Beat Frequency of Hamster Oviducts is Decreased in Vitro by Exposure to Solutions of Mainstream and Sidestream Cigarette Smoke¹," *Biol. Reprod.*, vol. 53, no. 1, pp. 29–37, Jul. 1995.
- [27] M. Knoll and P. Talbot, "Cigarette smoke inhibits oocyte cumulus complex pick-up by the oviduct in vitro independent of ciliary beat frequency," *Reprod. Toxicol.*, vol. 12, no. 1, pp. 57–68, Jan. 1998.
- [28] "Detecting a Cell Using Image Segmentation." [Online]. Available: <http://www.mathworks.com/help/images/%0Aexamples/detecting-a-cell-using-image-segmentation.html>.
- [29] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Trans. Syst. Man. Cybern.*, vol. 9, no. 1, pp. 62–66, Jan. 1979.
- [30] L. G. Shapiro and G. C. Stockman, *Computer Vision*. Pearson, 2001.
- [31] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, *Computational Geometry*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997.
- [32] Z. Zhao, S. Sharma, A. Anand, F. Morstatter, S. Alelyani, and H. Liu, "Advancing Feature Selection Research," *ASU Featur. Sel. Repos. Arizona State Univ.*, pp. 1–28, 2010.
- [33] X. V. Nguyen, J. Chan, S. Romano, and J. Bailey, "Effective global approaches for mutual information based feature selection," *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discov. data Min. - KDD '14*, pp. 512–521, 2014.
- [34] M. A. Hall and L. A. Smith, "Feature Selection for Machine Learning: Comparing a Correlation-Based Filter Approach to the Wrapper," in *Proceedings of the Twelfth International Florida Artificial Intelligence Research Society Conference*, 1999, pp. 235–239.

- [35] Huan Liu and R. Setiono, “Chi2: feature selection and discretization of numeric attributes,” in *Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence*, pp. 388–391.
- [36] M. L. Bermingham *et al.*, “Application of high-dimensional feature selection: evaluation for genomic prediction in man,” *Sci. Rep.*, vol. 5, no. 1, p. 10312, Sep. 2015.
- [37] R. O. Duda, P. E. Hart, and D. G. Stork, “Pattern Classification,” *New York: John Wiley, Section*. p. 654, 2000.
- [38] C. Gini, *Variabilità e mutabilità*. 1912.
- [39] T. M. Cover and J. A. Thomas, “Elements of Information Theory,” 1991.
- [40] G. C. Cawley, N. L. Talbot, and M. Girolami, “Sparse Multinomial Logistic Regression via Bayesian L1 Regularisation,” in *Advances in Neural Information Processing Systems 19*, B. Schölkopf, J. C. Platt, and T. Hoffman, Eds. MIT Press, 2007, pp. 209–216.
- [41] D. C. Montgomery, G. C. Runger, and N. F. Hubele, *Engineering statistics*. John Wiley & Sons, 2009.
- [42] L. J. Wei, “Asymptotic Conservativeness and Efficiency of Kruskal-Wallis Test for K Dependent Samples,” *J. Am. Stat. Assoc.*, vol. 76, no. 376, p. 1006, Dec. 1981.
- [43] Hanchuan Peng, Fuhui Long, and C. Ding, “Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.
- [44] X. Wang, W. Song, N. Kawazoe, and G. Chen, “Influence of cell protrusion and spreading on adipogenic differentiation of mesenchymal stem cells on micropatterned surfaces,” *Soft Matter*, vol. 9, no. 16, p. 4160, 2013.
- [45] R. McBeath, D. M. Pirone, C. M. Nelson, K. Bhadriraju, and C. S. Chen, “Cell shape, cytoskeletal tension, and RhoA regulate stem cell lineage commitment,” *Dev. Cell*, vol. 6, no. 4, pp. 483–95, Apr. 2004.
- [46] G. E. Uhlenbeck and L. S. Ornstein, “On the Theory of the Brownian Motion,” *Phys. Rev.*, vol. 36, no. 5, pp. 823–841, Sep. 1930.
- [47] P. Dieterich, R. Klages, R. Preuss, and A. Schwab, “Anomalous dynamics of cell migration,” *Proc. Natl. Acad. Sci.*, vol. 105, no. 2, pp. 459–463, 2008.

- [48] C. Huang, Z. Rajfur, C. Borchers, M. D. Schaller, and K. Jacobson, “JNK phosphorylates paxillin and regulates cell migration.,” *Nature*, vol. 424, no. 6945, pp. 219–23, Jul. 2003.
- [49] B. S. Fogh, H. A. B. Multhaupt, and J. R. Couchman, “Protein Kinase C, Focal Adhesions and the Regulation of Cell Migration,” *J. Histochem. Cytochem.*, vol. 62, no. 3, pp. 172–184, Mar. 2014.
- [50] H. Ke, J. Y. Zhang, S. K. Akiyama, and J. E. French, “BCL2 interaction with actin in vitro may inhibit cell motility by enhancing actin polymerization,” *Cell Adh. Migr.*, vol. 5, no. 1, pp. 6–10, Jan. 2011.
- [51] I. Barbaric *et al.*, “Time-Lapse Analysis of Human Embryonic Stem Cells Reveals Multiple Bottlenecks Restricting Colony Formation and Their Relief upon Culture Adaptation,” *Stem Cell Reports*, vol. 3, no. 1, pp. 142–155, Jul. 2014.
- [52] E. Maalouf, *Contribution to Fluorescence Microscopy, 3D Thick Samples Deconvolution and Depth-variant Psf Maalouf*. 2010.
- [53] D. J. Stephens and V. J. Allan, “Light Microscopy Techniques for Live Cell Imaging,” vol. 82, no. 2003, pp. 82–87, 2010.
- [54] A. Zahedi *et al.*, “Evaluating Cell Processes , Quality , and Biomarkers in Pluripotent Stem Cells Using Video Bioinformatics,” *PLoS One*, pp. 1–22, 2016.
- [55] L. Fass, “Imaging and cancer: A review,” *Mol. Oncol.*, vol. 2, no. 2, pp. 115–152, Aug. 2008.
- [56] S. Heerboth *et al.*, “EMT and tumor metastasis,” *Clin. Transl. Med.*, vol. 4, no. 1, p. 6, 2015.
- [57] L. Li *et al.*, “High-throughput Imaging: Focusing In On Drug Discovery in 3D,” pp. 97–102, 2017.
- [58] V. On, A. Zahedi, and B. Bhanu, “3D Reconstruction of Phase Contrast Images Using Focus Measures,” in *IEEE International Conference on Image Processing*, 2018.
- [59] A. Ray and A. K. Mitra, “Nanotechnology in Intracellular Trafficking, Imaging, and Delivery of Therapeutic Agents,” in *Emerging Nanotechnologies for Diagnostics, Drug Delivery and Medical Devices*, Elsevier, 2017, pp. 169–188.

- [60] D. Vaquero, N. Gelfand, M. Tico, K. Pulli, and M. Turk, “Generalized autofocus,” in *IEEE Workshop on Applications of Computer Vision (WACV)*, 2011, pp. 511–518.
- [61] J. Koseki *et al.*, “The development of an all-in-focus algorithm for endoscopy,” in *MHS2013*, 2013, no. 1, pp. 1–5.
- [62] M. Chen, L. Tian, and L. Waller, “3D differential phase contrast microscopy,” *Biomed. Opt. Express*, vol. 7, no. 10, p. 3940, 2016.
- [63] A. Ray and A. K. Mitra, *Nanotechnology in Intracellular Trafficking, Imaging, and Delivery of Therapeutic Agents*. Elsevier, 2017.
- [64] Y. Yao, B. Abidi, N. Doggaz, and M. Abidi, “Evaluation of sharpness measures and search algorithms for the auto focusing of high-magnification images,” 2006, vol. 6246, p. 62460G.
- [65] L. G. Brown, “A survey of image registration techniques,” *ACM Comput. Surv.*, vol. 24, no. 4, pp. 325–376, 1992.
- [66] W. Loh, “Regression Trees with Unbiased Variable Selection,” *Korean J. Appl. Stat.*, vol. 17, no. 3, pp. 459–473, Nov. 2004.
- [67] C.-H. Ho and C.-J. Lin, “Large-scale Linear Support Vector Regression,” *JMLR*, vol. 13, pp. 3323–3348, 2012.
- [68] J. H. Friedman, “Greedy Function Approximation: A Gradient Boosting Machine,” *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001.
- [69] P. J. Flynn and A. K. Jain, “On reliable curvature estimation,” in *Proceedings CVPR '89: IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 110–116.
- [70] J. J. Koenderink and A. J. van Doorn, “Surface shape and curvature scales,” *Image Vis. Comput.*, vol. 10, no. 8, pp. 557–564, 1992.
- [71] Y. Guo, T. Hastie, and R. Tibshirani, *Regularized linear discriminant analysis and its application in microarrays.*, vol. 8, no. 1. 2007.
- [72] E. L. Allwein, R. Schapire, and Y. Singer, “Reducing multiclass to binary: A unifying approach for margin classifiers,” *CrossRef List. Deleted DOIs*, vol. 1, pp. 113–141, 2000.

- [73] H. Chen and B. Bhanu, "3D free-form object recognition in range images using local surface patches," *Pattern Recognit. Lett.*, vol. 28, no. 10, pp. 1252–1262, 2007.
- [74] H. Chen and B. Bhanu, "Human ear recognition in 3D," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 718–737, 2007.
- [75] C. Dorai and A. K. Jain, "COSMOS-a representation scheme for 3D free-form objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 10, pp. 1115–1130, 1997.
- [76] B. Schiele and J. L. Crowley, "Recognition without Correspondence using Multidimensional Receptive Field Histograms," *Int. J. Comput. Vis.*, vol. 36, no. 1, pp. 31–50, 2000.
- [77] P. Caroni *et al.*, "Regulation of Actin Dynamics Through Phosphorylation of cofilin by LIM-kinase," *Nature*, vol. 393, no. 6687, pp. 805–809, Jun. 1998.
- [78] T. Sumi, K. Matsumoto, Y. Takai, and T. Nakamura, "Cofilin phosphorylation and actin cytoskeletal dynamics regulated by Rho- and Cdc42-activated LIM-kinase 2," *J. Cell Biol.*, vol. 147, no. 7, pp. 1519–1532, 1999.
- [79] E. Andrianantoandro and T. D. Pollard, "Mechanism of Actin Filament Turnover by Severing and Nucleation at Different Concentrations of ADF/Cofilin," *Mol. Cell*, vol. 24, no. 1, pp. 13–23, 2006.
- [80] C. Chan, C. C. Beltzner, and T. D. Pollard, "Cofilin Dissociates Arp2/3 Complex and Branches from Actin Filaments," *Curr. Biol.*, vol. 19, no. 7, pp. 537–545, 2009.
- [81] C. G. Pontrello, M.-Y. Sun, A. Lin, T. A. Fiacco, K. A. DeFea, and I. M. Ethell, "Cofilin under control of -arrestin-2 in NMDA-dependent dendritic spine plasticity, long-term depression (LTD), and learning," *Proc. Natl. Acad. Sci.*, vol. 109, no. 7, pp. E442–E451, Feb. 2012.
- [82] M. Bosch, J. Castro, T. Saneyoshi, H. Matsuno, M. Sur, and Y. Hayashi, "Structural and molecular remodeling of dendritic spine substructures during long-term potentiation.," *Neuron*, vol. 82, no. 2, pp. 444–59, Apr. 2014.
- [83] A. Rao and A. M. Craig, "Signaling between the actin cytoskeleton and the postsynaptic density of dendritic spines," *Hippocampus*, vol. 10, no. 5, pp. 527–541, 2000.

- [84] K. E. Sorra and K. M. Harris, “Overview on the structure, composition, function, development, and plasticity of hippocampal dendritic spines,” *Hippocampus*, vol. 10, no. 5, pp. 501–511, 2000.
- [85] H. Hering and M. Sheng, “Dendritic spines: structure, dynamics and regulation.,” *Nat. Rev. Neurosci.*, vol. 2, no. 12, pp. 880–888, Dec. 2001.
- [86] R. Yuste and T. Bonhoeffer, “Genesis of dendritic spines: insights from ultrastructural and imaging studies,” *Nat. Rev. Neurosci.*, vol. 5, no. 1, pp. 24–34, Jan. 2004.
- [87] M. Matsuzaki, G. C. Ellis-Davies, T. Nemoto, Y. Miyashita, M. Iino, and H. Kasai, “Dendritic spine geometry is critical for AMPA receptor expression in hippocampal CA1 pyramidal neurons.,” *Nat. Neurosci.*, vol. 4, no. 11, pp. 1086–92, Nov. 2001.
- [88] I. M. Ethell and E. B. Pasquale, “Molecular mechanisms of dendritic spine development and remodeling.,” *Prog. Neurobiol.*, vol. 75, no. 3, pp. 161–205, Feb. 2005.
- [89] J. R. Bamberg *et al.*, “ADF/Cofilin-Actin Rods in Neurodegenerative Diseases,” *Curr. Alzheimer Res.*, vol. 7, no. 3, pp. 241–250, May 2010.
- [90] A. Blangy, H. Touaitahuata, G. Cres, and G. Pawlak, “Cofilin Activation during Podosome Belt Formation in Osteoclasts,” *PLoS One*, vol. 7, no. 9, pp. 1–12, 2012.
- [91] C. a Schneider, W. S. Rasband, and K. W. Eliceiri, “NIH Image to ImageJ: 25 years of image analysis,” *Nat. Methods*, vol. 9, no. 7, pp. 671–675, 2012.
- [92] M. Halavi, K. A. Hamilton, R. Parekh, and G. A. Ascoli, “Digital reconstructions of neuronal morphology: three decades of research trends,” *Front. Neurosci.*, vol. 6, p. 49, 2012.
- [93] V. On, A. Zahedi, I. M. Ethell, and B. Bhanu, “Spatio-Temporal Pattern Recognition of Dendritic Spines and Protein Dynamics Using Live Multichannel Fluorescence Microscopy,” in *23rd International Conference on Pattern Recognition*, 2016.
- [94] M. E. Berginski, E. A. Vitriol, K. M. Hahn, and S. M. Gomez, “High-Resolution Quantification of Focal Adhesion Spatiotemporal Dynamics in Living Cells,” *PLoS One*, vol. 6, no. 7, p. e22025, Jul. 2011.

- [95] I. Smal, M. Loog, W. Niessen, and E. Meijering, “Quantitative Comparison of Spot Detection Methods in Fluorescence Microscopy,” *IEEE Trans. Med. Imaging*, vol. 29, no. 2, pp. 282–301, Feb. 2010.
- [96] T. Pecot, J. Boulanger, C. Kervrann, P. Bouthemy, and J. Salamero, “Estimation of the flow of particles within a partition of the image domain in fluorescence video-microscopy,” in *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, 2014, pp. 453–456.
- [97] T. Pecot, C. Kervrann, J. Salamero, and J. Boulanger, “Counting-Based Particle Flux Estimation for Traffic Analysis in Live Cell Imaging,” *IEEE J. Sel. Top. Signal Process.*, vol. 10, no. 1, pp. 203–216, Feb. 2016.
- [98] M. Théry *et al.*, “The extracellular matrix guides the orientation of the cell division axis,” *Nat. Cell Biol.*, vol. 7, no. 10, pp. 947–953, Oct. 2005.
- [99] A. Rodriguez, D. B. Ehlenberger, D. L. Dickstein, P. R. Hof, and S. L. Wearne, “Automated Three-Dimensional Detection and Shape Classification of Dendritic Spines from Fluorescence Microscopy Images,” *PLoS One*, vol. 3, no. 4, p. e1997, Apr. 2008.
- [100] I. Y. Y. Koh, W. B. Lindquist, K. Zito, E. A. Nimchinsky, and K. Svoboda, “An Image Analysis Algorithm for Dendritic Spines,” *Neural Comput.*, vol. 14, no. 6, pp. 1283–1310, Jun. 2002.
- [101] C. M. Weaver, P. R. Hof, S. L. Wearne, and W. B. Lindquist, “Automated algorithms for multiscale morphometry of neuronal dendrites,” *Neural Comput.*, vol. 16, no. 7, pp. 1353–83, Jul. 2004.
- [102] Y. Zhang *et al.*, “A novel tracing algorithm for high throughput imaging Screening of neuron-based assays,” *J. Neurosci. Methods*, vol. 160, no. 1, pp. 149–62, Feb. 2007.
- [103] J. Cheng *et al.*, “A novel computational approach for automatic dendrite spines detection in two-photon laser scan microscopy,” *J. Neurosci. Methods*, vol. 165, no. 1, pp. 122–34, Sep. 2007.
- [104] S. Basu *et al.*, “2dSpAn: semiautomated 2-d segmentation, classification and analysis of hippocampal dendritic spine plasticity,” *Bioinformatics*, p. 172, Apr. 2016.
- [105] Y. Zhang *et al.*, “A neurocomputational method for fully automated 3D dendritic spine detection and segmentation of medium-sized spiny neurons,” *Neuroimage*, vol. 50, no. 4, pp. 1472–1484, 2010.

- [106] C. Xu and J. L. Prince, “Gradient Vector Flow: A New External Force for Snakes,” *IEEE Proc. Conf. Comp. Vis. Patt. Recog*, 1997.
- [107] J. Han and B. Bhanu, “Individual Recognition Using Gait Energy Image,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 316–322, 2006.
- [108] M. Carter and J. Shieh, *Guide to Research Techniques in Neuroscience*. 2015.
- [109] G. Herberich, T. Würflinger, A. Sechi, R. Windoffer, R. Leube, and T. Aach, “Fluorescence microscopic imaging and image analysis of the cytoskeleton,” in *Conference Record - Asilomar Conference on Signals, Systems and Computers*, 2010, pp. 1359–1363.
- [110] B. Petrova, S. Dehler, T. Kruitwagen, J.-K. Hériché, K. Miura, and C. H. Haering, “Quantitative analysis of chromosome condensation in fission yeast,” *Mol. Cell. Biol.*, vol. 33, no. 5, pp. 984–98, Mar. 2013.
- [111] S. Frisken and R. Perry, “Adaptively sampled distance fields: a general representation of shape for computer graphics,” ... *Comput. Graph. ...*, pp. 249–254, 2000.
- [112] J. B. Slee and L. J. Lowe-Krentz, “Actin realignment and cofilin regulation are essential for barrier integrity during shear stress,” *J. Cell. Biochem.*, vol. 114, no. 4, pp. 782–795, Apr. 2013.
- [113] H. P. Mohan Kumar and H. S. Nagendraswamy, “LBP for gait recognition: A symbolic approach based on GEI plus RBL of GEI,” in *2014 International Conference on Electronics and Communication Systems (ICECS)*, 2014, pp. 1–5.
- [114] W. Kusakunniran, Q. Wu, H. Li, and J. Zhang, “Automatic Gait Recognition Using Weighted Binary Pattern on Video,” in *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2009, pp. 49–54.
- [115] T. Ojala, M. Pietikainen, and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.
- [116] R. E. Carlson and F. N. Fritsch, “Monotone Piecewise Bicubic Interpolation,” *SIAM J. Numer. Anal.*, vol. 22, no. 2, pp. 386–400, Apr. 1985.
- [117] Jie Cheng, Xiaobo Zhou, B. L. Sabatini, and S. T. Wong, “NeuronIQ: A novel computational approach for automatic dendrite spines detection and analysis,” in *2007 IEEE/NIH Life Science Systems and Applications Workshop*, 2007, pp. 168–171.

- [118] E. Montagna, M. M. Dorostkar, and J. Herms, “The Role of APP in Structural Spine Plasticity,” *Front. Mol. Neurosci.*, vol. 10, no. May, pp. 1–7, 2017.
- [119] E. Bertling and P. Hotulainen, “New waves in dendritic spine actin cytoskeleton: From branches and bundles to rings, from actin binding proteins to post-translational modifications,” *Mol. Cell. Neurosci.*, 2017.
- [120] C. K. Bradley *et al.*, “Derivation of Huntington’s Disease-Affected Human Embryonic Stem Cell Lines,” *Stem Cells Dev.*, vol. 20, no. 3, pp. 495–502, Mar. 2011.
- [121] J. D. Miller *et al.*, “Human iPSC-Based Modeling of Late-Onset Disease via Progerin-Induced Aging,” *Cell Stem Cell*, vol. 13, no. 6, pp. 691–705, Dec. 2013.
- [122] P. Grandjean *et al.*, “The Faroes Statement: Human Health Effects of Developmental Exposure to Chemicals in Our Environment,” *Basic Clin. Pharmacol. Toxicol.*, p. 070730044445006–???, Jul. 2007.

Appendix

S1 Fig. (A) Diagram showing workflow used to develop StemCellQC™. (B) Diagram showing feature selection methods for classification.

S2 Fig. Decision tree showing method for classifying hESC colonies into healthy, unhealthy or dying groups.

Red arrows show decisions resulting in classification of a colony as unhealthy or dying, green arrows show decisions resulting in classification as healthy, and black arrows indicate points where the classification process was continued.

S3 Fig. Ground truth verification of colony segmentation using ImageJ to manually segment 6 representative healthy, 6 unhealthy, and 6 dying colonies.

(A, B) Normalized area and perimeter values for healthy colonies extracted by StemCellQC compared to ground truth using ImageJ. 2-way ANOVA revealed no significant differences. (C, D) Normalized area and perimeter values for unhealthy colonies extracted by StemCellQC compared to ground truth using ImageJ. 2-way ANOVA revealed no significant differences. (E, F) Normalized area and perimeter values for dying colonies extracted by StemCellQC compared to ground truth using ImageJ. 2-way ANOVA revealed no significant differences, except for a portion of the normalized area of dying colonies. This corresponds with slight over-segmentation of software due to detection of cellular debris ejected from dying colonies after their death at 30hours (* = $P < 0.05$).

S4 Fig. Relationship between features and cell processes.

S5 Fig. Visual descriptors of extracted features related to area.

S6 Fig. Visual descriptors of extracted features related to morphology and area.

S7 Fig. Visual descriptors of extracted features related to motility.

S8 Fig. Visual descriptors of extracted features related to apoptosis.

S9 Fig. List of Extracted Features and Definitions.

S1 Video. Average intensity versus perimeter running plot shown for all individual healthy (green), unhealthy (blue), and dying (red) hESC colonies.

S2 Video. Mean-squared displacement versus area running plot shown for all individual healthy (green), unhealthy (blue), and dying (red) hESC colonies.

S3 Video. Phase contrast video of a representative healthy colony with the segmentation outlined in white.

S4 Video. Protrusions feature video of a representative healthy colony with the protrusions outlined in red.

S5 Video. Bright-to-total area ratio feature video with the bright dead cells of a representative unhealthy colony highlighted in white.

S6 Video. Solidity feature video of a representative dying colony with the convex hull shown in white and the colony segmentation outlined in red.