

ABSTRACT

Title of dissertation: SEGMENTATION AND INFORMATICS IN
MULTIDIMENSIONAL FLUORESCENCE
OPTICAL MICROSCOPY IMAGES

Kaustav Nandy, Doctor of Philosophy, 2015

Dissertation directed by: Professor Rama Chellappa
Department of Computer Science

Recent advances in the field of optical microscopy have enabled scientists to observe and image complex biological processes across a wide range of spatial and temporal resolution, resulting in an exponential increase in optical microscopy data. Manual analysis of such large volumes of data is extremely time consuming and often impossible if the changes cannot be detected by the human eye. Naturally it is essential to design robust, accurate and high performance image processing and analysis tools to extract biologically significant results. Furthermore, the presentation of the results to the end-user, post analysis, is also an equally challenging issue, especially when the data (and/or the hypothesis) involves several spatial/hierarchical scales (e.g., tissues, cells, (sub)-nuclear components). This dissertation concentrates on a subset of such problems such as robust edge detection, automatic nuclear segmentation and selection in multi-dimensional tissue images, spatial analysis of gene localization within the cell nucleus, information visualization and the development of a computational framework for efficient and high-throughput processing of large datasets.

Initially, we have developed 2D nuclear segmentation and selection algorithms which help in the development of an integrated approach for determining the preferential spatial localization of certain genes within the cell nuclei which is emerging as a promising technique for the diagnosis of breast cancer. Quantification requires accurate segmentation of 100 to 200 cell nuclei in each patient tissue sample in order to draw a statistically significant result. Thus, for large scale analysis involving hundreds of patients, manual processing is too time consuming and subjective. We have developed an integrated workflow that selects, following 2D automatic segmentation, a sub-population of accurately delineated nuclei for positioning of *fluorescence in situ hybridization labeled* genes of interest in tissue samples. Application of the method was demonstrated for discriminating normal and cancerous breast tissue sections based on the differential positioning of the HES5 gene. Automatic results agreed with manual analysis in 11 out of 14 cancers, all 4 normal cases and all 5 non-cancerous breast disease cases, thus showing the accuracy and robustness of the proposed approach.

As a natural progression from the 2D analysis algorithms to 3D, we first developed a robust and accurate probabilistic edge detection method for 3D tissue samples since several down stream analysis procedures such as segmentation and tracking rely on the performance of edge detection. The method based on multi-scale and multi-orientation steps surpasses several other conventional edge detectors in terms of its performance. Subsequently, given an appropriate edge measure, we developed an optimal graphcut-based 3D nuclear segmentation technique for samples where the cell nuclei are volume or surface labeled. It poses the problem as

one of finding minimal closure in a directed graph and solves it efficiently using the maxflow-mincut algorithm. Both interactive and automatic versions of the algorithm are developed. The algorithm outperforms, in terms of three metrics that are commonly used to evaluate segmentation algorithms, a recently reported geodesic distance transform-based 3D nuclear segmentation method which in turns was reported to outperform several other popular tools that segment 3D nuclei in tissue samples.

Finally, to apply some of the aforementioned methods to large microscopic datasets, we have developed a user friendly computing environment called *MiPipeline* which supports high throughput data analysis, data and process provenance, visual programming and seamlessly integrated information visualization of hierarchical biological data. The computational part of the environment is based on LONI Pipeline distributed computing server and the interactive information visualization makes use of several javascript based libraries to visualize an XML-based backbone file populated with essential meta-data and results.

SEGMENTATION AND INFORMATICS IN
MULTIDIMENSIONAL FLUORESCENCE OPTICAL
MICROSCOPY IMAGES

by

Kaustav Nandy

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2015

Advisory Committee:
Professor Rama Chellappa, Chair/Advisor
Professor David Jacobs
Professor Amitabh Varshney
Professor Alan Sussman
Professor Wolfgang Losert
Dr. Stephen Lockett

© Copyright by
Kaustav Nandy
2015

Dedication

... to my beloved son Kashyap, Anny, Ma, Baba and my family

Acknowledgments

It has been a long journey, but it would not have been possible without the constant support of my advisor Dr. Rama Chellappa. I would like to sincerely thank him for his guidance and mentoring. His suggestions and feedback have been invaluable in shaping up the dissertation. His support has been a constant source of encouragement for me. He was a great source of inspiration without whom my journey in graduate school would not have been as memorable. I also thank him for providing me the flexibility to work on my PhD while being full time employed.

I would also like to thank Dr. Amitabh Varshney, Dr. David Jacobs, Dr. Alan Sussman and Dr Wolfgang Losert for being on my dissertation committee. They have always been very encouraging and supportive. The meetings and discussions with them were very enlightening and in several occasions helped me in deciding the future course of action. I specifically attended courses with Dr. Jacobs and Dr. Sussman which were very instructive and enjoyable experiences and helped me in adjusting back to graduate school after the long hiatus.

My colleague Dr. Prabhakar R. Gudla and my supervisor Dr. Stephen J. Lockett at Frederick National Lab for Cancer research have been very supportive during the entire course of my PhD and without their encouragement and guidance the road would have been even more bumpy. Both Dr. Gudla and Dr. Lockett have provided me invaluable technical and generic guidance and suggestions. They have also helped me immensely in improving my presentation skills and preparing the manuscripts for conferences and journals. I specifically thank Dr. Lockett for

providing me the opportunity to work on my PhD while being full time employed under his supervision and being a part of my dissertation committee. His support and encouragement has been vital for the past several years.

I also thank my collaborators Dr. Tom Misteli and Dr. Karen Meaburn at the National Cancer Institute for their support and patience during our successful collaboration. It was the first large project handled by me and was a great learning experience both technically and professionally.

I sincerely thank my wife Debjani without whom my survival would have been a challenge during my PhD. She was always beside me during the ups and down of the past few years and has never failed to give me ample support and encouragement to see through the difficult times.

Last but not the least I am indebted to my parents and parents-in-law for having faith on me and being so supportive and enthusiastic during my PhD.

This project has been funded in whole or in part with Federal funds from the National Cancer Institute, National Institutes of Health, under Contract No. HHSN261200800001E. The content of this publication does not necessarily reflect the views or policies of the Department of Health and Human Services, nor does mention of trade names, commercial products, or organizations imply endorsement by the U.S. Government. The Office of Human Subjects Research (OHSR) at the National Institutes of Health, USA determined on January 2 2008 that federal regulations for the protection of human subjects do not apply to this research project. The human material used in this study had been de-identified before any of the authors received it.

Contents

List of Tables	viii
List of Figures	x
1 Introduction	1
1.1 Fluorescence and Confocal Microscopy	3
1.2 Challenges in Microscopic Image Analysis and Role of Computers . .	4
1.3 The Problem Statement	7
1.4 Contributions	8
1.5 Organization	10
2 Segmentation and Selection of Cell Nuclei in 2D Tissue Images for Breast Cancer Detection	11
2.1 Introduction	11
2.2 Samples and Images	16
2.3 Image Preprocessing	18
2.4 A Hybrid Algorithm for Nuclear Segmentation and Selection	19
2.4.1 Stacked Classifier-based Pattern Recognition Engine	23
2.5 Multistage Watershed Based Segmentation and Artificial Neural Net- work based PRE	25
2.5.1 Segmentation of nuclei	25
2.5.2 ANN based Pattern Recognition Engine: Feature Measure- ment and Selection	30
2.5.3 Object Classification using an Artificial Neural Network	31
2.5.4 Performance Assessment of Nuclear Segmentation and PRE . .	32
2.5.5 Segmentation and Classification Results	36
2.5.5.1 ANN Classification Performance	36
2.5.5.2 Segmentation Accuracy of Selected Nuclei using ANN	37
2.6 Application to Gene Localization for Breast Cancer Detection	39
2.6.1 Segmentation and Spatial Analysis of FISH labeled Gene signals	39
2.6.2 Manual Analysis	42
2.6.3 Results for Gene Localization using Hybrid Segmentation and Stacked Classifier	42
2.6.4 Results for Gene Localization using Multistage Watershed Segmentation and ANN	43
2.7 Discussion and Conclusion	46

3	Probabilistic Edge Detection in 2D and 3D Optical Microscopy Images	52
3.1	Introduction	52
3.2	Related Work	54
3.3	Probabilistic Edge Detection for Microscopy Images	56
3.3.1	Multiscale Brightness, Color and Texture Based Edge Measurements	58
3.3.2	Spectral Edge Measurements	61
3.3.3	Efficient Computation For 3D	62
3.4	Experimental Datasets and Methods	65
3.4.1	Experimental Datasets	65
3.4.1.1	Synthetic Datasets	65
3.4.1.2	Simulated Datasets	69
3.4.1.3	Original Microscopic Datasets	70
3.4.2	Experimental Methods	72
3.5	Experimental Results	75
3.5.1	2D Edge Detection Results	75
3.5.2	3D Edge Detection Results	77
3.6	Discussion and Conclusion	80
4	Graphcut Based Method For Nuclear Segmentation in 2D and 3D Optical Microscopy Tissue Images	87
4.1	Introduction	87
4.2	Related Work	89
4.3	Graph Cut Based Segmentation	91
4.3.1	Minimal Closure and Equivalent s-t Cut	92
4.3.2	Single Object Segmentation	93
4.3.3	Automatic Multi-Object Segmentation	98
4.3.3.1	Segmentation in 2D	98
4.3.3.2	Automatic Segmentation in 3D	101
4.3.4	Interactive Segmentation	103
4.4	Experimental Datasets and Methods	104
4.4.1	Segmentation Evaluation Metrics	105
4.4.2	Edge Measurement Methods	107
4.4.3	Comparison With Other Segmentation Methods	109
4.4.4	Implementation	109
4.5	Experimental Results	109
4.5.1	Results of 2D Automatic Segmentation	109
4.5.2	Results of 3D Automatic Segmentation	112
4.6	Discussion and Conclusions	114
5	MiPipeline (<u>Microscopy Pipeline</u>): A User Friendly Software Ecosystem for Microscopy Image Analysis and Informatics	118
5.1	Introduction	118
5.2	Related Work	121
5.3	MiPipeline Environment	123

5.3.1	LONI Pipeline	125
5.3.2	MiPipeline File Server	127
5.3.3	MiPipeline Data Provenance	128
5.3.4	MiPipeline Information Visualization	129
5.3.5	Microscopy Tool Integration	130
5.3.6	Worked Example and User Interaction	131
5.4	Case Studies	132
5.4.1	QuickPALM Case Study	133
5.4.2	Parallel Genome Organization Diagnosis Application (PAGODA) Case Study	135
5.4.2.1	Tissue Samples and Image Acquisition	135
5.4.2.2	Image Analysis and Workflow	137
5.4.2.3	PAGODA Results	138
5.4.2.4	Advanced Visual Exploration and Mining	140
5.5	Discussion and Conclusions	142
6	Conclusions and Future Directions	144
6.1	Summary	144
6.2	Future Directions of Work	146
A	Appendix	149
A.1	Nuclear Feature Set used for ANN PRE	149
	Bibliography	153

List of Tables

2.1	Performance comparison of stacked classifier and ANN classifier systems	38
2.2	Mean and standard deviation of accuracy parameters	38
2.3	Probability that the spatial FISH signals for NMF _M and NAF _A are similar using 1-D K-S test	42
2.4	Probability that the spatial FISH signals for NMF _M and NMFA are similar using 1-D K-S test	43
2.5	Manual and well segmented automatic nuclei count	44
2.6	Probability of similarity of FISH signal distribution between normal, cancer and non-cancerous breast disease tissue sections using manual and automatic processing	45
3.1	Maximum F-score values for 2D synthetic and simulated datasets using precision recall framework for evaluation	83
3.2	Maximum F-score values for 2D real images from MCF-10A, human breast tissue micro-array and mouse embryo samples	84
3.3	Maximum F-score values for 3D anisotropic synthetic and simulated datasets using precision recall framework for evaluation	85
3.4	Maximum F-score values for 3D real images from MCF-10A, human breast tissue micro-array and mouse embryo samples	86
3.5	Maximum F-score values for 3D synthetic (SNR 3 and blur 0.7), simulated (SNR 3 and blur 0.7) and real images from MCF-10A, human breast tissue micro-array and mouse embryo samples by full 3D and slice by slice MiPb calculation	86
4.1	Rand Index(RI), mean Overlap Ration(OR) and mean/standard deviation of Boundary Deviation (BD) (in terms of pixels) metrics comparing graphcut (GC) based segmentation and MINS segmentation maps with ground truth object boundaries of 2D simulated datasets.	107
4.2	Rand Index(RI), mean Overlap Ration(OR) and mean/standard deviation of Boundary Deviation (BD) (in terms of pixels) metrics comparing graphcut (GC) based segmentation and MINS segmentation maps with manually annotated object boundaries of 2D actual microscopic datasets.	110

4.3	Rand Index(RI), mean Overlap Ration(OR) and mean/standard deviation of Boundary Deviation (BD) (in terms of voxels) metrics comparing graphcut (GC) based segmentation and MINS segmentation maps with ground truth object boundaries of 3D simulated datasets.	112
4.4	Rand Index(RI), mean Overlap Ration(OR) and mean/standard deviation of Boundary Deviation (BD) (in terms of voxels) metrics comparing graphcut (GC) based segmentation and MINS segmentation maps with manually annotated object boundaries of 3D actual microscopic datasets.	113
5.1	Description of the worked example modules.	131

List of Figures

1.1	(a) Schematic showing working principle of a confocal microscope imaging a fluorescent sample. Image Source: http://microscopy.duke.edu/ (b) Illustrative excitation and emission spectra of a fluorescent tag. Image Source: http://www.olympusmicro.com/ (c) A 2D sample fluorescence image acquired on Zeiss confocal 510 microscope where 4',6 - diamidino - 2 - phenylindole (DAPI) stained cell nuclei are displayed in the red channel, FITC tagged actin fibers in the green channel and Alexa Flour 594 tagged mitochondria in the blue channel.	2
1.2	3D image of DAPI stained cell nuclei in a MCF10A acinus with single X-Y, X-Z and Z-Y slices displaying significant spatial clustering.	5
2.1	Sample human breast tissue images used for cancer detection. The DAPI stained cell nuclei are shown in the blue channel while the red and the green channels show the FISH tagged genes <i>HES5</i> and <i>FRA2</i> respectively as bright spots.	13
2.2	(a) Original blue DAPI channel (b) Wavelet-based preprocessing output	17
2.3	System Block Diagram	18
2.4	(a) Original DAPI channel after preprocessing (b) Thresholded image (c) Labeled image after morphological operations showing the input seeds for the level set segmentation algorithm with a single seed highlighted in red box A (d) Initial contour for level set algorithm for A (e) Level set evolved contour overlayed on the DAPI channel for A (f) Binary version of the evolved level set contour for A (g) Output image after applying watershed segmentation algorithm on the level set output region for A (h) Labeled version of the watershed output for A (i) Final segmentation on the entire image using the hybrid segmentation algorithm	21
2.5	Pattern recognition module showing the stacked classifier for identifying nuclei that can be used for the FISH analysis	22
2.6	Flow diagram showing the computational framework	24
2.7	Multistage watershed segmentation algorithm	26

2.8	Representative image and the corresponding outputs at different segmentation steps. (i) Original DAPI channel nuclei image (ii) Pre-processed nuclei channel (iii) Seeded intensity watershed output on image foreground (iv) Gray weighted distance transform (GDT) output (v) Merged output of intensity and GDT watershed (vi) Final segmentation output after the cluster breaking watershed and tree based merging	27
2.9	(a) Process of building up the merge tree for a node (b) Merged fragment and optimal ellipse fit (c) Non-overlapping (XOR) area . . .	29
2.10	(a) PRE for identifying accurately segmented nuclei (b) Details of the feature processing	31
2.11	(a) Example nucleus for boundary accuracy assessment (b) Example nucleus with control (C) (Green) and test (T) (Red) segmentation (c) The overlap area (purple) used to measure area similarity (d) Distance transform-based boundary accuracy calculation. The distance transform was calculated with respect to control segmentation (e) Normalized EDT calculation on control segmentation mask used to measure difference in relative distance measure. Control segmentation boundary is shown in white (f) The difference in normalized EDT based relative distance measure	34
2.12	PRE precision-recall plot for 1620 configurations and the best configuration (closest to (1,1)). Configurations with high precision and low recall are shown by the red box.	37
2.13	(a) Flow diagram showing steps for spatial gene localization analysis (b) Original image of segmented nucleus showing red and green FISH spots marked by arrows (c) Nucleus ROI showing segmented FISH spots (d) Euclidean distance transformed nucleus ROI showing normalized distance transform metric for each FISH spot (e) Histogram of FISH signal positions binned by normalized EDT values for aggregate cancers, aggregate normals and cancer samples C1, C10 and C12 (f) Cumulative distribution of FISH spots against normalized EDT values for aggregate cancers, aggregate normals and cancer samples C1, C10 and C12	40
3.1	(a) Texture filter 1 (b) Texture filter 2 (c) Texture filter 3 (d(i)) Schematic of portions of nuclei N1 and N2 with oriented cuboidal regions for cumulative histogram computation of B and T channels. Cuboids are exaggerated for illustrative purposes (d(ii)) Pre-rotated nuclei N1 and N2 with axis aligned cuboidal regions.	57
3.2	Synthetic 2D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7. Simulated 2D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7.	68

3.3	Single slice of synthetic 3D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7. Single slice of simulated 3D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7.	69
3.4	(a) Sample 2D image from human breast tissue dataset (b) Sample 2D image from MCF-10A dataset (c) Sample 2D image from mouse embryo dataset.	70
3.5	(a) Single slice of sample subvolume of 3D image from human breast tissue dataset (b) Single slice of sample subvolume of 3D image from MCF-10A dataset (c) Single slice of sample subvolume of 3D image from mouse embryo dataset.	70
3.6	MiPb response for synthetic 2D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7. Simulated 2D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7. .	75
3.7	MiPb response for single slice of synthetic 3D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7. Single slice of simulated 3D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7.	76
3.8	PR curves for 2D synthetic, simulated, MCF-10A Acini, Human breast tissue micro-array and Mouse embryo samples	77
3.9	MiPb response for (a) Sample 2D image from human breast tissue dataset (b) Sample 2D image from MCF-10A dataset (c) Sample 2D image from mouse embryo dataset.	78
3.10	MiPb response for (a) Single slice of sample subvolume of 3D image from human breast tissue dataset (b) Single slice of sample subvolume of 3D image from MCF-10A dataset (c) Single slice of sample subvolume of 3D image from mouse embryo dataset.	78
3.11	PR curves for 3D synthetic, simulated, MCF-10A Acini, Human breast tissue micro-array and Mouse embryo samples	80
3.12	PR curves for comparing 3D isotropic and anisotropic synthetic volumes. Both have maximum F-score of 0.86	80
4.1	(a) Object O embedded in a 3D volume in Cartesian coordinate system and corresponding variables r , θ and ϕ in spherical coordinate system. (b) Surface \mathcal{S} representing point convex object O in the spherical volume V_s (c) Edge connections of the graph shown for a single vertex $v'_{r,\theta,\phi}$ corresponding to voxel $I_{r,\theta,\phi}$ in (b). The red and green edges show connections in the θ and ϕ dimensions respectively	94

4.2	(a) Single slice of a subvolume of human breast tissue sample showing DAPI stained cell nuclei (b) Contrast stretched probabilistic edge map (c) Contours extracted using oriented watershed and ultra metric contour map. Gray scale values of the countours are proportional to their respective probabilistic edge strength (d) Blue contours and red dots corresponding to the edges and vertices of the 2D dynamic programming based segmentation graph (e) Sample graph around countour/edge a_{ij} having vertices v_i and v_j and red and green points at a distance of average sample radius on either side of the edge (f) Clockwise directed graph constructed using green point (g) Anti-clockwise directed graph constructed using red point (h) Red arrows showing gradient direction across edges a_1 , a_2 and a_3 . θ_1 and θ_2 show angle between gradient directions of parent edge a_1 to prospective child edges a_2 and a_3 (i) Angular deviation μ between parent edge a_1 and child edge a_2 (j) 2D initial rough segmentation shown as a white overlay on original image (k) Final 2D segmentation using the graphcut based method initialized using centroids of rough segmented objects.	99
4.3	Graph modifications for interactive segmentation	102
4.4	(a) Sample 2D nucleus with polar coordinates superimposed. O is the object inner seed point. (b) Polar transformed image (c) Radially outward directional gradient edge measurement in 2D. The red arrows show the negative edge measure produced by the target object and yellow arrows show the positive edge measure produced by neighboring objects which discourages the inclusion of those regions in the optimal cut.	106
4.5	Segmentation using graphcut based method of 2D simulated images with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7. Segmentation using MINS method of 2D simulated images with (f) SNR 2 Blur 0.4 (g) SNR 3 Blur 0.4 (h) SNR 3 Blur 0.7 (i) SNR 3 Blur 1.0 (j) SNR 4 Blur 0.7.	110
4.6	Segmentation using graphcut based method of 2D real images from (a) Human breast tissue (b) MCF-10A (c) Mouse embryo datasets respectively. Segmentation using MINS method of 2D real images from (d) Human breast tissue (e) MCF-10A (f) Mouse embryo datasets respectively.	111
4.7	(a,e) Single XY and YZ slices of original 3D simulated image at SNR 3 and blurring of 0.7 (b,f) Corresponding groundtruth available from synthetic object boundaries (c,g) Graphcut segmentation (d,h) MINS segmentation.	113
4.8	Segmentation using graphcut based method of 3D real images from (a) Human breast tissue (b) MCF-10A (c) Mouse embryo datasets respectively. Segmentation using MINS method of 3D real images from (a) Human breast tissue (b) MCF-10A (c) Mouse embryo datasets respectively. Only one slice of the segmentations are shown.	114

5.1	MiPipeline infrastructure showing various components of the MiPipeline environment	122
5.2	A simple ImageJ based workflow for performing nuclear segmentation and creating a backbone XML for visualization. The figure also explains the modules used for creating the workflow, the data connection lines and the module input output nodes. The red box on the left shows all pre-existing modules available on the LONI server which can be interconnected to create custom workflows.	125
5.3	(a) XML schema used or MiPipeline data provenance. (b) Visualization of a simple XML file generated by the worked example using the generic XML visualization capabilities of MiPipeline.	128
5.4	(a) ImageJ plugin based QuickPALM workflow using module <i>IJ_QuickPALM</i> (b) Average image from a sequence of PALM images of retinal pigment epithelium cells with tubules(c) Super-resolution PALM reconstruction from the sequence of images of retinal pigment epithelium cells with tubules	133
5.5	Figure showing MiPipeline PAGODA workflow	136
5.6	PAGODA standalone modules for FISH segmentation and nuclear segmentation	137
5.7	PAGODA hierarchical visualization and intermediate result display. .	139
5.8	PAGODA heat map display of dataset similarity measure along with tools for feature space visualization and filtering	140

Chapter 1: Introduction

Optical microscopy or light microscopy is a technology that uses visible light and a system of lenses to magnify, image and visualize micron and submicron level objects. Although the field has existed from the 17th century, the past few decades have seen impressive developments with applications in disciplines such as biology, material science, nanophysics, pharmaceuticals etc. Especially in cell and molecular biology, the development of a wide variety of fluorescent labels which can be used to visualize and image cellular and subcellular structures and proteins have accelerated the growth to an extent that such techniques are being routinely used now a days in laboratory, research and industrial applications. On the other hand, the development of digital imaging and video technologies was also a propelling force which resulted in rapid developments in optical microscopy-based imaging of a wider variety of biological samples including living and tissue samples. More recent advancements such as the development of super-resolution microscopy has enabled scientists to go beyond the diffraction limit of light and image nano-scale subcellular structures.

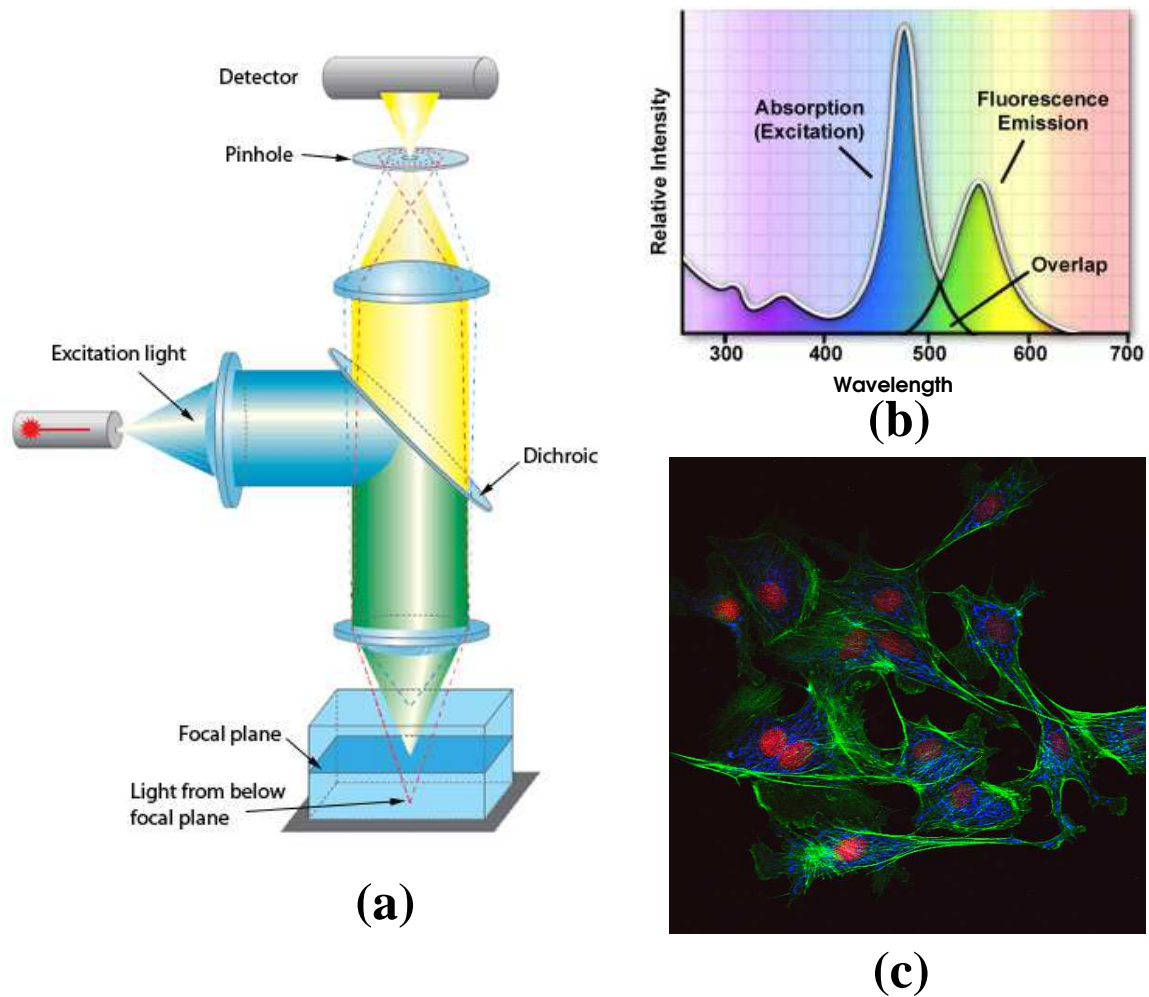


Figure 1.1: (a) Schematic showing working principle of a confocal microscope imaging a fluorescent sample. **Image Source:** <http://microscopy.duke.edu/> (b) Illustrative excitation and emission spectra of a fluorescent tag. **Image Source:** <http://www.olympusmicro.com/> (c) A 2D sample fluorescence image acquired on Zeiss confocal 510 microscope where 4',6 - diamidino - 2 - phenylindole (DAPI) stained cell nuclei are displayed in the red channel, FITC tagged actin fibers in the green channel and Alexa Fluor 594 tagged mitochondria in the blue channel.

1.1 Fluorescence and Confocal Microscopy

Biological objects of interest such as cells and sub-cellular objects are in general transparent or translucent and the amount of light transmitted or absorbed by them varies depending on their physical characteristics. Conventional optical microscopy visualizes biological samples of interest by placing them between a source of light and the image capturing device or observer and the objects are visualized as light is variably transmitted or absorbed by various internal components of the sample being imaged.

On the other hand in fluorescence microscopy the cells or sub-cellular objects of interest are tagged with fluorescent molecules with high degree of specificity which when illuminated with light of a certain wavelength emit light in a longer wavelength due to a spectral shift of the fluorescent molecules separating the intense excitation light from the dim emitted light (Fig. 1.1(b)). The use of a dichroic mirror prevents the excitation light to travel to the photon capturing device such as photo multiplier tubes or charge-coupled device(CCD) cameras while allowing the emitted light to reach the detectors (Fig. 1.1(a)). Multiple objects of interest such as cell membrane proteins, DNA, specific genes, mitochondria etc can be stained and imaged with multiple fluorophores simultaneously (Fig. 1.1(c)). When imaged with high resolution microscopes, the images provide a wealth of quantitative information such as spatial organization, protein localization, shape and morphological information not only in space but also in time when the microscopes have provisions for long term unsupervised automated motorized imaging of live samples. Characteristics of fluo-

rescence and associated tagging process such as linearity, high molecular specificity, extremely good single molecule sensitivity and ability to simultaneously detect multiple molecular species along with the fact that it allows 3D imaging of biological samples are some of the major advantages of fluorescent optical microscopy over conventional optical microscopy.

For 3D imaging of fluorescently-labeled samples, a technique called confocal microscopy (Fig. 1.1(a)) is used. Compared to a conventional wide-field microscope which captures light coming from the entire depth of a biological sample, including the out-of-focus light from outside the focal plane of the lens, a confocal microscope uses a spatial pin hole to reject all out-of-focus light. In essence, a confocal microscope in a single scan produces high resolution and sharp image of the sample across the focal plane of the imaging lens and a faithful 3D image of the sample can be acquired by moving the imaged focal plane across the depth of the sample. Hence, fluorescence and confocal microscopy in combination is a very powerful tool which is widely used in several biological applications.

1.2 Challenges in Microscopic Image Analysis and Role of Computers

For the past several decades researchers have developed methods for analyzing optical microscopy images. Studies involving cell culture models use images of cells in-vitro i.e. outside their natural biological tissue context. Images resulting from such studies are reasonably easier to analyze as the spatial density of cells can be controlled in most cases. However, analysis of tissue sample images is still a

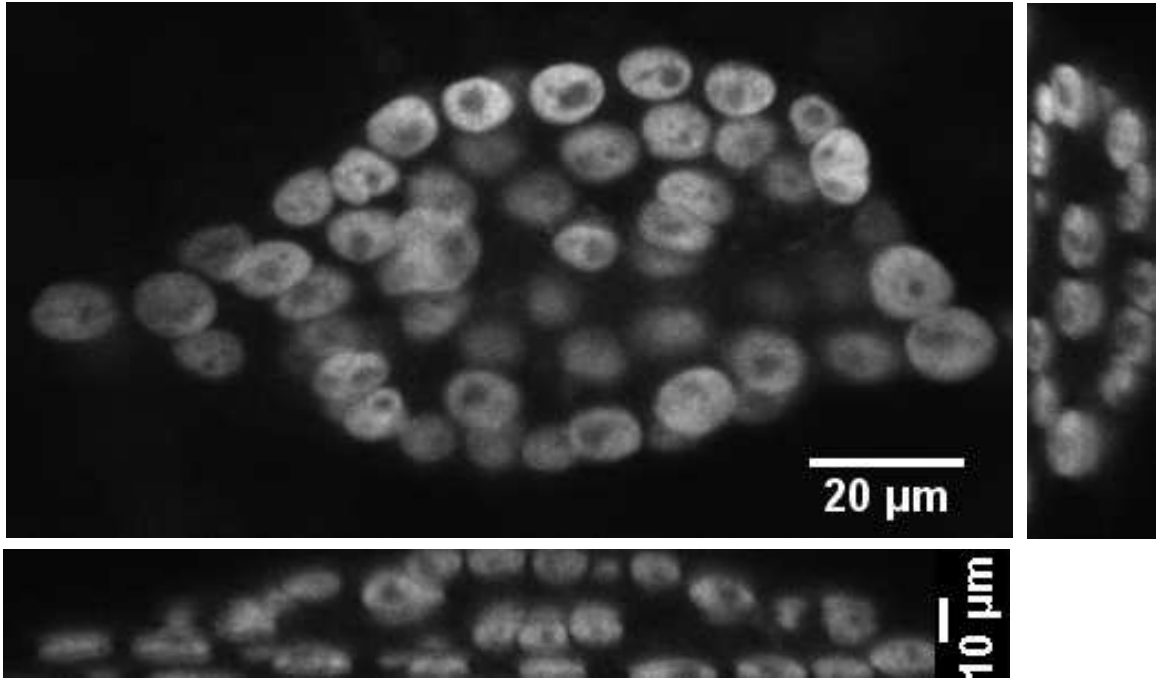


Figure 1.2: 3D image of DAPI stained cell nuclei in a MCF10A acinus with single X-Y, X-Z and Z-Y slices displaying significant spatial clustering.

challenge, especially in 3D. The practical issues in 3D include anisotropic voxel dimension due to low resolution of a confocal microscope in the depth or Z dimension, poor signal to noise ratio and blurring introduced by the inherent wave properties of light and the imaging components of the microscope. The biological tissue sample itself complicates the analysis due to issues such as non-uniform and improper staining of the sample, tight packing of the cells (Fig. 1.2) and photo-bleaching of the fluorescent molecules. Due to such issues, the object boundaries are often ill-defined and pose a formidable problem when the applications demand robust and accurate delineation of closely packed objects.

Additional challenges stem from the spectacular advances in optical microscopy in recent times which have enabled scientists to image biological samples across a

very wide range of spatial and temporal scales. A by-product of all these developments in the field is the creation of gigabyte or terrabyte size datasets. Also, the field of microscopy, for biological applications, is becoming increasingly quantitative where instead of making qualitative assessments, the multi-dimensional digital image datasets are being processed to extract, quantify and analyze biologically relevant information emphasizing the increasing need for fast, accurate, intelligent and robust automatic image analysis systems [1, 2]. However, manual processing and organization of such large volumes of data is time consuming, laborious and almost impossible in some applications where subtle feature variations are statistically significant, but visually undetectable (subvisual) [3, 4]. Also, manual processing is often subjective and introduces human bias resulting in inter and even intra observer variabilities. Hence, current state of the art applications almost always use computers or even clusters of computers to process and organize such large volumes of microscopic image data.

In general, several disciplines in computer science such as image analysis, computer vision, information and data visualization, human computer interaction and parallel and distributed computing have made significant contributions in the development of optical microscopy-based biological applications. Especially, image analysis and computer vision methods are extremely relevant since they enable us to quantify phenotypic modifications and signaling events in the spatial and temporal context. Some of the most common tasks for such an analysis are segmentation, screening and tracking of various objects of interest in 2D, 3D and 4D datasets. The objects of interest vary from cells, nuclei to subcellular structures in both cell

culture models and tissue samples.

1.3 The Problem Statement

This dissertation attempts to address the problem of **accurate understanding of individual cells in tissue context**. The problem is relevant and important because it studies cells in their natural tissue environment, the way they thrive in multicellular organisms. Understanding their coordination and interaction in such an environment is essential to understanding high level complex biological processes pertaining to cell behavior, developmental biology, disease progression, tumorigenesis and complex chemical pathways to name a few. In turn a good understanding of the aforementioned processes will help in developing effective methods for disease prognosis, diagnosis and treatment procedures. In a way it is the ultimate goal of all biological applications and optical microscopy in conjunction with rigorous quantitative analysis plays a vital role in studying such problems by enabling us to gather and understand visual information starting from the cellular to the molecular level without significantly disturbing the tissue environment.

However, the problem statement is very broad and from an image analysis perspective can be subdivided into several sub-problems such as segmentation, tracking and time resolved analysis, visualization of data and information etc. Developing the entire spectrum of methods for understanding individual cell coordination in tissue is beyond the scope of a single dissertation. Hence, in this dissertation we primarily concentrate on the sub-problem of segmenting objects of interest such

as cells, nuclei and individual genes in 2D and 3D microscopic images along with development of methods to seamlessly process, visualize and explore the wealth of information that can be extracted from large microscopic datasets. Development of reliable segmentation methods is one of the most important problems, the performance of which often dictates the performance of subsequent analysis methods such as tracking and spatial localization analysis. Also, as mentioned previously, extracting and exploring biologically relevant information and results from large microscopic datasets is a challenge and the development of effective, efficient and user friendly tools to address the issue helps in better understanding of cell behavior in terms of underlying data trends and patterns.

1.4 Contributions

The contributions of the dissertation are as follows:

- **Probabilistic Edge Detection:** Robust and accurate edge detection is a prerequisite for several image analysis and computer vision algorithms such as segmentation and tracking. We have developed a robust and accurate multi-scale, multi-orientation probabilistic edge detection method for both 2D and 3D tissue samples. To the best of our knowledge, this is the first such edge detector developed for microscopic images of tissue samples.
- **Robust 2D segmentation of cell nuclei:** We present a number of robust and accurate segmentation methods for segmenting cell nuclei in 2D tissue samples using methods such as dynamic programming, watershed and graph-

cuts. The segmentation results obtained using such algorithms were further utilized to develop a prospective human breast cancer diagnostic method by analyzing the spatial organization of specific genes within the cell nucleus. The developed diagnostic method mimics the manual processing of such datasets to detect cancer and shows that automation is possible for larger datasets obtained from real patient samples.

- **Identifying 'Well Segmented' Nuclei:** Since automatic segmentation results for tissue samples using even the best possible state-of-the-art algorithm are not 100% accurate, we introduce and implement the idea of using pattern analysis and machine learning tools to automatically identify and/or rank segmented cell nuclei in terms of the goodness of segmentation. Such information helps in improving the quality of results down stream as we can only process the subset of objects for which our confidence on the accuracy of segmentation is high.
- **Optimal 3D segmentation of cell nuclei:** Lack of effective tools to segment cell nuclei in 3D tissue samples has motivated us to develop an optimal graphcut-based segmentation algorithm for 3D microscopic images. Both automatic and interactive versions of the algorithms have been developed and experiments show the superior performance of the method when compared to a recently reported segmentation method for similar datasets.
- ***MiPipeline* for high throughput computing and informatics:** We have developed an user friendly environment called *MiPipeline* to process large mi-

croscopic datasets using distributed computing resources along with a seamless web based information visualization and informatics interface. It addresses the need for computing and visualization environments guaranteeing high throughput processing, data and process provenance, access to a wide variety of algorithms relevant for analyzing microscopic images, an environment for visual programming, workflow management and intuitive information visualization and mining capabilities.

1.5 Organization

The dissertation is organized as follows. Chapter 2 presents watershed algorithm-based segmentation methods for 2D tissue sample images along with the development of a supervised framework to identify or rank the segmented objects in terms of their goodness of segmentation. We also present a method for detecting human breast cancer by analyzing spatial organization of certain genes in the cell nucleus. Chapter 3 presents the development of a 2D and 3D probabilistic edge detection method for tissue samples. A graphcut-based 2D and 3D nuclear segmentation method is presented in chapter 4. Chapter 5 illustrates *MiPipeline*, an environment for high throughput computation and information visualization for large microscopic datasets. Concluding remarks and future research directions are included in chapter 6.

Chapter 2: Segmentation and Selection of Cell Nuclei in 2D Tissue Images for Breast Cancer Detection

2.1 Introduction

In this chapter, we present nuclear segmentation and selection algorithms in 2D images of human breast tissue samples which helped in the development of an automatic breast cancer detection methodology based on spatial localization analysis of certain genes in the cell nucleus. Due to large morphological and textural variations in sample images, standard segmentation algorithms (e.g., graph cuts, watershed on gradient-magnitude) failed to accurately segment a significant number of nuclei in each dataset. The nuclear segmentation algorithms presented here could handle the significant nuclear variations among datasets in a robust way, resulting in a satisfactory yield of well-segmented nuclei for analyzing the spatial positioning of the genes. In chapter 5 methods for applying this cancer detection technique to large microscopic image datasets will be illustrated.

Breast cancer is the second most common cancer in women. It is estimated that approximately 1 in 8 women in the US will be diagnosed with breast cancer and it is the primary reason behind the death of approximately 40,000 women, annually [5].

Nevertheless, there have been major improvements in the past decade which have caused these numbers to decline, largely, due to: (i) increased awareness, (ii) early detection/screening, and (iii) treatment advances. Early detection has been the focus of extensive research and there is accumulating evidence that if breast cancer is diagnosed early, the average survival rate can be extended to 98%, from only 23% if the cancer has already metastasized before being diagnosed [5].

Recently, it has been shown that genetic alterations to normal cells play a significant role in causing cancer [6]. Concurrently, due to improvements in optical microscopy and fluorescent labeling, it has been shown that the cell nucleus is compartmentalized into well-defined subregions and that the spatial position of genes in the nucleus correlates with their expression and cellular activities [7–9]. Furthermore, the positioning of certain genes, such as *HES5* and *FRA2*, has been shown to differ between normal and cancer cells in a cell culture model of cancer [10] and in patient tissue sections [11]. This discovery could emerge as a diagnostic and/or prognostic tool for breast cancer.

Determining gene positioning begins with fluorescent tagging of specific genes in the chromosome using a technique called fluorescence *in situ* hybridization (FISH) and subsequent multichannel optical microscopic imaging of DNA counterstained nuclei, where the FISH signals appear as punctate spots in a separate channel from the nuclei. Since discerning preferential gene positioning is virtually impossible by visual examination, as evident from the sample images displayed in Fig. 2.1, quantitative analysis of the images is required. This involves : (i) accurate nuclear segmentation, (ii) detection of FISH signals, and (iii) spatial localization of the FISH

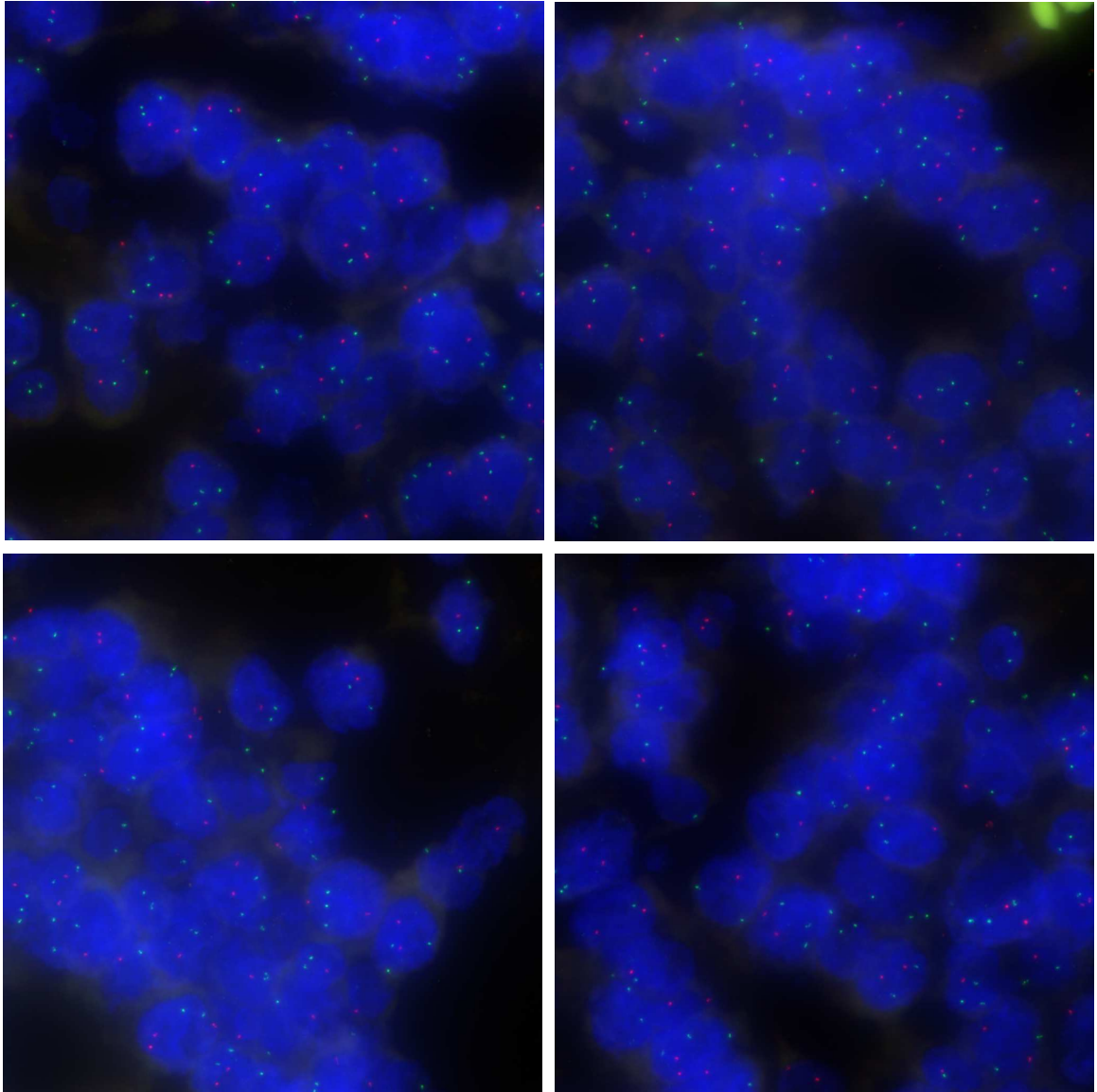


Figure 2.1: Sample human breast tissue images used for cancer detection. The DAPI stained cell nuclei are shown in the blue channel while the red and the green channels show the FISH tagged genes *HES5* and *FRA2* respectively as bright spots. signals with respect to the nuclear center and boundary. Meaburn et al., showed using manual image analysis of 100 to 200 nuclei per sample that this method reliably detected breast cancer across a panel of 11 normal and 14 cancer samples

[11], using the non-parametric Kolmogorov-Smirnov (KS) test to distinguish spatial gene localization between samples. Scaling up this approach for high-throughput clinical studies involves analysis of hundreds of thousands of nuclei across hundreds of normal and cancerous patient tissue samples. Analysis of this nature would be too time consuming, subjective and tedious if done manually, thus warranting automation. Hence the goal of this study was to automate the procedure of Meaburn et al. The method mimicked, as closely as possible, the manual analysis procedure presented in [11] in the form of an integrated workflow to automatically segment nuclei in tissue-section images followed by a supervised pattern recognition engine (PRE) to screen out well segmented nuclei, with a high degree of confidence.

Cell and nuclear segmentation in histopathology and fluorescence microscopy is an active area of research, resulting in the development of several automatic [12, 13] and semi-automatic [14, 15] strategies. The majority of these methods, in general, employ a pre-processing step for noise reduction and intensity/gradient-based thresholding for foreground identification. This is followed by independent or combined application of segmentation algorithms (e.g. gradient based methods [16, 17], active contours, watershed and dynamic programming [14, 18]) for delineation of individual objects within the foreground. Watershed-based algorithms [19, 20], which split thresholded objects into fragments, are often referred as "universal segmenters", but oversegmentation is a common problem requiring subsequent merging strategies [21]. Active contours are generally considered to be the state-of-the-art due to high accuracy, adaptability to image topology and ability to incorporate regularity features [22, 23]. However, computational load, precise initialization and numerical

instability are major concerns with this approach.

Most of the aforementioned segmentation techniques yield desirable results for specific cell culture images, but their extension to complex tissue sections have been less promising, especially for cancer tissue where there is considerable variation in the morphological and textural properties of nuclei along with severe nuclear clustering. However, the nuclear segmentation requirements for our application are different, because thousands of nuclei are available for imaging in each tissue section of which less than 10% (100 to 200 per sample) are needed for detecting cancer based on gene positioning. Thus our requirement is highly accurate segmentation of only a subset of nuclei rather than attempting to segment as many nuclei as possible. Therefore, we built a computational framework that uses a supervised pattern recognition engine (PRE) to perform the task of selecting accurately delineated nuclei from the segmentation algorithm.

Pattern recognition and machine-learning principles have been proven to be useful in several quantitative imaging applications related to cell biology [1,24] and more specifically in breast cancer (e.g., Wisconsin Breast Cancer Database [25,26]). Some of the relevant biological applications of PREs at the cellular and subcellular levels include classification of cells and nuclei based on their morphological, textural and appearance features [27,28] and interpreting and analyzing localization of proteins, antibodies and sub-cellular structures within the cell [29–32]. For instance, Hill et. al. [33] assessed the impact of imperfect segmentation on the quality of high content screening (HCS) data using a support vector machine (SVM)-based PRE to identify accurately delineated nuclei. In a similar line, to make the segmenta-

tion algorithm itself intelligent and data driven, Gudla et. al. [34] used classifiers interleaved with the segmentation algorithm to identify optimal parameters for segmenting and selecting accurately delineated nuclei in cell culture images.

The rest of the chapter is organized as follows. Section 2.2 provides a description of the samples and images followed by an explanation of the pre-processing steps in section 2.3. Section 2.4 outlines a hybrid algorithm for nuclear segmentation and a stacked classifier based PRE for identification of a subset of 'well-segmented' nuclei. Section 2.5 will describe a multi-stage watershed-based nuclear segmentation and artificial neural network based PRE which improves the nuclear yield for cancer detection purposes followed by the performance results of the reported PRE's and the boundary accuracy assessment in comparison to a 2D dynamic programming (DP)-based segmentation algorithm [18] that serves as a reference. The gene localization based breast cancer detection methodology using the nuclear segmentation and screening methods is described in section 2.6 along with associated performance results. Finally, section 2.7 provides discussion, draws conclusions and comments on the future directions for the work.

2.2 Samples and Images

Sample preparation and labeling was described in reference [11]. 4-5 μm thick formalin fixed, paraffin embedded human breast tissue sections were imaged using an Olympus IX70 microscope controlled by a Deltavision System (Applied Precision, Issaquah, WA, USA) with SoftWORX 3.5.1 (Applied Precision) and fitted with a

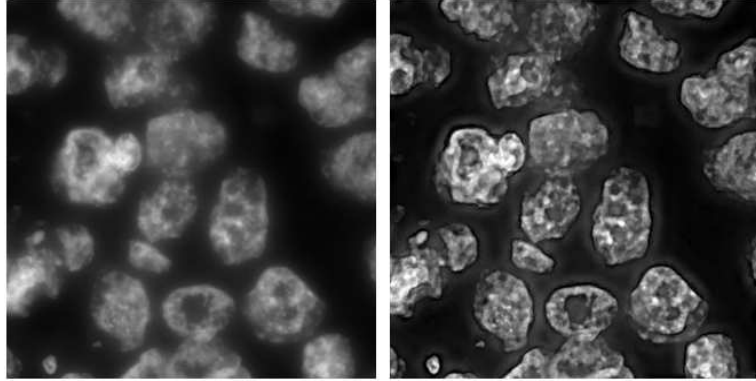


Figure 2.2: (a) Original blue DAPI channel (b) Wavelet-based preprocessing output

charge-coupled device camera (CoolSnap; Photometrics, Tucson, AZ, USA), using a 60X, 1.4 oil objective lens and an auxiliary magnification of 1.5. Non-confocal 3-D Z-stacks were acquired with a step size of 0.2 or 0.5 μm . The image size was 1024×1024 , with a pixel size of 0.074 μm in both X and Y directions. For this study, the fields of view to be acquired were manually selected and focused. Large regions of inter-connective tissue were not imaged, to increase the number of epithelial nuclei acquired. Beyond this, the fields of view were randomly selected over the tissue to reduce bias based on FISH signals or tissue morphology/heterogeneity.

The red and green FISH channels encoding the spatial localization of the genes were deconvolved using SoftWORX 3.5.1 which reduced background noise. Deconvolved version of the nuclear channel, though available, was not used due to increased texture that negatively impacted the segmentation algorithm. All three channels were reduced from 3D to 2D using maximum intensity projection (MIP). By manual analysis and inspection of the resulting MIPs, the increased step size of 0.5 μm gave identical results to 0.2 μm . Although 3D analysis would provide more

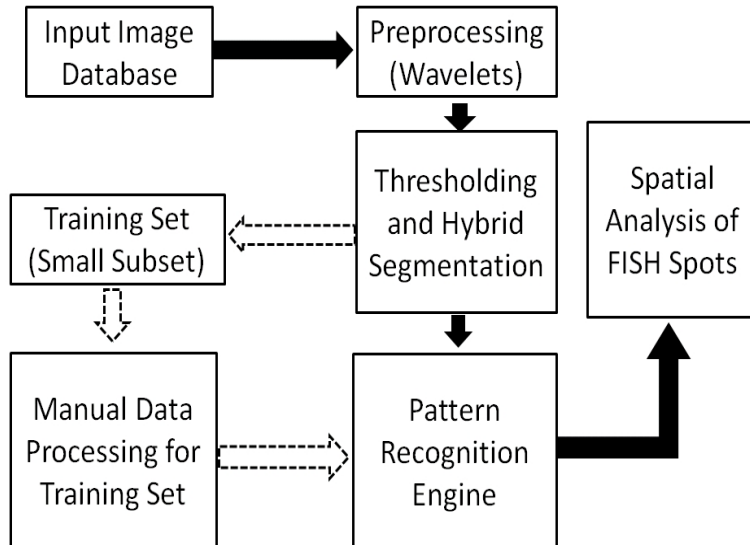


Figure 2.3: System Block Diagram

accurate gene position measurements the image acquisition method adopted here had been shown to produce accurate results from manual analysis [11].

Experiments involving the hybrid segmentation and stacked classifier-based PRE used 9 datasets (D1-D9) consisting of normal and cancerous breast tissue sections. For experiments involving the multistage watershed nuclear segmentation and ANN-based PRE, 23 datasets (500 images) were used of which 4 contained normal (N1-N4), 14 contained cancer (C1-C14) and the rest contained non-cancerous breast disease (fibroadenoma and hyperplasia) tissue section images (B1-B5).

2.3 Image Preprocessing

To improve nuclear segmentation accuracy, boundaries of foreground objects (e.g., cell nuclei) were enhanced using a modified version of Mallat-Zhong’s *extrema* algorithm [34, 35]. This wavelet preprocessing step involved: (i) using a bicubic

spline wavelet to decompose and identify the extrema in the DAPI-channel up to 5 scales; (ii) multiplying the chain-coded extremas (edges) in scales 2 to 4 by a factor of 3; (iii) and using the enhanced extrema in the wavelet reconstruction step. Since this processing was isotropic, noise and structures orthogonal to nuclear boundaries were also enhanced. This undesirable effect was ameliorated by smoothing with an edge preserving adaptive Gaussian filter [36] of standard deviation of 2 pixels and 0 pixels along the direction of edges and in the direction perpendicular to the edges, respectively. Fig. 2.2(a) shows an original nuclei channel image and Fig. 2.2(b) shows the same after preprocessing.

2.4 A Hybrid Algorithm for Nuclear Segmentation and Selection

The initial analysis method used a hybrid data-driven segmentation algorithm to accurately segment a subset of nuclei. The pattern analysis system was a stacked classifier which combined the output of multiple classifiers [37] to select the accurately segmented nuclei. The individual nuclei thus obtained were then used for automatic FISH segmentation and spatial statistical analysis to detect breast cancer.

Fig. 2.3 shows the block diagram of the proposed image analysis framework. The contrast enhanced images were binarized using a combination of the *isodata* and *triangle* thresholding algorithms available in DIPImage [38]. Morphological operations of binary closing and opening along with a size based screening removed small objects resulting from noisy background and texture within the nuclei.

Labeling the foreground objects in the processed thresholded image provided a good indication of the regions containing the nuclei in the image. However, the boundaries of the nuclei in the thresholded image were inaccurate and were not satisfactorily close to the actual object boundaries. To further improve the boundary accuracy a level set-based algorithm [39] was used in which each individual boundary from the thresholding operation evolved tightly around the visually perceived object boundaries. This method was a variational formulation for geometric active contours that forced the level set to be close to a signed distance function. The formulation consisted of an internal energy term that penalized the deviation of the level set function from a signed distance function and an external energy term in the form of the image gradient magnitude that drove the motion of the zero level set towards desired image features. Considering ϕ as a signed distance function plus a constant

$$\mathcal{P}(\phi) = \int_{\Omega} \frac{1}{2} (|\Delta\phi| - 1)^2 dx dy, \quad (2.1)$$

is a metric to measure how close ϕ is to a signed distance function in $\Omega \subset \mathbb{R}^2$. The variational formulation is

$$\mathcal{E}(\phi) = \mu \mathcal{P}(\phi) + \mathcal{E}_m(\phi), \quad (2.2)$$

where $\mu > 0$ controls the effect of penalizing the deviation of ϕ from a signed distance transform. $\mathcal{E}_m(\phi)$ is the energy term that drives the motion of the zero level curve of ϕ . The evolution equation

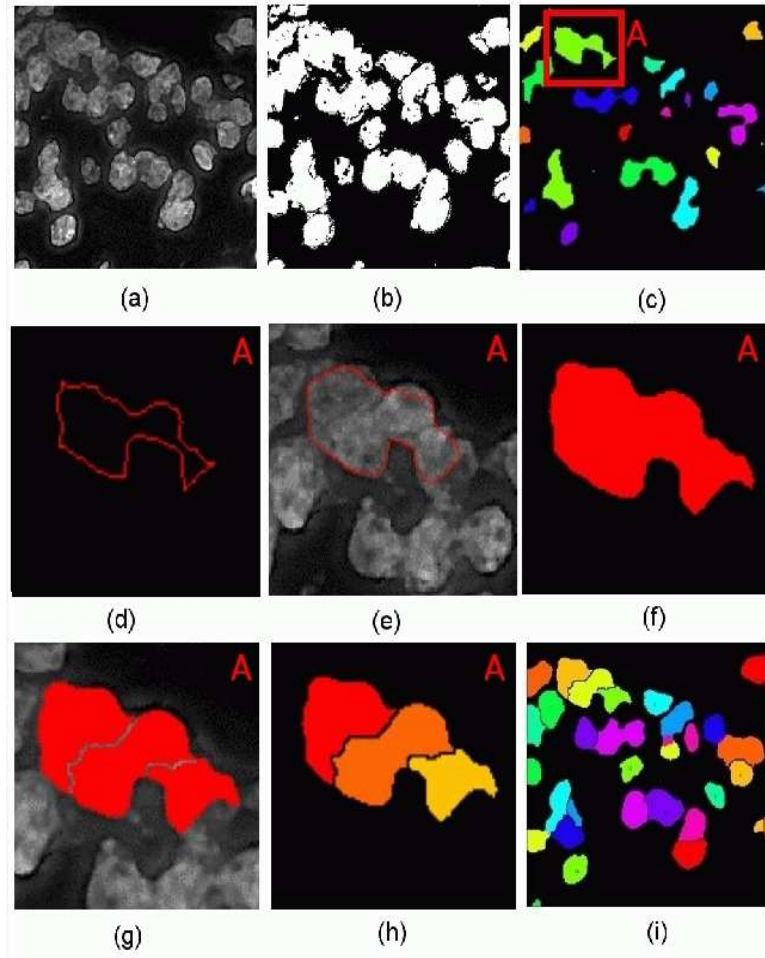


Figure 2.4: (a) Original DAPI channel after preprocessing (b) Thresholded image (c) Labeled image after morphological operations showing the input seeds for the level set segmentation algorithm with a single seed highlighted in red box A (d) Initial contour for level set algorithm for A (e) Level set evolved contour overlaid on the DAPI channel for A (f) Binary version of the evolved level set contour for A (g) Output image after applying watershed segmentation algorithm on the level set output region for A (h) Labeled version of the watershed output for A (i) Final segmentation on the entire image using the hybrid segmentation algorithm

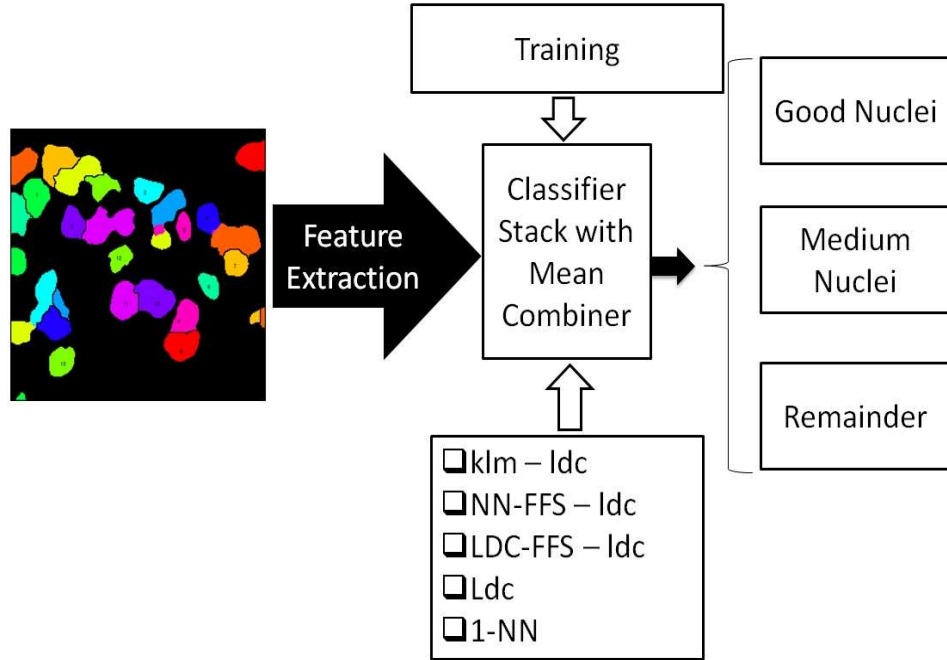


Figure 2.5: Pattern recognition module showing the stacked classifier for identifying nuclei that can be used for the FISH analysis

$$\frac{\partial \phi}{\partial t} = -\frac{\partial \mathcal{E}}{\partial \phi}, \quad (2.3)$$

is the gradient flow that minimizes the overall energy functional \mathcal{E} .

Until this point no effort was made to separate clustered nuclei and in a number of cases, the level set boundaries surrounded clusters of nuclei. A subsequent step attempted to break up the nuclei clusters into individual nuclei using the watershed algorithm [40]. As a post processing step the fragments of the watershed output were merged using a preset size value of nuclei. Fig. 2.4 shows the intermediate results for the segmentation procedure.

2.4.1 Stacked Classifier-based Pattern Recognition Engine

The pattern recognition engine selected the subset of accurately segmented nuclei. Fig. 2.5 shows the steps involved in the approach. We used a supervised classifier and the training was done on a subset of 5 segmented images from each of the 9 tissue section datasets. This was about 25% of the entire dataset. The training set was further partitioned such that 80% was to be used as training set and the remaining 20% as validation set. The segmented objects in the training and validation set were manually classified into 3 classes: 'Good Nuclei' (nuclei segmented almost perfectly), 'Medium Nuclei' (nuclei having small boundary inaccuracies) and the 'Remainder' (objects never used for subsequent analysis).

For any pattern recognition engine to work well, the feature space used for representing the object features has a vital role to play. In this case shape (perimeter to area ratio, Feret diameters), texture (mean intensity, intensity standard deviation), size (size, perimeter) and other morphological cues were used as the feature set to identify a well segmented nuclei. The dimensionality of the feature space was 24.

Fig. 2.5 shows the stacked classifier structure combining: (i) linear discriminant classifier based on the principal component analysis reduced space capturing 95% of the variance space (klm-ldc); (ii) linear discriminant classifier on the best 3 features selected by 1 nearest neighbor leave-one-out error (NN-FFS-ldc) method; (iii) linear discriminant classifier on the best 3 features selected by linear discriminant classifier leave-one-out error (LDC-FFS-ldc); (iv) linear discriminant classifier (Ldc); and (v) 1-Nearest neighbor classifier(1-NN). The stacked classifier was used

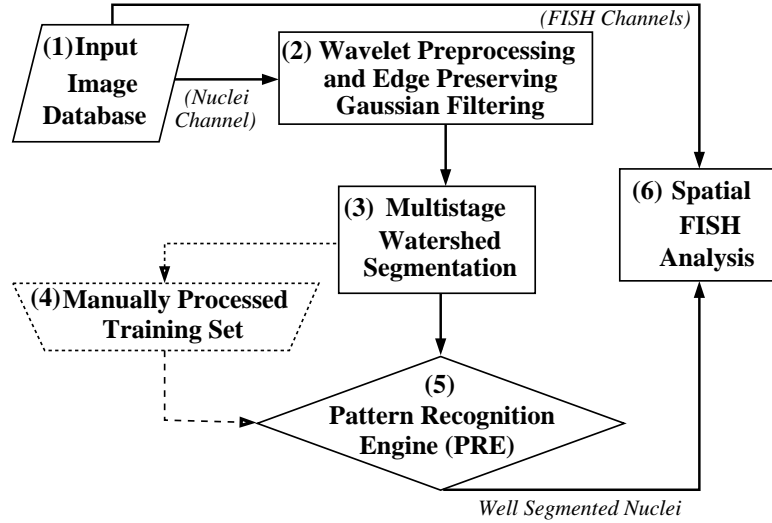


Figure 2.6: Flow diagram showing the computational framework

to harness the feature extraction power of all the classifiers which often show a complementary discriminating behavior. The classifier was trained on the manually classified training set and then validated on the validation set to identify the combiner to be used for the stacked classifier. Product, mean, median, maximum, minimum and voting combiners [37] were tested. Mean combiner performed the best among the 6 providing 93% correct classification on the validation set.

Our simulations with the training and validation set showed that the stacked classifier performed better than the individual classifiers. One major aim of this work was to design the pattern recognition engine so that it can identify the well segmented nuclei with a high degree of accuracy and confidence, since the immense variation in nuclear features makes it practically impossible to accurately segment every nucleus in an image.

2.5 Multistage Watershed Based Segmentation and Artificial Neural Network based PRE

As a more robust and accurate alternative to the hybrid levelset-watershed algorithm and stacked classifier workflow, a multistage watershed-based segmentation algorithm was subsequently developed along with an artificial neural network (ANN) based PRE. Fig. 2.6 illustrates in block diagram form the computational steps for nuclear segmentation and the associated PRE for the identification of "well-segmented" nuclei used in spatial FISH analysis. The general framework is very similar to the previously presented hybrid segmentation scheme. The nuclei channel of the original images (Fig. 2.6 - 1) were first wavelet preprocessed (Fig. 2.6 - 2) and segmented (Fig. 2.6 - 3). A small portion of the dataset was manually processed and used as the training set (Fig. 2.6 - 4) for the PRE (Fig. 2.6 - 5). The set of accurately segmented nuclei was then used for spatial analysis of the gene (Fig. 2.6 - 6).

2.5.1 Segmentation of nuclei

The multistage watershed nuclear segmentation algorithm is outlined in Fig. 2.7. It first identifies nuclear foreground regions to be used for watershed in subsequent steps, by performing an entropy-based texture filtering step followed by iterative isodata thresholding [41].

A seeded watershed algorithm [42] was used on the thresholded preprocessed

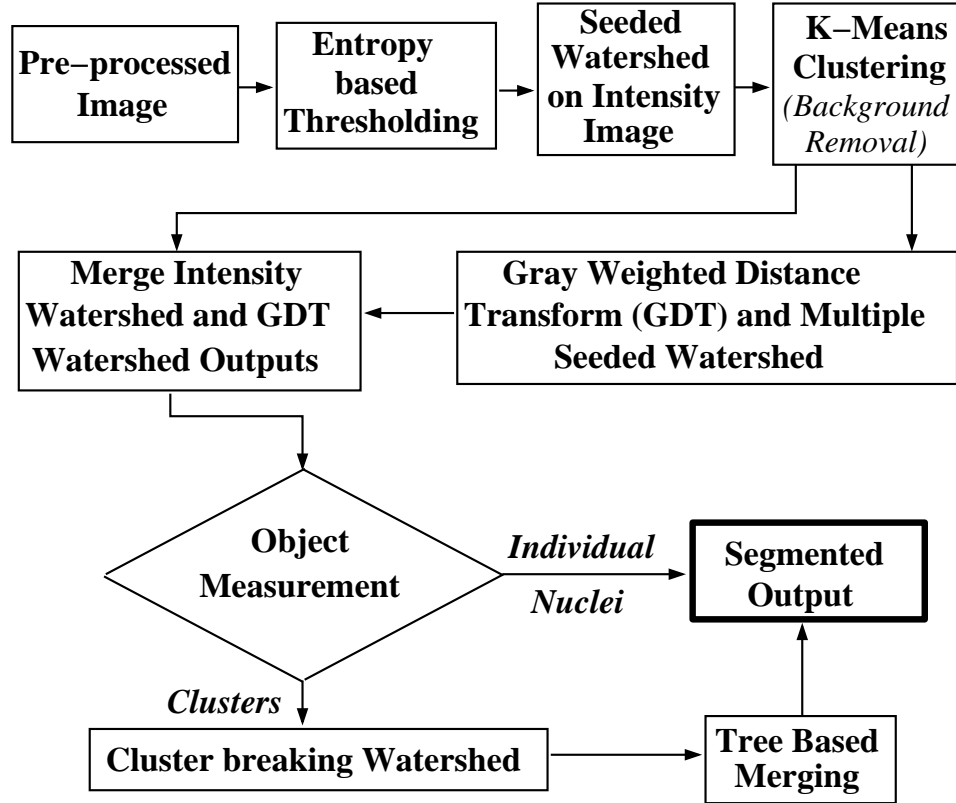


Figure 2.7: Multistage watershed segmentation algorithm

intensity image to split the nuclear foreground regions into individual objects. Although the intensity-based watershed technique found object boundaries accurately, it over-segmented most of the objects. The seeds were identified from the same image using an extended-maxima transform, which was the regional maxima of the morphological reconstruction based H -maxima transform [43]. Intensity variations within nuclei made it very difficult to identify unique local maxima for each individual nuclei, resulting in over or under segmentation. Hence for more reliable segmentation, we performed maxima identification using multiple values of H followed by seeded watershed. Watershed boundaries, that appeared at all H values were retained as the prospective edges (Fig. 2.8(iii)). To remove spurious fragments

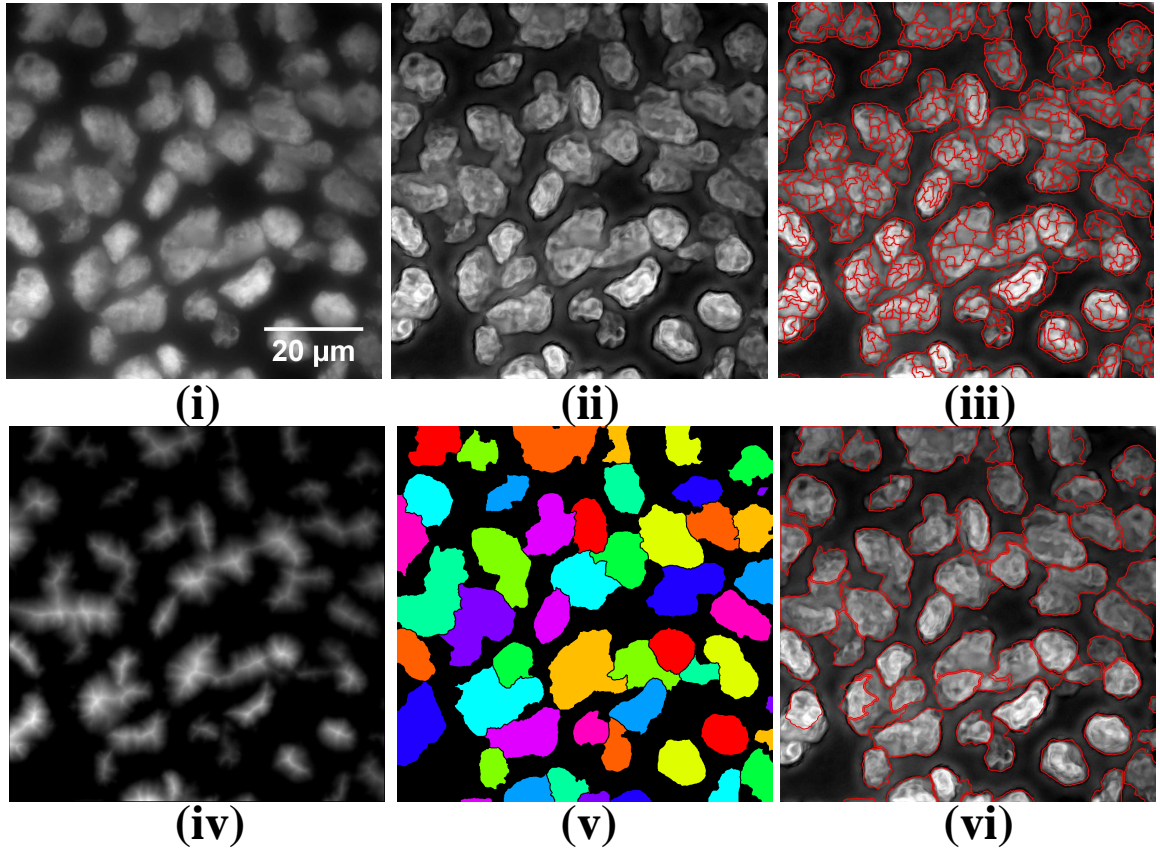


Figure 2.8: Representative image and the corresponding outputs at different segmentation steps. (i) Original DAPI channel nuclei image (ii) Preprocessed nuclei channel (iii) Seeded intensity watershed output on image foreground (iv) Gray weighted distance transform (GDT) output (v) Merged output of intensity and GDT watershed (vi) Final segmentation output after the cluster breaking watershed and tree based merging

in the background a k-means intensity based clustering [44] with 5 cluster centers was performed on the watershed output and the cluster having the lowest intensity average per pixel was rejected.

In order to merge nuclei fragments from the first watershed, we took advantage

of the known morphology of nuclei through the use of the gray weighted distance transform (GDT) [45]. After applying the GDT (Fig. 2.8(iv)), the aforementioned seeded watershed was repeated to identify high intensity GDT-transformed nuclei regions. However, in this case edges that appeared for more than 40% of the H values were retained as prominent edges. Although this method identified the general location and extent of the high intensity objects, boundary accuracy of the segmented objects was low since the method was not performed directly on the preprocessed image. Hence the output of the GDT and intensity watershed were combined as follows. Each intensity-based watershed fragment was associated with the GDT-based watershed fragment to which it had highest overlap. Then intensity-based watershed fragments associated with the same GDT-based fragment were merged into a single object. The resulting segmentation (Fig. 2.8(v)) was more accurate compared to the segmentation results from either the intensity or GDT watershed, which was visually verified for a large subset of the data.

The previous steps often failed to segment nuclei in large clusters. Thus, the next step identified such clusters using size and shape factor (normalized perimeter squared-to-area-ratio (P2A) which is 1.0 for a perfect circle) values. It was observed that the average size of a nucleus across datasets was around 10,000 pixels with considerable variation among datasets, and the average P2A value was 1.2 for well segmented nuclei and ranged between 1 and 1.4. A large and irregular cluster was hence defined as one having size $\geq 10,000$ pixels and $P2A > 1.4$. Clusters were split by application of the non-seeded watershed to the cluster ROI of the preprocessed image.

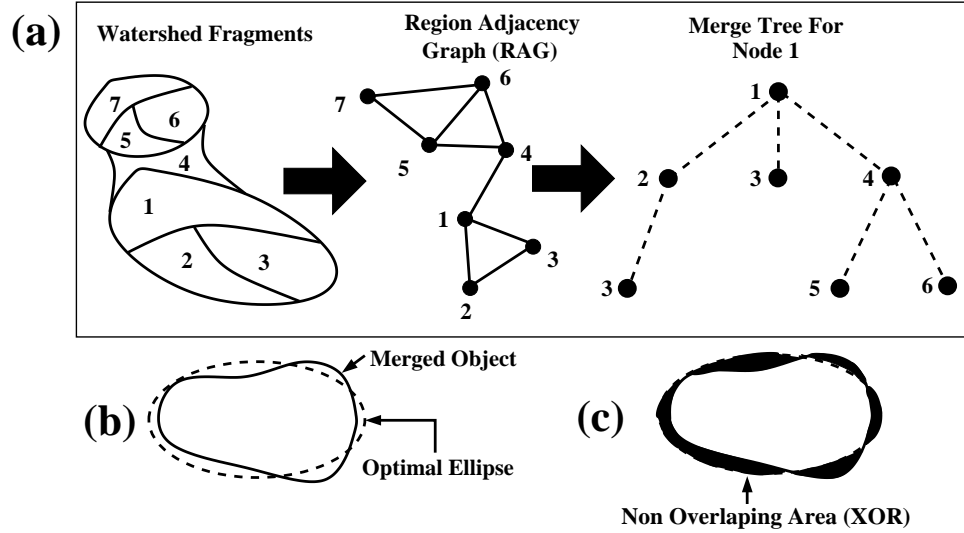


Figure 2.9: (a) Process of building up the merge tree for a node (b) Merged fragment and optimal ellipse fit (c) Non-overlapping (XOR) area

Due to nuclear size variations across datasets, the watershed algorithms, which partially depend on size criteria, over-segmented potentially good nuclei. Therefore a tree-based hierarchical merging strategy coupled with nuclear shape modeling [46] was used to merge oversegmented fragments. Briefly, the procedure built a region adjacency graph (RAG) of neighboring fragments from which a merging tree was created, for a given node (Fig. 2.9(a)). From the merge tree each combination of fragments were merged and an optimal ellipse fitting was performed (Fig. 2.9(b)). If the overlap of the object and the optimal ellipse ($1 - [\text{Non Overlapping Area (XOR)} / \text{Area of Merged Object}]$) (Fig. 2.9(c)) was more than 80%, the objects were merged. The final output of the segmentation module after merging is shown in Fig. 2.8(vi).

2.5.2 ANN based Pattern Recognition Engine: Feature Measurement and Selection

As an enhancement to the previously reported stacked classifier, an artificial neural network(ANN) based supervised PRE was developed to select well segmented nuclei. The problem was posed as a two-class classification problem with class-1 and class-2 representing accurately segmented nuclei and remaining segmented objects, respectively. The workflow for the PRE is shown in Fig. 2.10.

In order for the PRE to perform well, the feature set must capture pertinent characteristics of accurately segmented nuclei, which was done using a 64-dimensional feature set (refer appendix A.1 for the full list). The features comprised of 3 groups: shape-based (e.g. Feret diameters, P2A, PodczeckShapes [47], size, ratio of object area to convex hull area), intensity-based (e.g gray inertia, mean intensity, intensity standard deviation) and texture-based (e.g. Haralick texture features) [48].

Since ANNs require feature normalization to ensure numerical stability and to overcome problems such as neural network saturation while training with the backpropagation algorithm [44, 49], features were processed in the following 3 steps (Fig. 2.10(b)). In step one, five normalization techniques [50] were tested, namely (i) linear scaling to unit range, (ii) Z-Score scaling, (iii) linear scaling to unit variance, (iv) transformation to uniform distribution and (v) rank normalization. In step two, dependency ranking [51] was used to select features that to some extent correlated with the output classification. Dependency ranking was calculated using :

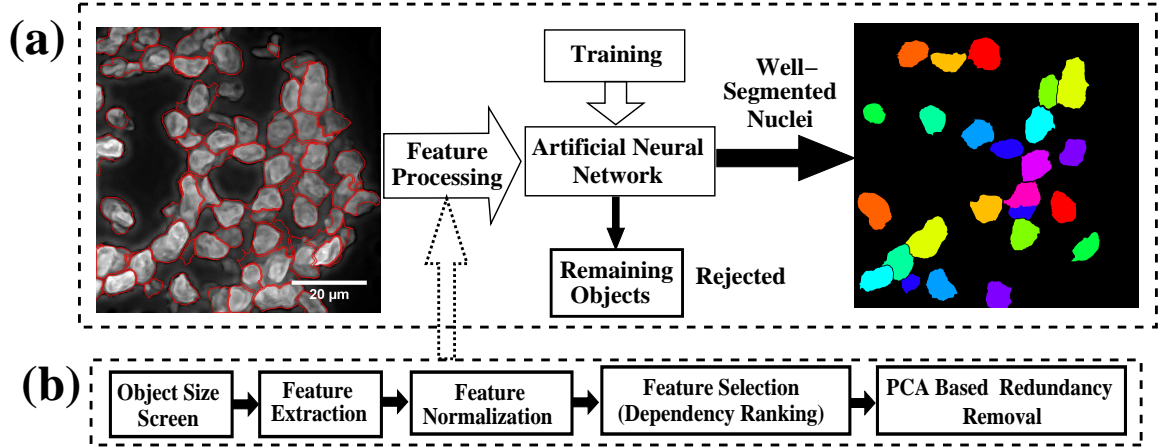


Figure 2.10: (a) PRE for identifying accurately segmented nuclei (b) Details of the feature processing

$$D(i) = \iint p(x_i, y) \left| \log \frac{p(x_i, y)}{p(x_i)p(y)} \right| dx_i dy,$$

where $D(i)$ is the dependency ranking score, x_i is the value of the i^{th} feature and y is the vector of output labels. The ranking provided a correlation score (dependency values) for each feature relative to the output classes which is based on the mutual information between the feature vectors and the output label vector. The third step reduced the number of features selected from dependency ranking using principal component analysis (PCA). This step removed redundancies caused by strong correlation between features.

2.5.3 Object Classification using an Artificial Neural Network

We had observed in 3D space considerable overlap of the three most significant features from PCA between correctly and incorrectly segmented nuclei, which in

turn meant that a nonlinear discriminant function was required to discriminate between the two object classes. Hence, we used an ANN that comprised 3 layers : input layer, single hidden layer containing neurons with *tansigmoidal* transfer-function and an output layer with a linear transfer-function. ANN training used 45 images (10% of the entire data), where correctly segmented nuclei had been manually identified. Training was performed by the *Levenberg-Marquardt* back-propagation training algorithm [52], the Conjugate gradient backpropagation with *Powell-Beale* restarts [53] or Resilient backpropagation [54] from the Neural Network Toolbox in MATLAB 2008a [55].

2.5.4 Performance Assessment of Nuclear Segmentation and PRE

Performance of the processing pipeline was assessed in 2 ways. The first measured the accuracy of identifying well segmented nuclei in a validation set of images using precision recall plots for all 1620 configurations of the PRE by varying the number of neurons in the hidden layer of the ANN (9 settings), the normalization method (5 normalizations and no-normalization), number of PCA dimensions (5 settings) and number of features selected using dependency ranking (6 settings). Precision and recall were defined as $TP/(TP + FP)$ and $TP/(TP + FN)$ respectively where TP = True Positive, FP = False Positive and FN = False Negative. Since ideally both precision and recall should equal 1, the PRE configuration closest (in terms of Euclidean distance) to the point (1, 1) was selected as the best possible configuration. This implicitly decided the cut off for the features used from the

feature selection procedure.

The second way measured the boundary delineation accuracy of automatically segmented nuclei by comparison to control segmentations generated by human interaction. However, given the notorious tedium in precisely delineating nuclei by hand, an efficient 2D DP based semi-automatic algorithm (SAA) [18] was used to generate control segmentations. Thus, the initial task was to validate the accuracy of SAA using synthetic control images of nuclei. Images of 20 synthetic nuclei were created by starting with 20 manually segmented nuclei from tissue sections, in order to capture the typical morphology of actual nuclei. Then known distortions of the image acquisition process, such as blurring and noise were estimated from the actual tissue images and were used to distort the idealized nuclei images. Background and nuclear intensities were set to 16 and 90 respectively. Next these bi-level images were Gaussian blurred by the lateral resolution of the microscope given by $0.51\lambda/NA$ where λ is the emission wavelength of the DAPI channel (450 nm) and NA (1.4) is the numerical aperture of the objective lens. Taking into consideration a pixel resolution of 74 nm, the standard deviation of the Gaussian was 0.9407 pixels. The noise level in tissue images was estimated by subtracting a Gaussian blurred version of the images from original images and calculating the variance, resulting in a standard deviation of 3.3 that was added to the synthetic images as Poisson noise.

Three parameters were used to measure boundary delineation accuracy. The first measured the overlap between test and control segmentations using area similarity (AS) [56–59] defined as $(2 \times A[T \cap C]) / (A[T] + A[C])$ where $A[\cdot]$ is the area of

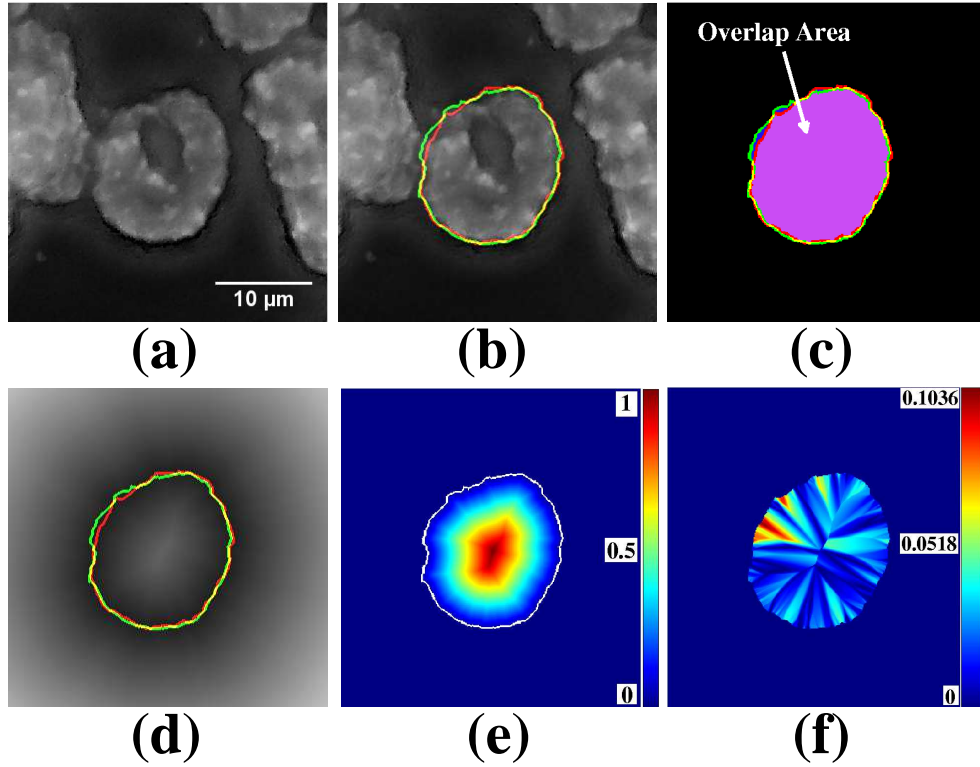


Figure 2.11: (a) Example nucleus for boundary accuracy assessment (b) Example nucleus with control (C) (Green) and test (T) (Red) segmentation (c) The overlap area (purple) used to measure area similarity (d) Distance transform-based boundary accuracy calculation. The distance transform was calculated with respect to control segmentation (e) Normalized EDT calculation on control segmentation mask used to measure difference in relative distance measure. Control segmentation boundary is shown in white (f) The difference in normalized EDT based relative distance measure

an object, \cap is the intersection of two objects, T is the test segmentation mask and C is the control segmentation mask. Thus, AS ranged from 1 for perfect agreement between test and control segmentations, down to 0 for no overlap between test and control and for cases where either the test or control segmentation did not exist.

This metric provided a combined accuracy from all the automatically selected nuclei including the false positives. Consequently, it was the true accuracy measure for the automatic analysis procedure. Fig. 2.11(a) shows a sample nucleus and Fig. 2.11(b) the two boundaries (control and test) overlaid on it. Fig. 2.11(c) shows the overlap area used to calculate AS. The second parameter was an Euclidean distance transform (EDT) based boundary metric explicitly designed to measure error at the nuclear boundary. For each nucleus the EDT was performed on the control segmentation with progressively higher values assigned to pixels farther from the control boundary. Pixel values in the EDT image at the position of the test segmentation boundary were averaged to calculate boundary error. Fig. 2.11(d) shows boundaries superimposed on the EDT image calculated with respect to the control segmentation boundary. The third parameter measured the normalized mean spatial deviation of all pixels in the overlapping area of the control and test segmentations of a nucleus. It was used to assess the effects of nuclear segmentation inaccuracies on the gene localization measurement. The EDT assigned a value of 0 to the boundary locations of the nucleus and increasing values to points further within the nucleus (Fig. 2.11(e)). The parameter was the mean of the absolute differences between the EDT images of the control and test segmentations where the two segmentations overlapped. Fig. 2.11(f) shows a heat map of the differences. Correlation analysis was performed between the 3 parameters to determine whether the parameters measured independent features of segmentation errors or alternatively whether one parameter would suffice.

2.5.5 Segmentation and Classification Results

The proposed image analysis steps were implemented in MATLAB (Release 2008a, Mathworks, Inc., Natick, MA, USA), except for wavelet-edge enhancement which was implemented in LastWave [35]. All the necessary code is available online (<http://ncifrederick.cancer.gov/atp/omal/flo/Ann.aspx>) through a license agreement with National Cancer Institute. Raw datasets (2-D MIP, R-G-B images only) and the output from the proposed workflow are also available through a material transfer agreement with National Cancer Institute.

2.5.5.1 ANN Classification Performance

We assessed the accuracy of identifying well segmented nuclei from precision recall plots (Fig. 2.12) for 1620 different configurations of the PRE using a manually annotated verification set of 133 images which were acquired from the same patient samples as the training set. The three ANN training algorithms mentioned earlier provided similar results in terms of performance and training time. The back-propagation algorithm was used for training the neural network. Prior to selection by the PRE, segmentation output had a precision of only 17% of segmented objects accurately representing individual nuclei based on visual inspection. The best PRE configuration used the 15 top features from dependency ranking, all 15 dimensions from PCA, rank normalization and 15 neurons in the hidden layer resulting in a precision = 71.5% and recall = 73.6%. Some false positive errors (debris, nuclear clusters etc.) closely resembled well segmented nuclei in shape, size and

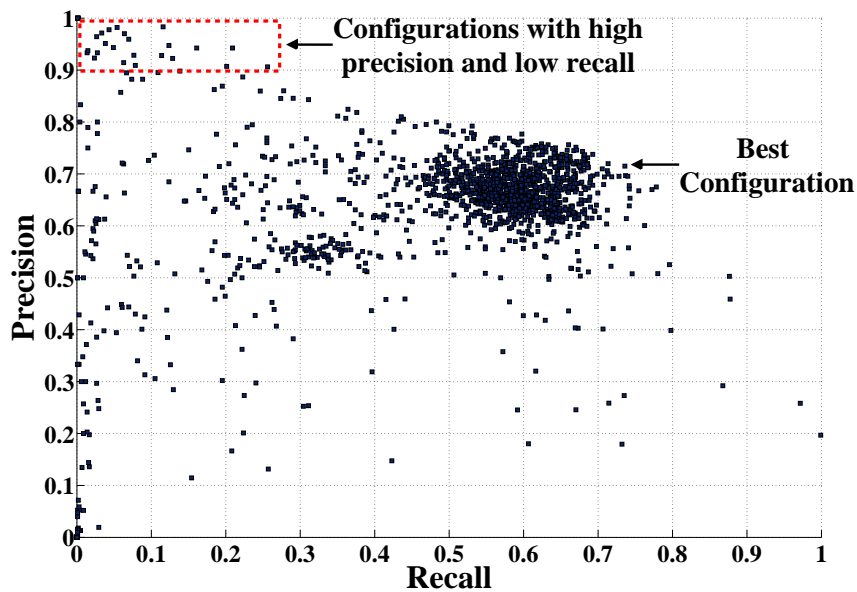


Figure 2.12: PRE precision-recall plot for 1620 configurations and the best configuration (closest to (1,1)). Configurations with high precision and low recall are shown by the red box.

other morphological features.

The classification results from the stacked classifier were also compared to those obtained from the ANN based PRE and they clearly displayed the superior performance of the latter in identifying a subset of 'well-segmented' nuclei in terms of the precision and recall. Table 2.1 shows the comparison between the stacked classifier and various configurations of the ANN classifier.

2.5.5.2 Segmentation Accuracy of Selected Nuclei using ANN

The first step in assessing segmentation accuracy of automatically selected nuclei was to ensure that the SAA was at least as accurate as hand delineation of

Table 2.1: Performance comparison of stacked classifier and ANN classifier systems

Performance	Stacked Classifier	ANN Classifier (6 Configurations)					
Recall	63.89%	5.4%	20.9%	24.63%	53.09%	59.76%	73.6%
Precision	67.11%	98.2%	94.3%	85.96%	79.21%	77.68%	71.5%

Table 2.2: Mean and standard deviation of accuracy parameters

Mean / Standard Deviation	Area Similarity per nuclei	Mean EDT based Boundary Error (in pixels)	EDT based Relative Distance Error per pixel
(I) By Hand Vs Control Boundary	0.9804 / 0.0036	1.2050 / 0.2407	0.0171 / 0.0031
(II) SAA Vs Control Boundary	0.9843 / 0.0034	1.042 / 0.0580	0.0137 / 0.0033
(III) Automatic Vs SAA	0.913 / 0.12	3.639 / 2.084	0.073 / 0.081

nuclei using synthetic control images. Measurements over 20 nuclei showed this was the case for all three segmentation accuracy parameters (Table 2.2). Therefore the SAA was used as the control for subsequent assessments of automatic segmentation accuracy.

For assessing the segmentation accuracy of the proposed workflow, AS was measured for all automatically selected nuclei, setting the value 0 for false positive nuclei. The mean value was 76% which as expected is lower than other reported accuracies that were measured over only true positive objects. When we evaluated AS for true positive objects only, to be consistent with other reported results, we obtained 91.3% (Table 2.2), which is equivalent to the accuracy we have achieved

for nuclei in cell culture [34] and significantly improved over reported accuracy in cancer tissue of 80% [60].

We calculated the mean error at the boundary and the mean normalized internal error for only true positive objects (Table 2.2) since these metrics are indeterminate for false positive objects. Both the mean boundary and internal errors of 3.6 pixels and 7.3% are approximately equivalent to the optical resolution limit.

Comparison between the segmentation accuracy metrics showed correlations of 96% between AS and internal pixel difference, 33% between AS and boundary parameter and 33% between internal pixel difference and boundary parameter. This shows that only one of the parameters AS or internal pixel difference is needed, while the boundary parameter does provide extra information not provided by either AS or internal pixel difference. However, depending on the application either AS or internal pixel difference may be more useful than the other.

2.6 Application to Gene Localization for Breast Cancer Detection

The set of nuclei identified as well segmented by the PRE were used for discriminating normal and cancer tissue sections using spatial analysis of the gene signals. In this study, we only analyzed *HES5*, a gene in the NOTCH pathway [61], which occupies different nuclear positions in normal and cancer tissues [11].

2.6.1 Segmentation and Spatial Analysis of FISH labeled Gene signals

Fluorescence *in situ* hybridization (FISH) is a technique for labeling specific gene sequences in the chromosome with a fluorescent tag which can be subsequently

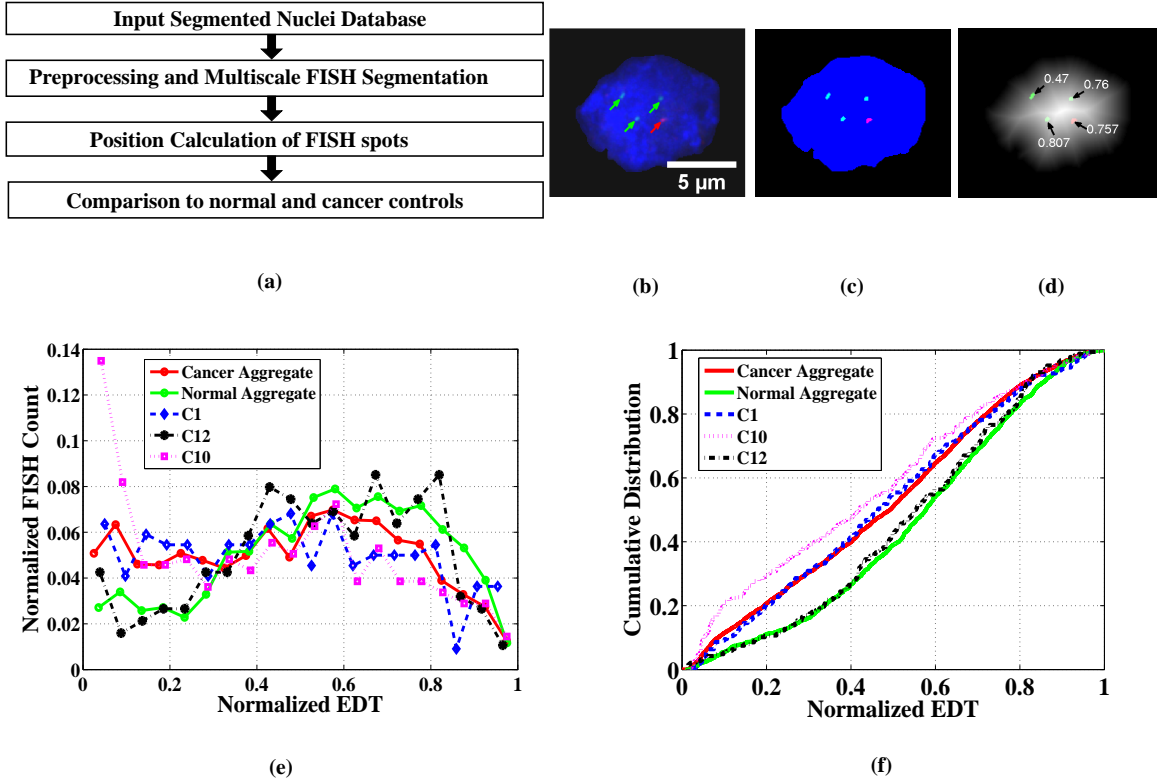


Figure 2.13: (a) Flow diagram showing steps for spatial gene localization analysis (b) Original image of segmented nucleus showing red and green FISH spots marked by arrows (c) Nucleus ROI showing segmented FISH spots (d) Euclidean distance transformed nucleus ROI showing normalized distance transform metric for each FISH spot (e) Histogram of FISH signal positions binned by normalized EDT values for aggregate cancers, aggregate normals and cancer samples C1, C10 and C12 (f) Cumulative distribution of FISH spots against normalized EDT values for aggregate cancers, aggregate normals and cancer samples C1, C10 and C12

imaged using an optical microscope to identify spatial localization of the tagged gene sequence which appear as bright spots (Fig. 2.13(b)). Fig. 2.13(a) shows the procedure for spatial analysis of FISH signals in segmented nuclei. The spot-like FISH signals (Fig. 2.13(b)) were segmented (Fig. 2.13(c)) using a derivative scale-space method [62].

First, the FISH signal channel \mathcal{F} , was convolved with Gaussian profiles of different widths to generate a Gaussian scale-space. Mathematically, this can be

described as:

$$\tilde{\mathcal{F}}_i = \mathcal{F} \otimes G(\sigma_i)_{2D} \quad \text{with } i = 0, \dots, (l - 1) \quad (2.4)$$

where l is the number of scales and \otimes is the convolution operator. $G(\cdot)_{2D}$ is the 2D Gaussian profile with width σ_i in the lateral direction. The Gaussian scale-space was generated by varying σ_i as a function of scale i using the following relation,

$$\sigma_i = \sigma_b * \sqrt{2^i} \quad (2.5)$$

where σ_b is the standard deviation of the Gaussian function at scale 0. To enhance the FISH signals, the difference between consecutive scale-spaces were multiplied as:

$$\tilde{\mathcal{F}}_{product} = (\mathcal{F} - \tilde{\mathcal{F}}_0)(\tilde{\mathcal{F}}_0 - \tilde{\mathcal{F}}_1) \dots (\tilde{\mathcal{F}}_{l-2} - \tilde{\mathcal{F}}_{l-1}) \quad (2.6)$$

To suppress noisy background from being detected as a FISH signals, $\tilde{\mathcal{F}}_{product}$ was convolved with a small Gaussian profile to create $\hat{\mathcal{F}}_{product}$ which was subsequently thresholded to detect the FISH signals.

The position of the spots with respect to the nuclear center and periphery were calculated using a shape independent Euclidean distance transform (EDT) based metric [63] and were normalized relative to the highest EDT value in the nucleus (Fig. 2.13(d)). The Kolmogorov-Smirnov (K-S) test was used to compare the distributions of gene positions in the nuclei of normal versus cancer specimens. Two samples were considered significantly different if the probability of them being from the same distribution obtained via the K-S test was less than or equal to 1%.

Table 2.3: Probability that the spatial FISH signals for NMFM and NAFA are similar using 1-D K-S test

Dataset	D1	D2	D3	D4	D5	D6	D7	D8	D9
G-Edt	0.43	0.56	0.81	0.87	0.88	0.16	0.98	0.05	0.67
G-EdtP	0.42	0.65	0.79	0.70	0.79	0.42	0.31	0.06	0.61
R-Edt	0.94	0.48	0.33	0.17	0.57	0.29	0.96	0.18	0.96
R-EdtP	0.98	0.66	0.29	0.16	0.47	0.11	0.76	0.80	0.74

2.6.2 Manual Analysis

For manual analysis, individual cell nuclei were manually delineated using the lasso tool in Photoshop 7.0 (Adobe Systems Incorporated, San Jose, CA, USA) and saved as separate image files. The red and green channels (FISH channels) of each nucleus were adjusted to reduce the background. After 130 nuclei were segmented no further tissue images were processed for a dataset. Spatial FISH analysis was performed using the same procedure as for the automated analysis. Data from the manual analysis of these tissues has previously been reported [11].

2.6.3 Results for Gene Localization using Hybrid Segmentation and Stacked Classifier

Experiments for evaluating the performance of the hybrid segmentation method were done in three stages resulting in 3 sets of outputs which differ in their degree of automation. The sets were the following : NMFM (Nuclei selection *manual*, FISH screening *manual*), NMFA (Nuclei selection *manual*, FISH screening *automatic*) and NAFA (Nuclei selection *automatic*, FISH screening *automatic*), where nuclei selection was done on the output of the automatic nuclei segmentation module and FISH screening was done from the automatic FISH segmentation module output.

Table 2.4: Probability that the spatial FISH signals for NMFm and NMFA are similar using 1-D K-S test

Dataset	D1	D2	D3	D4	D5	D6	D7	D8	D9
G-Edt	0.79	0.99	1.00	1.00	1.00	0.98	0.72	0.98	0.58
G-EdtP	0.64	0.72	0.99	0.99	0.88	0.83	0.20	0.80	0.63
R-Edt	0.51	0.43	0.27	1.00	1.00	1.00	0.99	0.82	1.00
R-EdtP	0.50	0.41	0.21	1.00	0.98	0.99	0.99	0.65	1.00

The degree of spatial similarity of the spots among the 3 output sets was used as the metric to evaluate the efficacy of the automation process. Table 2.3 shows the probability that the FISH distribution between NMFm and NAFA are similar. In the majority of cases the probability that the two methods calculated gave similar results was more than 50% and in only one instance was there a significant difference of 5% (D8 for green) level. This justifies the use of the hybrid segmentation and nuclei selection procedure for high throughput tissue screening.

As an intermediate step the automation efficacy of the FISH segmentation procedure was tested. Table 2.4 shows the probability of similarity of the FISH distribution between NMFm and NMFA. Most of the signal distributions were statistically very similar enabling us to use the existing automatic FISH segmentation procedure.

2.6.4 Results for Gene Localization using Multistage Watershed Segmentation and ANN

23 tissues were analyzed consisting of 4 normal (N1-N4), 5 non-cancerous breast disease (NCBD) (B1-B5) and 14 cancer samples (C1-C14). The set of nuclei selected by human experts and the PRE did not have a 100% correspondence.

Table 2.5: Manual and well segmented automatic nuclei count

Dataset	Number of Images	Manual Nuclei Count	Automatic Nuclei Count
N1-N4	114	536	676
C1-C14	257	1965	2588
B1-B5	129	699	736
Total	500	3200	4000

Table 2.5 shows the number of well-segmented nuclei selected both manually and automatically.

As expected, the copy number distribution of detected FISH signal per nucleus showed that the cancer samples had significantly more than 2 copies per nucleus. However, since it was rare for nuclei to have more than 10 FISH spots, such nuclei were rejected as potentially having spuriously detected spots.

Performance of the proposed processing pipeline was assessed by its ability to discriminate between normal, NCB and cancer, and its agreement to manual analysis. Fig. 2.13(e) shows the automatically-calculated distribution of gene positions aggregated for all cancers, aggregated for all normal and for 3 individual cancers (C1, C10 and C12) and Fig. 2.13(f) shows the equivalent cumulative distributions. From Fig. 2.13(e) we observe that cancer samples have significantly more nuclei where the *HES5* gene is closer to the periphery (normalized EDT ~ 0.2) than in normal samples. This is consistent with findings of the manual analysis of these tissues [11].

Comparison of normals with each other and with NCB showed no significant differences when analyzed automatically. Manual analysis reported similar results except one pair of normal samples (N2 Vs N3) were significantly different and one

Table 2.6: Probability of similarity of FISH signal distribution between normal, cancer and non-cancerous breast disease tissue sections using manual and automatic processing

	N1	N2	N3	N4	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	B1	B2	B3	B4	B5
MANUAL ANALYSIS																							
N1	1	0.46	0.1	0.66	0	0	0.7	0	0	0	0	0	0	0.01	0	0.01	0	0.4	0.29	0.2	0.05	0.44	0.13
N2	*	1	0	0.51	0	0	0.12	0	0	0	0	0	0	0	0	0	0	0.13	0.4	0.51	0.01	0.87	0.56
N3	*	*	1	0.18	0	0	0.47	0	0	0	0	0	0	0.15	0	0.39	0.05	0.15	0	0.02	0.78	0.01	0
N4	*	*	*	1	0	0	0.83	0	0	0	0	0	0	0	0	0.01	0	0.5	0.17	0.24	0.06	0.26	0.18
AUTOMATIC ANALYSIS																							
N1	1	0.41	0.67	0.12	0.00	0.00	0.81	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.96	0.00	0.02	0.17	0.43	0.64	0.49	0.30
N2	*	1	0.12	0.20	0.00	0.01	0.41	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.40	0.00	0.03	0.25	0.80	0.12	0.55	0.96
N3	*	*	1	0.06	0.00	0.00	0.92	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.78	0.00	0.00	0.02	0.13	0.22	0.12	0.11
N4	*	*	*	1	0.04	0.10	0.29	0.10	0.00	0.00	0.00	0.01	0.00	0.00	0.08	0.34	0.01	0.00	0.01	0.21	0.19	0.04	0.16

^a Two samples were considered significantly different if the probability from the K-S test was less than or equal to 1%.

^b Green cells denote cases where normal or NCBd samples were not similar to normal samples. ^c Red cells denote cases where cancer samples were similar to normal samples.

normal sample (N3) was significantly different from two (B1 and B5) out of five NCBDs (*green* cells in Table 2.6). The *red* cells in Table 2.6 denote the cases where cancer samples were not significantly different from the normal samples. Among 56 cancer versus normal comparisons, the results of the manual and the proposed method are in accord for more than 80% of cases. When the majority vote of normal specimens versus a cancer specimen was used, manual and automatic analysis concur for 11 out of 14 cancers. In the remaining 3 cases (C2, C10 and C14), either manual or automatic analysis gave an uncertain result and there were no outright contradictions.

2.7 Discussion and Conclusion

We have demonstrated integrated workflows featuring automatic nuclei segmentation methods and supervised pattern recognition engines for nuclei screening which, along with a statistical analysis of spatial localization of *HES5* gene in cell nuclei, has the potential to detect breast cancer from tissue sections. Since manual analysis of tissue sections is subjective and time consuming, our supervised method is essential and opens up the possibility of a future procedure for diagnosis and/or prognosis of breast cancer through reliable and robust automation.

Aspects of the analysis strategy adopted in this study warrant further discussion. Although, the use of 2D MIPs of the original 3D DAPI channel showed that the segmentation algorithm could be potentially used for segmenting non-confocal 2D images acquired on conventional fluorescence microscopes, a future, separate study will answer which is the best acquisition mode. Confocal, non-confocal 3D

followed by deconvolution (done in this case) and non-confocal 2D are all technically feasible. We would want to first determine which is the most accurate and then determine how much performance degrades by using conventional fluorescence microscopy. Given the fact that the method works successfully in a 2D setting, as shown by the reported experiments, a full 3D analysis, which is more accurate, will be considered next.

We compared methods to assess the boundary accuracy of nuclei screened by the PRE in terms of the unique requirements for gene localization analysis. The assessment was aided by a dynamic programming-based semi-automatic segmentation to rapidly and accurately generate validation data. Several methods have been devised previously to assess boundary accuracy of segmented objects, of which manual identification of over, under and correctly segmented nuclei [14, 15] is the most common. However, as reported earlier, utilization of simulated objects [14] enables quantitative identification of the performance limits of a segmentation algorithm. For our work, we used three accuracy parameters: area intersection between control and test segmentation, mean EDT-based boundary deviation of the test segmentation from the control segmentation [34] and a novel EDT-based relative distance error per pixel to assess the impact of boundary inaccuracies on FISH localization measurements. As is evident from Table 2.2, the average error of 0.073 per pixel corresponds to area similarity of 91.3% and mean boundary error of 3.639 pixels. A close examination of Fig. 2.13(e) reveals that cancer detection results are not adversely affected by an error of 0.073 per pixel. This is further justified by the fact that a left shifted normal cumulative plot or a right shifted cancer cumulative plot

(Fig. 2.13(f)), by normalized EDT value of 0.073, does not affect the statistically significant difference between the normals and the cancers obtained by K-S test.

Sensitivity analysis was qualitative at this stage. Many iterations of the nuclear segmentation algorithm were tested during development and in particular the multistage watershed method described herein out performed the hybrid levelset-watershed based algorithm [64]. In general, it was observed that the multistage watershed segmentation algorithm satisfactorily handled significant intensity variations in the DAPI channel which included cases of inter-image intensity variation across datasets and also inter-nucleus intensity variation within a single image. Although we considered that all our samples were of the same quality because they had been labeled and imaged under the same protocol, the quality of segmentation did depend on certain sample characteristics. As an example, the segmentation results for normal datasets were not as good as cancer datasets, because normals showed significantly more nuclear clustering compared to cancer samples. Although, the robustness of the multistage watershed algorithm to morphological, textural and size variations of the nuclei was further verified by the consistent results in a much larger tissue micro-array dataset containing approximately 1,700 images (results not shown), we envision additional opportunities for improvements by merging results from multiple segmentation algorithms. Often different algorithms successfully segment different subsets of nuclei, thus merging the output from different algorithms should increase yield of well segmented nuclei. As mentioned above, the ease of segmentation did correlate to some extent to the disease progression. However, we have no evidence that the automated analysis is biasing the data by using only easily

segmentable nuclei. Future studies with the system will address such issues further.

In the case of the PRE the future improvement would involve the use of online learning systems [65] to enable the system to learn from its mistakes. Similarly, visual inspection of automatically selected nuclei in order to admit only true positives would further improve performance. A small subset of new data can be provided to a user and the manual decisions made on that subset can be used to further improve the discriminating power of the PRE. Although the supervised PRE was trained on breast cancer tissue images that had a range of nuclear morphologies, we expect, it would successfully screen other tissue images as long as the dominant features identifying well-segmented nuclei remain the same. On the other hand, cancers from different organ sites likely have a different set of distinctive features and therefore the PRE would need to be retrained. In this study, an ANN was used to identify the well segmented set of nuclei. However, other pattern classification methods (e.g. support vector machines, random forests) could provide improved precision-recall performance. Furthermore, to enhance the accuracy and effectiveness of this cancer diagnostic system, spatial analysis of multiple genes could be combined [11].

One of the strengths of the workflow is that it is modular. That is, each step (e.g., nuclear segmentation, pattern analysis) can be substituted by an improved algorithm providing scope for continuous improvement of the processing pipeline. The same workflow can be applied to a broader spectrum of cell biology applications (e.g., cancer malignancy classification and grading from histopathology sections [66], DNA ploidy analysis [67]) where the quality of nuclei segmentation is paramount and where a larger pool of nuclei are acquired than are required for drawing a statistically

significant conclusion. In this context, we point out that the segmentation algorithm required no manual intervention for segmenting nuclei in the reported tissue section datasets. The robustness of the algorithm was achieved by the use of multiple stages of watershed-based segmentation and pattern classification which can be viewed as the merger of a bottom-up (intensity watershed followed by GDT based and tree based merging) and a top-down (cluster identification and further nuclear segmentation within the clusters) method. For users having minimal experience in image analysis and pattern recognition, a user friendly software tool is being developed which can be used to perform all the analysis steps illustrated in the framework with minimal manual intervention. We have made the software for the reported work available at: (<http://ncifrederick.cancer.gov/atp/omal/flo/Ann.aspx>), which will enable image analysis experts to further advance the method. However, the value of the work does not lie solely in the utility of the software. A key value is the discovery that automatically selecting cell nuclei is not only feasible, but can lead to valuable biomedical results. Also, another goal of this study was to show that a successful manual method can be automated. Further characterization such as quality of nuclei selected/rejected, quality of FISH signal selected/rejected, sensitivity analysis is for a future study.

The results from our workflow, both in terms of cancer detection and nuclei screening, are very encouraging. Comparison of spatial distribution of FISH spots for the *HES5* gene obtained by manual and automatic procedure show very good correspondence justifying the effectiveness of the automation. We have also shown that the system selects almost 70% of the well segmented nuclei and the screened

nuclei have high boundary accuracy when compared to a validated semi-automatic nuclear segmentation. In summary, these very promising results show for the first time the potential of a supervised learning based high-throughput and objective test for breast cancer using localization statistics of certain genes within the cell nucleus. With further validation across large numbers of samples, it could have a subsidiary role in diagnosis, prognosis or further understanding of the mechanisms of cancer development.

Chapter 3: Probabilistic Edge Detection in 2D and 3D Optical Microscopy Images

3.1 Introduction

This chapter discusses the design of an automatic probabilistic edge detection method trained by a supervised learning framework for both 2D and 3D optical microscopy images of tissue samples. Experiments involving 2D tissue sample images presented in the previous chapter incorporated a wavelet based boundary enhancement step which inadvertently enhanced the internal structures of the cell nuclei causing degradation of the final segmentation output. Hence we developed the probabilistic edge detection method using supervised training framework to target application specific edges of interest and avoid enhancing structures irrelevant to the application. The impact on the segmentation results for using such an edge detector will be illustrated in chapter 4. The method is inspired by one of the most accurate and robust 2D edge detection methods developed for images of natural scenes reported in [68]. To the best of our knowledge, the method proposed in this chapter is the first such probabilistic edge detector developed for 3D optical microscope images of tissue samples. Although in this work we specifically target the computation of reliable edge maps for volume labeled cell nuclei in tissue samples, with appropriate training the method can be applied for detecting edges of a wider variety of target

objects such as cells and sub-cellular or sub-nuclear organelles.

Edges are known to be some of the most significant information encoding elements in images and decades of multi-disciplinary research activities have been focused towards the design of robust and accurate edge detectors. Object boundaries or contours, which unlike edges convey pixel ownership information, can be defined in terms of the detected edges and the success of several computer vision procedures such as object detection, segmentation, recognition and tracking depend on the performance of edge and contour detection procedures. There exists a large body of work targeted towards edge and contour detection in natural images of which early methods mostly concentrated on the use of local derivative filters in order to detect edges. However, in recent times, several supervised learning-based edge detection methods have been proposed which can be trained to handle edge and contour detection requirements for images which are not from natural scenes. The ideas proposed in these recent methods are particularly attractive for detecting edges in optical microscopy images.

Analysis of optical microscopy images, which exhibit some unique characteristics in brightness, color and texture channels and are distinct from natural images, is a particularly challenging area of research where accurate edge detection plays a vital role. Most optical microscopy image analysis-based biological applications involve detection and/or segmentation of cellular or sub-cellular bodies as preliminary steps and their success depends substantially on accurate identification of edges and contours of objects of interest. On the other hand, some applications such as drug screening require accurate quantification of protein localization on the cell or

nuclear membrane (boundary or edge) for which a probabilistic measure of edge strength is particularly attractive. Accurate and reliable edge detection is especially important for 3D tissue samples for which object identification and segmentation is significantly more difficult compared to its 2D counterpart of cell culture samples, due to tight packing of cells in the tissue environment, poor signal to noise ratio and poor resolution in the depth direction. Also a supervised framework is particularly attractive for edge detection in microscopic images due to wide variations in brightness, color and textural properties of objects of interest for which several different staining and imaging protocols produce considerable variations in the images acquired using optical microscopes.

The chapter is organized as follows. Section 3.2 provides a literature survey of related works. Subsequent sections outline the details of the developed methods followed by a description of the experimental protocols and datasets used for quantitative evaluation of the accuracy and robustness of the proposed method. In the results section, we demonstrate the superior quality of the edge maps obtained by the proposed method compared to those obtained from other conventional and advanced edge detectors.

3.2 Related Work

Early classical edge detection techniques such as Sobel [44], Prewitt [69], Roberts [70] and Frei-Chen [71] operators use derivative filters to identify local grayscale changes to identify step edges. However, due to the lack of any smoothing step they are sensitive to noise. Marr and Hildreth [72] identify edges by detect-

ing zero crossings of the Laplacian of Gaussian function. The Canny edge detector [73] which also uses a gradient operator along with non-maximal suppression and hysteresis thresholding to identify edges is one of the most popular edge detection techniques. Several methods based on statistical approaches have also been developed for edge detection [74] [75] [76]. Konishi et al [76] pose the edge detection problem as a statistical inference problem and design a data driven method. They learn probability distributions on filter responses from a set of pre-segmented training images and detect edges as a discrimination task using the likelihood ratio test. Santis and Sinisgalli [75] use a linear stochastic model for edge detection for which Bayesian methods are used for parameter estimation. The edge detection task was framed as a hypothesis test based on the likelihood ratio. Several machine learning-based techniques [77] [78] [79] [80] use tools like fuzzy logic, neural networks, support vector machines and genetic algorithms for performing edge detection. Several multi-orientation, multi-resolution methods [81] [82] [83] [84] [85] [86] [87] [88] have also been developed for performing robust edge detection by combining edge detection results across an appropriate range of scales. Lindeberg [89] proposed a filter-based method with an automatic scale selection step.

Among several other advanced techniques currently available, those reported in [90], [91] and [68] for natural images are most relevant to the method presented here. Dollar *et al.* [90] attempt to learn an edge classifier in the form of a probabilistic boosting tree from thousands of wide aperture multiscale features computed on image patches and term the method as Boosted Edge Learning (BEL). On the other hand, Martin *et al.* [91] combine carefully designed multi-orientation edge measure-

ments from brightness, color and texture channels using a logistic regression in a supervised learning framework. [68] extends the edge detection method presented by Martin *et al.* by using multiscale versions of the multi-channel edge features and introducing a spectral decomposition-based edge globalization technique.

In the context of optical microscopy, Geback and Koumoutsakos [92] report a discrete curvelet transform-based method for edge detection. They extract a directional field for edges which is subsequently processed by non-maximal suppression and thresholding. [93] reports an algorithm fusion method using multiscale Gabor wavelet filters to locate edges of various sizes and under varying noise and contrast conditions.

3.3 Probabilistic Edge Detection for Microscopy Images

The probabilistic edge detection algorithm for optical microscopy images is inspired by previous works of Martin *et al.* [91] and Arbelaez *et al.* [68] in 2D natural images, who compute a posterior probability of edge strength at each image location. In our work we define a similar oriented edge strength probability function $MiPb(\mathcal{S}, \vartheta)$, specifically designed for 2D and 3D optical microscopy images. \mathcal{S} denotes a spatial location in terms of a pixel (xy location) or a voxel (xyz location) in 2D or 3D space respectively and ϑ denotes a 2D orientation in terms of a rotation θ with respect to the x-axis (in our case, $\theta \in \{0, \pi/4, \pi/2, 3\pi/4, \pi, 5\pi/4, 3\pi/2, 7\pi/4\}$ resulting in 8 2D orientations) or a particular 3D orientation in terms of a combination of Euler angles [94] α , β and γ (in our case, $\alpha, \beta, \gamma \in \{0, \pi/4, \pi/2, 3\pi/4\}$ resulting in 64 3D orientations). In short, $MiPb(\mathcal{S}, \vartheta)$ was calculated by measuring

local image brightness, color (optional since optical microscopy images often require segmentation of target objects in a single color channel) and texture in multiple scales followed by a global edge measurement using the spectral decomposition of the data. It also incorporates edge features that intentionally target certain edge properties that are unique for microscopy images and hence improve edge detection performance in such images compared to the previously reported probabilistic edge measures developed for natural images. The following subsections present the $MiPb(\mathcal{S}, \vartheta)$ measurement framework for 3D volumes. The 2D counterpart is a simplification of the 3D case and although it closely resembles the work reported in [68], the uniqueness in adapting the method for microscopy images will be highlighted as appropriate.

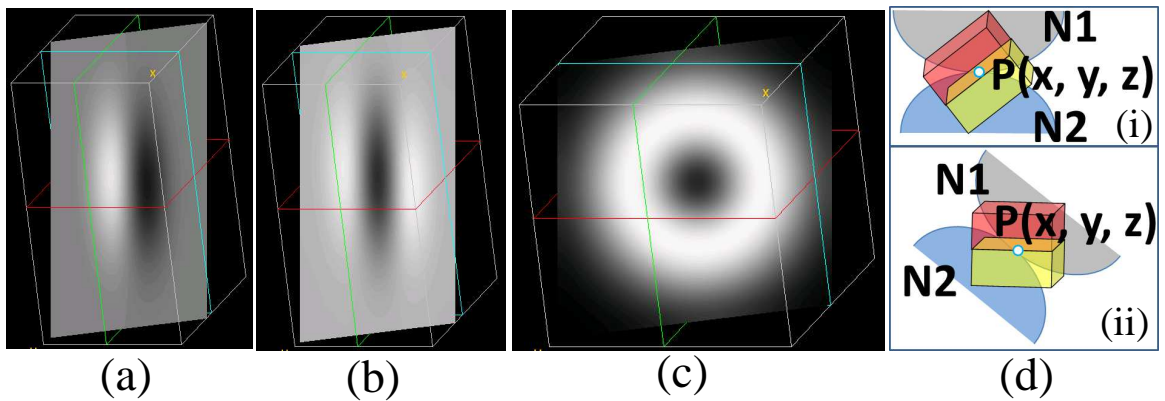


Figure 3.1: (a) Texture filter 1 (b) Texture filter 2 (c) Texture filter 3 (d(i)) Schematic of portions of nuclei N1 and N2 with oriented cuboidal regions for cumulative histogram computation of B and T channels. Cuboids are exaggerated for illustrative purposes (d(ii)) Pre-rotated nuclei N1 and N2 with axis aligned cuboidal regions.

3.3.1 Multiscale Brightness, Color and Texture Based Edge Measurements

Multi-scale and multi-orientation brightness, color and texture-based edge measurements were computed from separate volumes created before the actual measurement computation. The original 3D volume was transformed to the *CIE Lab* color space, designed to approximate the human visual system, resulting in the brightness(L), color a and color b volumes which were used for edge measure computation. In optical microscopy, often individual objects of interest such as cell nuclei are imaged in one separate red, green or blue channel. Hence, for nuclear segmentation it was often sufficient to use a single channel volume which when converted to the *CIE Lab* color space resulted in the L channel only, while the a and b channels were empty. In such cases, only the brightness volume was used and the color channels were ignored for edge measurement purposes. In the following description, the color channels are ignored.

The texture channel was created by using volume specific textons [68] extracted from the L channel. Texton computation was initiated by first filtering the volume using 64 orientations of 3D filters shown in Figure 3.1. Texture Filter 1 was created by taking a first order derivative of a Gaussian (odd-symmetric) while Filter 2 was created by taking a second order derivative (even-symmetric). Filter 3 is the difference of Gaussian (DoG) filter. For each voxel $P(x, y, z)$, responses from 64 orientations of filters 1 and 2 and a single response of filter 3 created a 129-dimensional vector of responses, which were subsequently clustered using *K-means* [44] clustering with K cluster centers (a value of $K = 32$ was used in our implementation since tex-

tural properties of microscopy images can be reasonably captured using 32 cluster centers). The K cluster centers created the volume specific textons and each volume location $P(x, y, z)$ was assigned a texton id (the closest cluster center) ranging from 1 to K , thus creating the texture volume (T) used for edge measure computation.

For a particular image location, $P(x, y, z)$ in Figure 3.1(d), the edge measurements for each individual channel (L, a, b and T) were obtained by calculating the distance between histograms of voxel values within ϑ oriented cuboidal regions on either side of the location shown in colors red and yellow in Fig Figure 3.1(d(i)). Multiscale measurements were obtained by using three ($D/2, D, 2D$) sizes of the cuboidal regions used to calculate the histograms, where D was the length of a side of the cuboid. For the brightness and texture volumes the D values were 5 and 10 respectively. The distance between the two histograms was measured by the χ^2 distance which is defined as

$$\chi^2(r, y) = \frac{1}{2} \sum_{n=1}^B \frac{(r(n) - y(n))^2}{(r(n) + y(n))} \quad (3.1)$$

where B is the number of bins (8 in our case) and r and y are the histograms of the red and yellow colored regions respectively. This results in a set of edge measurement volumes $V_{s,c}(x, y, z, \vartheta)$ corresponding to scales $s \in [D/2, D, 2D]$, channels $c \in [L, T]$ and orientations ϑ of the measurement cuboid. We subsequently used the Savitzky-Golay [95] filter to enhance the local maxima of edge measurements in a direction orthogonal to ϑ .

For microscopic images, an additional channel was created in order to identify

edges between touching objects which had similar brightness and textural properties. Such touching objects are encountered quite frequently in microscopic images, especially in tissue samples where the cells are arranged in a tight packing. As can be easily seen, such edges which are very typical of microscopic images cannot be detected reliably using brightness and texture based measurements outlined above. In order to identify such edges an additional channel was created by filtering the original volume with a symmetric Gaussian double derivative filter as shown in Fig 3.1(b) (texture filter 2) in 64 ϑ orientations resulting in a set of filtered volumes $G(x, y, z, \vartheta)$. Such a filter can specifically target edges between touching objects having similar brightness and textural properties.

The multi-scale multi-orientation and multi-channel edge measurements were then combined as,

$$mMiPb(x, y, z, \vartheta) = \sum_s \sum_c \eta_{s,c} \cdot V_{s,c}(x, y, z, \vartheta) + \nu \cdot G(x, y, z, \vartheta) \quad (3.2)$$

where $mMiPb(x, y, z, \vartheta)$ denotes an initial multi-scale edge measurement and $\eta_{s,c}$ and ν are coefficients used to combine the measurements. The coefficients $\eta_{s,c}$ and ν were learned from a edge delineated training set of 3D volumes using logistic regression which was demonstrated [91] to be as effective as any other more sophisticated classifier such as support vector machines.

The final edge measurement was obtained for each volume location (x, y, z) by taking the maximum edge response from $mMiPb(x, y, z, \vartheta)$ over all orientations ϑ , as

$$mMiPb(x, y, z) = \max_j mMiPb(x, y, z, \vartheta) \quad (3.3)$$

3.3.2 Spectral Edge Measurements

Measuring spectral edges is a strategy for the globalization of edge measurements by capturing the dominant edges of global structures in a volume. For computing the eigen decomposition-based spectral edge measurements, a sparse symmetric affinity matrix W was computed first. The affinity between two locations (i and j) in the 3D volume was measured in terms of the maximum $mMiPb(x, y, z)$ value on the line segment connecting the two locations, which is termed as the *intervening contour cue*. In our case all locations within a maximum radius (\bar{r}_{max}) of 5 voxels was used to construct the affinity matrix. $W(i, j)$, the ij^{th} entry of the affinity matrix was then defined as,

$$W(i, j) = \exp \left(- \frac{\max_{\bar{r}} \{mMiPb(x, y, z)\}}{\rho} \right) \quad (3.4)$$

where $\bar{r} \leq \bar{r}_{max}$ was the line segment connecting locations i and j in the 3D volume and ρ was a constant (with value 0.1). Subsequently, the generalized eigen value decomposition was performed for the following system,

$$(D - W)\mathbf{v} = \lambda D\mathbf{v} \quad (3.5)$$

where $D(i, i) = \sum_j W(i, j)$, λ were the eigen values and \mathbf{v} were the corresponding eigen vectors (volumes). n (16 in our case) eigen vectors (volumes) \mathbf{v}_1 to \mathbf{v}_n corresponding to the smallest eigen values λ_1 to λ_n , leaving out the smallest one λ_0 (which

is 0) were used to compute the spectral edge measurements $sMiPb(x, y, z, \vartheta)$. The global edge information contained in the eigen volumes were identified by performing a Gaussian derivative (∇_{ϑ}) in 64 ϑ orientations. The derivatives were subsequently combined as

$$sMiPb(x, y, z, \vartheta) = \sum_{i=1}^n \frac{1}{\sqrt{\lambda_i}} \nabla_{\vartheta} \mathbf{v}_i(x, y, z) \quad (3.6)$$

to obtain the complete set of spectral edge measurements.

Finally, the multiscale brightness, color and texture-based measurements and the spectral measurements were combined to obtain the multi-oriented global 3D edge information :

$$MiPb(x, y, z, \vartheta) = \sum_s \sum_c \varepsilon_{s,c} \cdot V_{s,c}(x, y, z, \vartheta) + \mu \cdot G(x, y, z, \vartheta) + \kappa \cdot sMiPb(x, y, z, \vartheta) \quad (3.7)$$

where coefficients $\varepsilon_{s,c}$, μ and κ were learned from a edge delineated training set of 3D volumes using logistic regression. As in the case of multiscale brightness, color and texture-based measurements, the final edge measurement was obtained by taking the maximum over all orientations

$$MiPb(x, y, z) = \max_{\vartheta} MiPb(x, y, z, \vartheta) \quad (3.8)$$

3.3.3 Efficient Computation For 3D

In spite of the fact that conceptually and theoretically the computation of 2D and 3D edge detectors look very similar to each other, the 3rd dimension adds

significantly to the computational complexity of the edge detection method when compared to its 2D counterpart. Hence, some modifications were made to improve the computational efficiency of the 3D edge detector.

- **Cumulative Histogram Computation:** Martin *et al.* [91] in their original development of the 2D edge detector for natural images used semi-circular regions (centered at each pixel location) on either sides of a θ oriented line to calculate the cumulative histograms used for L, a, b and T based measurements. This operation is computationally expensive and takes $\mathcal{O}(Nr^2)$ time for an N pixel image and a semi-circle radius of r . However, Arbelaez *et al.* [68] developed a computationally more efficient 2D algorithm which takes $\mathcal{O}(N)$ time by pre-rotating the image and using integral images.

Along similar lines, for 3D volumes, we took advantage of volume pre-rotation and 3D integral volumes to compute the cumulative histograms efficiently. Instead of rotating the cuboids for histogram computation, the volume was first rotated to ϑ orientation (Fig. 3.1 (d(ii))). For a particular histogram bin b an indicator volume was created from the rotated volume where a voxel was set to 1 if the intensity value was within bin b and 0 otherwise. An integral volume was created from the indicator volume by summing the voxel values along the 3 dimensions. A linear combination of the voxel values at the 8 corners of the axis aligned 3D cuboid on the integral image provided the count of voxels within bin b . This computation was repeated for each of the B bins to compute the complete edge measurements. Such a computation is

very efficient and took $\mathcal{O}(N)$ time where N was the total number of voxels and was independent of the size (D) of the cuboidal region.

- **Volume Filtering:** Computations such as the texton calculation and Gaussian derivative calculation of the eigen volumes involved filtering of 3D volumes. Fast fourier transform was used to efficiently perform all such operations. Also, instead of rotating the filters in 3D, the volumes to be filtered were pre-rotated to ϑ orientation before the filtering operation, as was done for the cumulative histogram computation.
- **Spectral Decomposition:** Computation of the eigen vectors of the generalized eigen system $(D - W)\mathbf{v} = \lambda D\mathbf{v}$ becomes extremely memory and computation intensive for 3D volumes. A small $100 \times 100 \times 100$ voxel volume results in an affinity matrix W which is of size $10^6 \times 10^6$ and understandably such eigen computation soon becomes impractical for progressively larger volumes. Hence to reduce the computational load, each individual slice of the $mMiPb(x, y, z)$ volume was first segmented into 2D superpixels using a simple watershed algorithm [43]. The ij^{th} entry (corresponding to the i^{th} and j^{th} superpixel) $W(i, j)$ of the affinity matrix was then defined as in (3.4), where \bar{r} was the line segment connecting the centroids of the two superpixels. To impose the local constraint as imposed by \bar{r}_{max} in (3.4), a superpixel was morphologically dilated \bar{r}_{max} times in 3D and the affinity calculation was performed for all superpixels that overlapped atleast 1 voxel with the original dilated superpixel.

- **Multiprocessor Computation:** Multiprocessor computation was performed to further improve the efficiency of the algorithm. The computational parallelism was implemented on the 3D orientation ϑ since computations at each orientation was independent of computations for any other orientation.

3.4 Experimental Datasets and Methods

Experiments were performed to quantify the performance of edge detection algorithm. For both 2D and 3D algorithms, three kinds of datasets were used for the experiments, (i) synthetic images, (ii) simulated images created using nuclei in original microscopic datasets and (iii) original microscopic images. The first 2 datasets helped in accurate quantification of the strengths and weaknesses of the algorithms as original object boundaries used as ground truth were available for quantitative comparison. For original microscopic images, the object boundaries were subjective since they depended on the perception of the human annotator who manually or semi-automatically identified the object boundaries.

3.4.1 Experimental Datasets

3.4.1.1 Synthetic Datasets

Dataset of synthetic images were created by forming bright synthetic elliptical (in 2D) or ellipsoidal (in 3D) objects on a dark background as often observed in microscopic images of biological samples with volume stained cell nuclei. The objects were of comparable dimensions to cell nuclei in real reference sample microscopic images with a particular pixel (in 2D) or voxel (in 3D) size. The brightness of the objects and the background were also estimated from the reference images. The

objects were spatially positioned such that there was a mix of isolated objects as well as closely packed nuclear objects, simulating their packing in a real tissue sample. Subsequently, two kinds of distortions were introduced, the blurring and the noise generated by the optical imaging and image capturing system of the microscope. The blurring kernel was simulated as a 2D or 3D Gaussian function (which is a reasonable assumption) for which the width was calculated using the theoretical expressions of the point spread function (PSF) of a confocal microscope.

The x-y resolution (R_{x-y}) of a confocal microscope is often defined in terms of the *Full Width Half Maximum* (FWHM) of the central disc of the *Airy pattern* [96] created by the microscope for a point source of light and for a confocal microscope $R_{x-y} = 0.4 \lambda_{ex}/(NA)$, where λ_{ex} is the wavelength of the excitation light and NA is the *numerical aperture* of the imaging lens. To find the width of a Gaussian function of standard deviation σ_{XY} which closely simulated the 2D PSF of a confocal microscope, the FWHM of the Gaussian approximated by $2\sqrt{2 \ln 2} \sigma_{XY}$ was equated with R_{x-y} . The z-resolution of a confocal microscope is much worse compared to its x-y resolution and as a reasonable assumption, for simulating the 3D PSF, the standard deviation of the Gaussian in z-direction (σ_Z) was taken to be 3 times that of the sigma in x-y direction.

Although in reality the noise introduced by a confocal microscope is a combination of Poisson and Gaussian noise, for simulation purposes the noise was assumed to be additive Gaussian. When the *signal to noise ratio* (SNR) is defined as

$$\text{SNR} = \frac{|\mu(\mathcal{F}) - \mu(\mathcal{B})|}{\sqrt{\sigma^2(N_{\mathcal{F}}) + \sigma^2(N_{\mathcal{B}})}} \quad (3.9)$$

where \mathcal{F} is the foreground region, \mathcal{B} is the background region, $\mu(\mathcal{F})$ is the mean foreground intensity, $\mu(\mathcal{B})$ is the mean background intensity, $\sigma(N_{\mathcal{F}})$ is the foreground noise standard deviation and $\sigma(N_{\mathcal{B}})$ is the background noise standard deviation, the SNR of a standard confocal image is around 3.

For 2D images, human breast tissue sample images acquired on a Zeiss 710 confocal microscope from a tissue micro-array was used as the reference images. The pixel resolution of the reference images was $0.08\mu m$ per pixel in the x and y directions and typical nuclear diameter varied from $6\mu m$ to $14\mu m$. The mean foreground and background intensities measured were 90 and 23 respectively. Nuclear objects were simulated by ellipses with intensity 90 on a background of intensity 23. σ_{XY} of the blurring Gaussian function was calculated to be 0.7 considering the information that the wavelength of the excitation light for imaging the reference sample was $405nm$ and the NA of the imaging lens was 1.4. Subsequently Gaussian noise was added to the blurred images in order to create a final synthetic image. In order to test the robustness of the algorithms with respect to blurring and noise, σ_{XY} was varied as $\{0.4, 0.7, 1.0\}$ and the SNR was varied as $\{2, 3, 4\}$ to create the complete synthetic dataset.

For 3D synthetic images, the simulated objects were ellipsoids with dimensions comparable to that of real cell nuclei assuming voxel size of $0.08\mu m$ in the x and y directions. Due to reduced resolution in the z direction, real confocal volumes have

anisotropic voxel size with the z dimension of the voxel size being around 3 times that of the x and y dimensions. To measure the effects of voxel anisotropy, both isotropic and anisotropic volumes were created. The isotropic volumes (voxel size of $0.08\mu m$ for all 3 dimensions) initially created using ellipsoids were subsampled by a factor of 3 to create an anisotropic volume (voxel size of $0.08\mu m$ for x and y dimensions and $0.24\mu m$ for z dimension). To introduce the blurring of the imaging system, for isotropic volumes σ_Z was equal to σ_{XY} . However, for anisotropic volumes, σ_Z was 3 times σ_{XY} . Subsequently, Gaussian noise was added to both blurred isotropic and anisotropic volumes. For 3D too, σ_{XY} was varied as $\{0.4, 0.7, 1.0\}$ (σ_Z was varied accordingly as mentioned) and the SNR was varied as $\{2, 3, 4\}$ to create the two sets of synthetic datasets. Fig. 3.2(a-e) shows sample synthetic 2D and Fig. 3.3(a-e) shows single slice of sample synthetic 3D images with different SNR and blur values.

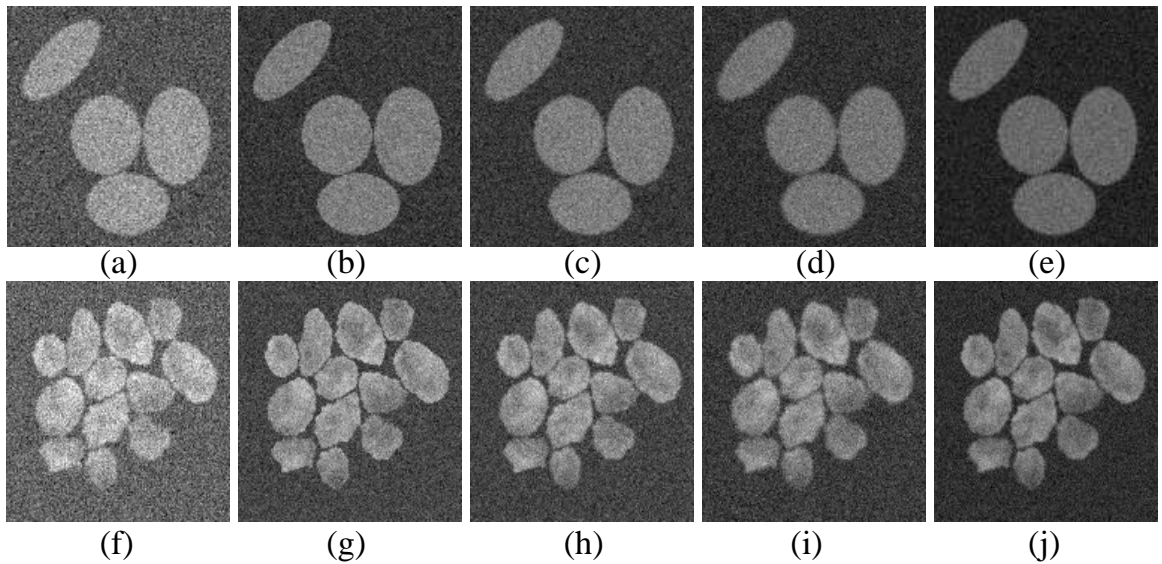


Figure 3.2: Synthetic 2D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7. Simulated 2D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7.

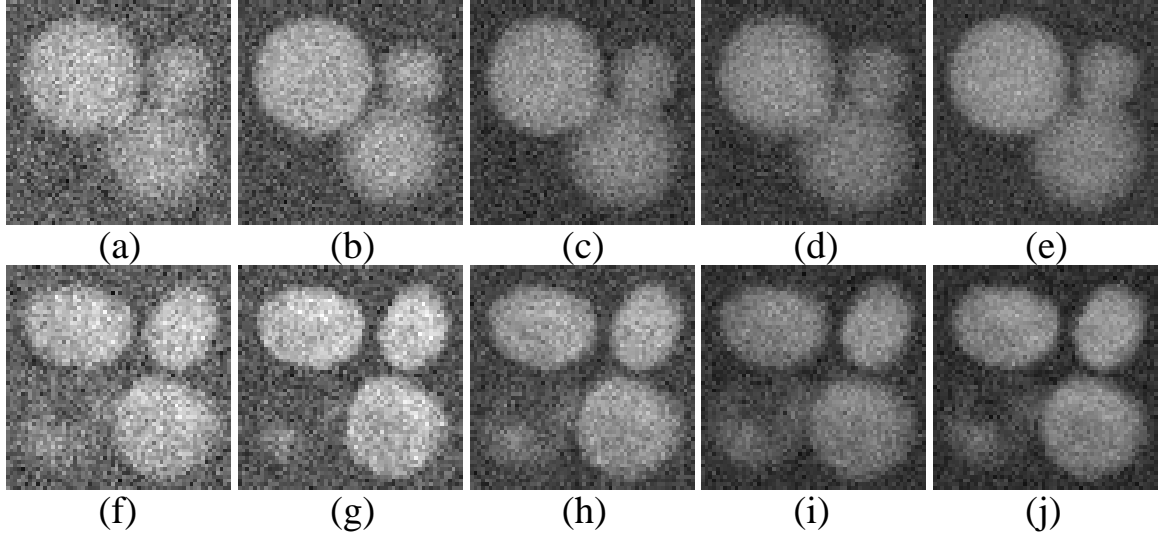


Figure 3.3: Single slice of synthetic 3D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7. Single slice of simulated 3D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7.

3.4.1.2 Simulated Datasets

Objects simulating the cell nuclei in synthetic images were of constant intensity and hence lacked the low frequency texture present within the cell nuclei in real microscopic images. Hence, simulated images were created from cell nuclei in real microscopic images to quantify the effects of the low frequency texture on the accuracy of the algorithms. 2D and 3D nuclei segmented interactively from Gaussian smoothed versions of the reference images using the semi-automatic graph cut-based segmentation interface were spatially positioned on a background of intensity 23 to simulate isolated as well as tightly packed cell nuclei. Subsequently, multiple levels of blurring ($\sigma_{XY} \in \{0.4, 0.7, 1.0\}$ and σ_Z was 3 times σ_{XY}) and noise (SNR $\in \{2, 3, 4\}$) distortions were introduced as discussed in the previous section. Fig. 3.2(f-j) shows sample synthetic 2D and Fig. 3.3(f-j) shows single slice of sample

synthetic 3D images with varied SNR and blur values.

3.4.1.3 Original Microscopic Datasets

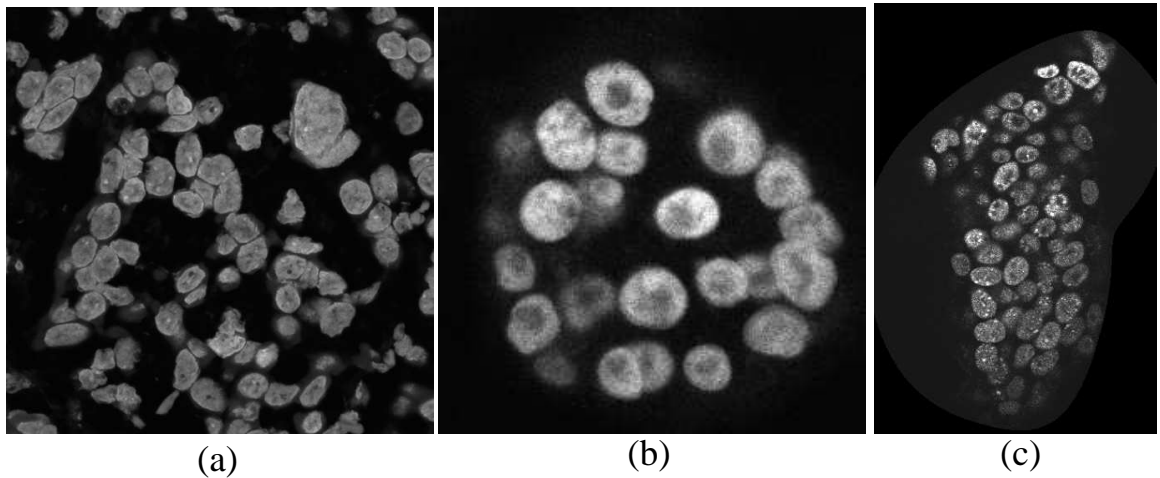


Figure 3.4: (a) Sample 2D image from human breast tissue dataset (b) Sample 2D image from MCF-10A dataset (c) Sample 2D image from mouse embryo dataset.

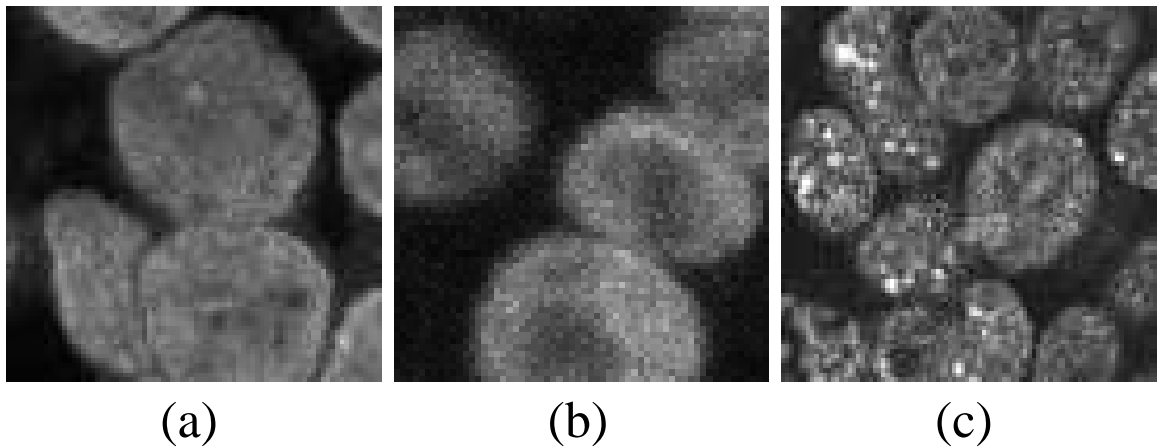


Figure 3.5: (a) Single slice of sample subvolume of 3D image from human breast tissue dataset (b) Single slice of sample subvolume of 3D image from MCF-10A dataset (c) Single slice of sample subvolume of 3D image from mouse embryo dataset.

Original microscope datasets consisted of images from three kinds of samples.

The sample information are as follows,

- **Human Breast Tissue Micro-array:** The tissue micro-array sample consisted of 150 cores including 75 cases of normal, reactive, premalignant and various grades of malignant tissues of the human breast. The cores were $4\mu m$ thick and $1.1mm$ in diameter and labeled with the DNA dye, 4-in, 6-diamidino-2-phenylindole (DAPI). A Zeiss LSM 710 (Carl Zeiss, Inc.) was used to image 10 such cores at multiple locations with a 63X, 1.4 NA oil objective lens and a pinhole of 1 airy unit. The original voxel size was $0.08\mu m$ in the x and y dimensions and $0.32\mu m$ in the z dimension. For experimentation, the original stacks were subsampled by a factor of 3.4 resulting in a voxel size of $0.27\mu m$ in x and y dimensions and $1.00\mu m$ in the z dimension. Fig. 3.4(a) shows a single slice of a human breast tissue sample which was used for 2D edge detection experiments and Fig. 3.5(a) shows a single slice of a subvolume of a human breast tissue sample which was used for 3D edge detection experiments.
- **MCF-10A Acini:** Mature acini were grown for 14 days from single MCF-10A cells on Basement Membrane Extract (Trevigen Inc., Gaithersburg, MD), fixed in 4% paraformaldehyde in PBS for 10 min at RT and labeled with the DNA dye, 4-in, 6-diamidino-2-phenylindole (DAPI). Samples were mounted in Vectashield. Images were acquired using a 63X, 1.4 NA oil objective lens, and pinhole of 1 airy unit on an LSM 510 confocal microscope (Carl Zeiss, Inc.). Excitation was with $405nm$ laser light and emitted light between 420 and $480nm$ was acquired. Voxel size was $0.14\mu m$ in the x and y dimensions and $0.50\mu m$ in the z dimension, resulting in 10-20 z slices per nucleus. For

experimentation, three such acini image stacks were used and the original stacks were subsampled by a factor of 2 resulting in a voxel size of $0.28\mu m$ in x and y dimensions and $1.00\mu m$ in the z dimension. Fig. 3.4(b) shows a single slice of a MCF-10A acinus which was used for 2D edge detection experiments and Fig. 3.5(b) shows a single slice of a subvolume of a MCF-10A acinus which was used for 3D edge detection experiments.

- **Mouse Embryo:** Mouse embryos at E5.5 were dissected in ice-cold PBS with 0.02% BSA (PBS-BSA), fixed for 20 min in 2% PFA, washed and permeabilized in 0.1% Triton, 100 mM glycine in PBS for 10 min at room temperature. Following primary antibodies were used; anti-Oct4 (C-10; SC-5279; Santa Cruz Biotechnology, CA); anti-Cdx2 (Cdx2-88; AM392; Biogenex, CA); anti-histone H3 (tri methyl K27; ab108245, Abcam, MA). Embryos were incubated with secondary antibodies conjugated with Alexa Fluor 488 and Alexa Fluor 633-conjugated Phalloidin (both 1:400 dilution; Molecular Probes). Images were acquired using a Zeiss LSM510 confocal microscope with Plan-Neofluor 40x/1.3 oil objective. Two volumes of MCF-10A and human breast tissue samples and a single volume of mouse embryo tissue were used for experiments.

3.4.2 Experimental Methods

The edge detection accuracy for both 2D and 3D algorithms was tested using the precision-recall (PR) curve framework reported in [91]. For synthetic and simulated datasets, the actual object boundaries were available and they were used as the *true* ground truth edge map of the cell nuclei. However, for original microscopic

images, a *true* ground truth is not available for testing the algorithms. Hence, for both 2D and 3D datasets semi-automatic human segmentations using the interactive graph cut-based segmentation interface (reported in section 4.3.4 of the thesis) were used as the ground truth segmentation. However, such a ground truth is not completely accurate and depends on the the human annotator’s subjective perception of an object edge.

For generating the PR curve, the output edge map was thresholded at progressively increasing values, generating a binary edge map at each individual threshold. This binary map was compared to the ground truth segmentation voxel by voxel in order to calculate the precision $P = TP/(TP+FP)$ and recall $R = TP/(TP+FN)$, where TP is the number of true positive voxels, FP is the number of false positive voxels and FN is the number of false negative voxels. As a comprehensive measure, the F-score defined as $PR/(\alpha P + (1 - \alpha)R)$ is also reported at $\alpha = 0.5$. For a particular application, α can be used as a trade-off between true signal requirement (precision) and noise tolerance (recall) and the highest F-score at that α can provide the best possible operating point on the PR curve. However, this particular framework is very stringent and heavily penalizes even minor edge localization inaccuracies from algorithms which otherwise produce reasonably accurate object edge maps. As a more practical approach for quantifying edge accuracy, we allow a 2 voxel wide region of trust on either side of the ground truth boundaries where an edge voxel identification is granted as correct. In practice, the ground truth object boundaries are dilated twice before the calculation of the PR curve. This method also attempts to mitigate accuracy measurement issues due to the ground truth

inaccuracies present in a human annotated real microscopic image dataset.

Specific experiments performed to test the algorithms were as follows,

- **Comparison with conventional edge detectors** Results from the reported 2D and 3D edge detection algorithms were compared with that of conventional edge detectors namely, Sobel [44], Prewitt [69], Canny [73] and the probabilistic edge detector (gPb) without the adaptation for microscopy images using the above mentioned PR curve framework. For Sobel and Prewitt detectors, the outputs were thresholded at progressively higher values to plot the PR curve. However, for Canny the hysteresis threshold values were progressively increased to obtain multiple levels of binary edge responses from which the PR curves were derived. For 3D real microscopic datasets, edge measurements obtained by using multi-scale multi-orientation FIJI [97] based *Trainable WEKA Segmentation*(TWS) (version 2.1.0 with *FastRandomForest* classifier having 200 trees and 2 features per node) was also used for comparison.
- **Anisotropy** The effects of anisotropic voxels was measured for 3D synthetic datasets by comparing the PR curves for corresponding isotropic and anisotropic synthetic volumes.
- **Slice-by-slice and Full 3D** The 3D edge detection method is computationally intensive and to test whether a 2D slice-by-slice edge detection method was good enough, the PR curves generated by the two methods were compared for a synthetic and a simulated data volume.
- **Varying Original Samples** To demonstrate the performance of 2D and 3D

edge detectors on real microscopy data, three kinds of biological sample images were utilized and the performance of the edge detectors were quantified against boundaries obtained by human semi-automatic segmentation results.

3.5 Experimental Results

In this section we report the performance results of our experiments with the 2D and 3D edge detection algorithms.

3.5.1 2D Edge Detection Results

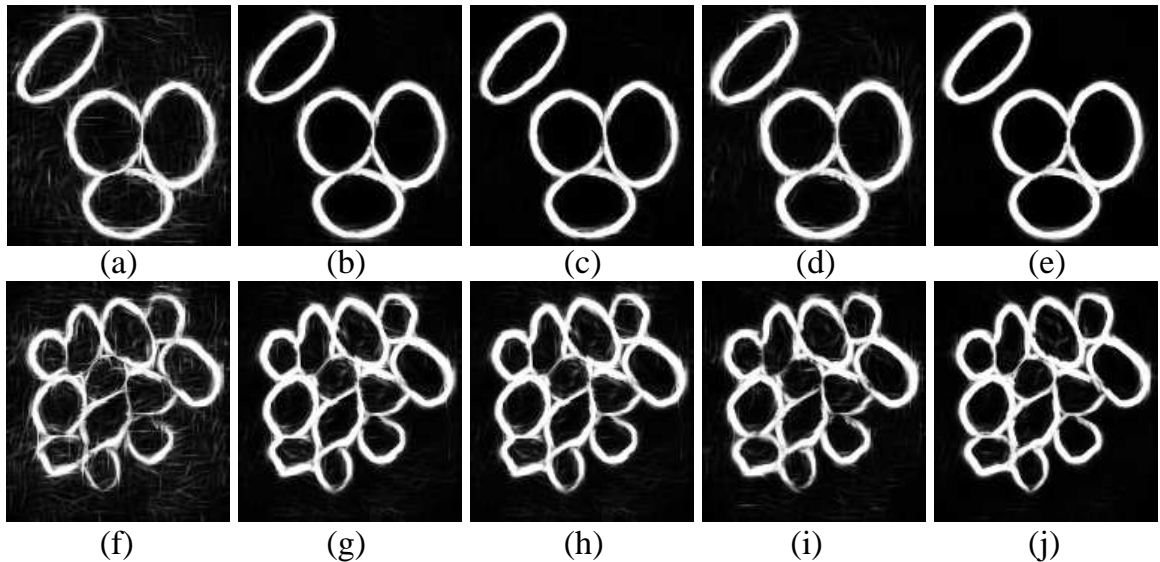


Figure 3.6: MiPb response for synthetic 2D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7. Simulated 2D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7.

The basic framework for experimentation was using the PR curves as mentioned in the previous section. Table 3.1 reports the F-score values for synthetic and simulated 2D images for varies values of SNR and blurring as obtained using Sobel(S), Prewitt(P), Canny(C), probabilistic edge detector without microscopy

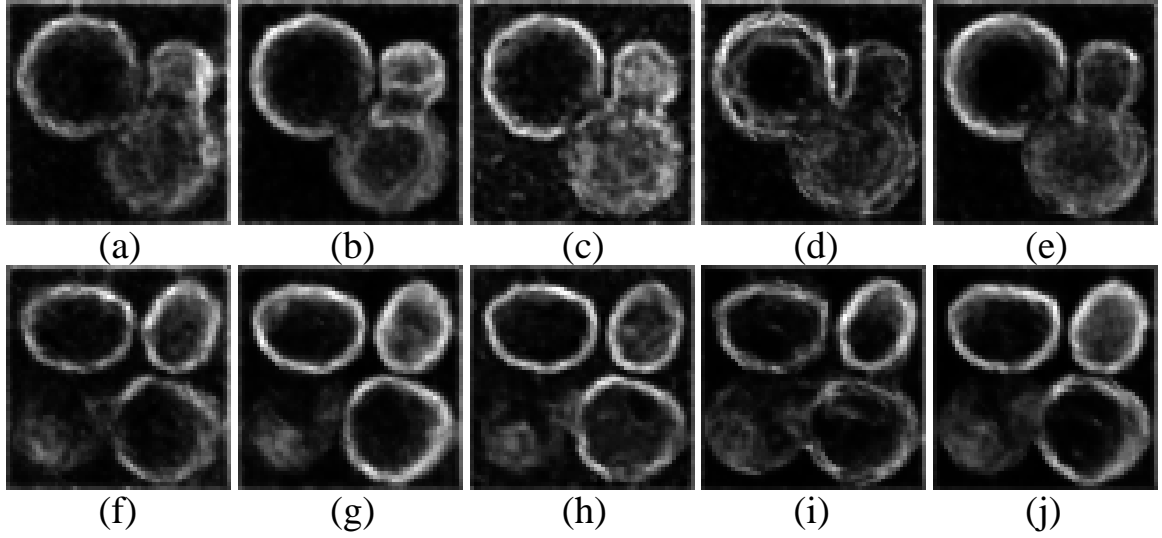


Figure 3.7: MiPb response for single slice of synthetic 3D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7. Single slice of simulated 3D image with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7.

adaptation (gPb) and the reported method(MiPb). MiPb consistently works better than the other edge detectors as reflected by the F-score values. MiPb shows reasonably robust results against increasing blur and decreasing SNR. It should also be noted that the additional channel used in MiPb to adapt gPb for microscopic images boosts the performance of the edge detector in almost all cases. Fig. 3.6 shows the contrast stretched MiPb response for 2D synthetic and simulated images corresponding to Fig. 3.2.

The maximum F-scores for the real microscopic images from MCF-10A, human breast tissue micro-array and mouse embryo sample are shown in Table 3.2. In this case too, MiPb works consistently better than the other edge detectors. Fig 3.8 shows the PR plots for 2D synthetic and simulated datasets at SNR 3 and blurring of 0.7 (settings which are closest to an average confocal microscope image) and real

2D images from MCF-10A, human breast tissue micro-array and mouse embryo samples. Fig. 3.9 shows the contrast stretched MiPb response for real microscopic 2D images corresponding to Fig. 3.4.

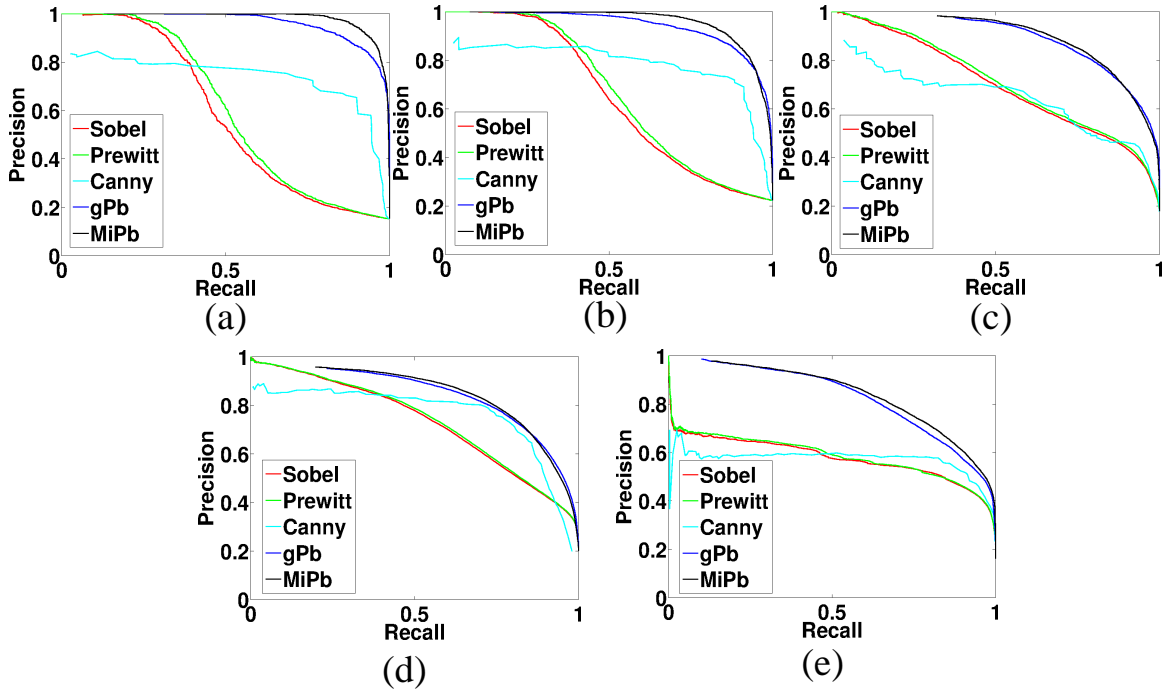


Figure 3.8: PR curves for 2D synthetic, simulated, MCF-10A Acini, Human breast tissue micro-array and Mouse embryo samples

3.5.2 3D Edge Detection Results

The edge detector performance for the 3D case was also measured using the PR framework. Table 3.3 shows maximum F-score values for Sobel(S), Prewitt(P), Canny(C), probabilistic edge detector without microscopy adaptation (gPb) and the reported 3D method(MiPb) for synthetic (having anisotropic voxel size) and simulated datasets at various levels of noise corruption and blurring. For 3D too, MiPb works consistently better than the other edge detectors in terms of F-score. It should also be noted that the F-scores for the simulated datasets are in general

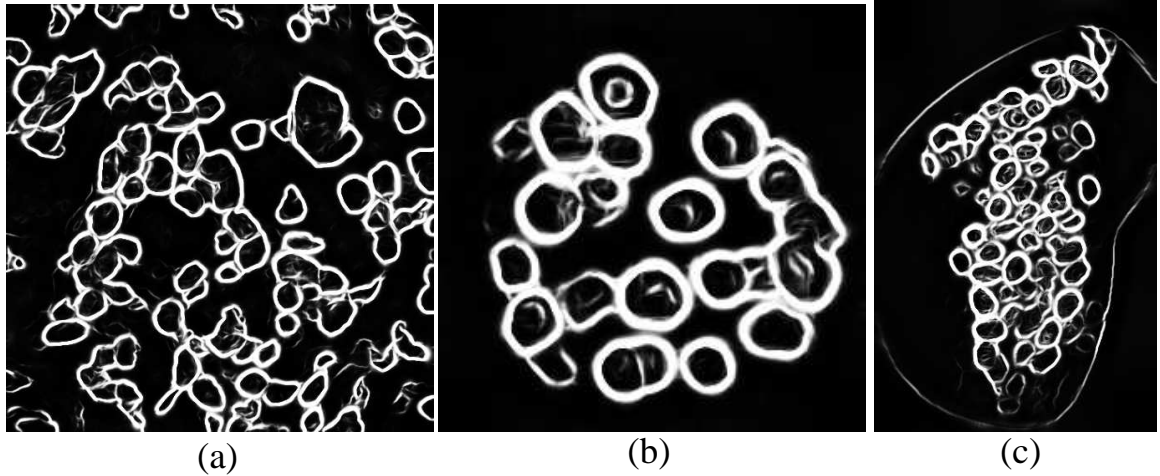


Figure 3.9: MiPb response for (a) Sample 2D image from human breast tissue dataset (b) Sample 2D image from MCF-10A dataset (c) Sample 2D image from mouse embryo dataset.

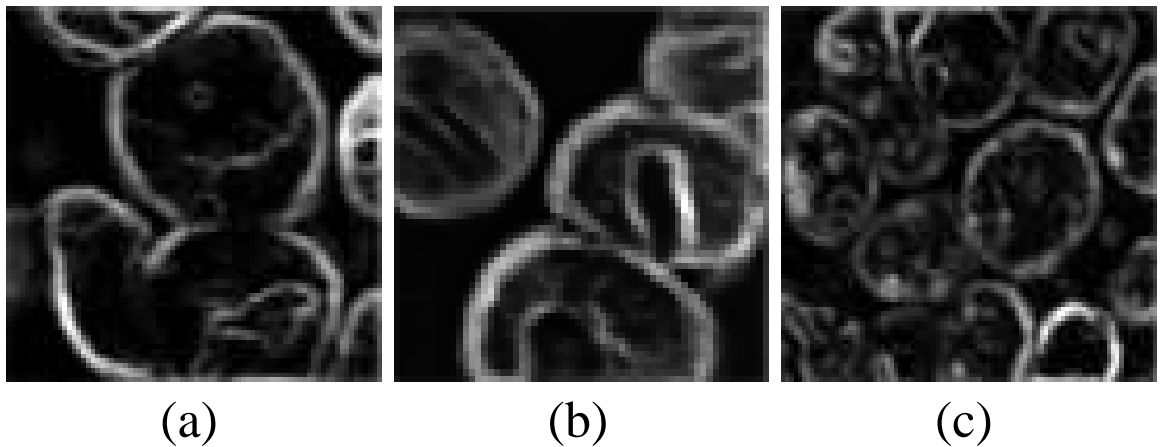


Figure 3.10: MiPb response for (a) Single slice of sample subvolume of 3D image from human breast tissue dataset (b) Single slice of sample subvolume of 3D image from MCF-10A dataset (c) Single slice of sample subvolume of 3D image from mouse embryo dataset.

higher than for the synthetic datasets, which is counter intuitive since the simulated datasets incorporate the low frequency texture components present in real nuclear volumes which should ideally hamper the performance of the edge detection algorithm. MiPb F-scores are reasonably robust across variations in SNR and blurring.

Fig. 3.7 shows the contrast stretched MiPb response for a single slice of 3D synthetic

and simulated images corresponding to Fig. 3.3. Table 3.4 shows the maximum F-score values observed for actual microscopic datasets. As previously mentioned, the edge maps against which the performance of the detector responses were measured were obtained from a semi-automatic graphcut based interactive segmentation method reported in the following chapter and depends on the visual perception of the annotator. Hence the F-score values are not as reliable as those obtained from synthetic and simulated datasets. Fig. 3.11 shows the precision recall plots for all the edge detectors for synthetic with anisotropic voxel (at SNR 3 and blur 0.7), simulated (at SNR 3 and blur 0.7) and real microscopic datasets. Fig. 3.10 shows the contrast stretched MiPb response for a single slice of real 3D microscopic images corresponding to Fig. 3.5.

Fig 3.12 shows the PR plot for isotropic and anisotropic synthetic volumes at SNR 3 and blurring 0.7. The maximum F-score value for both the cases was 0.86. It should be noted that MiPb accuracy and robustness is largely retained for even anisotropic voxel sizes (with z dimension three times the x-y dimension) implying that the method can be reliably used for most 3D optical microscopy applications without significant degradation of its performance.

Table 3.5 presents the maximum F-score values for 3D synthetic (SNR 3 and blur 0.7), simulated (SNR 3 and blur 0.7) and real images from MCF-10A, human breast tissue micro-array and mouse embryo samples for full 3D MiPb edge detection and 2D slice by slice edge detection. As expected the 3D method performs consistently and significantly better than the slice-by-slice method.

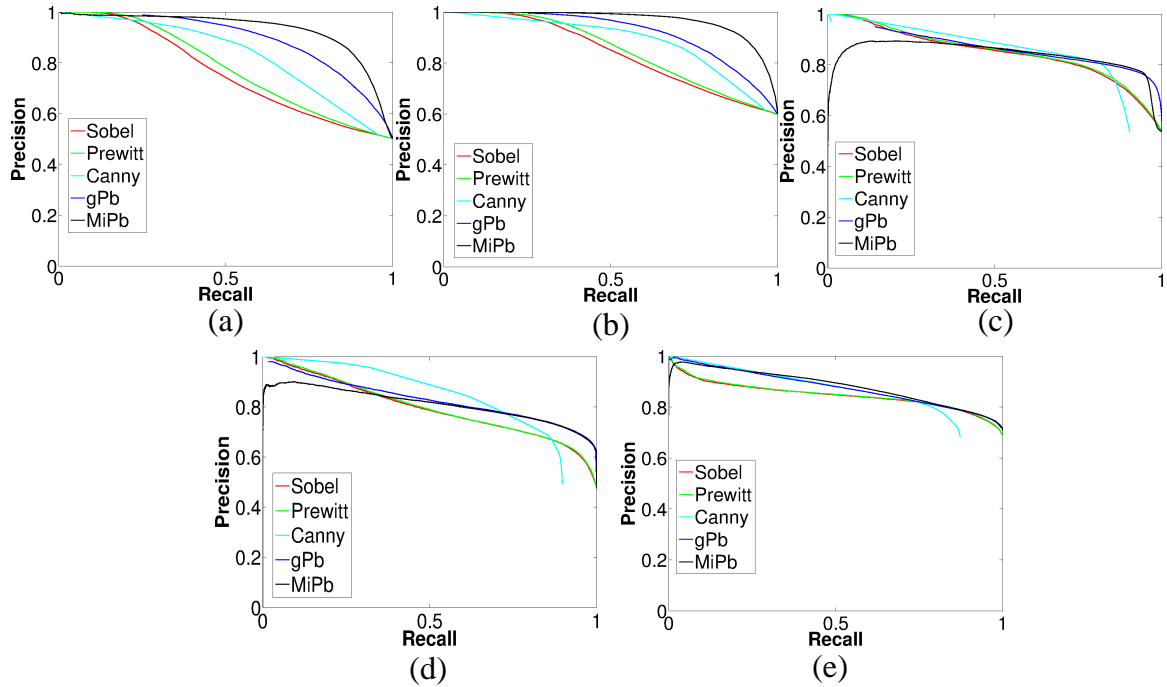


Figure 3.11: PR curves for 3D synthetic, simulated, MCF-10A Acini, Human breast tissue micro-array and Mouse embryo samples

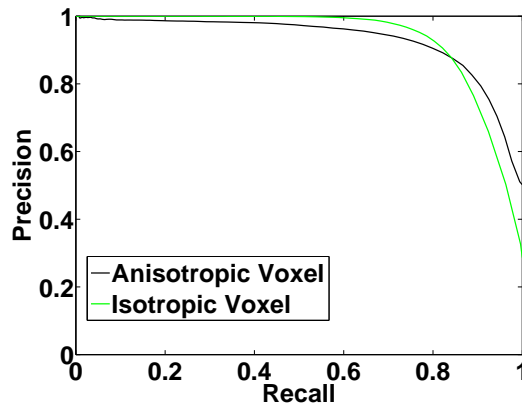


Figure 3.12: PR curves for comparing 3D isotropic and anisotropic synthetic volumes. Both have maximum F-score of 0.86

3.6 Discussion and Conclusion

The chapter presented a robust and accurate probabilistic edge detection method developed for 3D microscopic volumes. The inspiration for the work was one of the best known 2D edge detection methods developed for natural images. The

method utilizes brightness, texture and spectral decomposition based edge features to train a logistic regression based classifier in a supervised learning framework. It also incorporates modifications for adapting the original 2D method for natural images to work better in 2D microscopic images first and then the ideas were extended to 3D.

The supervised learning framework is particularly attractive for edge detection in microscopic images due to considerable variations in objects of interest and their brightness and textural properties when imaged under a confocal microscope. Although the method was used only to detect nuclear edges in the reported work, the performance of the developed edge detector is envisioned to be reliable and accurate for other objects of interest due to the supervised learning framework used to train the classifier. In the reported work too, the MiPb edge detector was trained on a single subvolume of the MCF-10A tissue sample and the resulting logistic regression coefficients were used for all the reported experiments. Excellent performance of the edge detector for all the experiments (including the synthetic and simulated datasets) emphasize the robustness and reliability of the method boosting the confidence that it will perform reasonably well for other target objects of interest.

Performance of all the edge detectors were tested using the PR framework. However, the PR framework might not be the best possible way to measure the accuracy and robustness of edge detectors for microscopic images. The edge detection step is only an intermediate step for the development of reliable segmentation methods for the target images. Hence the ultimate test for the method is to characterize the impact of the developed method in improving segmentation algorithms. Chapter

4 will illustrate one such method based on graphcuts which can take full advantage of robust and accurate edge detection and convert the pixel level information to the object level by successfully segmenting the target objects.

The computational load of the method is a limiting factor and future work will be focused towards mitigating this limitation. All the experiments were conducted on volumes which were relatively small due to the complexity constraints. However, the method can be used for computing edge maps for larger volumes by tiling several overlapping subvolumes and computing the edge maps for each subvolume separately. Although the current implementation uses a 3D orientation-based parallelism to improve the run time of the algorithm, the computational performance of the method can be further improved by using more fine grained parallelism. Also, in line with the 2D algorithm, the 3D algorithm can be ported to a GPGPU platform which should significantly improve the running time of the algorithm.

As part of future work we also want to apply the method to a wider variety of target objects and compare the results with more recent edge detection methods in order to further identify the strengths and weaknesses of the method.

Table 3.1: Maximum F-score values for 2D synthetic and simulated datasets using precision recall framework for evaluation

	SNR	2					3					4				
Blur		S	P	C	gPb	MiPb	S	P	C	gPb	MiPb	S	P	C	gPb	MiPb
0.4	Syn	0.43	0.44	0.65	0.88	0.90	0.52	0.54	0.85	0.89	0.92	0.57	0.60	0.90	0.89	0.93
	Sim	0.48	0.49	0.72	0.84	0.85	0.55	0.58	0.83	0.87	0.89	0.60	0.63	0.86	0.88	0.89
0.7	Syn	0.40	0.41	0.63	0.87	0.89	0.53	0.56	0.75	0.88	0.93	0.61	0.63	0.85	0.90	0.93
	Sim	0.45	0.47	0.67	0.82	0.82	0.56	0.58	0.79	0.86	0.89	0.62	0.64	0.82	0.87	0.90
1.0	Syn	0.37	0.38	0.46	0.86	0.88	0.55	0.57	0.67	0.87	0.91	0.66	0.68	0.80	0.89	0.93
	Sim	0.44	0.46	0.56	0.80	0.80	0.56	0.58	0.72	0.84	0.87	0.64	0.65	0.76	0.86	0.89

Table 3.2: Maximum F-score values for 2D real images from MCF-10A, human breast tissue micro-array and mouse embryo samples

Sample	S	P	C	gPb	MiPb
MCF-10A	0.62	0.63	0.65	0.78	0.80
Human Breast Tissue	0.65	0.66	0.75	0.76	0.78
Mouse Embryo	0.62	0.62	0.67	0.73	0.75

Table 3.3: Maximum F-score values for 3D anisotropic synthetic and simulated datasets using precision recall framework for evaluation

Blur	SNR	2					3					4				
		S	P	C	gPb	MiPb	S	P	C	gPb	MiPb	S	P	C	gPb	MiPb
0.4	Syn	0.66	0.67	0.69	0.82	0.84	0.66	0.67	0.75	0.84	0.85	0.67	0.69	0.77	0.90	0.90
	Sim	0.75	0.75	0.75	0.88	0.89	0.75	0.75	0.80	0.91	0.93	0.75	0.76	0.84	0.89	0.93
0.7	Syn	0.67	0.68	0.68	0.81	0.83	0.67	0.68	0.71	0.80	0.86	0.69	0.70	0.78	0.86	0.88
	Sim	0.75	0.75	0.75	0.83	0.87	0.74	0.75	0.79	0.82	0.89	0.75	0.76	0.82	0.89	0.90
1.0	Syn	0.67	0.67	0.67	0.78	0.80	0.67	0.68	0.70	0.79	0.81	0.70	0.71	0.75	0.82	0.83
	Sim	0.75	0.75	0.75	0.83	0.85	0.74	0.75	0.77	0.84	0.86	0.75	0.76	0.79	0.86	0.89

Table 3.4: Maximum F-score values for 3D real images from MCF-10A, human breast tissue micro-array and mouse embryo samples

Sample	S	P	C	TWS	gPb	MiPb
MCF-10A	0.79	0.79	0.81	0.85	0.83	0.85
Human Breast Tissue	0.75	0.75	0.76	0.78	0.80	0.81
Mouse Embryo	0.83	0.83	0.80	0.82	0.84	0.85

Table 3.5: Maximum F-score values for 3D synthetic (SNR 3 and blur 0.7), simulated (SNR 3 and blur 0.7) and real images from MCF-10A, human breast tissue micro-array and mouse embryo samples by full 3D and slice by slice MiPb calculation

Sample	Synthetic	Simulated	MCF-10A	Tissue Micro-array	Mouse Embryo
Slice-by-slice	0.82	0.84	0.78	0.74	0.83
Full 3D	0.86	0.88	0.85	0.81	0.85

Chapter 4: Graphcut Based Method For Nuclear Segmentation in 2D and 3D Optical Microscopy Tissue Images

4.1 Introduction

In this chapter we present a robust and accurate nuclear segmentation algorithm based on graphcuts. The resulting combinatorial optimization algorithm is mainly targeted towards the segmentation of cell nuclei in 3D tissue samples of higher order organisms. We also present a 2D version of the algorithm which is a straight forward simplification of the 3D algorithm and can be used as an independent 2D algorithm in its own merit. The aim of the presented 2D algorithm was to segment as many cell nuclei as possible unlike the 2D segmentation methods presented in chapter 2 where the main aim was to identify only a subset of well segmented nuclei. However, the main focus of the chapter is the 3D segmentation algorithm.

In several biological studies and clinical applications attempting to understand biological processes such as morphogenesis [98] [99], tumorigenesis [100], disease progression and diagnosis [11], protein dynamics [101] [102] and drug screening [103], detection and segmentation of cells, cell nuclei, sub cellular and sub nuclear structures is the stepping stone for extracting biological information from microscopic image datasets. The importance of robust and accurate segmentation is paramount

in a large proportion of such studies and hence decades of research has focused on developing such methods. Although significant progress has been made towards the development of robust 2D segmentation algorithms, in 3D it still remains a challenge and there is a dearth of effective algorithms to reliably segment objects of interest. In this context it should be noted that, compared to 2D cell culture, the problem of segmentation is particularly difficult for 3D images of tissue samples due to tight packing of the cells, low signal to noise ratio and poor resolution in the depth direction.

Inspired by previous works in open pit mining [104] [105] and medical image segmentation [106], the proposed method transforms the microscopic volumes to a geometric volume in the spherical space with respect to a point of reference internal to a target nuclei and finds the globally optimal surface in that geometric volume which separates the target cell nuclei from the rest of the volume. The segmentation problem is posed as one of finding a minimal *closure* [107] in a directed graph and is solved efficiently using the maxflow-mincut algorithm [108]. The algorithm has several desirable properties one of which is gracefully handling the unique problem of anisotropic voxel dimensions. An interactive version of the algorithm requiring minimal manual interaction for segmentation initialization and correction has also been developed for low throughput studies where the accuracy of segmentation is of utmost importance. As opposed to the work reported in [106], this work develops the segmentation method for ellipsoidal objects in microscopic volumes such as cell nuclei, extends the method to handle multiple target objects and also proposes an interactive version of the algorithm. Experiments comparing our segmentation maps

to those obtained from a recently reported state-of-the art segmentation algorithm [109] shows improved robustness and accuracy of the proposed method.

The chapter is organized as follows. Section 4.2 discusses relevant works in the area of nuclear segmentation. Section 4.3 is going to present a detailed description of the graphcut based segmentation method. Section 4.4 provides a description of the experimental datasets and methods followed by the experimental results in section 4.5. Section 4.6 discusses relevant issues, observations regarding the reported method, explores the future avenues of improvements and concludes the chapter.

4.2 Related Work

Automatic nuclear segmentation in 2D fluorescence optical microscopy images has been an area of research for several decades. Some of the most prominent works in this area use methods such as thresholding [110] [111], watershed based algorithm [112] [113] [19], template matching [114], active contours and levelsets [22] [23] [115] [116] to segment 2D images of volume or surface labeled cell nuclei in a wide variety of cell types. Although each of the aforementioned methods work well under certain conditions, there is no single robust and generic algorithm to handle nuclear segmentation in all kinds of microscopic images. Thresholding-based methods work well when the nuclei are well separated and all the images have uniform background illumination. The widely used watershed-based methods suffer from the short coming of over-segmentation which is a severe drawback and although several post processing methods have been developed to post process and merge over segmented watershed outputs, the results are often far from satisfactory.

Template matching-based methods also work well on images having well separated nuclei but fail to handle cases where there is close clustering of nuclei. The active contours and levelsets are often considered state-of-the-art for 2D nuclear segmentation. Computational complexity, careful designing of the energy function to be optimized and requirement of careful initialization of such algorithms are significant drawbacks.

Automatic nuclear segmentation in 3D is significantly more challenging compared to its 2D counter part. Several methods and algorithms have been proposed to perform detection and segmentation for the 3D case using the generalized hough transform [117] [17], watershed algorithm [111] [118] [17] and active contour and levelset based energy minimizing deformable models [119] [120] [56] [121]. However, problems similar to those mentioned for the 2D case also plague the 3D segmentation methods such as the watershed-based methods suffer from over-segmentation and the deformable models suffer from significant computational complexity and requirement for careful initialization. Appleton *et al* [122] reported a 3-D continuous maximal flow based segmentation technique to find globally minimal surfaces. Some other methods that worth mentioning include those based on generic classification [123], gradient flow tracking [124], atlas-based methods [125], geometric filtering and curvature-based partitioning [126], seeded geodesic image segmentation method [109] and graph regularization [127]. For nuclear segmentation in 2D and 3D several interactive segmentation methods have also been developed for applications where the quality and accuracy of segmentation is of paramount importance. Methods reported in [18] [14] use dynamic programming and combinatorial opti-

mization to perform interactive segmentation. [128] report a combination of random walk and the geodesic graph-based method for performing interactive segmentation in microscopic images.

The use of graph-based methods for segmentation in microscopic images is far more limited compared to that of natural scenes. Graph-based image segmentation methods are very versatile and often segmentation algorithms are based on the use of minimum spanning tree [129] [130], shortest paths [131] [132] [18] and graph cuts [133] [134] [135] [136] [137]. The minimum s-t cut [138] method is most relevant to the algorithm presented in this chapter. [139] [15] [117] report graph-based methods developed for nuclear segmentation in microscopic images. The method proposed in [15] is limited to 2D. [139] uses graphcuts to identify nuclear foreground followed by convexity and concavity analysis to split clumped nuclei. However, the method fails to split nuclear clumps which are themselves convex in shape.

4.3 Graph Cut Based Segmentation

The algorithm for segmenting 2D and 3D nuclei in microscopic images is posed as finding optimal surfaces separating each object from the rest of the volume. For a single nuclei, a minimum s-t graph cut is used to efficiently solve the problem on a edge weighted graph derived from the original volume of interest. The method was then extended to handle the common problem of segmenting multiple nuclei of interest in a microscopic volume.

4.3.1 Minimal Closure and Equivalent s-t Cut

The problem of finding an optimal surface enclosing the object of interest is first reduced to the problem of finding the minimal *closure* on a directed graph (digraph) $G(V, E)$, where $V = \{v_1, v_2, \dots, v_n\}$ is the set of n nodes and $E = \{(v_i, v_j)\}$ is the set of directed arcs where (v_i, v_j) is the arc from node v_i to v_j . Also the digraph is node weighted i.e. each node v_i in G has an associated real valued cost w_i . A *closure* \mathcal{C} on graph G is defined as a set of nodes whose successors are also within \mathcal{C} . The minimal *closure* \mathcal{C}_{min} is the *closure* for which the sum of the costs of all nodes in the *closure* ($\sum_{v_i \in \mathcal{C}} w_i$) is minimum.

Picard [107] showed that the problem of finding \mathcal{C}_{min} for a node weighted digraph G can be solved by finding the minimal cut (min-cut) on a edge weighted source-sink (s-t) digraph which is derived from G . The minimal closure problem for digraph G can be represented as

$$\text{Min } g(z) = \sum_{i=1}^n -w_i z_i + \sum_{i=1}^n \sum_{j=1}^n \kappa q_{ij} z_i (1 - z_j) \quad (4.1)$$

where z_i is an indicator variable which is 1 if $v_i \in \mathcal{C}$ and 0 otherwise, q_{ij} is the ij^{th} entry of the incidence matrix of G i.e. $q_{ij} = 1$ if $(v_i, v_j) \in E$ and 0 otherwise and κ is a constant which is large enough to ensure that $\sum_{i=1}^n \sum_{j=1}^n q_{ij} z_i (1 - z_j) = 0$ which in turn ensures that \mathcal{C} is a closure in G by imposing the condition $z_i \leq z_j$ for $(v_i, v_j) \in E$.

Considering another graph $G'(V', E')$ with vertices $V' = \{v'_0, v'_1, \dots, v'_n, v'_{n+1}\}$, v'_0 and v'_{n+1} being the source and sink nodes respectively, a cut (C, \bar{C}) where $(v'_0 \in$

$C, v'_{n+1} \in \bar{C}, C \cup \bar{C} = V, C \cap \bar{C} = \emptyset$) can be expressed as a vector $(1, y_1, y_2, \dots, y_n, 0)$ [140], where $y_0 = 1, y_{n+1} = 0, y_i = 0$ or 1 for $i = 1, 2, \dots, n, C = \{v'_i | y_i = 1\}$ and $\bar{C} = \{v'_i | y_i = 0\}$. The capacity of the cut can be expressed as

$$c(Y) = \sum_{j=1}^{n+1} c_{0,j} + \sum_{i=1}^n (c_{i,n+1} - c_{0,i})y_i + \sum_{i=1}^n \sum_{j=1}^n c_{i,j}y_i(1 - y_j) \quad (4.2)$$

where $c_{i,j}$ is the capacity of arc $(v'_i, v'_j) \in E'$ and $Y = (y_1, y_2, \dots, y_n)$.

Equations 4.1 and 4.2 are similar in form with the exception of a constant term $\sum_{j=1}^{n+1} c_{0,j}$. Hence the minimal closure problem on G can be solved by performing a min-cut on a derived equivalent graph G' . Let $T^+ = \{i | w_i > 0; i = 1, 2, \dots, n\}$ and $T^- = \{i | w_i < 0; i = 1, 2, \dots, n\}$. The graph G' is then constructed as follows,

- **Nodes:** Retain all existing nodes and add two additional nodes, a source v_0 also called s and a sink v_{n+1} also called t to G
- **∞ Edges:** Assign a very high capacity κ (can be set to infinity) to the existing arcs in graph G i.e. $c_{i,j} = \kappa = \infty$.
- **s-t Edges:** $\forall i \in T^+$, add arcs from v_0 to v_i with capacity $c_{0,i} = w_i$ and $\forall i \in T^-$, add arcs from v_i to v_{n+1} with capacity $c_{i,n+1} = -w_i$

Subsequently, performing a min-cut on G' provides the minimal closure for G .

4.3.2 Single Object Segmentation

The segmentation algorithm is illustrated for a 3D object embedded in a volume. The 2D case is a straight forward simplification of the same. The segmentation method assumes the object to be *point convex* i.e. all the lines connecting boundary

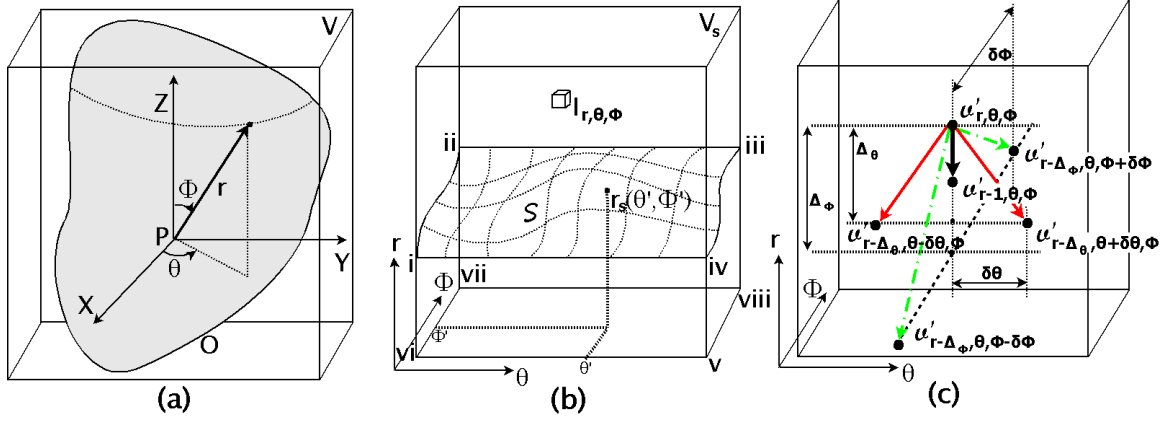


Figure 4.1: (a) Object O embedded in a 3D volume in Cartesian coordinate system and corresponding variables r , θ and ϕ in spherical coordinate system. (b) Surface S representing point convex object O in the spherical volume V_s (c) Edge connections of the graph shown for a single vertex $v'_{r,\theta,\phi}$ corresponding to voxel $I_{r,\theta,\phi}$ in (b). The red and green edges show connections in the θ and ϕ dimensions respectively

points of the object to atleast one internal point are completely enclosed by the object, which is a valid assumption for most frequently encountered cell nuclei in microscopic images.

Let V be a volume where intensity at each voxel is inversely related to the likelihood (based on a suitable pre-defined notion of edge strength) of being on the surface of an object O with an internal point P with respect to which O is point convex as shown in Fig. 4.1(a). Defining a $\{\mathbf{x}, \mathbf{y}, \mathbf{z}\}$ Cartesian coordinate system with P as the origin, a spherical transform is performed to convert volume V to a volume V_s in $\{\mathbf{r}, \boldsymbol{\theta}, \boldsymbol{\phi}\}$ space (Fig. 4.1(b)) where $r \in \mathbf{r} = \{0, 1, 2, \dots, r_{max}\}$ is the radial distance upto a certain distance r_{max} from the origin, $\theta \in \boldsymbol{\theta} = \{0, \delta\theta, 2\delta\theta, \dots, 2\pi\}$ is the azimuthal angle and $\phi \in \boldsymbol{\phi} = \{0, \delta\phi, \dots, \pi\}$ is the polar angle. Conceptually, the object O is unwrapped with respect to the origin P and the enclosing surface of the object in Cartesian coordinate system is transformed to a terrain-like surface

(\mathcal{S}) in the spherical domain as illustrated in Figure 4.1(b). Since object O is point convex with respect to P each column of the spherical volume contains one and only one voxel belonging to \mathcal{S} i.e. for a particular value of θ and ϕ , the radial distance of a surface point $r_{\mathcal{S}}(\theta, \phi)$ is unique and the cost of the surface \mathcal{S} defined as the sum of intensities ($I_{r,\theta,\phi}$) of all voxels $\in \mathcal{S}$ ($\sum_{\mathcal{S}} I_{r,\theta,\phi}$) is the global minimum among all such terrain-like surfaces.

In order to formulate the problem as a minimal closure problem, a new volume V'_S is derived from V_S such that

$$w'_{r,\theta,\phi} = \begin{cases} I_{r,\theta,\phi} & \text{if } r = 0, \\ I_{r,\theta,\phi} - I_{r-1,\theta,\phi} & \text{if } r = 1, 2, \dots, r_{max} \end{cases} \quad (4.3)$$

where $I_{r,\theta,\phi}$ are voxel intensities of V_S and $w'_{r,\theta,\phi}$ are voxel intensities of V'_S . Due to such a construction of V'_S , the cost of \mathcal{S} in V_S is equal to the sum of intensities of set of all voxels enclosed by the plane $r = 0$ and surface \mathcal{S} in V'_S i.e. $\sum_{\mathcal{S} \in V_S} I_{r,\theta,\phi} = \sum_{\theta,\phi \in V'_S} w'_{0 \leq r \leq r_{\mathcal{S}}(\theta,\phi),\theta,\phi}$. Thus the problem of finding the surface \mathcal{S} in volume V_S with minimal cost is modified into that of finding the set of voxels enclosed by the $r = 0$ plane and surface \mathcal{S} in V'_S with an equal minimal cost. To find this set of voxels, an equivalent edge weighted graph G' is subsequently constructed so that the minimal closure C^* corresponds to the same set of voxels enclosed by the $r = 0$ plane and surface \mathcal{S} in V'_S

The equivalent graph G' to find the minimal closure of V'_S is created as follows:

- **Nodes:** Each voxel $v'_{r,\theta,\phi}$ in volume V'_S in spherical domain represent a node

in the graph $G'(V', A')$. In addition, a source node s and a sink node t are added to the set of nodes i.e. $V' = \{s, t, v'_{r,\theta,\phi}\}$.

- ∞ **Edges** The aforementioned problem formulation requires that if voxel $v'_{r,\theta,\phi}$ belongs to the minimal closure, all voxels $v'_{0 \leq r' \leq r, \theta, \phi}$ also belongs to the minimal closure. The following ∞ -capacity edges of G' impose such a condition,

$$(v'_{r,\theta,\phi}, v'_{r-1,\theta,\phi}) \in A' \quad \text{where } r > 0$$

The next set of edges control the flexibility of the minimal surface by imposing certain geometric constraints which can be application specific. The ∞ -capacity edges are defined as,

$$(v'_{r,\theta,\phi}, v'_{\max(0, r-\Delta_\theta), \theta-\delta\theta, \phi}) \in A'$$

$$(v'_{r,\theta,\phi}, v'_{\max(0, r-\Delta_\theta), \theta+\delta\theta, \phi}) \in A'$$

$$(v'_{r,\theta,\phi}, v'_{\max(0, r-\Delta_\phi), \theta, \phi-\delta\phi}) \in A'$$

$$(v'_{r,\theta,\phi}, v'_{\max(0, r-\Delta_\phi), \theta, \phi+\delta\phi}) \in A'$$

where the parameters Δ_θ and Δ_ϕ control the maximum radial variation allowed for a feasible surface in θ and ϕ directions respectively. In other words, if a voxel $v'_{r,\theta,\phi}$ is on the minimal surface, it's nearest neighbors in θ direction, $v'_{r,\theta-\delta\theta,\phi}$ and $v'_{r,\theta+\delta\theta,\phi}$, cannot be lower or higher than Δ_θ relative to $v'_{r,\theta,\phi}$. The same explanation holds true for the ϕ direction. All edge connections for a single node is shown in Fig. 4.1(c).

- **s-t Edges** Defining two sets of voxels $T^+ = \{v'_{r,\theta,\phi} | w'_{r,\theta,\phi} > 0\}$ and $T^- = \{v'_{r,\theta,\phi} | w'_{r,\theta,\phi} < 0\}$, the s-t edges of G' are defined as,

$$(s, v_{r,\theta,\phi}^+) \in A' \quad \text{where} \quad v_{r,\theta,\phi}^+ \in T^+ \quad \text{and capacity} \quad + w'_{r,\theta,\phi}$$

$$(v_{r,\theta,\phi}^-, t) \in A' \quad \text{where} \quad v_{r,\theta,\phi}^- \in T^- \quad \text{and capacity} \quad - w'_{r,\theta,\phi}$$

- **Translation of Base Set** The set of nodes corresponding to the plane $r = 0$ is termed as the base set and by construction of G' this set of nodes is contained by all non empty closures in G' . However, an empty set is also a closure in G' and in order to avoid the minimum closure to be an empty set, all nodes in the base set are assigned an arbitrary negative node weight in V'_S such as -1 . This is called a translation and ensures that the minimal closure is a non-empty set of nodes.
- **Additional Constraints** For maintaining continuity of the surface of object O in the Cartesian coordinates, the vertices on edge planes $\theta = 0$ and $\theta = 2\pi$ are connected using the same connectivity constraints in θ so that on the optimal surface, the edge nodes on $\theta = 0$ plane is within Δ_θ distance from the nodes on $\theta = 2\pi$. Also, for $\phi = 0$ and $\phi = \pi$ planes, the value of r should be a constant corresponding to single points on the north and south poles of the object O . Hence for nodes on those two edge planes, $\Delta_\theta = 0$ which in turn ensures that r is a constant on those points.

The min-cut of G' corresponds to the set of voxels enclosed by the $r = 0$ plane and surface \mathcal{S} in V'_S and the upper envelope (in terms of radial distance r) of this set corresponds to the minimal surface \mathcal{S} of object O .

4.3.3 Automatic Multi-Object Segmentation

The method for segmenting single objects, as outlined above, was subsequently extended to handle multi-object segmentation in 2D and 3D microscopic images. For both 2D and 3D cases, the most important step was to identify a point internal to each individual object, with respect to which the spherical transform was performed in order to create the edge weighted graph G' on which the min-cut problem was solved. The procedure was repeated for each individual object, one at a time, in order to find the best surface separating that object from the rest of the volume without overlapping with segmentations of all previously segmented objects.

4.3.3.1 Segmentation in 2D

Identification of object internal point can be tackled in a number of ways depending on the application specific sample staining and quality of images available. For example, in a number of applications, the cell nucleus and the cell membrane are tagged with fluorescent dyes and the goal is to segment individual cells using the membrane stain. Although the cells can be tightly packed, the nuclei are often well separated so that individual nuclear bodies can be identified by simple operations such as thresholding and they can then act as cellular internal points which can be used to initialize the graph cut segmentation.

In our work we developed a 2D object model-based graph search technique to

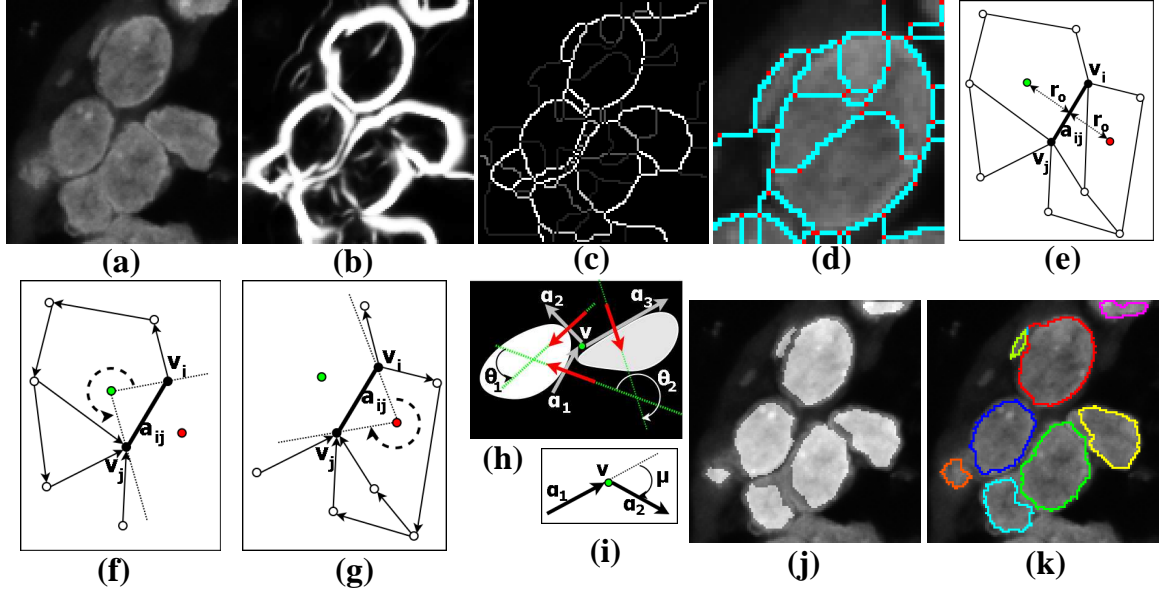


Figure 4.2: (a) Single slice of a subvolume of human breast tissue sample showing DAPI stained cell nuclei (b) Contrast stretched probabilistic edge map (c) Contours extracted using oriented watershed and ultra metric contour map. Gray scale values of the contours are proportional to their respective probabilistic edge strength (d) Blue contours and red dots corresponding to the edges and vertices of the 2D dynamic programming based segmentation graph (e) Sample graph around contour/edge a_{ij} having vertices v_i and v_j and red and green points at a distance of average sample radius on either side of the edge (f) Clockwise directed graph constructed using green point (g) Anti-clockwise directed graph constructed using red point (h) Red arrows showing gradient direction across edges a_1 , a_2 and a_3 . θ_1 and θ_2 show angle between gradient directions of parent edge a_1 to prospective child edges a_2 and a_3 (i) Angular deviation μ between parent edge a_1 and child edge a_2 (j) 2D initial rough segmentation shown as a white overlay on original image (k) Final 2D segmentation using the graphcut based method initialized using centroids of rough segmented objects.

estimate object internal points which can handle more sophisticated nuclear segmentation tasks. Figure 4.2(a) shows a small region of interest of a human breast tissue sample visualizing closely packed cell nuclei, which are to be segmented. Simple thresholding and morphological operations were unable to reliably identify internal nuclear points. In this case, the object edges were first quantified using the 2D probabilistic edge detector illustrated in chapter 3. Since the edge detector was

supervised, an user had to delineate objects of interest in a small subset of training images. Two levels of information were extracted from the training data, (i) the pixel level information used for training the edge detector and (ii) the object level information in the form of features(capturing shape, morphology, texture etc) which were used to train a logistic regression(LR) based object model. Using the object edges detected in the previous step (Fig. 4.2(b)), the probabilistic edge strength weighted dominant contours of the image were then extracted using the oriented watershed transform [68] followed by the generation of the ultra metric contour map [68] as shown in Fig. 4.2(c). A graph $G = (V, A)$ was created from the weighted contours where $V = \{\text{set of all contour junctions}\}$ and $A = \{\text{set of all edge weighted contours}\}$ (shown as red dots and blue contours) (Fig.4.2(d)). For each edge $a_{ij}(= (v_i, v_j) \in A)$ in the graph (Fig. 4.2(e)), assuming that two objects of interest might be located on either side of the edge, two directed acyclic graphs (DAGs) (Fig. 4.2(f) and Fig. 4.2(g) respectively) were created by assigning the rest of the contours clockwise or anti-clockwise directionality with respect to two points situated at object mean radial distance ($= r_O$ extracted from training data) on either side of the original contour a_{ij} . Hence, a path originating from v_i and ending at v_j created a closed loop in clockwise or anti-clockwise directions respectively. Dynamic programming(DP) was used to find the optimal clockwise and counter clockwise loops on either side of a_{ij} . For performing DP, three kinds of equally weighted costs were used, (i) edge transition costs penalizing large angular deviation (μ in Fig. 4.2(i)) from parent (a_1) to child edge(a_2), (ii) probabilistic edge strength encouraging inclusion of edges having higher edge probability and (iii) gradient direction costs penalizing transi-

tions where angular deviation of gradient direction (θ_1 and θ_2 in Fig. 4.2(h)) across parent and child edges were high. The objects created by those loops were subsequently scored by the LR based object model to quantify the goodness of fit with respect to the target objects. This procedure was repeated for all edges in A and for each pixel in the 2D image the highest LR score was retained. Thresholding the model suggested LR scores at a very low level (1^{-6} in our case) resulted in a rough segmentation of the original image (Fig. 4.2(j)). The centroids of the objects found in the previous segmentation were subsequently used to initialize the graph cut based segmentation for a more accurate result (Fig. 4.2(k)).

4.3.3.2 Automatic Segmentation in 3D

The multi-object 3D segmentation was again initialized by a set of object internal points identified with the help of the 2D segmentation. At first, the volume was segmented slice by slice using the 2D segmentation method outlined in the previous section and LR based 2D object model scores s_i and centroids $\mathbf{c}_i = (c_x c_y c_z)$ where $i \in \{\text{set of 2D segmented objects } O_{2D}\}$ were retained for all segmented objects. Using these scores, we calculate one more score for each 2D object indicating the likelihood of the 2D object to be close to the center of a 3D object in z direction. Calculation of the score assumed that for a particular 3D object, the corresponding 2D slice objects around the central region of the 3D object in z direction are well segmented since they conformed well with the 2D object model (usually in 2D imaging of a 3D object, the central slices in z direction are taken to be a faithful representation of the object in 2D) and the centroids of all 2D objects belonging to

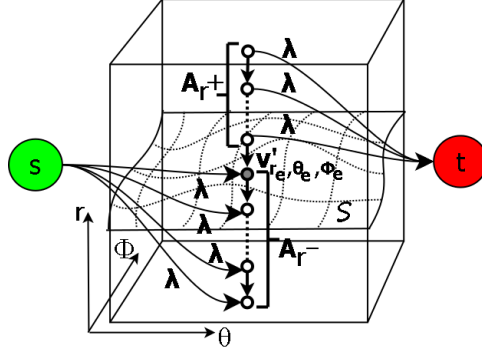


Figure 4.3: Graph modifications for interactive segmentation

a particular 3D object were close to each other within the 3D volume. The score τ_i for each segmented object in the slice by slice segmentation was defined in terms of a 3D Gaussian kernel as,

$$\tau_i = \sum_{j \in O_{2D}} s_j e^{-\mathbf{d}_{ij}^T \Sigma \mathbf{d}_{ij}} \quad \text{with} \quad \mathbf{d}_{ij} = (\mathbf{c}_i - \mathbf{c}_j) \quad \Sigma = \begin{bmatrix} \frac{1}{\sigma_x^2} & 0 & 0 \\ 0 & \frac{1}{\sigma_y^2} & 0 \\ 0 & 0 & \frac{1}{\sigma_z^2} \end{bmatrix} \quad (4.4)$$

where σ_x , σ_y and σ_z were user specified parameters depending on the x , y and z extent of a typical object in the 3D volume. A greedy approach was utilized to perform the actual 3D segmentation by sorting τ_i in descending order for all objects in O_{2D} and initializing the first 3D segmentation with the centroid of the highest scored object. Once the first object was segmented, all 2D objects having centroids within the 3D segmentation were discarded and the second 3D segmentation was initialized by the centroid of the highest scored object among all remaining objects. This was continued till all 2D objects were used up.

4.3.4 Interactive Segmentation

The graph cut segmentation method was also adapted to perform interactive segmentation of 2D and 3D objects. As mentioned earlier, identification of an internal object point was crucial for a reliable multi-object segmentation. For interactive segmentation, the user provided this internal point by using a single mouse click. Once the initial segmentation was completed, an user could add additional edit points in order to correct the initial segmentation and could keep on doing the edits till he/she was satisfied with the segmentation. Once done with one object, they could start the same procedure with other objects in the microscopic image. In this case too, once an object was segmented, all nodes of G' contained within the object were assigned a large negative value resulting in non-overlapping segmentation of the objects. When an edit point was added to an initial segmentation, the corresponding spherical domain node $v'_{r_e, \theta_e, \phi_e}$ in G' was first calculated. The edge weighted graph G' was subsequently modified by adding two sets of additional edges A_{r-} and A_{r+} where,

$$A_{r-} = (s, v'_{r \leq r_e, \theta_e, \phi_e}) \quad \text{with cost } \lambda = \infty$$

$$A_{r+} = (v'_{r > r_e, \theta_e, \phi_e}, t) \quad \text{with cost } \lambda = \infty$$

as shown in Fig. 4.3. The corresponding graph cut solution forced all nodes $v'_{r \leq r_e, \theta_e, \phi_e}$ to be included in the minimal closure of G' and all nodes $v'_{r > r_e, \theta_e, \phi_e}$ were excluded from the minimal closure so that the minimal surface \mathcal{S} included the node

$v'_{r_e, \theta_e, \Phi_e}$. The final segmentation of an object was computed using the modified G' which included all the additional edges resulting from all additional edit points incorporated by the user.

For cases in which multiple edit points are required for a successful segmentation, there remains a potential for addition of edit points which violate the convexity and smoothness constraints of the graph. In those case the solution is the best possible segmentation of the nucleus with inclusion of the maximal number of edit points without violating the convexity and smoothness constraints of the graph. Although the convexity constraint has to be maintained in all cases, the permissible smoothness of the graph can be easily adapted to the application requirements by altering the Δ_θ and Δ_ϕ parameters of the graph.

4.4 Experimental Datasets and Methods

Experiments were performed on two kinds of datasets. The simulated datasets created using real microscopic objects and the real microscopic volumes illustrated in chapter 3 were used for quantifying the performance of the segmentation algorithm in both 2D and 3D. For the simulated datasets the ground truth was available and the performance quantification was accurate. However, for the the actual microscopic volumes, the ground truth was generated by humans using the interactive graph cut based segmentation interface and hence the performance quantification is not as accurate as the simulated volumes since the ground truth segmentation is subjective and depends on the perception of the human annotator.

4.4.1 Segmentation Evaluation Metrics

Three kinds of metrics were used for quantifying the accuracy of the segmentation method. The metrics were as follows:

- **Rand Index (RI):** Rand index [141] is a method for computing similarity between two clusterings of a set of points $S = \{s_1, s_2, \dots, s_n\}$ with possibly different number of classes. The index is computed by looking at pairwise label relationship of all pairs of points in the original set. If S^1 and S^2 are two valid labelings of the set S with corresponding labels $\{l_i^1\}$ and $\{l_i^2\}$ with $i = 1, 2, \dots, n$, the Rand Index measuring similarity between the two labelings is defined as

$$RI(S^1, S^2) = \frac{1}{\binom{n}{2}} \sum_{\substack{i,j \\ i \neq j}} [\mathbf{I}(l_i^1 = l_j^1 \cap l_i^2 = l_j^2) + \mathbf{I}(l_i^1 \neq l_j^1 \cap l_i^2 \neq l_j^2)] \quad (4.5)$$

where \mathbf{I} is the indicator function. Effectively, it is the ratio of the number of point pairs having same labeling relationship and the total number of unique point pairs. The Rand Index can be computed in an efficient way using the following expression if the number of labels is much smaller than the number of points

$$RI(S^1, S^2) = 1 - \frac{\left[\frac{1}{2}(\sum_p n_{p\bullet}^2 + \sum_q n_{\bullet q}^2) - \sum_{p,q} n_{pq}^2 \right]}{N(N-1)/2} \quad (4.6)$$

where $n_{p\bullet}$ is the number of points having label p in S^1 , $n_{\bullet q}$ is the number of points having label q in S^2 and n_{pq} is the number of points having label p in

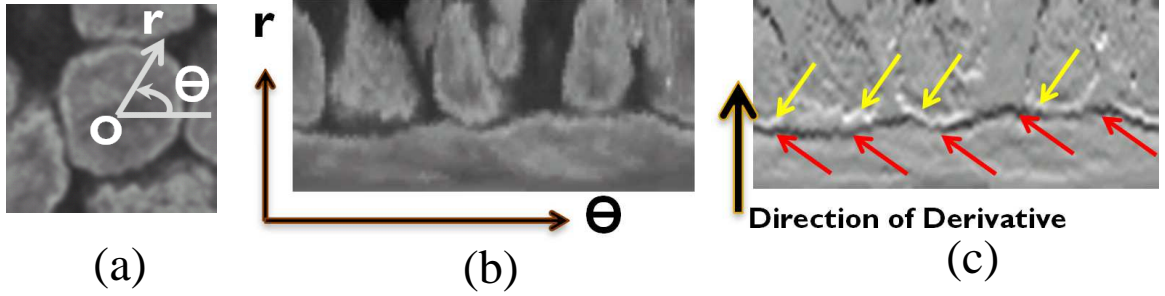


Figure 4.4: (a) Sample 2D nucleus with polar coordinates superimposed. O is the object inner seed point. (b) Polar transformed image (c) Radially outward directional gradient edge measurement in 2D. The red arrows show the negative edge measure produced by the target object and yellow arrows show the positive edge measure produced by neighboring objects which discourages the inclusion of those regions in the optimal cut.

S^1 and q in S^2 respectively.

The Rand Index was used to find the similarity between the ground truth segmentation and the test segmentation for both 2D and 3D cases. The value of the index varies from 0 to 1 with 0 signifying no correspondence between S^1 and S^2 and 1 signifying perfect correspondence between the two labelings.

- **Average Overlap Ratio (OR):** The overlap ratio measured the overlap between a segmented region in the test segmentation and the best possible corresponding region in the ground truth segmentation based on the maximum overlap. For segmented objects O_1 in the test segmentation and O_2 in the ground truth segmentation, the overlap ratio was defined as

$$OR(O_1, O_2) = \frac{2\mathbf{A}(O_1 \cap O_2)}{\mathbf{A}(O_1) + \mathbf{A}(O_2)} \quad (4.7)$$

where $\mathbf{A}(O)$ is the area or volume of a region O and \cap is the intersection of the two regions O_1 and O_2 . The metric was averaged over all the test segmentation

Table 4.1: Rand Index(RI), mean Overlap Ratio(OR) and mean/standard deviation of Boundary Deviation (BD) (in terms of pixels) metrics comparing graphcut (GC) based segmentation and MINS segmentation maps with ground truth object boundaries of 2D simulated datasets.

Blur	SNR	2			3			4		
		RI	OR	BD	RI	OR	BD	RI	OR	BD
0.4	GC-Gradient	0.95	0.95	1.4/0.3	0.96	0.95	1.3/0.2	0.96	0.95	1.2/0.07
	MINS	0.89	0.87	1.7/0.8	0.92	0.92	1.4/0.6	0.95	0.94	1.3/0.6
0.7	GC-Gradient	0.95	0.92	1.6/0.7	0.96	0.96	1.3/0.2	0.97	0.95	1.3/0.1
	MINS	0.9	0.9	1.7/0.8	0.92	0.92	1.5/0.5	0.95	0.93	1.4/0.6
1.0	GC-Gradient	0.94	0.87	2.8/3.0	0.96	0.95	1.3/0.1	0.95	0.95	1.4/0.2
	MINS	0.9	0.87	1.7/0.7	0.92	0.84	3.8/5	0.93	0.9	1.5/0.6

objects to obtain a consolidated measure of overlap accuracy.

- Mean Boundary Deviation (BD):** The mean boundary deviation has been defined in chapter 3 for the 2D case. We extended it for the 3D case here. It measures the average deviation of each object boundary pixel/voxel of the test segmentation from the closest boundary point of the corresponding object (based on maximum overlap) on the ground truth segmentation in terms of pixels/voxels calculated using an Euclidean distance transform which assigned progressively increasing distances to pixels/voxels as one moved away from the groundtruth boundary. We report the mean value for all boundary pixels/voxels and the corresponding standard deviations.

4.4.2 Edge Measurement Methods

For all 2D experiments using the graphcut (GC) segmentation method, a radially outward normalized gradient measure with respect to the inner seed point of an object was used as the edge measure. This measure encodes the directional slope information of the edges which increase the robustness of the segmentation by preventing the boundaries from jumping on boundaries of adjacent nuclei as il-

illustrated in Fig. 4.4. Briefly, the directional gradient calculated with respect to an inner seed point for an object is negative for the object edges itself (shown by red arrows in Fig. 4.4(c)) and positive for edges belonging to adjacent objects (shown by yellow arrows in Fig. 4.4(c)). Since the GC method finds a min cut on the graph derived from the directional gradient image, the cut is discouraged to include the positive gradient regions which discourages the segmentation boundary to jump on to neighboring objects.

For the 3D GC based method, two kinds of edge measurements were used for the simulated datasets. The first one was based on MiPb, the 3D probabilistic edge measure presented in the previous chapter. The other edge measure was a directional gradient (termed as Gradient in tables 4.1 and 4.3) calculated radially outwards from the inner seed point of an object as illustrated in 2D. Experiments were performed to evaluate whether the MiPb based robust and accurate edge detection method improved the segmentation accuracy. In practice, instead of simple MiPb values, a linear combination of MiPb and normalized directional gradient information was used as the edge measure to take advantage of the directional information encoded in the gradient which is not available from MiPb measurements. A linear combination of the two edge measures was advantageous since it combined the robust edge measurements from MiPb and the directional information encoded in the gradient and discouraged cuts to jump onto neighboring objects. However, due to complexity constraints, MiPb was not calculated for real microscopic 3D volumes and all experiments with GC segmentation were conducted with the radial directional gradient information as the edge measure for those datasets.

4.4.3 Comparison With Other Segmentation Methods

Segmentation maps obtained from the reported 2D and 3D methods were compared to that obtained from a recently reported seeded geodesic image segmentation [109] based 2D and 3D segmentation tool (modular interactive nuclear segmentation - MINS) using the three aforementioned metrics. It was also shown [109] that the segmentation maps obtained using their software out performed in terms of accuracy those obtained from other segmentation methods such as ilastik [142], FARSIGHT [143], and CellSegmentation3D [124]. For both MINS and GC method, the only parameter input was an average nuclear diameter.

4.4.4 Implementation

The proposed algorithms were implemented in Matlab 2010b (The Mathworks Inc., Natick, MA) along with the DipImage Toolbox [38], Matlab Boost Graph Library [144] and pre-existing implementations for mincut-maxflow algorithm [145] and 2D probabilistic edge detection with oriented watershed [68].

4.5 Experimental Results

This section provides quantitative results showcasing the performance of the 2D and 3D segmentation methods.

4.5.1 Results of 2D Automatic Segmentation

Experiments using 2D simulated datasets were performed to quantify the quality of segmentation maps obtained using the graphcut (GC) based segmentation and MINS method to the groundtruth object boundaries. For the simulated datasets the

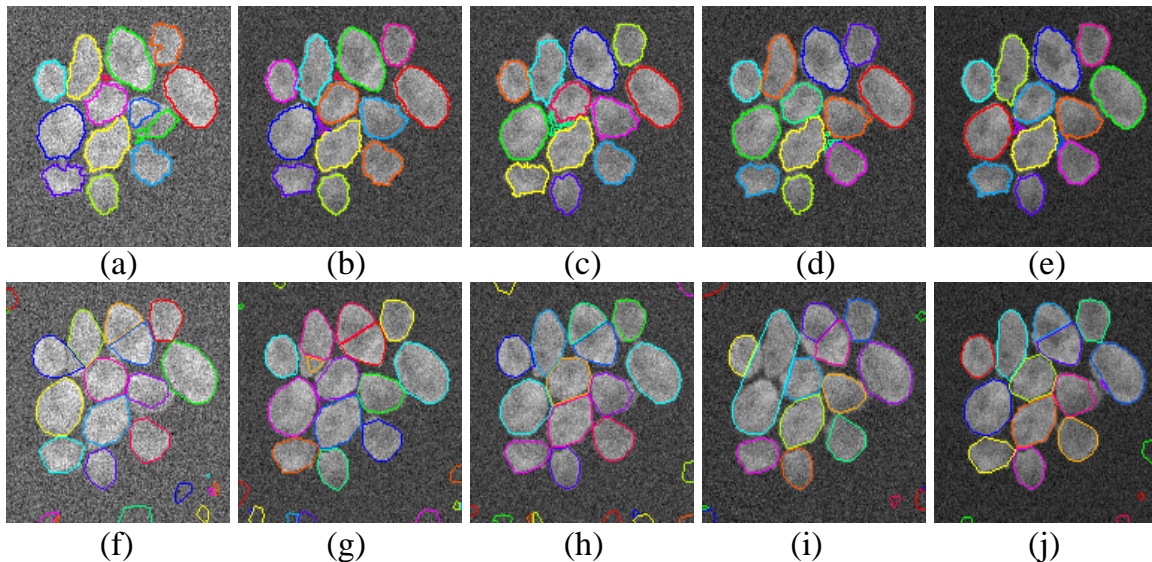


Figure 4.5: Segmentation using graphcut based method of 2D simulated images with (a) SNR 2 Blur 0.4 (b) SNR 3 Blur 0.4 (c) SNR 3 Blur 0.7 (d) SNR 3 Blur 1.0 (e) SNR 4 Blur 0.7. Segmentation using MINS method of 2D simulated images with (f) SNR 2 Blur 0.4 (g) SNR 3 Blur 0.4 (h) SNR 3 Blur 0.7 (i) SNR 3 Blur 1.0 (j) SNR 4 Blur 0.7.

Table 4.2: Rand Index(RI), mean Overlap Ration(OR) and mean/standard deviation of Boundary Deviation (BD) (in terms of pixels) metrics comparing graphcut (GC) based segmentation and MINS segmentation maps with manually annotated object boundaries of 2D actual microscopic datasets.

Dataset	RI		OR		BD	
	GC	MINS	GC	MINS	GC	MINS
MCF-10A	0.96	0.9	0.91	0.81	1.9/2.0	3.2/3.7
Human Breast Tissue	0.92	0.84	0.84	0.73	3.1/3.4	4.0/3.5
Mouse Embryo	0.94	0.86	0.82	0.68	2.3/3.3	3.4/4.2

true object boundaries were available and hence the performance quantification is accurate. Table 4.1 provides the three metrics measured for 2D simulated datasets. In almost all cases, GC method outperforms the MINS method in terms of all the three metrics. It should also be noted that the GC method is particularly robust across various values of SNR and blur. Also in most cases the standard deviation of pixel deviation is lower compared to the MINS method. This metric is particularly important for applications where the boundary accuracy of object delineation is of

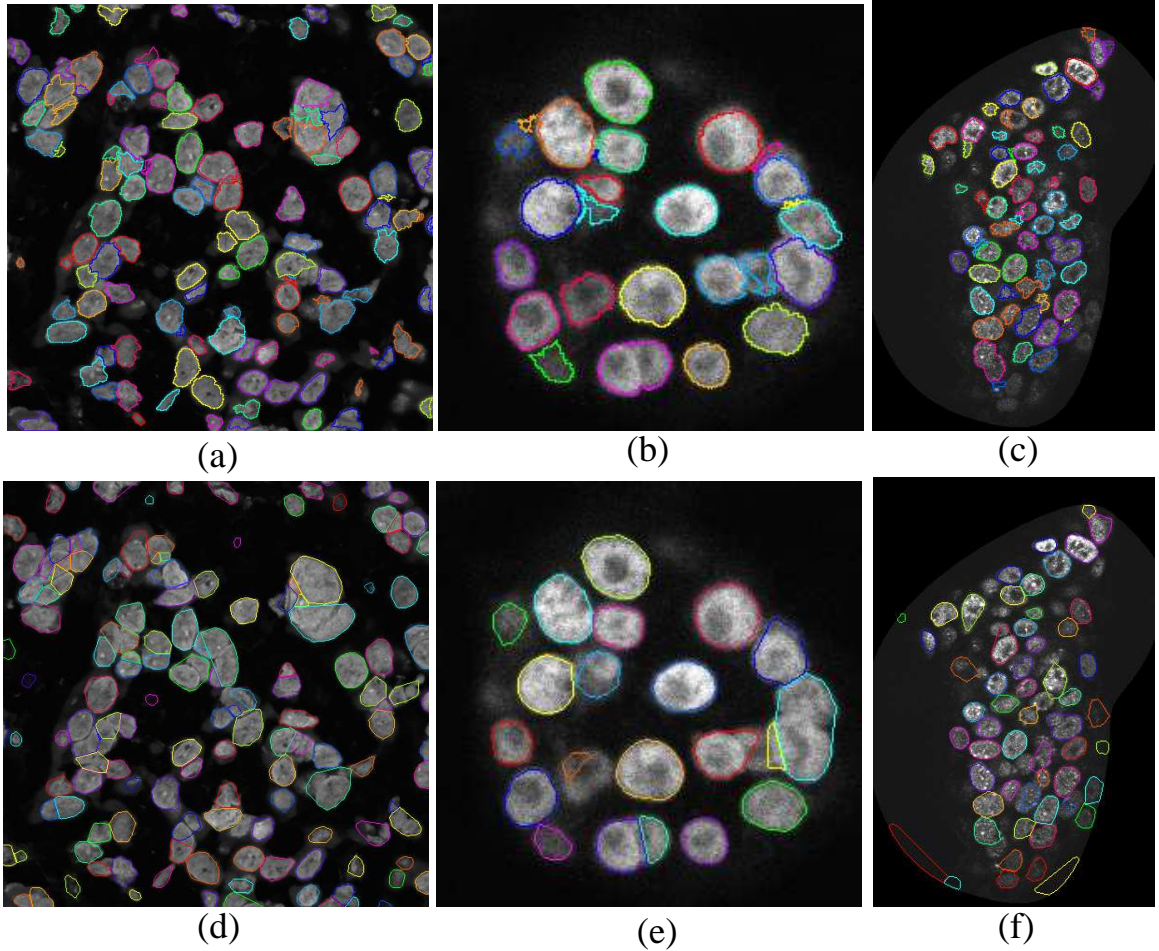


Figure 4.6: Segmentation using graphcut based method of 2D real images from (a) Human breast tissue (b) MCF-10A (c) Mouse embryo datasets respectively. Segmentation using MINS method of 2D real images from (d) Human breast tissue (e) MCF-10A (f) Mouse embryo datasets respectively.

paramount importance.

Fig. 4.5 illustrates a few sample segmentations of the simulated images across various values of SNR and blur. Visual inspection shows that the quality of segmentation for GC method is better than the MINS method as also shown by the quantitative metrics.

Results for experiments involving real microscopic images are shown in Table 4.2. The segmentation maps obtained by the GC and MINS method were compared

Table 4.3: Rand Index(RI), mean Overlap Ratio(OR) and mean/standard deviation of Boundary Deviation (BD) (in terms of voxels) metrics comparing graphcut (GC) based segmentation and MINS segmentation maps with ground truth object boundaries of 3D simulated datasets.

Blur	SNR	2			3			4		
		RI	OR	BD	RI	OR	BD	RI	OR	BD
0.4	GC-MiPb	0.93	0.89	1.0/0.04	0.94	0.91	1.1/0.04	0.94	0.9	1.0/0.03
	GC-Gradient	0.92	0.86	1.1/0.1	0.93	0.89	1.1/0.2	0.94	0.9	1.1/0.02
	MINS	0.82	0.76	1.5/0.3	0.88	0.83	1.3/0.19	0.91	0.85	1.2/0.10
0.7	GC-MiPb	0.90	0.83	1.3/0.2	0.9	0.83	1.2/0.2	0.9	0.85	1.2/0.2
	GC-Gradient	0.88	0.80	1.40/0.1	0.89	0.83	1.2/0.2	0.9	0.85	1.28/0.1
	MINS	0.80	0.74	1.64/0.3	0.86	0.79	1.4/0.2	0.89	0.82	1.3/0.3
1.0	GC-MiPb	0.88	0.83	1.4/0.3	0.87	0.81	1.2/0.1	0.88	0.80	1.3/0.15
	GC-Gradient	0.83	0.78	1.5/0.1	0.84	0.80	1.2/0.1	0.87	0.79	1.2/0.1
	MINS	0.78	0.70	1.73/0.21	0.83	0.75	1.5/0.3	0.84	0.76	1.5/0.2

to that obtained by interactive segmentation of the images by a human annotator. In this case too, GC outperforms the MINS method in terms of all the metrics in almost all cases. In general the mean boundary deviation and standard deviations are higher for both GC and MINS compared to the simulated datasets. However, for the GC method the boundaries are more accurate compared to the MINS method. Fig. 4.6 illustrates a few sample segmentations of the real microscopic images for a human breast tissue , MCF-10A and a mouse embryo sample. Visual inspection shows that the quality of segmentation for GC method is better than the MINS method as also demonstrated by the quantitative metrics.

4.5.2 Results of 3D Automatic Segmentation

The results of segmentation performance evaluation for 3D simulated datasets are shown in Table 4.3 for various SNR and blur values. As mentioned previously, the two kinds of edge measures were used for the GC-based method. The GC based method using both the edge measurements outperformed the MINS based segmentation method in almost all cases. A comparison of the metrics for GC-

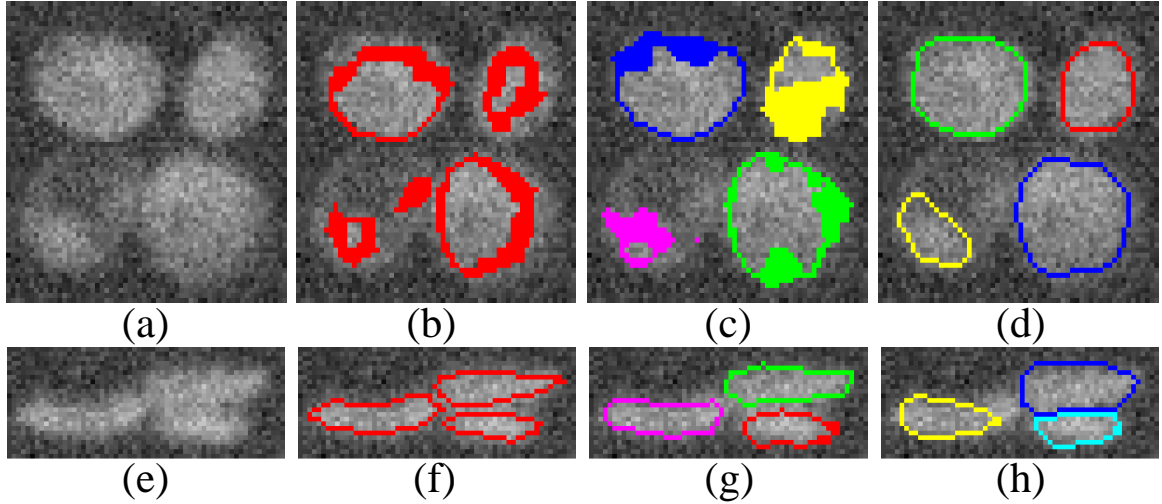


Figure 4.7: (a,e) Single XY and YZ slices of original 3D simulated image at SNR 3 and blurring of 0.7 (b,f) Corresponding groundtruth available from synthetic object boundaries (c,g) Graphcut segmentation (d,h) MINS segmentation.

Table 4.4: Rand Index(RI), mean Overlap Ration(OR) and mean/standard deviation of Boundary Deviation (BD) (in terms of voxels) metrics comparing graphcut (GC) based segmentation and MINS segmentation maps with manually annotated object boundaries of 3D actual microscopic datasets.

Dataset	RI		OR		BD	
	GC-Gradient	MINS	GC-Gradient	MINS	GC-Gradient	MINS
MCF-10A	0.92	0.89	0.76	0.7	2.1/1.2	3.2/2.6
Human Breast Tissue	0.89	0.85	0.76	0.65	2.5/2.9	3.6/3.8
Mouse Embryo	0.87	0.85	0.6	0.6	2.3/1.6	2.8/2.5

MiPb and GC-Gradient shows that in most cases the MiPb-based edge measure did improve the performance of the segmentation algorithm marginally compared to the radial gradient measure. However, for none of the cases the improvement was drastic. Fig. 4.7 shows a single slice of the 3D segmentation boundaries overlaid on the simulated data for various SNR and blur values.

Table 4.4 shows the performance metrics for real microscopic datasets from human breast tissue, MCF-10A and mouse embryo samples. In this case only the GC-Gradient method was used for experimentation due to complexity constraints of MiPb computation. However, the GC-Gradient method itself performed better than

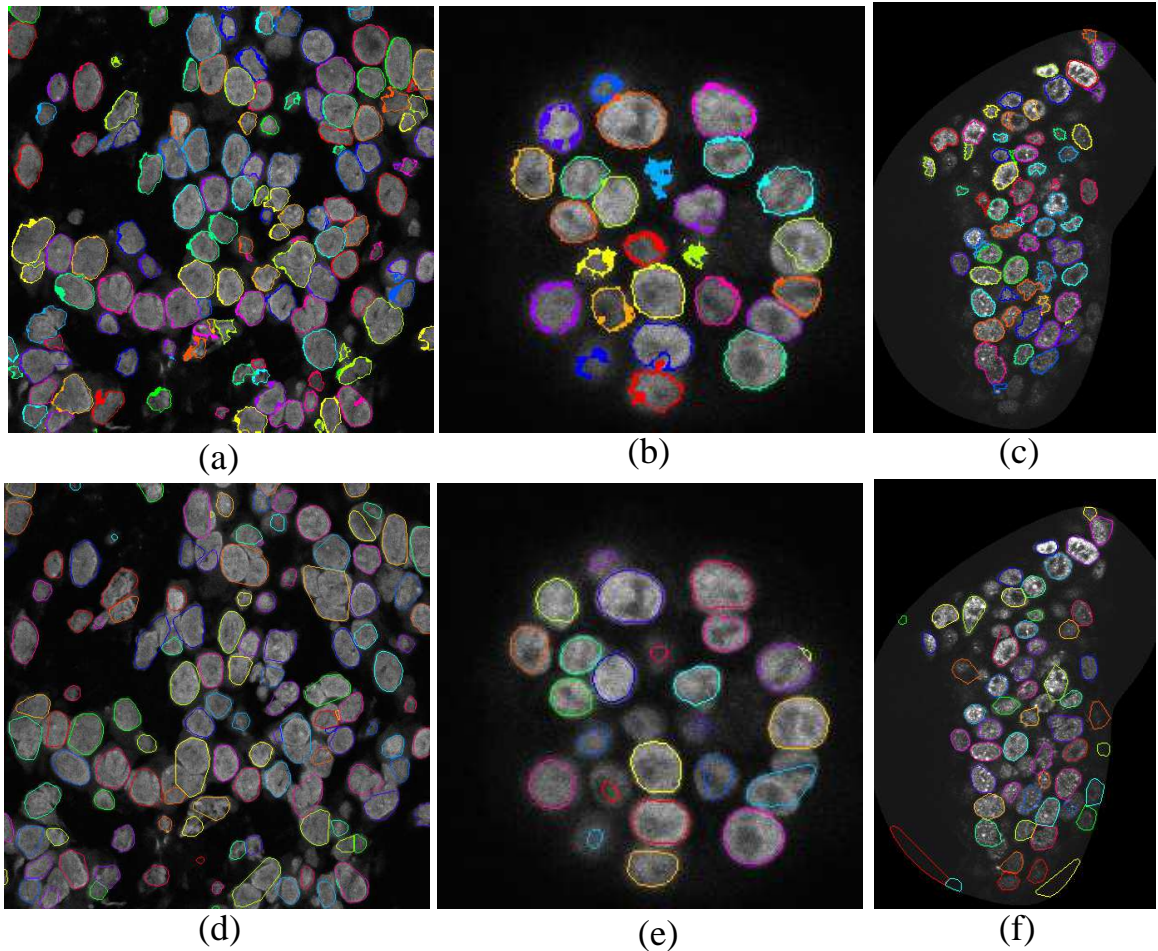


Figure 4.8: Segmentation using graphcut based method of 3D real images from (a) Human breast tissue (b) MCF-10A (c) Mouse embryo datasets respectively. Segmentation using MINS method of 3D real images from (a) Human breast tissue (b) MCF-10A (c) Mouse embryo datasets respectively. Only one slice of the segmentations are shown.

the MINS segmentation method in almost all the cases in terms of all the metrics.

Fig. 4.8 shows a single slice of the 3D segmentation boundaries overlaid on real microscopic data from various datasets.

4.6 Discussion and Conclusions

The current chapter presents a graphcut-based nuclear segmentation algorithm for both 2D and 3D microscopic datasets. Given a measure of the object

edge strengths, the method finds an optimal path (in 2D) or surface (in 3D) which separates an object from rest of the volume. The 2D and 3D segmentation algorithms presented in this chapter showed improved performance compared to a recently reported state-of-the art segmentation tool for similar datasets. However, the algorithm did fail to segment several nuclei in test volumes due to inaccurate seed detection and lack of edge response between two juxtaposed nuclei especially in the z-direction in which the imaging resolution is significantly poor compared to the x and y dimensions. Inaccurate seed point detection contributed to a number failures of the segmentation algorithm. A more robust and accurate seed detection method will improve the segmentation performance. With more confidence on the detected seeds, additional constrains can be incorporated in the graph to successfully avoid cases such as inclusion of multiple seeds within a single object. The second source of segmentation error was poor edge response especially with closely packed nuclei. The improvement due to the use of MiPb based edge measure over using a radial directional derivative was marginal. Due to computational constrains MiPb was calculated in 64 3D orientations. It may be the case that with increase in the number of orientations of MiPb calculation and improved training would result in better edge response which in turn should improve segmentation performance. Also, possibly the robustness of the optimal segmentation method renders the use of a sophisticated edge measure superfluous.

The segmentation method presented is limited to point convex objects. However, for nuclear segmentation this constraint introduced due to the problem formulation using the spherical transform is advantageous since it inherently incorporates

the ellipsoidal structure of cell nuclei observed in most cell types. However, this constraint can be potentially removed by using multiple seed points within a non point convex object and developing a mapping of the space to convert the object to a point convex one. Future studies will look into development of such methods.

A 3D interactive segmentation method using a combination of dynamic programming and combinatorial optimization was presented in [14]. The reported method overcomes quite a few constraints imposed by the dynamic programming-based method. Firstly, for interactive segmentation the previous method needed manual identification of two points, one on the boundary of the object and an internal seed point. The graphcut-based method reduces such manual interaction to a single internal seed point identification. Also the solution derived by [14] is suboptimal. On the other hand, for the method reported in this chapter, the solution is globally optimal in terms of the edge measure for the object of interest. It should be noted that both the methods suffer from sampling artifacts when converting the image volume from Cartesian to spherical domain especially close to the poles. Also to reduce the computational complexity of the algorithm, the authors of [14] searched for a surface in the spherical domain where the neighboring voxels are within the 26-connected neighborhood of each other. The graphcut-based method relaxes this constraint and allows the surface to be more flexible due to a careful interconnection of the ∞ -weight edges of the graph via the parameters Δ_θ and Δ_ϕ which control the extent of flexibility. This in fact also helps in better handling of the anisotropic voxel dimensions inherent in optical microscopy datasets. In this formulation the ϕ dimension in the spherical domain roughly corresponds to the z or depth direction

in Cartesian coordinate system. Hence providing a larger value of Δ_ϕ would give more flexibility to the spherical domain surface in ϕ dimension which in turn would better accommodate for the increased voxel dimension in the z direction.

In future, further experimentation would be done to quantify the strengths and weaknesses of the proposed method in a more comprehensive manner. Simulations will surely play a significant role in such experiments since for real microscopy datasets the true ground truth is not available. Future work will also concentrate on development of pattern classification method to remove outliers and identify 'well-segmented' nuclei in similar line to the 2D study presented in chapter 2. The development of such quality-based nuclear screening or ranking method will further boost the usable final segmented objects obtained using the segmentation method. Also, since the current segmentation method runs one nucleus at a time, the same process can be parallelized to perform segmentation for multiple nuclei at a time. This can be achieved by first using a graph coloring algorithm to color individual seeds/nuclei and running the segmentation for nuclei having the same color at a time since nuclei of same color would not be neighbors in the colored graph.

Chapter 5: MiPipeline (Microscopy Pipeline): A User Friendly Software Ecosystem for Microscopy Image Analysis and Informatics

5.1 Introduction

In this chapter we present *MiPipeline*, a comprehensive compute and data exploration environment for large microscopic image datasets. We have ported the cancer detection technique presented in chapter 2 to the *MiPipeline* environment to analyze a human breast tissue micro-array dataset consisting of 1700 sample images and present a case study on the same. Other case studies showcase the importance of the proposed compute environment by importing essential microscopic image analysis tools such as BioFormats and ImageJ.

The past few years have witnessed unprecedented improvements in optical microscopes particularly in terms of spatial resolution [146] and the automated generation of highcontent inevitably led to an explosion in the quantity of acquired data (MBytes to GBytes to TBytes). Concomitantly, there is an increasing need for rigorous quantification of the complex biological interactions often at hierarchical scales (tissues, cells, subcellular/nuclear, and molecular) that are captured in microscopic images [147] [1] [148] [149]. Thus, analysis and visualization of large, information rich microscopic bio-image datasets is becoming increasingly important and challenging. Since, manual processing and interpretation of such datasets is

impractical and subjective, automatic analysis is essential. However, the variations in the implementations of image processing algorithms and the adaptation of analysis procedures due to differences in samples and/or acquisition platforms lead to results that are poorly reproducible. This brings forth with it an urgency for data and process provenance.

To address the analysis needs of large bio-image datasets, a comprehensive computational environment should ideally provide:

- Access to algorithms and libraries from multiple disciplines (image processing, computer vision, machine learning, and biostatistics [10] [11] [113])
- Data and process provenance for accurate tracking of data and image processing and analysis procedures
- A collaborative workflow management so that multiple participants can seamlessly contribute to data analysis and development of analysis tools
- An environment for visual programming and workflow management requiring minimal programming knowledge to work with
- Integration with high performance computing (HPC) clusters, and
- Information visualization and analytics.

The goal of this study was to provide a comprehensive computing environment for analysis of biological samples imaged with optical microscopy. The environment, *MiPipeline* is based on the LONI Pipeline [150] for image processing and analysis coupled with a new web browser based visual analytics. LONI

Pipeline [150] (<http://www.loni.usc.edu/Software/Pipeline>) is a computational tool originally built for processing neuroimaging data. It has two components: A distributed processing server (LONI-DPS), which is deployed on a central machine and controls high performance computational resources via a cluster management software. The second component, the *Pipeline client*, available for popular operating systems (OS), enables a user to connect to the LONI-DPS. The client provides a rich visual programming environment where individual data processing steps, represented as modules, can be interconnected to create complex analysis workflows. The reported *MiPipeline* environment provides:

1. A collaborative environment with independence from specific application programming interface or API (e.g., Java or Python)
2. End-user friendly visual workflows
3. Tight integration with existing HPC resources
4. Data provenance and reproducibility through provision of an XML [151] backbone that tracks image data and its progress through processing and analysis algorithms
5. Rapid transition from prototyping to production
6. Availability of Bioformats [152] for importing images in formats commonly used in biomicroscopy and ImageJ/Fiji for image processing and analysis
7. Preexisting pool of multidisciplinary algorithms included in LONI Pipeline that can be used in creating visual workflows

8. Seamless information exploration and analytics via a web interface

The subsequent sections of this chapter are organized as follows: The next section discusses some related work. The following section describes the MiPipeline environment infrastructure. The next section provides a simple, worked example of its utilization for segmenting cell nuclei from images. Subsequent sections showcase MiPipeline for two case studies: super-resolution localization of single, photo-activated fluorescence molecules, and breast cancer detection by measuring the position of specific genes in the cell nuclei in patient biopsy tissue sections. The chapter concludes with discussion of future extensions of MiPipeline.

5.2 Related Work

Several promising data processing platforms for bio-image data have emerged in the past few years that each address a subset of aforementioned needs. The principal platforms include: OME/OMERO [153] [154], Farsight Toolkit [143], Cell Profiler [155] and ICY [156]. OME/OMERO provides data provenance, collaborative image database capabilities along with server side script execution from the local client, and it supports grid computing through the use of the IceGrid [157] framework. The Farsight Toolkit provides multiple image processing modules that can be toolchained using a scripting language, but it does not currently provide tools for managing visual workflows. Cell Profiler is similar in that it provides a wide array of image processing and machine learning algorithms, including data provenance and workflow sharing capabilities. It, however, has minimal HPC integration capabilities and does not provide an environment for visual workflow creation. ICY, in contrast,

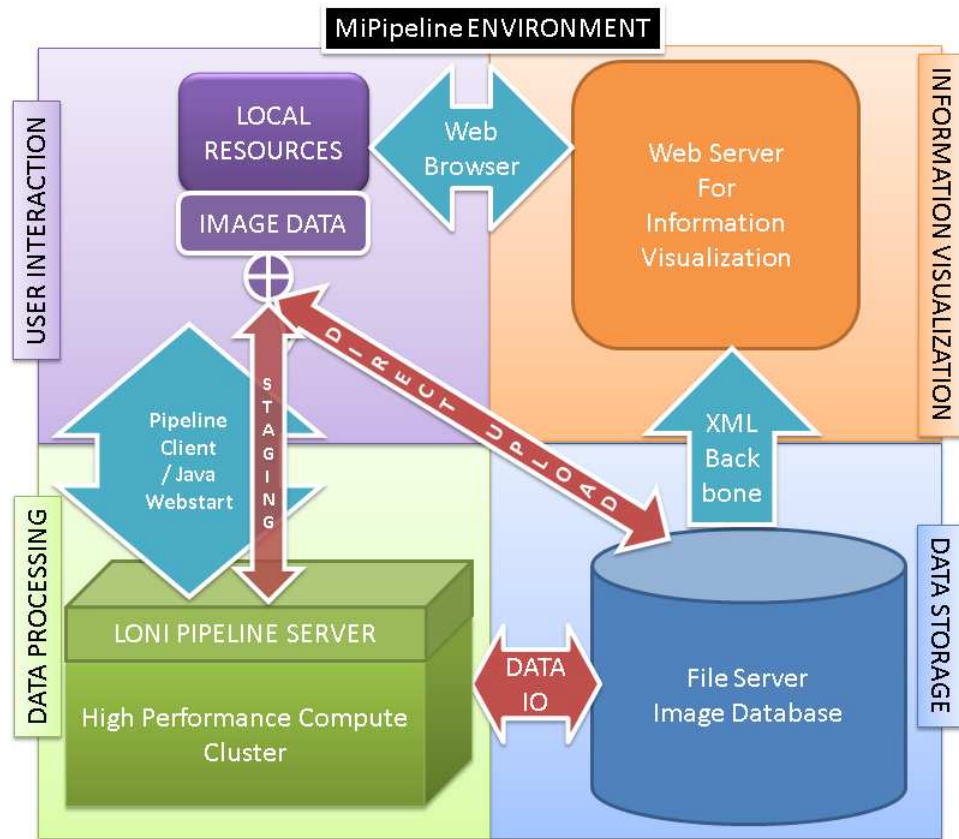


Figure 5.1: MiPipeline infrastructure showing various components of the MiPipeline environment

is an environment for visual workflow creation, provides easy integration with ImageJ [158], Micromanager [159] and Java/Eclipse based application programming interface (API) for creating custom plugins. However, ICY lacks support for data provenance and since ICY does not follow a client server model, it lacks capabilities for cluster computation.

From a workflow management standpoint, several additional opensource and commercial platforms called workflow management systems (WMS) exist, e.g., KNIME [160], TAVERNA [161], KEPLER [162], TRIANA [163], LONIPipeline environment [150], PSOM [164], Galaxy [165] [166] [167] and Pipeline Pilot (Accelrys,

Inc., San Diego, CA, USA). These platforms provide access to algorithms from widearray of disciplines and can be executed on existing HPCs. Most of them also provide visual data processing and reporting. TAVERNA, KEPLER and TRIANA are versatile general purpose open source WMS which supports desired facilities such as grid computing, visual workflow management, workflow sharing, web based interaction and process provenance. However, support for integration of system executables and bio-image informatics is limited. Pipeline Pilot on the other hand is a commercial product that also supports a comprehensive set of desired facilities.

Among these Galaxy, PSOM and KNIME are most similar to the work that we report here. Galaxy is a web based WMS for a wide variety of bioinformatics applications which supports grid computing and visual programming. PSOM provides script based pipeline creation and management tools, parallel execution capabilities and data provenance. KNIME on the other hand provides a visual programming interface for workflow creation and management, data provenance, a JAVA based API for custom modules and provision for integration of popular scripting languages (R, Python and Matlab). However, it has minimal control when incorporating system executables into workflows. Furthermore, KNIMEs server side products (e.g., centralized workflow management and cluster compute engine) are part of their commercial offering.

5.3 MiPipeline Environment

Figure 5.1 illustrates the four components of the MiPipeline environment along with their interactions, namely, *USER INTERACTION*, *DATA PROCESSING*,

DATA STORAGE and INFORMATION VISUALIZATION.

The first component, *USER INTERACTION* includes organizing and transferring images of biological samples and associated data for processing, software development activities if needed for custom applications, setting up the processing pipeline and manual data mining activities with the interactive information visualization component: *INFORMATION VISUALIZATION* once the analysis is completed.

The *DATA PROCESSING* component is based on LONI-DPS and provides a user friendly interface (via LONI Pipeline Client) to high performance computing resources. Once the user sets up their processing pipeline, this component gets the input data either directly from the user ('STAGING', Fig. 5.1) or from a central file server (*DATA STORAGE*). Once the data processing is completed, the results are streamed back to the user ('STAGING') or saved on the central storage.

The third component is *DATA STORAGE*, which is a central data server accessible from the user local resources and all compute nodes of the *DATA PROCESSING* component. A user can directly upload ('DIRECT UPLOAD' in Fig. 5.1) their raw data onto the *DATA STORAGE* which is subsequently processed by the *DATA PROCESSING* component. An user can also create an XML information backbone post data processing, which is subsequently used for information visualization and data provenance.

The *INFORMATION VISUALIZATION* component of the MiPipeline environment is web technology based and uses the XML backbone to visualize metadata and results for exploration and mining purposes. The interaction is completely web

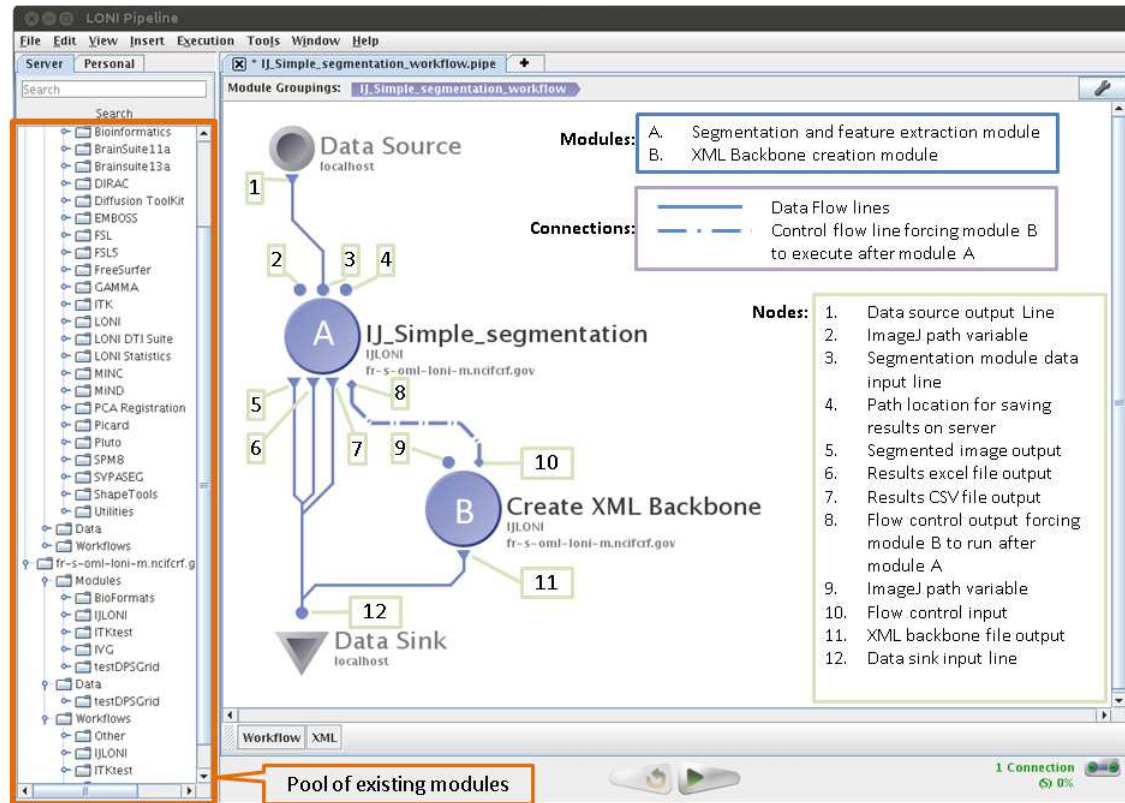


Figure 5.2: A simple ImageJ based workflow for performing nuclear segmentation and creating a backbone XML for visualization. The figure also explains the modules used for creating the workflow, the data connection lines and the module input output nodes. The red box on the left shows all pre-existing modules available on the LONI server which can be interconnected to create custom workflows.

enabled and is performed via a web browser.

5.3.1 LONI Pipeline

The LONI Pipeline tool is well suited for deployment of intricate and involved data analysis pipelines and warrant minimal maintenance and support overhead. Once a module/workflow is deployed on the main server, a user with minimal or no programming background can easily import their data and process it using a rich set of pre-existing workflows or their own custom made workflows created by

interconnecting the available server or local custom-made modules (see Fig. 5.2 for a simple example). Along with a large pool of central server based repositories, catering the needs of a wide user base, local repositories of custom-made modules, which cater to the requirements of focused research groups, can be maintained. Moreover, content management of modules and workflows become very centralized as they themselves can hold a wealth of associated information such as licensing, links to publications, tags and documentation.

The visual programming environment offered by LONI Pipeline client is also conducive for rapid application prototyping, deployment and sharing. The client can either be installed by downloading it from the LONI Pipeline website or can be started from the web using the webstart version of the application which requires no local installation. Once developed, generic as well as custom workflows can not only be easily accessed on the server via the LONI pipeline client, but can also be shared as '.pipe' files which are XML documents encoding the workflow details. Such an environment in turn fosters reproducible research in the form of server deployed standardized or sharable workflows that can be easily accessed and executed through a Pipeline client.

LONI Pipeline is a versatile platform where users can easily incorporate and interconnect softwares developed in disparate languages as modules as long as they can be invoked from the command line prompt on the server. For example, a single pipeline can incorporate modules developed using R, Perl, Java, Python, C, C++, Octave or Matlab (using Matlab Runtime Component) and more. Furthermore, the LONI pipeline, which itself runs on a Linux OS is inherently OS independent

and can be accessed through the pipeline client, which is available on all popular operating systems.

The Pipeline server provides a seamless integration of the modules to existing high performance computing infrastructure available locally or at the main server. The biggest advantage for a user is that they can simply use the visual programming environment to tap into the enormous resource of large compute grids without going into the intricate details of configuring, scheduling and using a parallel computing environment

The current deployment of MiPipeline *DATA PROCESSING* module is hosted on LONI Pipeline Server at **cranium.loni.usc.edu** and provides (guest) access to LONI DPS installed on top of Oracle/Sun Grid engine (SGE) The main high performance computation (HPC) cluster include a Linux (CentOS 6.4, 64bit) computer cluster with one head node and thousands of multicore compute nodes (<http://pipeline.loni.usc.edu/>). The HPC cluster has the capability to run application executables developed in a wide variety of programming languages.

5.3.2 MiPipeline File Server

The file server (*DATA STORAGE* in Fig. 5.1) is the main data hub for the MiPipeline environment. A user can upload their acquired images onto the server which can then be processed by LONI DPS. The analysis results can also be stored on the server. The file server can be a simple data server such as a NFS share, image databases such as OMERO, XNAT, or can be very generic cloud sources such as Dropbox or Amazon S3.

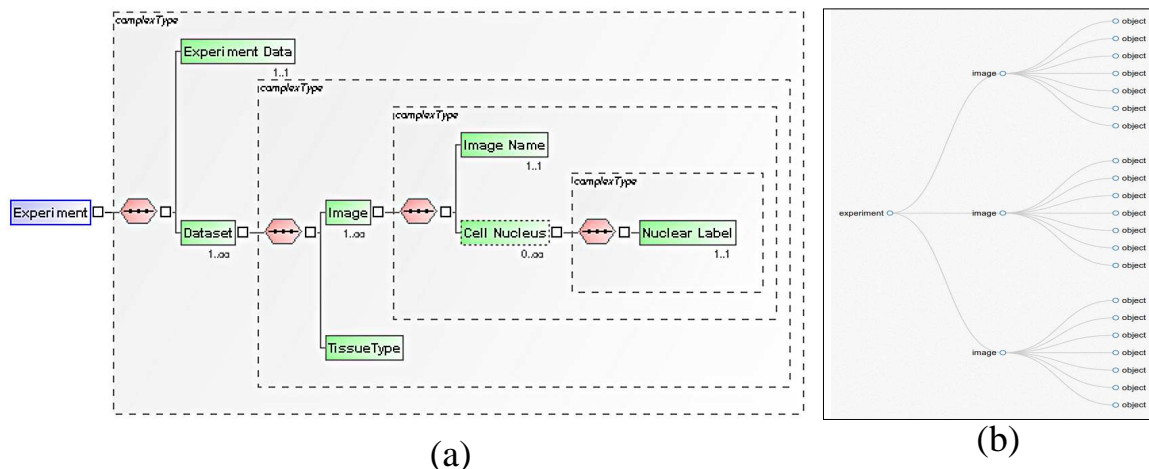


Figure 5.3: (a) XML schema used or MiPipeline data provenance. (b) Visualization of a simple XML file generated by the worked example using the generic XML visualization capabilities of MiPipeline.

5.3.3 MiPipeline Data Provenance

MiPipeline provides two layers of data provenance. The LONI Pipeline tool (*DATA PROCESSING*) itself provides the first layer of module and workflow provenance in the form of internal XML files.

In MiPipeline, we added a second layer of application specific data provenance in the form of an XML information backbone. Figure 5.3.(a) shows a simple XML schema illustrating the method of capturing often inherent hierarchical structure in biological datasets. The sample schema has four levels: (i) Experiment (an instance of an image database capturing images from a multi-patient biological experiment), (ii) Dataset (which might correspond to individual patients), (iii) Image (Multiple images captured from one or more patient samples) and (iv) Nucleus (nuclear objects identified by image analysis algorithms). The backbone needs to be created by the user depending on the needs of the application. Comprehensive application

data provenance, such as the one offered by MiPipeline, is essential for complex biological applications and enables extraction of complicated and hard to ascertain trends and results while ensuring reproducible research. The XML is populated with information such as meta-data about the biological sample, steps taken in the image processing and results. This information, can be seamlessly visualized, explored and mined using the MiPipeline information visualization component.

5.3.4 MiPipeline Information Visualization

In the case of large image datasets, it is a challenge for users to explore, navigate and extract connections, trends, and distributions from the meta data and experimental output. Thus we have developed a custom web-browser based flexible yet powerful dynamic information visualization framework. It was built using open source JavaScript visualization techniques (*d3.js* [168]) and noSQL database (MongoDB [169]). The frontend of this framework provides a web application for efficient data interaction and navigation, which can be accessed using any modern web browser. To facilitate metadata storage and search capability within the web interface, the XML information backbone is parsed into the noSQL database (MongoDB). MongoDB is flexible, thus allowing us to quickly adopt the dynamic visualization framework for different types of data sets . MongoDB also has a powerful and flexible search capability.

Data visualization is done with minimal overhead development by integrating multiple JavaScript libraries such as *d3.js*, jQuery [170] and Bootstrap [171]. Initial data preparation and processing (preceding the database injection) for visualization

and analysis tasks is done on the server side by a mixture of C/C++ binaries and PHP scripts.

In its most generic version, this system can render any XML file as an interactive hierarchical tree structure, in lines with the sample XML schema displayed in Figure 5.3.(a), using *d3.js* functions. To draw the tree structure, we pull metadata from MongoDB database and create the tree structure in JavaScript Object Notation (JSON) format, which is required by *d3.js* for rendering. This dynamic tree structure generation method also allows us to generate subtree structure created from search results for rendering (Fig. 5.3.(b)). In the MiPipeline information visualization framework we added additional visual exploration and mining capabilities of image display, data filtering, feature visualization and searching. These capabilities are incorporated based on the nature of information to be visualized for custom applications.

5.3.5 Microscopy Tool Integration

We integrated some popular and powerful microscopy tools into the MiPipeline environment: LOCI Bio-Formats [152] library and ImageJ/Fiji [158]. LOCI Bio-Formats library enables reading of metadata as well as image data of more than one hundred file types including proprietary microscopy image formats. Whereas ImageJ/Fiji provides a wide array of microscopy image analysis tools that can furthermore be used in a parallel computing infrastructure. Advanced image analysis workflows can be created by interconnecting ImageJ tools where each one is an ImageJ macro. The only restriction is that the macro must be launched from the

Table 5.1: Description of the worked example modules.

Modules	User Input	Description
Data Source	Input files on user’s local machine	This pre-existing Pipeline module acts as a source for input data residing on the local computer and transfers (stages) the input data on to the server before processing.
IJ_Simple_segmentation	(i) ImageJ path variable locating the path of ImageJ jar file if not already populated (ii) Location for saving the output files on the server	This custom made ImageJ macro based module reads input images, splits the channels for RGB images, selects the B channel, performs Gaussian filtering on B, thresholds the channel, runs watershed based segmentation, measures several segmented object features and saves the segmented image and the feature measurements in a comma separated value (CSV) ASCII files. The ImageJ macro is called from a BASH script file which is executed from the command line by the server.
Create XML Backbone	Location where output files are located on the server	This custom made BASH script based module reads the output CSV files and creates an XML file, used for visualizing, filtering and searching the results from the segmentation. The XML file has three tags, <i>experiment</i> at the top, <i>image</i> with an image identification attribute <i>id</i> and <i>object</i> assigning labels to each individual segmented object.
Data Sink	Local folder on user’s machine where final results are staged back.	This pre-existing Pipeline module acts as a sink for output results and transfers (stages) them from the server to the local machine after processing is done.

command line prompt and therefore must be able to execute as a headless, batch mode operation. The integration of such Java based applications is seamless and can be easily accomplished by making the appropriate jar files available on the Pipeline server.

5.3.6 Worked Example and User Interaction

To illustrate the user interactions necessary to run applications and visualize results in MiPipeline, in this section, we provide a small worked example, which segments cell nuclei in 2D images. Figure 5.2 shows the workflow in the LONI Pipeline client interface which has four modules: *Data Source*, *IJ_Simple_segmentation*, *Create XML Backbone* and *Data Sink*. The modules were dragged and dropped from the left hand side panel, which contains the prepopulated pool of existing modules available on the Pipeline server. Table 1 describes the individual modules.

Once the modules are dragged onto the workflow area, they are interconnected as shown in Fig. 5.2. When the workflow is executed, first the input

data is staged from the local computer to a temporary folder on the server and the *IJ_Simple_segmentation* module starts a distributed execution depending on the number of input images and available cores on the server. On execution completion of the segmentation module, the XML creation module reads the output comma separated values (CSV) files from a pre-specified server location and creates the XML information backbone file to enable visualization, filtering and searching of the results. Finally, all the output files including the XML is staged back to a folder on the users local machine by the *Data Sink*.

The visualization system is web based and the XML file is first transferred to a pre-specified folder on the web server, which is intermittently polled to check for new XML files to be visualized. The web interface lists the available XML files and on selecting the XML file specific to the example workflow produces a dynamic visualization of the xml file (Fig. 5.3(b)). In this particular example, there were 3 images and the tree had three hierarchies: the experiment at the top, the sample images and the segmented objects. Application specific visualization components can also be used in addition to the interactive tree visualization and all such advanced components will be discussed as part of a case study later in the paper.

5.4 Case Studies

In this section we discuss two case studies to demonstrate the strength and capabilities of the MiPipeline environment.

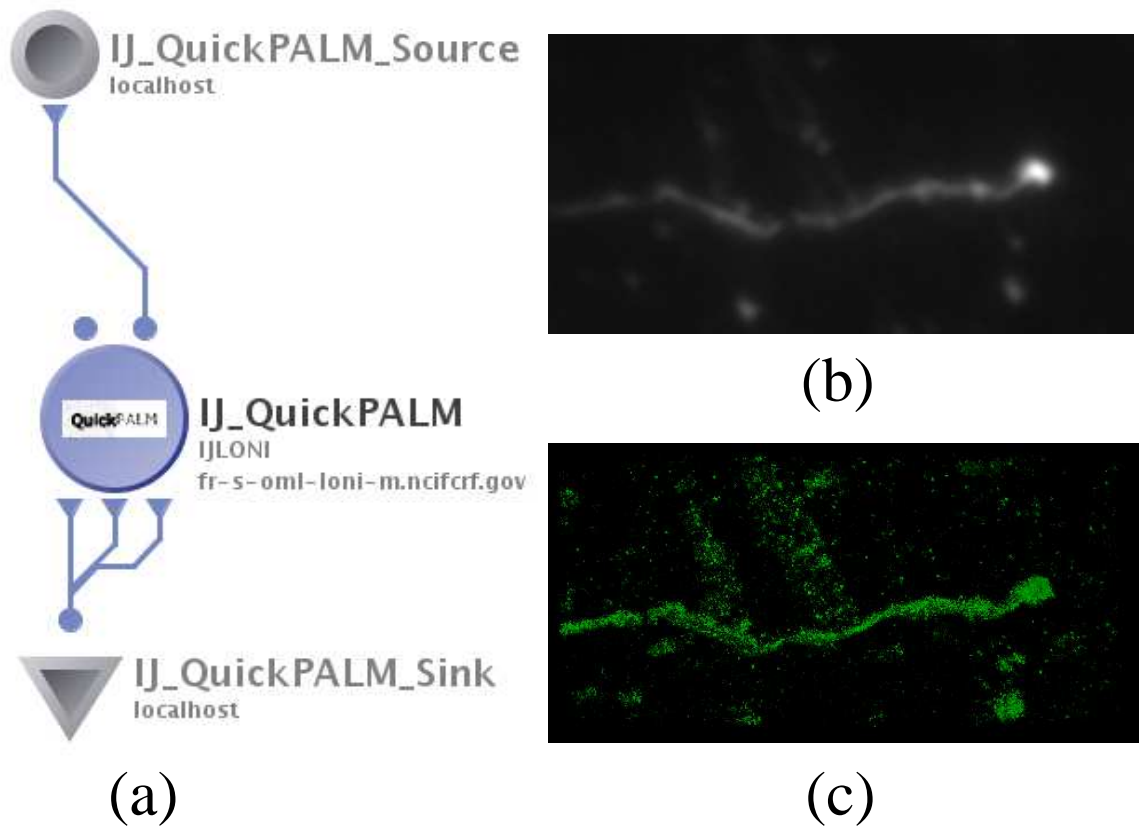


Figure 5.4: (a) ImageJ plugin based QuickPALM workflow using module *IJ_QuickPALM* (b) Average image from a sequence of PALM images of retinal pigment epithelium cells with tubules (c) Super-resolution PALM reconstruction from the sequence of images of retinal pigment epithelium cells with tubules

5.4.1 QuickPALM Case Study

The first case study illustrates the integration of an ImageJ plugin, QuickPALM [172] which is an algorithm for reconstruction and analysis of super resolution photoactivated localization microscopy (PALM) [173] and stochastic optical resolution microscopy (STORM) [174] based 2D and 3D datasets. Both the methods utilize fluorescence molecules that can be activated to fluoresce, thus enabling only a small fraction of molecules can be made fluorescent at a given time. In this situation each fluorescence molecule produces a point spread function from which the exact

position of these molecules can be localized to 10s of nanometers of precision. By sequentially imaging different sparse subsets of molecules, a complete picture of the molecules in the sample can be constructed. QuickPALM as an ImageJ plugin provides algorithms for real time reconstruction of 2D images along with methods to do 3D reconstruction, drift correction and acquisition. As the reconstruction process for such datasets is computationally intensive, adequate use of high performance computing machinery is essential for large datasets.

QuickPALM integration with MiPipeline was performed as part of ImageJ. Figure 5.4(a) shows the MiPipeline QuickPALM module *IJ_QuickPALM* in a simple workflow, which is in the form of an ImageJ macro. MiPipeline QuickPALM is called using a BASH script, which handles the custom input and output operations. All the analysis and reconstruction parameters of QuickPALM can be integrated into the module. One such parameter, the display update rate has been currently integrated to showcase the methodology.

To illustrate *IJ_QuickPALM*, fixed RPE (Retinal Pigment Epithelium) cells with tubules labeled by transfection with EHD1- mEos plasmid was imaged with an Olympus TIRF 3 based PALM system that included an IX-82 inverted microscope, and a 100x, 1.49 NA TIRFM oil objective lens. Before PALM imaging, the green fluorescence of mEos was excited with a 488 nm laser (SAPH, Coherent, Santa Clara, CA) and emission was collected with a quad-band emission filter (LF405/488/561/635-A, Semrock, Rochester, New York). The green image was used to locate the cell for PALM imaging. During PALM imaging. A 405-nm laser (CUBE, Coherent, Santa Clara, CA) set to a low intensity converted mEos from

green to red emission. The red fluorescence of mEos was continuously excited with a 561 nm DPSS laser (85-YCA-020-115, Melles Griot, Carlsbad, CA) with an excitation intensity of 1 kW/cm^2 . Emission was collected continuously at an acquisition time of 40 msec with the same quad-band emission filter and an additional 561 longpass to remove background. Images were acquired with an EMCCD camera (C9100-13, Hamamatsu, Middlesex, NJ) and a total of 8000 frames were collected for each PALM image. The average image of the 8000 frames is shown in Fig. 5.4(b) and the super-resolution image resulting from MiPipeline QuickPALM is shown in Fig. 5.4(c).

5.4.2 Parallel Genome Organization Diagnosis Application (PAGODA) Case Study

The second MiPipeline case study deals with analysis of spatial positioning of genes within cell nuclei as a potential method to detect breast cancer. The software builds on recent studies showing that spatial organization of candidate genes, for instance *HES5* and *FRA2*, differ between normal and breast cancer cells both in cell culture models [10] and in human biopsied tissue sections [11]. This technology could potentially emerge as a diagnostic and/or prognostic tool for breast cancer.

5.4.2.1 Tissue Samples and Image Acquisition

The experimental data used for testing PAGODA was from a tissue micro array (TMA) slide which comprised of 43 human breast biopsy samples (40 breast cancer and 3 non cancerous breast disease samples). *HES5* and *FRA2* DNA sequences were labeled by fluorescence in situ hybridization (FISH), imaged and pseudocolored as

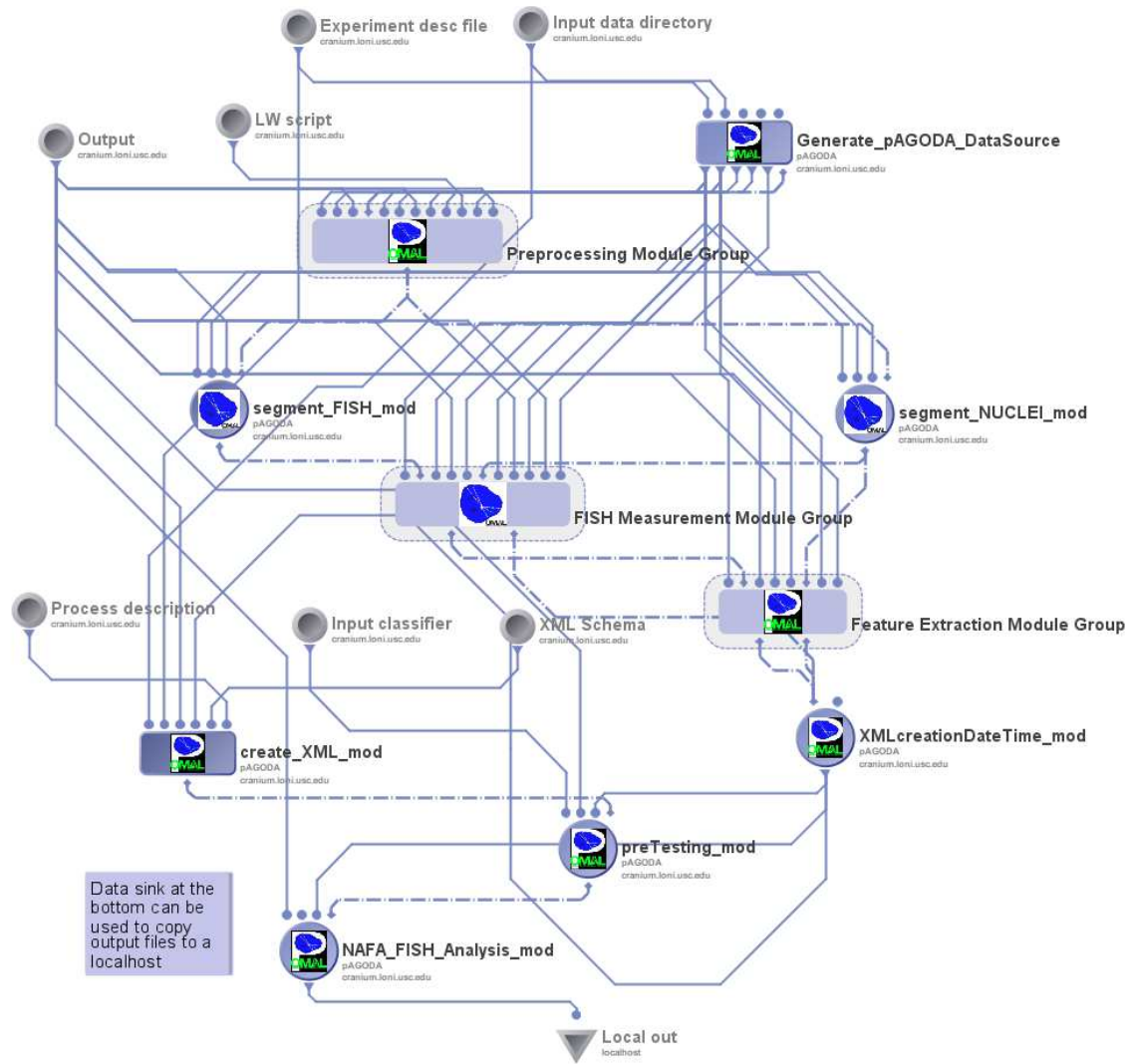


Figure 5.5: Figure showing MiPipeline PAGODA workflow

red and green color channels, respectively [11]. The 4',6 diamidino2phenylindole counter stained cell nuclei were imaged and assigned to the blue channel. On average, each patient sample was imaged in 3D at 40 different locations by visually selecting regions-of-interest (ROI) on a Deltavision microscope (equipped with deconvolution software). The input to PAGODA was approximately 5.1 GB (maximum intensity projected 1703 2D RGB images of size 1024x1024 pixels with 3 8-bit channels). A small portion of the TMA data is available on cranium.loni.usc.edu

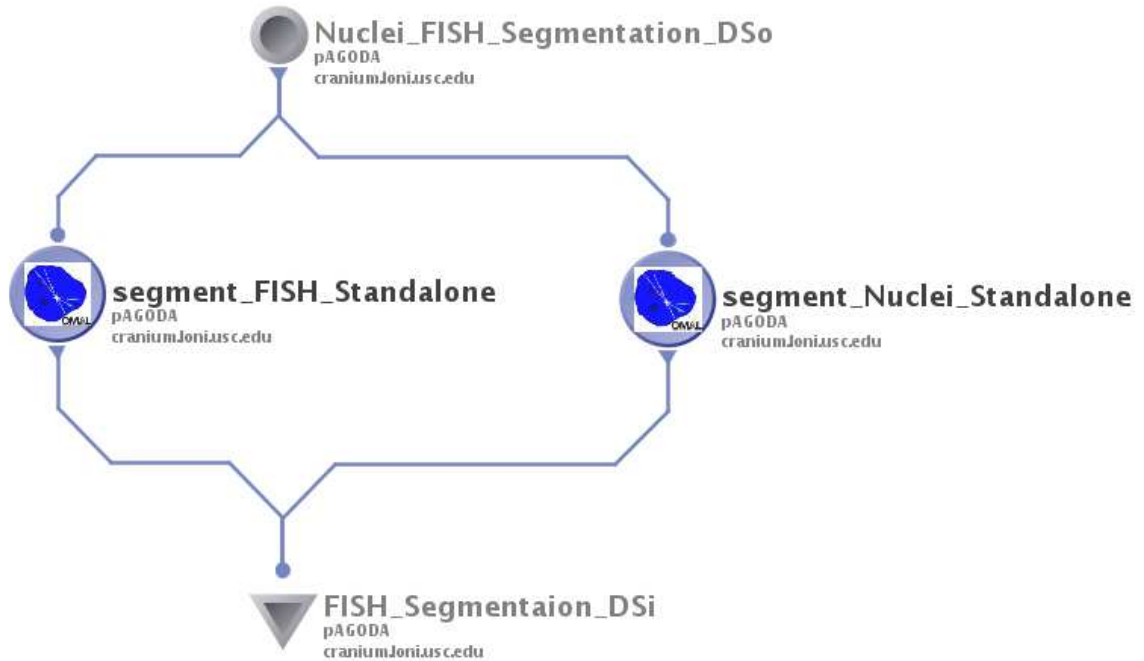


Figure 5.6: PAGODA standalone modules for FISH segmentation and nuclear segmentation

(TMA_DEMO) for testing and the current PAGODA workflow is configured to demonstrate its capabilities on this small dataset.

5.4.2.2 Image Analysis and Workflow

The analysis of tissue section images involved the following steps [113]:

- Background removal
- Wavelet-based edge enhancement of the nuclear channel
- Automatic segmentation of cell nuclei
- Selection of accurately-segmented nuclei using a logistic regression [63] based pattern recognition engine
- Detection of fluorescence in situ hybridization (FISH) labeled gene sequences

- Measurement of the gene locations (FISH signals) with respect to the nuclear periphery. For a statistically significant result, 100–200 nuclei had to be analyzed per patient tissue sample [11].

Fig. 5.5 shows the PAGODA workflow. Core PAGODA modules were implemented using MATLAB 2010b [175] making use of functions from commercial MATLAB toolboxes: Image Processing and Statistics toolboxes. In addition, Lastwave [35], DipImage [38], PRTOOLS [176] and XML IO Tools [177] were also used for performing some specialized image processing and XML related operations. The modules were converted to executables (using MATLAB compiler and requires MATLAB Component Runtime for execution) with appropriate wrappers (BASH scripts) for accommodating custom input output requirements. The aforementioned image analysis steps were offered as MiPipeline PAGODA modules. Two modules, one for FISH signal segmentation and another for nuclear segmentation are also available as modules (Fig. 5.6) that can be used in custom workflows.. The details of the algorithms were reported in [113] [63].

An XML backbone was used for data provenance which captured a comprehensive set of meta-data and experimental results.

5.4.2.3 PAGODA Results

The validation and accuracy of the segmentation module and machine learning modules of PAGODA have been reported previously [113] [63]. Using MiPipeline PAGODA, the TMA slide data with 38 patient samples was compared to a manually pooled benchmark dataset using Kolmogorov-Smirnov (KS) test with a 1% level of

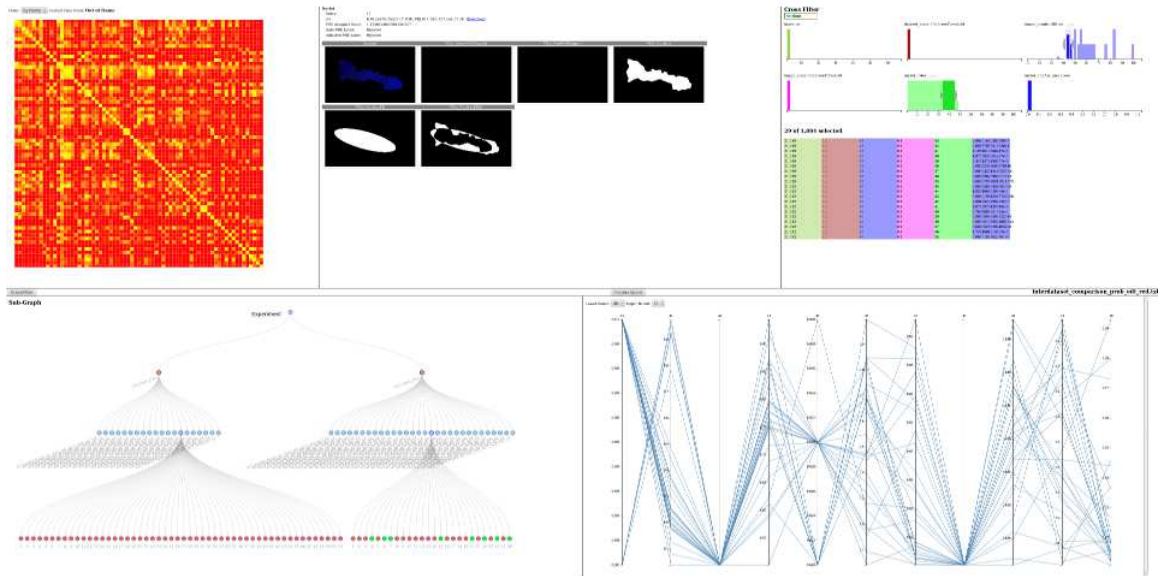


Figure 5.8: PAGODA heat map display of dataset similarity measure along with tools for feature space visualization and filtering

results (input images, segmentation boundaries, feature vectors, plots/graphs, cross comparison tables, etc.) on the network file server.

5.4.2.4 Advanced Visual Exploration and Mining

Processing of the TMA data using the PAGODA visual workflow produced over 700k files which included intermediate images from various image processing modules, plots, tables, and numerical data files. Understandably, it is a challenge for users to explore, navigate and extract connections, trends, and distributions from the experimental output. Utilizing the XML metadata backbone, we have developed a custom webbrowser based flexible yet powerful dynamic information visualization framework. The basic visualization of the XML was the hierarchical tree structure as illustrated in Figure 5.7. There are four different levels in TMA data, namely experiment, dataset, image and nuclei. When user places the mouse cursor over a

tree node, the event is captured and node properties and details get displayed to the user. Color encoding is implemented on nodes to help users understand the data set. For example, at the dataset level, nodes are color-coded based on cancerous or noncancerous tissue type.

We have also implemented search capabilities for users to narrow down examination scope to target data points of interest. Search functions could be performed at specific tree levels or in the entire tree structure. Upon opening a new data set, we extract node properties and list them as tags such that user could search based on a specific node property. Fig. 5.7 shows the search user interface built in our visualization framework. Currently, four predetermined parameters are provided for search functionality: *Class Name*, *Tag Name*, *Condition*, and *Keyword*. *Class Name* specifies the hierarchical level in the tree at which the search is performed. The *Tag Name* is based on the properties of the specific *Class Name*. The *Condition* is the type of search criteria used. The *Keyword* is what is going to be searched. There are three types of search capabilities: Global Search, Search by Class Name, and Search by Tag Name. Global Search searches the entire data set with given *Condition* and *Keyword*. Search by Class Name will limit scope of search to the selected level in the tree structure. Search by Tag Name will further limit scope of search on the selected node property in addition to the selected Class Name. All search capabilities are based on the presumption that a proper Keyword and Condition have been provided by user. The possible choice of Condition are: =, >, <, >=, <=, Not Equal, and Regular Expression. The result of executing a search is a new graph of relevant results with proper child and parent relationship of any relevant node.

Figure 5.8 shows the data browsing and analysis capability implemented in our visualization framework. The interactive heat map matrix displays cross correlation values between any of the 71 datasets in the experiment. Numerical value of each cell can be revealed by simply placing mouse cursor over each cell. Users can click each individual matrix cell in the heat map chart to further examine the pair of datasets associated with the selected cell which are displayed as a tree structure. Users can check properties and detailed information for each tree node by simply placing mouse cursor over these nodes. For the TMA experiment dataset, each subtree has thousands of nuclei. Each nucleus has several properties such as analysis score, index, and other numerical values. We have used CrossFilter [178] to manipulate this set of multivariate data. CrossFilter is a JavaScript library featuring fast interaction on large multivariate datasets. Lastly, we used *d3.js* for rendering parallel coordinates from the filtered subset of nuclei for feature space visualization.

5.5 Discussion and Conclusions

The current chapter reports the development of a visual programming based high throughput analysis and information visualization environment called MiPipeline for handling large microscopy based image datasets. The LONI pipeline distributed processing server which interacts with general purpose cluster management systems for job handling, is used as the basis for the high throughput data processing. On the other hand, the LONI pipeline client provides a rich visual programming environment with a pre-existing pool of algorithms available as modules which can be inter-connected to create custom processing workflows. MiPipeline

also has provisions for a comprehensive process and data provenance via the use of XML backbones which are further utilized for information visualization. The information visualization module of MiPipeline is webbrowser based and utilizes javascript based libraries such as *d3.js* to visualize meta data and experimental results. Several other tools for feature visualization, intermediate results display, searching and data exploration have been integrated with MiPipeline information visualization framework. A number of tools which are very useful in handling microscopy based datasets such as LOCI- Bioformats and ImageJ have been ported to MiPipeline environment and future work will concentrate on broadening the array of modules that are available on MiPipeline. On the other hand, several improvements are planned for the information visualization framework to make it more generic and intuitive by incorporating several other information visualization tools which would help in understanding and interpreting underlying trends of the data and results. On the whole MiPipeline is a software environment targeted towards processing and visualizing information from microscopic datasets which attempts to provide a number of attributes such as access to multi-disciplinary algorithms, provision for process and data provenance, visual programming environment, cluster based high-throughput processing and seamless information visualization that are desirable for modern day research involving large microscopic datasets.

Chapter 6: Conclusions and Future Directions

In this dissertation we presented methods for analyzing and exploring images and information extracted from biological samples imaged using fluorescence optical microscopes. Recent advances in optical microscopy have led to a data and information explosion warranting the development of reliable, accurate and automatic image analysis techniques for the extraction of biologically relevant information that can be used for applications such as disease prognosis, diagnosis and drug discovery. Furthermore, the presentation of the results to the end-user, post analysis, is also an equally challenging issue, especially when the data (and/or the hypothesis) involves several spatial/hierarchical scales (e.g., tissues, cells, (sub)-nuclear components).

6.1 Summary

Chapter 1 provided a brief introduction to optical microscopy and the problem of analyzing optical microscopy-based image datasets for **accurate understanding of individual cell coordination in tissue context**. It also outlines the novel contributions of the dissertation.

Chapter 2 discussed a prospective method for human breast cancer prognosis and diagnosis based on spatial localization of specific genes within the cell nuclei. In order to automate the method and develop a computer-based system to mimic

manual analysis procedure, two 2D nuclear segmentation methods were developed based on levelsets and the watershed algorithm along with a supervised pattern classification based screening procedure to identify 'well-segmented' nuclei. A comparison of manual and automatic analysis results for human breast tissue sample images showed successful automation of the cancer detection procedure.

As a natural extension to 3D and to overcome the shortcomings of the 2D nuclear segmentation methods presented in chapter 2, we developed robust and accurate methods for probabilistic edge detection and nuclear segmentation for 3D tissue datasets analyzing which is considerably more challenging compared to its 2D cell culture counterpart. The multi-scale and multi-orientation edge detection method reported in chapter 3 outperformed several conventional edge detection methods and results from experiments involving both synthetic and real microscopic datasets demonstrated the robustness and accuracy of the method across a wide range of noise, blurring and sample variations. Unlike the boundary enhancement procedure presented in chapter 2, which enhanced the unwanted internal structures of the cell nuclei, the probabilistic edge detector specifically targets object edges which are relevant for the application.

Chapter 4 presented the design of a graphcut-based optimal segmentation method for cell nuclei in 3D datasets. The method proved to be robust and accurate across variations in noise and blurring in synthetic datasets. Experiments using a wide variety of real datasets also demonstrated superior results compared to other methods for 3D segmentation of cell nuclei. We also presented a 2D version of the algorithm which is a straight forward simplification of the 3D case.

Finally, chapter 5 outlined the development of *MiPipeline*, an environment for high-throughput processing and seamless information visualization targeted towards handling large microscopy datasets. The environment provides several other essential benefits in the form of comprehensive data and process provenance, an user friendly visual programming environment, platform independence and reproducibility. Several tools such as *BioFormats* and *ImageJ* which are highly relevant for analyzing microscopic datasets have been ported to the environment for easy user access. An XML backbone used for data provenance provides the basis for the information visualization framework which has several exploratory and mining functionalities.

6.2 Future Directions of Work

Although the dissertation outlines a variety of tools and methods for analyzing optical microscopy based datasets, considerable amount of future work is required to address the problem of **accurate understanding of individual cell coordination in tissue context**. We outline several such prospective areas of work:

- **Super Resolution Datasets:** Development of super resolution optical microscopy techniques has enabled scientists to go beyond the diffraction limit of light and image nano scale objects. With higher resolution comes large amounts of information. There are several associated challenges such as data handling since super resolution datasets are inherently large, faithful and efficient data reconstruction and development of appropriate image analysis algorithms that can analyze such datasets while taking full advantage of the

enhanced resolution.

- **High Performance Computing:** Although we presented the *MiPipeline* environment for high throughput processing and information visualization targeted towards microscopy datasets, further enhancements in this area are warranted. It will be a challenging and interesting task to incorporate GPU-based computing for improving the computational performance of algorithms such as reconstruction of super resolution datasets, edge detection and segmentation, especially in 3D. Also it is essential to port several of the existing algorithms to a multicore or multigpu computing environments so that legacy applications become more efficient.
- **Multidimensional Data Visualization and Annotation:** Multidimensional data visualization and annotation come in as a challenging problem. Manual annotation and screening plays a big role in the development of accurate and robust segmentation and screening algorithms in the form of an essential quality control step. A number of commercial and open source tools are available for performing such tasks. However, their flexibility and accuracy in providing desired level of multidimensional visualization and especially annotation capabilities leave significant scope for improvements.
- **Tracking in Multidimensional Data:** Time lapse microscopic imaging is a very powerful tool which has the ability to visualize and capture cellular and sub-cellular dynamics in time and space. However, the addition of time as an additional dimension significantly increases the complexity of the analysis.

The most fundamental problem with time lapse datasets is that of tracking which can provide us a wealth of information about cell proliferation, differentiation and migration, characterization of which is extremely important in developmental biology. Although several methods [179] have been reported for tracking cellular and sub-cellular objects of interest, further research and development of robust and accurate algorithms is essential. As a future work, we would like to develop robust, efficient and accurate tracking algorithms for 2D and 3D time lapse microscopic datasets.

Chapter A: Appendix

A.1 Nuclear Feature Set used for ANN PRE

The 64 dimensional feature set used to characterize segmented objects for classification with the ANN based PRE in chapter 2 is provided below.

1. CCBendingEnergy
2. Feret (3 dimensions)
3. GreyInertia (2 dimensions)
4. GreyMu (3 dimensions)
5. Inertia (2 dimensions)
6. Mass
7. Mean object intensity
8. Mu (3 dimensions)
9. P2A
10. Perimeter
11. PodczeckShapes (5 dimensions)

12. Size
13. StdDev
14. DimensionsEllipsoid - major axes
15. DimensionsEllipsoid - minor axes
16. DimensionsEllipsoid - eccentricity
17. DimensionsEllipsoid - minor/major ratio
18. MajorAxes (4 dimensions)
19. ConvexRatio - Object Area/Convex Hull Area
20. ConvexRatio - Convex Hull Area - Object Area
21. RadiusStats - entropy of all radii
22. RadiusStats - range of all radii
23. RadiusStats - variance of all radii
24. GradientStats - magnitude sum
25. GradientStats - mean
26. GradientStats - range
27. GradientStats - variance
28. Autocorrelation

29. Contrast
30. Correlation (2 dimensions)
31. Cluster Prominence
32. Cluster Shade
33. Dissimilarity
34. Energy
35. Entropy
36. Homogeneity (2 dimensions)
37. Maximum probability
38. Sum of squares: Variance
39. Sum average
40. Sum variance
41. Sum entropy
42. Difference variance
43. Difference entropy
44. Information measure of correlation1
45. Information measure of correlation2

46. Inverse difference (INV)

47. Inverse difference normalized (INN)

Bibliography

- [1] J. Rittscher. Characterization of biological processes through automated image analysis. *Ann. Rev. Biomed. Eng.*, 12:315–344, 2010;.
- [2] J. G. Lock and S. Strömblad. Systems microscopy: An emerging strategy for the life sciences. *Exp. Cell Res.*, 316(8):1438–1444, 2010;.
- [3] R. Montironi, M. Scarpelli, A. Lopez-Beltran, R. Mazzucchelli, D. Alberts, J. Ranger-Moore, H. G. Bartels, P. W. Hamilton, J. Einspahr, and P. H. Bartels. Chromatin phenotype karyometry can predict recurrence in papillary urothelial neoplasms of low malignant potential. *Cell. Oncol.*, 29(1):47–58, 2007;.
- [4] A. R. Cohen, F. L. Gomes, B. Roysam, and M. Cayouette. Computational prediction of neural progenitor cell fates. *Nat. Methods*, 7(3):213–218, 2010;.
- [5] American Cancer Society. Cancer facts & figures 2011. *Atlanta: American Cancer Society*, 2011.
- [6] J. M. Bishop. The molecular genetics of cancer. *Science*, 235(4786):305–311, 1987;.
- [7] S.M. Gasser. Positions of potential:nuclear organization and gene expression. *Cell*, 104(5):639–642, 2001;.
- [8] K. J. Meaburn and T. Misteli. Cell biology: Chromosome territories. *Nature*, 445(7126):379–381, 2007;.
- [9] L. A. Parada and T. Misteli. Chromosome positioning in the interphase nucleus. *Trends Cell Biol.*, 12(9):425–432, 2002;.
- [10] K. J. Meaburn and T. Misteli. Locus-specific and activity-independent gene repositioning during early tumorigenesis. *J. Cell Biol.*, 180(1):39–50, 2008;.
- [11] K. J. Meaburn, P. R. Gudla, S. Khan, S. J. Lockett, and T. Misteli. Disease-specific gene repositioning in breast cancer. *J Cell Biol.*, 187(6):801–812, 2009;.

- [12] V. R. Korde, H. Bartels, J. Barton, and J. Ranger-Moore. Automatic segmentation of cell nuclei in bladder and skin tissue for karyometric analysis. *Anal Quant Cytol Histol.*, 31(2):83–89, 2009;.
- [13] F. Li, X. Zhou, J. Zhu, J. Ma, X. Huang, and S.T.C. Wong. High content image analysis for human H4 neuroglioma cells exposed to CuO nanoparticles. *BMC Biotechnol.*, 7(66), 2007;.
- [14] D. P. McCullough, P. R. Gudla, B. S. Harris, J. A. Collins, K. J. Meaburn, M. Nakaya, T. P. Yamaguchi, T. Misteli, and S. J. Lockett. Segmentation of whole cells and cell nuclei from 3-D optical microscope images using dynamic programming. *IEEE Trans Med Imaging.*, 27(5):723–734, 2008;.
- [15] Y. Al-Kofahi, W. Lassoued, W. Lee, and B. Roysam. Improved automatic detection and segmentation of cell nuclei in histopathology images. *IEEE Trans Biomed Eng.*, 57(4):841–852, 2010;.
- [16] G. Li, T. Liu, J. Nie, L. Guo, J. Chen, J. Zhu, W. Xia, A. Mara, S. Holley, and S. T. Wong. Segmentation of touching cell nuclei using gradient flow tracking. *J Microsc.*, 231(Pt 1):47–58, 2008;.
- [17] C. O. De Solorzano, R. Malladi, S. A. Lelièvre, and S. J. Lockett. Segmentation of nuclei and cells using membrane related protein markers. *J Microsc.*, 201(Pt 1):404–415, 2001;.
- [18] D. Baggett, M. Nakaya, M. McAuliffe, T. P. Yamaguchi, and S. Lockett. Whole cell segmentation in solid tissue sections. *Cytometry A*, 67(2):137–143, 2005;.
- [19] F. Raimondo, M. A. Gavrielides, G. Karayannopoulou, K. Lyroudia, I. Pitas, and I. Kostopoulos. Automated evaluation of Her-2/neu status in breast tissue from fluorescent in situ hybridization images. *IEEE Trans Image Process.*, 14(9):1288–1299, 2005;.
- [20] J. Cheng and J. C. Rajapakse. Segmentation of clustered nuclei with shape markers and marking function. *IEEE Trans Biomed Eng.*, 56(3):741–748, 2009;.
- [21] G. Lin, M. K. Chawla, K. Olson, C. A. Barnes, J. F. Guzowski, C. Bjornsson, W. Shain, and B. Roysam. A multi-model approach to simultaneous segmentation and classification of heterogeneous populations of cell nuclei in 3D confocal microscope images. *Cytometry A*, 71(9):724–736, 2007;.
- [22] A. Hafiane, F. Bunyak, and K. Palaniappan. Fuzzy clustering and active contours for histopathology image segmentation and nuclei detection. *Lect Notes Comput Sci.*, 5259:903–914, 2008;.

- [23] H. Fatakdawala, J. Xu, A. Basavanhally, G. Bhanot, S. Ganesan, M. Feldman, J. E. Tomaszewski, and A. Madabhushi. Expectation - maximization - driven geodesic active contour with overlap resolution (EMaGACOR): Application to lymphocyte segmentation on breast cancer histopathology. *IEEE Trans Biomed Eng.*, 57(7):1676–1689, 2010;.
- [24] C. C. Li and K. S. Fu. Machine-assisted pattern classification in medicine and biology. *Annu Rev Biophys Bioeng.*, 9:393–436, 1980;.
- [25] OL Mangasarian, R Setiono, and WH Wolberg. Pattern recognition via linear programming: Theory and application to medical diagnosis. In *Large-scale numerical optimization*, pages 22–30. SIAM Publications, Philadelphia, 1990.
- [26] W. H. Wolberg and O.L. Mangasarian. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the National Academy of Sciences, U.S.A.*, 87:9193–9196, 1990;.
- [27] A. Daub, P. Sharma, and S. Finkbeiner. High-content screening of primary neurons: ready for prime time. *Curr Opin Neurobiol.*, 19(5):537–543, 2009;.
- [28] J. Wang, X. Zhou, F. Li, P. L. Bradley, S. Chang, N. Perrimon, and S. T. C. Wong. An image score inference system for RNAi genome-wide screening based on fuzzy mixture regression modeling. *J Biomed Inform.*, 42(1):32–40, 2009;.
- [29] X. Chen and R. F. Murphy. Automated interpretation of protein subcellular location patterns. *Int Rev Cytol.*, 249:193–227, 2006;.
- [30] A. Chebira, Y. Barbotin, C. Jackson, T. Merryman, G. Srinivasa, R. F Murphy, and J. Kovačević. A multiresolution approach to automated classification of protein subcellular location images. *BMC Bioinformatics*, 8:210, 2007;.
- [31] R. Hiemann, T. Büttner, T. Krieger, D. Roggenbuck, U. Sack, and K. Conrad. Challenges of automated screening and differentiation of non-organ specific autoantibodies on HEp-2 cells. *Autoimmun Rev.*, 9(1):17–22, 2009;.
- [32] J. A. Helmuth, G. Paul, and I. F. Sbalzarini. Beyond co-localization: inferring spatial interactions between sub-cellular structures from microscopy images. *BMC Bioinformatics*, 11:372, 2010;.
- [33] A. A Hill, P. LaPan, Y. Li, and S. Haney. Impact of image segmentation on high-content screening data quality for SK-BR-3 cells. *BMC Bioinformatics*, 8:340, 2007;.
- [34] P. R. Gudla, K. Nandy, J. Collins, K. J. Meaburn, T. Misteli, and S. J. Lockett. A high-throughput system for segmenting nuclei using multiscale techniques. *Cytometry A*, 73(5):451–466, 2008;.

- [35] E. Bacry. Lastwave. <http://www.cmap.polytechnique.fr/~bacry/LastWave/>, 1998-2004.
- [36] L. Haglund. *Adaptive Multidimensional Filtering*. PhD thesis, Linköping University, Sweden, 1992.
- [37] J. Kittler, M. Hatef, R.P.W. Duin, and J. Matas. On combining classifiers. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(3):226–239, mar 1998.
- [38] DIPImage: A MATLAB toolbox for scientific image processing and analysis, release 1.6 (academic research version). [<http://www.diplib.org>], May 11, 2007.
- [39] Chunming Li, Chenyang Xu, Changfeng Gui, and M.D. Fox. Level set evolution without re-initialization: a new variational formulation. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 430 – 436 vol. 1, june 2005.
- [40] B. Rieger, C. Luengo, M. van Ginkel, G. van Kempen, and L. van Vliet. DIPImage Toolbox for MATLAB, 2009.
- [41] T.W. Ridler and S. Calvard. Picture thresholding using an iterative selection method. *IEEE Trans System, Man and Cybernetics*, 8(8):630–632, 1978.
- [42] F. Meyer. Topographic distance and watershed lines. *Signal Processing*, 38(1):113–125, 1994;.
- [43] P. Soille. *Morphological Image Analysis: Principles and Applications*. Springer-Verlag, 2 edition, 2002.
- [44] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. Wiley Interscience, 2001.
- [45] B.J.H. Verwer, P.W. Verbeek, and S.T. Dekker. An efficient uniform cost algorithm applied to distance transforms. *IEEE Trans Pattern Anal Mach Intell.*, 11(4):425–429, 1989;.
- [46] G. Lin, M. K. Chawla, K. Olson, J. F. Guzowski, C. A. Barnes, and B. Roysam. Hierarchical, model-based merging of multiple fragments for improved three dimensional segmentation of nuclei. *Cytometry A*, 63(1):20–33, 2005;.
- [47] F. Podczec. A shape factor to assess the shape of particles using image analysis. *Powder Technology*, 93(1):47–53, 1997;.
- [48] R. M. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, 1979;.
- [49] I. A. Basheer and M. Hajmeer. Artificial neural networks: fundamentals, computing, design, and application. *J Microbiol Methods.*, 43(1):3–31, 2000;.

- [50] S. Aksoy and R. Haralick. Feature normalization and likelihood-based similarity measures for image retrieval. *Pattern Recognition Letters*, 22(5):563–582, 2001;.
- [51] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3:1157–1182, 2003;.
- [52] D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963;.
- [53] M. J. D. Powell. Restart procedures for the conjugate gradient method. *Mathematical Programming*, 12(1):241–254, 1977;.
- [54] M. Riedmiller and H. Braun. A direct adaptive method for faster backpropagation learning: The RPROP algorithm. *Proceedings of the IEEE International Conference on Neural Networks, San Fransisco*, pages 586–591, 1993;.
- [55] Matlab2008a. Neural network toolbox, 2008. <http://www.mathworks.com>.
- [56] G. Srinivasa, M. C. Fickus, Y. Guo, A. D. Linstedt, and J. Kovačević. Active mask segmentation of fluorescence microscope images. *IEEE Trans Image Process.*, 18(8):1817–1829, 2009;.
- [57] M. Halter, D. R. Sisan, J. Chalfoun, B. L. Stottrup, A. Cardone, A. A. Dima, A. Tona, A. L. Plant, and John T. Elliott. Cell cycle dependent TN-C promoter activity determined by live cell imaging. *Cytometry A*, 79A(3):192–202, 2011;.
- [58] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971;.
- [59] L. R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945;.
- [60] S. Di Cataldo, E. Ficarra, A. Acquaviva, and E. Macii. Automated segmentation of tissue images for computerized IHC analysis. *Comput Methods Programs Biomed*, 100(1):1–15, 2010.
- [61] M. Baron. An overview of the notch signalling pathway. *Seminars in Cell and Developmental Biology*, 14(2):113–119, 2003;.
- [62] B. J. Vermolen, Y. Garini, I. T. Young, R. W. Dirks, and V. Raz. Segmentation and analysis of the three-dimensional redistribution of nuclear components in human mesenchymal stem cells. *Cytometry A*, 73(9):816–824, 2008;.
- [63] William Cukierski, Kaustav Nandy, Prabhakar Gudla, Karen Meaburn, Tom Misteli, David Foran, and Stephen Lockett. Ranked retrieval of segmented nuclei for objective assessment of cancer gene repositioning. *BMC Bioinformatics*, 13(1):232, 2012.

- [64] K. Nandy, P. R. Gudla, K. J. Meaburn, T. Misteli, and S. J. Lockett. Automatic nuclei segmentation and spatial fish analysis for cancer detection. *Conf. Proc. IEEE Eng. Med. Biol. Soc.*, pages 6718–6721, 2009;.
- [65] J. Kivinen, A. J. Smola, and R. C. Williamson. Online learning with kernels. *IEEE Trans Signal Processing*, 52(8):2165–2176, 2004;.
- [66] Y. Nakazato, Y. Minami, H. Kobayashi, K. Satomi, Y. Anami, K. Tsuta, R. Tanaka, M. Okada, T. Goya, and M. Noguchi. Nuclear grading of primary pulmonary adenocarcinomas: correlation between nuclear size and prognosis. *Cancer*, 116(8):2011–2019, 2010;.
- [67] J. A. Beliën, H. A. van Ginkel, P. Tekola, L. S. Ploeger, N. M. Poulin, J. P. Baak, and P. J. van Diest. Confocal DNA cytometry: a contour-based segmentation algorithm for automated three-dimensional image segmentation. *Cytometry A*, 49(1):12–21, 2002;.
- [68] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(5):898–916, May 2011.
- [69] J. M. S. Prewitt. Object enhancement and extraction. In B. Lipkin and Eds A. Rosenfeld, editors, *Picture Processing and Psychopictorics*, pages 75–149. New York: Academic Press, 1970.
- [70] L. G. Roberts. *Machine perception of three-dimensional solids*. MIT Press, Cambridge, MA, 1965.
- [71] Werner Frei and Chung-Ching Chen. Fast boundary detection: A generalization and a new algorithm. *Computers, IEEE Transactions on*, C-26(10):988–998, Oct 1977.
- [72] D. Marr and E. Hildreth. Theory of edge detection. *Proceedings of the Royal Society of London B: Biological Sciences*, 207(1167):187–217, 1980.
- [73] John Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8(6):679–698, Nov 1986.
- [74] J.C. Bezdek, R.A. Chandrasekhar, and Y. Attikouzel. A geometric approach to edge detection. *Fuzzy Systems, IEEE Transactions on*, 6(1):52–75, Feb 1998.
- [75] A. De Santis and C. Sinisgalli. A bayesian approach to edge detection in noisy images. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 46(6):686–699, Jun 1999.

- [76] S. Konishi, A.L. Yuille, J.M. Coughlan, and Song-Chun Zhu. Statistical edge detection: learning and evaluating edge cues. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(1):57–74, 2003.
- [77] Jinbo Wu, Zhouping Yin, and Youlun Xiong. The fast multilevel fuzzy edge detection of blurry images. *Signal Processing Letters, IEEE*, 14(5):344–347, May 2007.
- [78] Siwei Lu, Ziqing Wang, and Jun Shen. Neuro-fuzzy synergism to the intelligent system for edge detection and enhancement. *Pattern Recognition*, 36(10):2395 – 2409, 2003.
- [79] Sheng Zheng, Jian Liu, and Jin Wen Tian. A new efficient SVM-based edge detection method. *Pattern Recognition Letters*, 25(10):1143 – 1154, 2004.
- [80] Suchendra M. Bhandarkar, Yiqing Zhang, and Walter D. Potter. An edge detection technique using genetic algorithm-based optimization. *Pattern Recognition*, 27(9):1159 – 1180, 1994.
- [81] Andrew P. Witkin. Scale-space filtering. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence - Volume 2, IJCAI'83*, pages 1019–1022, San Francisco, CA, USA, 1983. Morgan Kaufmann Publishers Inc.
- [82] Fredrik Bergholm. Edge focusing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-9(6):726–741, Nov 1987.
- [83] V. Lacroix. The primary raster: a multiresolution image description. In *Pattern Recognition, 1990. Proceedings., 10th International Conference on*, volume i, pages 903–907 vol.1, Jun 1990.
- [84] Donna J. Williams and Mubarak Shah. Edge contours using multiple scales. In O. Faugeras, editor, *Computer Vision ECCV 90*, volume 427 of *Lecture Notes in Computer Science*, pages 66–70. Springer Berlin Heidelberg, 1990.
- [85] Ardeshir Goshtasby. On edge focusing. *Image and Vision Computing*, 12(4):247 – 256, 1994.
- [86] P. Perona and J. Malik. Detecting and localizing edges composed of steps, peaks and roofs. In *Computer Vision, 1990. Proceedings, Third International Conference on*, pages 52–57, Dec 1990.
- [87] M.C. Morrone and R.A. Owens. Feature detection from local energy. *Pattern Recognition Letters*, 6(5):303 – 313, 1987.
- [88] W.T. Freeman and E.H. Adelson. The design and use of steerable filters. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 13(9):891–906, Sep 1991.

- [89] T. Lindeberg. Edge detection and ridge detection with automatic scale selection. In *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR '96, 1996 IEEE Computer Society Conference on*, pages 465–470, Jun 1996.
- [90] P. Dollar, Zhuowen Tu, and S. Belongie. Supervised learning of edges and object boundaries. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1964–1971, 2006.
- [91] D.R. Martin, C.C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(5):530–549, May 2004.
- [92] Tobias Geback and Petros Koumoutsakos. Edge detection in microscopy images using curvelets. *BMC Bioinformatics*, 10(1):75, 2009.
- [93] Ashit Talukder and David P. Casasent. Multiscale gabor wavelet fusion for edge detection in microscopy images. volume 3391, pages 336–347, 1998.
- [94] D. Hearn and M. P. Baker. *Computer Graphics*. Prentice Hall, 1994.
- [95] Srinivasan Rajagopalan and Richard A. Robb. Image smoothing with savitzky-golai filters. volume 5029, pages 773–781, 2003.
- [96] J. Pawley(Editor). *Handbook of Biological Confocal Microscopy*. Springer, 3rd edition, 2006.
- [97] J Schindelin, I Arganda-Carreras, E Frise, V Kaynig, M Longair, T Pietzsch, S Preibisch, C Rueden, S Saalfeld, B Schmid, JY Tinevez, DJ White, V Hartenstein, K Eliceiri, P Tomancak, and A. Cardona. Fiji: an open-source platform for biological-image analysis. *Nature Methods*, 9:676–682, 2012.
- [98] Yoh-suke Mukoyama, Jennifer James, Joseph Nam, and Yutaka Uchida. Whole-mount confocal microscopy for vascular branching morphogenesis. In Xu Peng and Marc Antonyak, editors, *Cardiovascular Development*, volume 843 of *Methods in Molecular Biology*, pages 69–78. Humana Press, 2012.
- [99] P. Pantazis and W. Supatto. Advances in whole-embryo imaging: a quantitative transition is underway. *Nature Reviews Molecular Cell Biology*, 15:327–339, 2014.
- [100] Kenneth A. Giuliano. Actin-cytoskeleton and glial cell transformation: dissecting the morphological and molecular dynamics of tumorigenesis using light microscopy. *Proc. SPIE*, 2678:151–164, 1996.
- [101] Samantha Stehbens, Hayley Pemble, Lyndsay Murrow, and Torsten Wittmann. Chapter fifteen - imaging intracellular protein dynamics by spinning disk confocal microscopy. In P. Michael conn, editor, *Imaging and Spectroscopic Analysis of Living Cells Optical and Spectroscopic Techniques*, volume 504 of *Methods in Enzymology*, pages 293 – 313. Academic Press, 2012.

- [102] Joanna M. Kwiatek, Elizabeth Hinde, and Katharina Gaus. Microscopy approaches to investigate protein dynamics and lipid organization. *Molecular Membrane Biology*, 31(5):141–151, 2014.
- [103] V Starkuviene and R Pepperkok. The potential of high-content high-throughput microscopy in drug discovery. *British Journal of Pharmacology*, 152(1):62–71, 2007.
- [104] I.G.H. Lerchs. Optimum design of open-pit mines. *Trans. CIM*, 58:17–24, 1965.
- [105] B. Denby and D. Schofield. Genetic algorithms: A new approach to pit optimization. In *Proc. 24th Intl APCOM Symp.*, pages 126–133, 1993.
- [106] Kang Li, Xiaodong Wu, D.Z. Chen, and M. Sonka. Optimal surface segmentation in volumetric images - a graph-theoretic approach. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(1):119–134, Jan 2006.
- [107] Jean-Claude Picard. Maximal closure of a graph and applications to combinatorial problems. *Management Science*, 22(11):1268–1272, 1976.
- [108] Thomas H. Cormen, Clifford Stein, Ronald L. Rivest, and Charles E. Leiserson. *Introduction to Algorithms*. McGraw-Hill Higher Education, 2nd edition, 2001.
- [109] Xinghua Lou, Minjung Kang, Panagiotis Xenopoulos, Silvia Mu noz Descalzo, and Anna-Katerina Hadjantonakis. A rapid and efficient 2D/3D nuclear segmentation method for analysis of early mouse embryo and stem cell image data. *Stem Cell Reports*, 2(3):382 – 397, 2014.
- [110] P.K Sahoo, S Soltani, and A.K.C Wong. A survey of thresholding techniques. *Computer Vision, Graphics, and Image Processing*, 41(2):233 – 260, 1988.
- [111] C. WÄHLBY, I.-M. SINTORN, F. ERLANDSSON, G. BORGEFORS, and E. BENGTSSON. Combining intensity, edge and shape information for 2d and 3d segmentation of cell nuclei in tissue sections. *Journal of Microscopy*, 215(1):67–76, 2004.
- [112] L. Vincent and P. Soille. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 13(6):583–598, Jun 1991.
- [113] K Nandy, P.R. Gudla, R. Amundsen, K.J. Meaburn, T. Misteli, and S.J. Lockett. Automatic segmentation and supervised learning-based selection of nuclei in cancer tissue images. *Cytometry A.*, 81(9):743–754, 2012.
- [114] Cheng Chen, Wei Wang, John A. Ozolek, and Gustavo K. Rohde. A flexible and robust approach for segmenting cell nuclei from 2D microscopy images using supervised learning and template matching. *Cytometry Part A*, 83A(5):495–507, 2013.

- [115] D.P. Mukherjee, N. Ray, and S.T. Acton. Level set analysis for leukocyte detection and tracking. *Image Processing, IEEE Transactions on*, 13(4):562–572, April 2004.
- [116] T.F. Chan and L.A. Vese. Active contours without edges. *Image Processing, IEEE Transactions on*, 10(2):266–277, Feb 2001.
- [117] Dagmar Kainmueller, Florian Jug, Carsten Rother, and Gene Myers. Active graph matching for automatic joint segmentation and annotation of *C. elegans*. In Polina Golland, Nobuhiko Hata, Christian Barillot, Joachim Hornegger, and Robert Howe, editors, *Medical Image Computing and Computer-Assisted Intervention MICCAI 2014*, volume 8673 of *Lecture Notes in Computer Science*, pages 81–88. Springer International Publishing, 2014.
- [118] Gang Lin, Umesh Adiga, Kathy Olson, John F. Guzowski, Carol A. Barnes, and Badrinath Roysam. A hybrid 3D watershed algorithm incorporating gradient cues and object models for automatic segmentation of nuclei in confocal image stacks. *Cytometry A*, 56A:23–36, 2003.
- [119] R. Chinta and M. Wasser. Three-dimensional segmentation of nuclei and mitotic chromosomes for the study of cell divisions in live drosophila embryos. *Cytometry A*, 81(1):52–64, 2012.
- [120] A. Dufour, R. Thibeaux, E. Labruyere, N. Guillen, and J.-C. Olivo-Marin. 3-D active meshes: Fast discrete deformable models for cell tracking in 3-D time-lapse microscopy. *Image Processing, IEEE Transactions on*, 20(7):1925–1937, July 2011.
- [121] C. Zanella, M. Campana, B. Rizzi, C. Melani, G. Sanguinetti, P. Bourguine, K. Mikula, N. Peyrieras, and A. Sarti. Cells segmentation from 3-D confocal images of early zebrafish embryogenesis. *Image Processing, IEEE Transactions on*, 19(3):770–781, March 2010.
- [122] B. Appleton and H. Talbot. Globally minimal surfaces by continuous maximal flows. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(1):106–118, Jan 2006.
- [123] Jaza Gul-Mohammed, Ignacio Arganda-Carreras, Philippe Andrey, Vincent Galy, and Thomas Boudier. A generic classification-based method for segmentation of nuclei in 3D images of early embryos. *BMC Bioinformatics*, 15(1):9, 2014.
- [124] Gang Li, Tianming Liu, Ashley Tarokh, Jingxin Nie, Lei Guo, Andrew Mara, Scott Holley, and Stephen Wong. 3D cell nuclei segmentation based on gradient flow tracking. *BMC Cell Biology*, 8(1):40, 2007.
- [125] Lei Qu, Fuhui Long, Xiao Liu, Stuart Kim, Eugene Myers, and Hanchuan Peng. Simultaneous recognition and segmentation of cells: application in *c.elegans*. *Bioinformatics*, 27(20):2895–2902, 2011.

- [126] Cemal Cagatay Bilgin, Sun Kim, Elle Leung, Hang Chang, and Bahram Parvin. Integrated profiling of three dimensional cell culture models and 3D microscopy. *Bioinformatics*, 29(23):3087–3093, 2013.
- [127] Christian Widmer, Stephanie Heinrich, Philipp Drewe, Xinghua Lou, Shefali Umrania, and Gunnar Rätsch. Graph-regularized 3D shape reconstruction from highly anisotropic and noisy images. *Signal, Image and Video Processing*, 8(1):41–48, 2014.
- [128] Cheng-Jin Du, Marco Marcello, David G. Spiller, Michael R. H. White, and Till Bretschneider. Interactive segmentation of clustered cells via geodesic commute distance and constrained density weighted nystrom method. *Cytometry Part A*, 77A(12):1137–1147, 2010.
- [129] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *Int. J. Comput. Vision*, 59(2):167–181, September 2004.
- [130] Ying Xu and Edward C. Uberbacher. 2D image segmentation using minimum spanning trees. *Image and Vision Computing*, 15(1):47 – 57, 1997.
- [131] Eric N. Mortensen and William A. Barrett. Interactive segmentation with intelligent scissors. *Graphical Models and Image Processing*, 60(5):349 – 384, 1998.
- [132] A.X. Falcao, J.K. Udupa, and F.K. Miyazawa. An ultra-fast user-steered image segmentation paradigm: live wire on the fly. *Medical Imaging, IEEE Transactions on*, 19(1):55–62, Jan 2000.
- [133] Y.Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 105–112 vol.1, 2001.
- [134] Song Wang and J.M. Siskind. Image segmentation with ratio cut. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(6):675–690, June 2003.
- [135] Y. Boykov and V. Kolmogorov. Computing geodesics and minimal surfaces via graph cuts. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 26–33 vol.1, Oct 2003.
- [136] I.H. Jermyn and H. Ishikawa. Globally optimal regions and boundaries as minimum ratio weight cycles. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(10):1075–1088, Oct 2001.
- [137] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, August 2000.

- [138] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(9):1124–1137, Sept 2004.
- [139] J. QI. Dense nuclei segmentation based on graph cut and convexity-concavity analysis. *Journal of Microscopy*, 253(1):42–53, 2014.
- [140] P. L. Hammer and S Rudeanu. *Boolean Methods in Operations Research*. Springer-Verlag, New York, 1968.
- [141] Ranjith Unnikrishnan and Martial Hebert. Measures of similarity. In *Proceedings of the Seventh IEEE Workshops on Application of Computer Vision (WACV/MOTION'05) - Volume 1 - Volume 01*, WACV-MOTION '05, pages 394–, Washington, DC, USA, 2005. IEEE Computer Society.
- [142] C. Sommer, C. Straehle, U. Kothe, and F.A. Hamprecht. Ilastik: Interactive learning and segmentation toolkit. In *Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on*, pages 230–233, March 2011.
- [143] The FARSIGHT toolkit. <http://www.farsighttoolkit.org>.
- [144] MatlabBGL. https://www.cs.purdue.edu/homes/dgleich/packages/matlab_bgl/.
- [145] Maxflow v3.0. <http://vision.csd.uwo.ca/code/>.
- [146] J. T. Wessels, K. Yamauchi, R. M. Hoffman, and F. S. Wouters. Advances in cellular, subcellular, and nanoscale imaging in vitro and in vivo. *Cytometry A*, 77A(7):667–676, 2010;.
- [147] Mohammad Fallahi-Sichani, Saman Honarnejad, Laura M Heiser, Joe W Gray, and Peter K Sorger. Metrics other than potency reveal systematic variation in responses to cancer drugs. *Nature Chemical Biology*, 9:708714, 2013.
- [148] Aabid Shariff, Robert F. Murphy, and Gustavo K. Rohde. A generative model of microtubule distributions, and indirect estimation of its parameters from fluorescence microscopy images. *Cytometry Part A*, 77A(5):457–466, 2010.
- [149] J.J. Roix, P.G. McQueen, P.J. Munson, L.A. Parada, and T. Misteli. Spatial proximity of translocation-prone gene loci in human lymphomas. *Nat Genetics*, 34(3):287–291, 2003.
- [150] Ivo Dinov, Kamen Lozev, Petros Petrosyan, Zhizhong Liu, Paul Eggert, Jonathan Pierce, Alen Zamanyan, Shruthi Chakrapani, John Van Horn, D. Stott Parker, Rico Magsipoc, Kelvin Leung, Boris Gutman, Roger Woods, and Arthur Toga. Neuroimaging Study Designs, Computational Analyses and Data Provenance using the LONI Pipeline. *PLoS ONE*, 5(9):e13070, 09 2010.

- [151] T. Bray, J. Paoli, and C.M. Sperberg-McQueen. *Extensible Markup Language (XML) 1.0*. W3C Recommendation, Feb. 1998.
- [152] Melissa Linkert, Curtis T. Rueden, Chris Allan, Jean-Marie Burel, Will Moore, Andrew Patterson, Brian Loranger, Josh Moore, Carlos Neves, Donald MacDonald, Aleksandra Tarkowska, Caitlin Sticco, Emma Hill, Mike Rossner, Kevin W. Eliceiri, and Jason R. Swedlow. Metadata matters: access to image data in the real world. *The Journal of Cell Biology*, 189(5):777–782, 2010.
- [153] J. Johnston, A. Nagaraja, H. Hochheiser, and I. Goldberg. A flexible framework for web interfaces to image databases: supporting user-defined ontologies and links to external databases. In *Biomedical Imaging: Nano to Macro, 2006. 3rd IEEE International Symposium on*, pages 1380–1383, April 2006.
- [154] Chris Allan, Jean-Marie Burel, Josh Moore, Colin Blackburn, Melissa Linkert, Scott Loynton, Donald MacDonald, William J Moore, Carlos Neves, Andrew Patterson, Michael Porter, Aleksandra Tarkowska, Brian Loranger, Jerome Avondo, Ingvar Lagerstedt, Luca Lianas, Simone Leo, Katherine Hands, Ron T Hay, Ardan Patwardhan, Christoph Best, Gerard J Kleywegt, Gianluigi Zanetti, and Jason R Swedlow. OMERO: flexible, model-driven data management for experimental biology. *Nature Methods*, 9:245–253, 2012.
- [155] Anne Carpenter, Thouis Jones, Michael Lamprecht, Colin Clarke, In Kang, Ola Friman, David Guertin, Joo Chang, Robert Lindquist, Jason Moffat, Polina Golland, and David Sabatini. CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biology*, 7(10):R100, 2006.
- [156] Fabrice de Chaumont, Stéphane Dallongeville, Nicolas Chenouard, Nicolas Hervé, Sorin Pop, Thomas Provoost, Vannary Meas-Yedid, Praveen Pankajashan, Timothée Lecomte, Yoann Le Montagner, Thibault Lagache, Alexandre Dufour, and Jean-Christophe Olivo-Marin. Icy: An open bioimage informatics platform for extended reproducible research. *Nature Methods*, 9:690–696, 2012.
- [157] IceGrid. <http://www.zeroc.com/icegrid/>.
- [158] C.A. Schneider, W.S. Rasband, and K.W Eliceiri. NIH Image to ImageJ: 25 years of image analysis. *Nature Methods*, 9:671–675, 2012.
- [159] Arthur Edelstein, Nenad Amodaj, Karl Hoover, Ron Vale, and Nico Stuurman. *Computer Control of Microscopes Using Manager*. John Wiley & Sons, Inc., 2010.
- [160] Michael R. Berthold, Nicolas Cebren, Fabian Dill, Thomas R. Gabriel, Tobias Kötter, Thorsten Meinl, Peter Ohl, Christoph Sieb, Kilian Thiel, and Bernd

- Wiswedel. KNIME: The Konstanz Information Miner. In *Studies in Classification, Data Analysis, and Knowledge Organization (GfKL 2007)*. Springer, 2007.
- [161] Katherine Wolstencroft, Robert Haines, Donal Fellows, Alan Williams, David Withers, Stuart Owen, Stian Soiland-Reyes, Ian Dunlop, Aleksandra Nenadic, Paul Fisher, Jiten Bhagat, Khalid Belhajjame, Finn Bacall, Alex Hardisty, Abraham Nieva de la Hidalga, Maria P. Balcazar Vargas, Shoaib Sufi, and Carole Goble. The Taverna workflow suite: designing and executing workflows of web services on the desktop, web or in the cloud. *Nucleic Acids Research*, 41(W1):W557–W561, 2013.
- [162] Bertram Ludäscher, Ilkay Altintas, Chad Berkley, Dan Higgins, Efrat Jaeger, Matthew Jones, Edward A. Lee, Jing Tao, and Yang Zhao. Scientific workflow management and the Kepler system: Research articles. *Concurr. Comput. : Pract. Exper.*, 18(10):1039–1065, August 2006.
- [163] David Churches, Gabor Gombas, Andrew Harrison, Jason Maassen, Craig Robinson, Matthew Shields, Ian Taylor, and Ian Wang. Programming Scientific and Distributed Workflow with Triana Services: Research articles. *Concurr. Comput. : Pract. Exper.*, 18(10):1021–1037, August 2006.
- [164] Pierre Bellec, Sébastien Lavoie-Courchesne, Phil Dickinson, Jason Lerch, Alex Zijdenbos, and Alan C Evans. The pipeline system for Octave and Matlab (PSOM): a lightweight scripting framework and execution engine for scientific workflows. *Frontiers in Neuroinformatics*, 6(7), 2012.
- [165] Jeremy Goecks, Anton Nekrutenko, James Taylor, and The Galaxy Team. Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biol*, 11(8):R86, 2010.
- [166] Daniel Blankenberg, Gregory Von Kuster, Nathaniel Coraor, Guruprasad Ananda, Ross Lazarus, Mary Mangan, Anton Nekrutenko, and James Taylor. Galaxy: A web-based genome analysis tool for experimentalists. *Current protocols in molecular biology*, pages 19–10, 2010.
- [167] Belinda Giardine, Cathy Riemer, Ross C Hardison, Richard Burhans, Laura Elnitski, Prachi Shah, Yi Zhang, Daniel Blankenberg, Istvan Albert, James Taylor, Webb C Miller, W James Kent, and Anton Nekrutenko. Galaxy: a platform for interactive large-scale genome analysis. *Genome research*, 15(10):1451–1455, 2005.
- [168] M. Bostock, V. Ogievetsky, and J. Heer. D3: Data-Driven Documents. *Visualization and Computer Graphics, IEEE Transactions on*, 17(12):2301–2309, dec. 2011.

- [169] Eelco Plugge, Tim Hawkins, and Peter Membrey. *The Definitive Guide to MongoDB: The NoSQL Database for Cloud and Desktop Computing*. Apress, Berkely, CA, USA, 1st edition, 2010.
- [170] jQuery. <http://www.jquery.com>.
- [171] Bootstrap. <http://twitter.github.io/bootstrap/>.
- [172] Ricardo Henriques, Mickael Lelek, Eugenio F Fornasiero, Flavia Valtorta, Christophe Zimmer, and Musa M Mhlanga. QuickPALM: 3D real-time photoactivation nanoscopy image processing in ImageJ. *Nature Methods*, 7:339–340, 2010.
- [173] Eric Betzig, George H. Patterson, Rachid Sougrat, O. Wolf Lindwasser, Scott Olenych, Juan S. Bonifacino, Michael W. Davidson, Jennifer Lippincott-Schwartz, and Harald F. Hess. Imaging intracellular fluorescent proteins at nanometer resolution. *Science*, 313(5793):1642–1645, 2006.
- [174] Michael J Rust, Mark Bates, and Xiaowei Zhuang. Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM). *Nature Methods*, 3:793–796, 2006.
- [175] Matlab, release 2010b, 2010. <http://www.mathworks.com>.
- [176] Prtools. www.prttools.org.
- [177] XML IO Tools. <http://www.mathworks.com/matlabcentral/fileexchange/12907-xml-io-tools>.
- [178] Crossfilter. <http://square.github.io/crossfilter/>.
- [179] Erik Meijering, Oleh Dzyubachyk, Ihor Smal, and Wiggert A. van Cappellen. Tracking in cell and developmental biology. *Seminars in Cell and Developmental Biology*, 20(8):894 – 902, 2009.

CURRICULUM VITAE

Kaustav Nandy

Leidos Biomedical Research, Inc./Frederick National Laboratory for Cancer Research,
Frederick, Maryland, 21702

Work: 301-846-6109

Email: nandyk@mail.nih.gov, kaustav.nandy@gmail.com

EDUCATION

- **Ph.D:** Candidate with the Department of Computer Science, *University of Maryland College Park*, expected graduation 2015
- **Master of Science:** Electrical Engineering, *University of Maryland, College Park* (Dec 2006)
- **Master of Technology** Electrical Engineering, *Indian Institute of Technology(IIT), Kanpur*, India (May 2004)
- **Bachelor of Electrical Engineering(Hons)** Electrical Engineering, *Jadavpur University, Kolkata, India* (May 2001)

WORK EXPERIENCE

- **Scientist 1** *Optical Microscopy and Analysis Lab , Leidos Biomedical Research, Inc.*, Developing image processing, pattern recognition and classification algorithms for segmentation and analysis of microscopic images, operation of confocal microscopes and software support [Jan 2007 – till date]
- **Summer Internship** *SET Corporation, Greenbelt*, Real time face detection and tracking using intensity, contour and skin color segmentation, used for identity hiding in video(Summer 2006)
- **Graduate Research Assistant** *University of Maryland, Advisor: Dr. Rama Chellappa*: Human gait recognition, Modeling of Human gait using articulated dynamic models, Extracting gait data from video
- **Graduate Research Assistant** *Indian Institute of Technology, Kanpur, Advisor: Dr. Govind Sharma*: Image Enhancement Using Curve Indicator Random Field for 2D and 3D image enhancement applications
- **Graduate Teaching Assistant** *Indian Institute of Technology, Kanpur*

PUBLICATIONS

- MiPipeline (Microscopy Pipeline): A User Friendly Software Ecosystem for Microscopy Image Analysis and Informatics, **Kaustav Nandy**, P.R. Gudla, W. Cukierski, K.J. Meaburn, T. Misteli and S.J. Lockett, *manuscript close to submission*
- Graphcut Based 3D Nuclear Segmentation in Optical Microscopy Tissue Images, **Kaustav Nandy**, Rama Chellappa and Stephen Lockett, *submitted to IEEE Signal Processing Society journal on Selected Topics in signal processing, March 2015*
- Probabilistic Edge Detection In 3D Optical Microscopy Images Of Tissue Samples, **Kaustav Nandy**, P. R. Gudla, Rama Chellappa and Stephen J. Lockett, *accepted at IEEE International Symposium on Biomedical Imaging*, NYC, April 2015
- Segmentation and Quantitative Analysis of Individual Cells in Developmental Tissues, **Kaustav Nandy**, Jusub Kim, Dean P. McCullough, Matthew McAuliffe, Karen J. Meaburn, Terry P. Yamaguchi, Prabhakar R. Gudla and Stephen J. Lockett, book chapter published in *Mouse Molecular Embryology, Methods in Molecular Biology*, Volume 1092, Chapter 16, 2014
- Quantitative analysis of F-actin redistribution in astrocytoma cells treated with candidate pharmaceuticals, S. Lockett, C. Verma, A. Brafman, P. Gudla, **Kaustav Nandy**, Y. Mimaki, P. L. Fuchs, J. Jaja, K. M. Reilly, J. Beutler, T. J. Turbyville, *Cytometry A*, 85 (6), 512-521, 2014
- Ranked retrieval of segmented nuclei for objective assessment of cancer gene repositioning, W. J. Cukierski, **Kaustav Nandy**, P. R. Gudla, K. J. Meaburn, T. Misteli, D. J. Foran and S. J. Lockett, *BMC Bioinformatics*, 2012 Sep 12;13:232
- Automatic Segmentation and Supervised Learning Based Selection of Nuclei in Cancer Tissue Images, **Kaustav Nandy**, P. R. Gudla, R. Amundsen, Karen Meaburn, Tom Misteli and Stephen J. Lockett, *Cytometry A*, 2012 Sep;81(9):743-54.
- Interactive Segmentation and Tracking in Optical Microscopic Images, **Kaustav Nandy**, *Cytometry A*, 2012 May;81:357-9
- Supervised Learning Framework for Screening Nuclei in Tissue sections, **Kaustav Nandy**, Prabhakar R. Gudla, Ryan Amundsen, Karen J. Meaburn, Tom Misteli and Stephen J. Lockett, accepted for oral presentation at *33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '11)*
- Automatic Nuclei Segmentation and Spatial FISH Analysis for Cancer Detection, **Kaustav Nandy**, Prabhakar R. Gudla, Karen Meaburn, Tom Misteli and Stephen Lockett, *Proceedings of 31st Annual International IEEE EMBS Conference*, September, 2-6, 2009, Minneapolis, Minnesota, USA
- Quantifying the astrocytoma cell response to candidate pharmaceutical from F-ACTIN image analysis, Chi Cui, Joseph Jaja, Thomas Turbyville, John Beutler, Prabhakar R. Gudla, **Kaustav Nandy** and Stephen Lockett, *Proceedings of 31st Annual International IEEE EMBS Conference*, September, 2-6, 2009, Minneapolis, Minnesota, USA
- Quantification of F-Actin Structures in Astrocytoma Cells in Response to Candidate Pharmaceutical, Chi Cui, Tommy Turbyville, Prabhakar Gudla, **Kaustav Nandy**, John Beutler, Joseph Jaja and Stephen Lockett, abstract for poster presentation at *International Workshop on Microscopic Image Analysis with Applications in Biology*, 2008
- Solid Tumor Image Registration By Hierarchical Decomposition Assisted Automatic Landmark Detection, Alvaro Jorge, Prabhakar Gudla, **Kaustav Nandy**, Stephen J. Lockett, *International Workshop on Microscopic Image Analysis with Applications in Biology*, 2008
- A High-Throughput System for Segmenting Nuclei Using Multiscale Techniques, Prabhakar R. Gudla, **Kaustav Nandy**, J. Collins, K. J. Meaburn, T. Misteli, and S. J. Lockett, *Cytometry A*, May 2008

- Automatic Segmentation of Cell Nuclei in 2D using Dynamic Programming, **Kaustav Nandy**, P R Gudla, Stephen J Lockett, *International Workshop on Microscopic Image Analysis with Applications in Biology*, 2007
- Simulation and Analysis of Human Walking Motion, **Kaustav Nandy** and Rama Chellappa, Proceedings of *32nd IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Volume1*, 2007
- Application of Wavelets in Detection and Classification of Microcalcification in Digital Mammograms - Some Recent Advances, Soma Biswas and **Kaustav Nandy**, proceedings of *International Conference on Mathematical Biology*, IIT, Kanpur, February 2004

POSTERS, ABSTRACTS AND TECHNICAL PAPER PRESENTATION

- Web-centric Software for Studying Tissue Across Multiple Spatial/Hierarchical Scales, **Kaustav Nandy**, Yanling Liu, David Mott, Karen Meaburn, Tom Misteli, Stephen J. Lockett and Prabhakar R Gudla, *Turning Images to Knowledge: Large-Scale 3D Image Annotation, Visualization and Management Conference, Janelia Farms, HHMI, Oct 2012*
- Selecting and Verifying Accurately Segmented Nuclei From Tissue Images, **Kaustav Nandy**, Prabhakar R. Gudla, Ryan Amundsen, Karen Meaburn, Tom Mistelli and Stephen Lockett, *abstract accepted for oral presentation at CYTO 2011, XXVI Congress of the International Society for Advancement of Cytometry, Baltimore, MD, May 2011*
- Automatic Nuclei Segmentation and Spatial FISH Analysis for Cancer Detection, **Kaustav Nandy**, Prabhakar R. Gudla, Karen Meaburn, Tom Mistelli and Stephen Lockett, *Proceedings of 31st Annual International IEEE EMBS Conference*, September, 2-6, 2009, Minneapolis, Minnesota, USA

HONORS AND AWARDS

- Dean's Fellowship from University of Maryland for PhD in Computer Science
- SAIC Technical Fellows Council publication award 2010 for the paper Automatic Nuclei Segmentation and Spatial FISH Analysis for Cancer Detection, **Kaustav Nandy** et al, *Proceedings of 31st Annual International IEEE EMBS Conference*, September, 2-6, 2009, Minneapolis, Minnesota, USA
- SAIC Technical Fellows Council publication award 2009 for the paper *A High-Throughput System for Segmenting Nuclei Using Multiscale Techniques*, Prabhakar R. Gudla, **Kaustav Nandy**, et al, *Cytometry A*, May 2008
- Merit Certificate for academic excellence from Jadavpur University Alumni Association
- B.N. Paul Memorial Gold Centered Silver Medal, Jadavpur University, 2001
- All India Rank 29th (99.76 percentile) in GATE 2002 Examination which is the all India qualifying examination for engineering masters education
- State 19th in Madhyamik (Secondary) Examination

RESEARCH INTERESTS

Computer Vision/Image Processing, Pattern Recognition, Signal Processing, Video Processing, Machine Learning, Software Programming, Data Analysis and Modeling, Data Visualization, Biophysical applications to cell biology, Microscopy applications

PROFESSIONAL AFFILIATION

- ❖ Member *Institute of Electrical and Electronics Engineers (IEEE)*
- ❖ Member *The International Society for Optical Engineering (SPIE)*
- ❖ Member *Association for Computing Machinery (ACM)*

OTHER PROFESSIONAL ACTIVITIES

- Reviewer for journals : Cytometry A, Cytometry B, IEEE Transactions on Biomedical Engineering, Analytical Cellular Pathology, IEEE Transactions on Medical Imaging, BioTechniques, Biomedical Signal Processing and Control, and Computer Vision and Image Understanding
- Mentored students in high school, undergraduate and advanced graduate levels (both MS and PhD) for summer and year long internships

COMPUTER SKILLS

- *Software* : C, C++, OpenCV, VC++, Fortran, Visual Basic, FoxPro, Qt, HTML, PHP, JavaScript, Perl, SQL, MPI, OpenMP
- *Academic Software* : Matlab, DIP Image, MIPAV, VTK, ITK, ImageJ, Latex
- *Platform* : Windows, Linux, Unix, Sun

COURSE WORK

- **PhD**: High Performance Computing, Image Segmentation, Analysis of algorithms.
- **M.S.**: Random Processes in Communication and Control, Digital Image Processing, Information Theory, Statistical and Neural Pattern Recognition, Machine Learning, Estimation and Detection Theory, Scientific Computing
- **M. Tech.**: Introduction to Signal Processing and Communication, Statistical Signal Processing, Advanced Topics in Digital Filtering (Multirate Filtering and Wavelets), Introduction to Modern Control Systems, Robust Control
- **Undergraduate**: Digital Signal Processing, Microprocessor Architecture and programming, Numerical Analysis, Control Systems