To the Graduate Council:

I am submitting herewith a dissertation written by Charles H. Margraves entitled “From Nearfield Nanoparticle Tracking To Intracellular Vesicle Tracking.” I have examined the final electronic copy of the dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Mechanical Engineering.

Dr. Kenneth D. Kihm,
Major Professor

We have read this dissertation
and recommend its acceptance:

Dr. Anthony English
Dr. Seung Joon Baek
Dr. Dongjun Lee

Accepted for the Council:

Carolyn R. Hodges
Vice Provost and Dean of Graduate School

(Original signatures are on file with the official student records.)

From Nearfield Nanoparticle Tracking To Intracellular Vesicle Trafficking

A Dissertation

Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Charles H. Margraves

August, 2008

Copyright © 2008 by Charles H. Margraves
All rights reserved.

DEDICATION

I would like to dedicate this work to several family members who are not here to share in the joy of its completion. To my grandparents: Frank and Fleeta Moore, and Henriette Margraves. No words can express how blessed I feel to have had such wonderful people in my life. To my mother in law Maria Christina Rutz, a wonderful and giving person who I wish I could have known longer. To my father Charles W. Margraves, while I wish he could have seen the completion, I am thankful to the Lord for allowing him to see the start. I hope one day that I will be able to pass along the wisdom, love, and encouragement he provided to me to my own children.

Finally, to my Lord and savior Jesus Christ, may any glory received from this work be passed to Him, while any errors show proof of my need for Him.

“The fear of the Lord is the beginning of wisdom, and the knowledge of the holy is understanding.”

Proverbs 9:10 KJV

“Do not be anxious about anything, but in everything, by prayer and petition, with thanksgiving, present your request to God. And the peace of God, which transcends all understanding, will guard your hearts and your minds in Christ Jesus.”

Philippians 4:6-7 NIV

“If you want to make God laugh, tell Him *your* long term plans.”

Charles W Margraves

ACKNOWLEDGEMENTS

I would like to first thank Dr. Kenneth Kihm for his patience and support. He has been an invaluable resource for the last several years not only professionally but personally as well. I would also like to thank my committee members, Dr. Anthony E. English, Dr. Seung J. Baek, and Dr. Dongjun Lee for their willingness to serve and guide me through this process. A special thanks also to all of the members of Dr. Baek's lab who have gone above and beyond in helping me complete this work. Specifically I would like to recognize Dr. Seong-Ho Lee, Jason Liggett, and Jae Hoon Bahn.

I would also like to recognize the members of the MINSFET lab who I have worked with for the last four years: Dr. Chanhee Chon, Dr. Seonghwan Kim, and Iltai Kihm. Also to Dr. Changkyoung Choi who I spent countless hours with behind a microscope. Thank you for all your help both professionally, but much more importantly personally. It has been my honor to get to know and spend time with you and your family. I would also like to say a special thanks to Joey Tipton. Your encouragement and sense of humor have many times made a difficult situation much easier. I pray blessings on you and your family.

To Dr. Mike Jones, my first inspiration to pursue this path. I hope one day to have the same impact on students that you had on me.

To my mother, Martha Margraves. A special thanks for dinners at 10:00 p.m. with no complaints, and for always being there even when it was not easy. I thank God for such a wonderful mother, and now grandmother.

To my daughter Christina, and our newest member of the family who should be here in February. I have known few greater joys in this life than being your father.

Finally to my wife Ivette: A man could not have a better partner to share his life with. Of all of the many blessings I have received, next to my Lord, you are the greatest. Your love and support have been an endless source of strength. I look forward to spending the rest of our lives together.

ABSTRACT

The initial goal of this research was to measure the hindered Brownian motion of nanoparticles (100 nm to 500 nm in radii), in varying salinities of water, in order to compare the normal and tangential motion with existing theory. Using techniques developed from this work, brain cancer cells containing vesicles loaded with Nonsteroidal anti-inflammatory drug-Activated Gene-1 (NAG-1), tagged with a Green Fluorescent Protein (GFP), were examined to see what effect the glass cover slip played in hindering their motion.

Several microscopy techniques have been used in this work including Total Internal Reflection Fluorescent Microscopy (TIRFM) and Differential Interference Reflection Microscopy (DICM). TIRFM in is a method used to examine an area approximately 1micrometer from the cover slip, while DICM can be used to examine the coverage area of a cell. Included in this work are several digital image processing techniques that were developed for tracking nanoparticles and biological vesicles, as well as software to examine cellular and focal adhesion coverage area using DICM and Interference Reflection Contrast Microscopy (IRCM), a technique that is especially useful in examining cell substrate interactions.

Results of nanoparticle tracking showed that the tangential motion of the particles followed very closely to the theory proposed by Goldman et. Al., while the normal motion was substantially different than that proposed by Brenner. However it should be noted that Brenner's theory does not include electrostatic forces that are significant in this work and therefore it should not be concluded that the theory is incorrect. Rather it was concluded that an additional term is needed to account for this added force.

For intracellular vesicle tracking in cancer cells three types of motion were apparent: directional, Brownian, and caged. It was observed that the majority of the motion was either directional or caged and that hindrance values due to hydrodynamic effects were small compared with other hindrance effects. Also a method was established to estimate the average vesicle size base on the observed motion. This

method is believed to have potential for use in determining fluid viscosity as well as nanoparticle sizes in future studies.

Table of Contents Page

CHAPTER I INTRODUCTION 1
1.1 Purpose of Study 1
1.2 Literature Review 2
1.2.1 Near-Wall Micro/Nanoparticle Tracking 2
1.2.2 Intracellular Vesicle Tracking 4
1.2.3 Nonsteroidal anti-inflammatory drug-Activated Gene-1 (NAG-1) 6

CHAPTER II NANOPARTICLE TRACKING 8
2.1 Total Internal Reflection Fluorescent Microscopy (TIRFM) 8
2.1.1 Total Internal Reflection and Evanescent Waves 8
2.1.2 Fluorescence and the Ratiometric Method 11
2.1.3 Objective vs. Prism Based Systems 13
2.2 Theory 15
2.2.1 Free and Hindered Brownian Diffusion 15
2.2.2 Electric Double Layer Forces 16
2.3 Experimental Methods 18
2.3.1 Imaging System 20
2.3.2 Sample Preparation and Data Acquisition 20
2.4 Digital Image Processing 23
2.4.1 Photobleaching 23
2.4.2 Intensity Variations and Dark Noise Corrections 24
2.4.3 Reference Particle Intensity Determination 28
2.4.4 Particle Tracking 28
2.5 Results 32
2.5.1 Hindered Tangential Diffusion 32
2.5.1 Hindered Normal Diffusion 35
2.5.3 Measurement Uncertainty 38
2.6 Conclusions and Future Work 41

CHAPTER III DIGITAL IMAGE PROCESSING OF LIVE CELLS 44
3.1 Differential Interference Contrast Microscopy Image Analysis 44
3.1.1 Purpose of Experiment 44
3.1.2 Microscopy Technique 45
3.1.3 Algorithm 49
3.1.4 Results 54
3.1.5 Experimental Conclusions and Future Work 54
3.2 Interference Reflection Contrast Microscopy Image Analysis 57
3.2.1 Purpose of Experiment 57
3.2.2 Microscopy Technique 58
3.2.3 Algorithm 64
3.2.4 Results 67
3.2.4 Experimental Conclusions and Future Work 70

CHAPTER IV Intracellular Vesicle Formation and Tracking 71
4.1 Purpose 71
4.2 Cell Handling 72
4.2.1 Cell Culturing and Constructs 72
4.2.2 Transfection 73
4.3 Vesicle Formation 73
4.3.1 Controls 73
4.3.2 NAG-1 DNA Modifications and Mutants 75
4.3.3 Results and Conclusions 75
4.4 Intracellular Vesicle Tracking 77
4.4.1 Types of motion 77
4.4.2 Calculation of Vesicle Center 79
4.4.3 Particle Tracking Algorithm 81
4.5 Analysis of Vesicle Motion 83
4.5.1 Division of Vesicle Motion 83
4.5.2 Examination of Near Wall Hindrance 87
4.5.3 Examination of Particle Size and Hindrance Forces 89
4.6 Results 95
4.6 Conclusions and Future Research 98

LIST OF REFERENCES 100

Appendix A 108
Appendix B 123
Appendix C 134
Appendix D 138

Vita 153

LIST OF TABLES

Table Page

2.1 Sample Preparation Conditions for Hindered Brownian Motion Experiments 22
3.1 Effect of multiple reflections on the intensity and phase of an incident light source. 60
4.1 Average parameters for varying types of vesicle motion. 84
4.2 Estimated vesicle size and effective fluid viscosity for vesicles experiencing Brownian or caged motion. 97

LIST OF FIGURES

Figure Page

2.1 100 nm radii polystyrene beads in water. (a) Non-TIRFM conditions. (b) TIRFM conditions 9

2.2 Comparison of incident angle to refracted angle. (a) When n_t is greater than n_i, θ_t is less than θ_i. (b) When n_t is less than n_i, θ_t is greater than θ_i. The magnitude of the transmitted wave can be calculated using Snell's law (Equation (1)). 10

2.3 Particles closer to the surface appear brighter due to the higher intensity of the evanescent wave near the surface. By taking a ratio of the intensity of a moving particle and an attached particle, equation 5 can be used to measure absolute elevations. 12

2.4 (a) Schematic of objective based TIRFM system. Angle of the wave is adjusted by changing the location it enters the prism. Imaging is completed from above the sample. (b) Objective based TIRFM system. Incident angle is set by the micrometer and the dichroic mirror is used to split the excitation and emission signals allowing imaging to occur below the sample. 14

2.5 Normal and tangential hindered diffusion coefficients for 100 nm and 500 nm radii particles. The hindered effects for larger particles are experienced much further from the wall than for smaller particles. 17

2.6 Effect of adding ions to the electric double layer. As the ionic concentration is increased, the Debye length is reduced, which in turn reduces the elevations over which the double layer forces are felt. This infers there should be less hindrance to the motion of a particle, at a given elevation, for higher ionic concentrations. 19

2.7 Picture of TIRFM imaging system. 21

2.8 Photobleaching rates for various excitation intensities for 100 nm (solid symbols) and 500nm (regular symbols) radius fluorescent particles. The inset caption illustrates the illumination powers. 25

2.9 Intensity corrections for uneven illumination. (a) A single image of fluorescent die dissolved in water under the same testing conditions used for moving particle measurements. (b) Average of 100 single images, normalized by the average intensity of all images. (c) Effect of dividing (a) by (b). Note no pattern is observed and illumination is almost completely even. 27

2.10 Flow chart for the Brownian Motion Code (BMC). This algorithm was developed to track particle motion and calculate MSD values both normally and tangentially to the surface. 30

2.11 (a) Initial frame. (b) Selected window. Individual windows were chosen from the initial images in order to track single particles. Each window was examined over all frames to insure only one particle was being tracked at a given time. This prevented the possibility of mistaking displacement measurements with distance between particles. 32

2.12 Comparison of theoretical and experimental tangential MSD values for three particle sizes. All three particle sizes follow the theoretical values very closely. 33

2.13 Comparison of theoretical and experimental tangential MSD values for 250 nm radii particles at three different concentrations of NaCl. Little change is observed for MSD values at different NaCl concentrations. 34

2.14 Tangential MSD values versus time step. Each data point represents measurements made for a given particle size at an elevation near the penetration depth. The linearity of the MSD values versus time step indicate the Brownian nature of the movement. 36

2.15 Comparison of theoretical and experimental normal MSD values for three particle sizes. Larger particles tend to follow the theoretical values more closely, particularly further from the surface. NaCl concentration is at 10 mM for all particle sizes. 37

2.16 Comparison of theoretical and experimental normal MSD values for 250 nm particles at NaCl concentrations of .5, 1, and 10 mM. As the amount of NaCl decreases, the hindered effect very near the surface increases. 39

2.17 250 nm radii particle motion is not linear with respect to time step, evidence that the motion is not Brownian. NaCl concentration is at 10 mM. 40

2.18 Uncertainty in elevation measurements due to micrometer accuracy and intensity fluctuations for moving and stuck particles. 42

3.1 Indium-Tin Oxide coated coverglass. ITO is both electrically conductive and optically transparent. These properties make it an excellent choice for conducting optical and electrical measurements simultaneously. 45

3.2 Schematic of light path and optical components necessary for DICM imaging. White light from a halogen lamp is polarized before entering the Nomarski prism. It is then split into two components, normal to each other, and is passed through the condenser and into the sample. The two beams pass through the sample

approximately 200 nm apart from each other. The beams then travel through the objective and are recombined into a single beam as they pass through the second Nomarski prism. They are then passed through the analyzer to remove any extraneous light and finally inter the CCD for image capturing. 46

3.3 DICM image of an endothelial cell. Note variations in the intensity are created by the change in phase between the two beams of light passing through adjacent sections of the sample. 48

3.4 DIP algorithm developed to calculate the coverage area of endothelial cells growing on an ITO electrode. 50

3.5 Digital image processing results. (a) Original image. (b) Deconvoluted image. (c) Edge detection image. (d) Image after thresholding filter. (e) Image after diagonal filter. (f) Image after stitch filter. (g) Image after removal filter. (h) Overlay image. This displays the pixels determined to be part of the cell coverage area (red) on the original image (grey)..... 51

3.6 Comparison of manual and automatic area calculations. Manual calculations were made by tracing the boundary of cells using Wasabi imaging software..... 55

3.7 Comparison of normalized resistance vs. normalized coverage area. Resistance continues to increase even after the electrode is completely covered. 56

3.8(a) Schematic of light path and optical components necessary for IRCM. (b) Schematic of reflection of incident light at the glass medium interface and the medium cell interface. 59

3.9(a) A bright fringe is created when the two reflected waves are in phase, resulting from an optical path difference equal to one quarter of the incident wavelength in this case. (b) A dark fringe is created when the two reflected beams are out of phase, resulting from an optical path difference equal to one half of the incident wavelength in this case. 62

3.11 Relationship between pixel grey level and cell-substrate gap distance as calculated by equations 6-8. 66

3.12 (a) DICM images. (b) IRCM images. (c) Overlay of DICM (Red) and IRCM (Green) images. (d) IRCM images with colors representing pixle grey levels of 1000. Dark blue represents the smallest values with dark red the largest. (e) Blue values represent focal contacts determined from IRCM images. The first column shows the pretoxic state of the cells. The second and third column show changes in the cell 10 minutes and 1 hour after the addition of the toxic agent. The last three columns show cell recovery 20 minutes, 1 hour 50 minutes, and five hours after replacement with new fresh medium. 68

3.13 (a) Changes in the focal contact area over time. (b) Changes in the normalized cell-covered area over time. (c) Ratio of the focal contact area to the total cell covered area over time. 69

4.1 Row 1 DICM images, Row 2 TIRFM images, Row 3 overlay DICM (Red) and TIRFM (Green). (a) Control 1 with wild type NAG-1 tagged with GFP. Vesicle formation is present. (b) Control 2 with GFP directly transfected into the cell. No vesicles present with a cloudy formation distributed throughout the cell. 74

4.2 Row 1 DICM images, Row 2 TIRFM images, Row 3 overlay DICM (Red) and TIRFM (Green). (a) Δ29 modification. (b) RXXR mutation. (c) Active form of NAG-1 only. 76

4.3 (a) Digital image of a single vesicle located inside a brain cancer cell. The cell was transfected with the wild-type NAG-1 tagged with a GFP. (b) Two dimensional plot of the intensity distribution versus pixel for a single vesicle. (c) One dimensional curve fit of the particle intensity to a Gaussian Curve. 80

4.4 Flow Chart for vesicle tracking algorithm. 82

4.5 Tangential MSD values vs. varying time steps. (a) Brownian motion. (b) Directional motion. (c) Caged motion. 85

4.6 Maximum, minimum, and average tangential MSD values vs. varying time steps. (a) Brownian motion. (b) Directional motion. (c) Caged motion. 86

4.7 Examination of tangential hindrance effect as vesicles approach a glass coverslip. .. 88

4.8 Examination of normal hindrance effect as vesicles approach a glass coverslip. 90

4.9 Curve fit of Goldman's hindered tangential diffusion coefficient to experimental data, with particle diameter as the only variable. 92

4.10 (a) Tangential hindered coefficient for .1,1, and 10 µm sized particles. (b) Mean square displacement values for a .1µm sized particle at viscosities of .001, .0015, and .002 $N*s/m^2$ 94

4.11 Curve fit of Goldman's hindered tangential diffusion coefficient to experimental data, with both particle diameter and viscosity left as variable. 96

CHAPTER I INTRODUCTION

1.1 Purpose of Study

The purpose of this research was to use engineering techniques that could provide physiological information not previously observed by methods used in the biological community. Specifically the movement of vesicles, located inside brain cancer cells, was examined and categorized using Total Internal Reflection Fluorescent Microscopy (TIRFM), a technique that examines only the near wall region (~1 μm) of a sample. These vesicles contained the Non-steroidal anti-inflammatory drug-Activated Gene-1 (NAG-1) tagged with a Green Fluorescent Protein (GFP). It is believed that this particular protein (NAG-1) is important in promoting the growth of healthy cells while inhibiting the growth of cancerous cells. During testing a method was developed to estimate the average size of the vesicles, and an effective viscosity within the cell, based on their movement. It is hoped that this method will provide a way to determine the effectiveness of drug treatments on the production of NAG-1, without having to destroy the cell. Current methods, such as Western Blot analysis, require the cells be fixed before examining the amount of a specific protein.

Before beginning work with cancer cells an intermediate goal was to examine the movement of fluorescent nanoparticles, in various ionic concentration solutions, using TIRFM. Both the normal and tangential motion of the particles were examined and compared to existing theories. Because these experiments are more controlled than when working with living biological samples, they provided an excellent chance to develop reliable experimental procedures for use with cancer cells.

Included in this work is a section on digital image processing of endothelial cells using two different types of microscopial images. While not directly related to vesicle or nanoparticle tracking, methods were established during this phase that proved helpful in later work. Of particular help was the creation of software which enabled two different images to be overlaid. Using this method it was possible to examine where vesicles were

located within a cell as well as to create movies showing the movement of the vesicles with regard to particular cellular structures such as the nuclear region.

The next section will provide a literature review on previous work completed in both micro/nano particle tracking, and intracellular vesicle trafficking. Also included is research examining the NAG-1 protein and its response to different drug treatments. Chapters II, III, and IV, provide the details of the work completed for this research.

1.2 Literature Review

1.2.1 Near-Wall Micro/Nanoparticle Tracking

A great deal of work has been completed over the last twenty years examining the movement of micro and/or nano-sized particles. Much of this work has focused on Brownian or random Diffusion, also referred to as Brownian Motion, which was first observed by Gray while studying glass globules suspended in a fluid [1]. Originally Brownian Diffusion was attributed to the motion of living substances until the botanist Robert Brown observed the motion in inorganic grains [2]. The next step in characterizing the motion of these particles was completed by Gouy who proved that the motion was due to thermal fluctuation of the surrounding fluid molecules and not due to external vibrations. Gouy was also responsible for showing that the motion of these particles were dependant on their size and the surrounding fluid viscosity [3].

Credit for the development of the theory behind Brownian Diffusion is given to Albert Einstein, who used the kinetic theory of gases purposed by Boltzmann to obtain the well known diffusion coefficient [4]. This theory was later validated experimentally by Jean Perrin who examined the motion of known particle sizes in a given fluid [5]. To this point all work only considered particles moving very far from a surface in which case the motion was isotropic. However later work showed that as a particle approached the surface this motion became anisotropic with motion normal to the surface significantly different than motion tangential to the surface. This type of motion became known as Hindered Brownian Diffusion.

In 1961 Brenner completed work relating a particles change in Brownian Diffusion, normal to the wall, as it approached the surface [6]. The analytical expression, based on an infinite series, was a function of particle size and elevation. In 1967 Goldman et al. developed an asymptotic solution to relate the change in Brownian diffusion tangential to the wall [7]. As with Brenner's solution the function was only a relationship between the particle size and its distance from the surface.

In order to examine a particles behavior very near a wall, as well as confirm the accuracy of the solutions proposed by Brenner and Goldman, it was necessary to develop a measurement technique capable of accurately examining the near wall region. In the 1980's Dr. Daniel Axelrod, at the University of Michigan, began developing a technique known as Total Internal Reflection Fluorescence Microscopy (TIRFM) to examine biological samples within 1μm of a surface. This work included, but was not limited to, the examination of serum-albumin absorption dynamics [8], cell-substrate contacts [9], and immunoglobulin surface binding kinetics [10].

The development of this tool made it possible to examine several different near-wall effects including hindered Brownian motion. In the late 1980's and 1990's one of the leading researchers in the area was Dr. Dennis Prieve at Carnegie Mellon University. Unlike Dr. Axelrod who used fluorescent materials for his studies, Dr. Prieve measured the scattering of the evanescent wave, produced by the total internal reflection condition, to measure the movement of micron size spheres. This technique is commonly referred to as TIRM, or Total Internal Reflection Microscopy. Dr. Prieve's focus was to use this method to examine colloidal forces, such as electrostatic forces arising from the electric double layer created on a charged particle and surface [11,12,13]. Later work also examined the hindered motion of the particles as discussed above [14, 15, 16]. For an excellent review of this work, as well as other experiments conducted using TIRM, reference [17] is highly recommended.

As mentioned previously Dr. Prieve's work dealt mainly with micron sized particles. While it is possible to use TIRM with nano-sized particles [18,19] typically TIRFM is used as the fluorescent material produces a much stronger signal than the scattering of the evanescent wave by these very small particles. One of the first efforts to

examine the hindered diffusion of fluorescent nanoparticles was completed by Kihm et al. [20]. This work used TIRFM to measure average values for both the normal and tangential hindered diffusion of nano-sized fluorescent particles. Margraves et al. used the same technique to measure changes in the approach distance of fluorescent nanoparticle in solutions of different ionic concentration [21]. Finally Choi et al. used TIRFM to compare the theories proposed by Goldman and Brenner by measuring elevation dependant hindered diffusion values both normal and tangential to the surface of a glass cover slip [22].

More recent work using TIRFM has involved examination of a flow field seeded with fluorescent particles. Unlike Brownian diffusion studies where the fluid is quiescent and particle motion is due to thermal fluctuations of the fluid, in these studies the particle motion is due to the induced motion of the fluid. Several researchers have examined the so called "slip" velocity on both hydrophilic and hydrophobic surfaces [23, 24]. Numerical simulations have also been conducted to examine the feasibility of dividing TIRFM images into multiple levels above the glass surface [25] in order to achieve a better discrimination of the flow field.

1.2.2 Intracellular Vesicle Tracking

Because intracellular vesicles and/or granules are important in carrying out a multitude of biological functions, a great deal of effort has been spent on examining their interaction with the cell membrane. One specific area examined by several researchers is the docking, fusion, and exocytosis of proteins carried by the vesicles into the plasma membrane. Toomre et al. first used a combination of confocal and epifluorescence microscopy, to examine the movement of vesicular stomatitis virus G protein tagged with a GFP (VSVG3-GFP) from the trans-Golgi network to the plasma membrane [26]. Later they combined epifluorescence with TIRFM to examine the docking and fusion of similar vesicles with the plasma membrane [27]. Other studies examining the fusion of specific vesicles with the plasma membrane include work by Schmoranzer et al., and Toonen et al. [28, 29]. Nofal et al. used TIRFM to analyze the movement of vesicles in order to distinguish between vesicles that had simply docked with the membrane and those that

were primed i.e. ready for fusion [30]. For a review of these processes reference [31] is suggested.

Along with membrane interaction, a significant effort has been made to characterize the movement of these vesicles within different types of cells. These studies are often referred to as single particle tracking or SPT. Schutz et al. used fluorescence microscopy to conduct both two and three dimensional tracking of secretory vesicles along neurites in PC12 cells (Rat adrenal Medula) [32]. Similarly Lemke and Klingauf employed Laser Scanning Confocal Microscopy (LSCM) to conduct SPT experiments on vesicles inside of Hippocampal Boutons [33]. In a study by Yoshina-Ishii et al., phospholipid vesicles were tethered to a lipid bilayer using DNA hybridization [34]. The diffusive dynamics of these vesicles were then examined using SPT and epi-fluorescence microscopy. Unlike previous studies these vesicles were not located inside of a cell. One final example of single particle tracking was conducted by Li et al., who like Schutz et al., examine the three-dimensional motion of single secretory granules inside PC12 cells [35]. Wide field fluorescence images were taken at various elevations, with respect to the coverslip, by moving the objective using a piezoelectric controller. The images were then combined using a deconvolution algorithm to form a three-dimensional image and thus provide three-dimensional tracking. From this data attempts were successfully made to observe a change in the vesicle motion both close to and far from the coverslip.

In order to get better resolution of vesicle movement in the near-wall region TIRFM is typically the tool of choice. The Glucose Transporter 4 (GLUT4) vesicle, which plays an important role in transferring insulin to both fat and muscle cells, was tracked three dimensionally using TIRFM inside of 3T3-L1 cells (Mouse embryonic fibroblasts) [36]. TIRFM was also used for the three-dimensional tracking of large dense-core secretory vesicles within the actin cortex of PC12 cells [37]. While the majority of particle tracking has been completed on human or animal cells other attempts have been made with plant cells. One interesting example is Wang et al.'s use of TIRFM to study the motion of secretory vesicles inside of pollen tubes [38].

A new method to investigate the movement of proteins and or vesicles within a cell is through the use of quantum dots [39]. Quantum dots are small semiconductors

which provide a fluorescent signal that is directly related to their size. By functionalizing the surface of these materials it is possible to attach them to specific proteins. This quantum dot protein combination may then be contained within vesicles [40], or the quantum dot itself may act as a vesicle moving the attached proteins from one location to the next [41]. There are several benefits to using quantum dots, rather than another fluorescent material such as GFP. One of the major benefits is that no modification to the DNA of the protein being examined is needed. Another benefit is that quantum dots tend to have lower photobleaching rates than GFP which makes long term imaging more viable. However without the functionalized coating quantum dots may be toxic to the cells. Also quantum dots, along with GFP, are not inherent to cells and thus add an unknown, and unphyisological, variable to the cell.

Because statistical data is important in all experiments, particularly those of a biological nature, several attempts have been made to automate particle tracking in order to increase the data size. Ku et al. used TIRFM images to automatically track, two dimensionally, secretory vesicles inside of PC12 cells [42], while Singh et al tracked GLUT 4 vesicles tagged with GFP, also two dimensionally, using confocal microscopy images [43]. One of the keys in any tracking algorithm is determining the center point of the vesicle such that distances between two frames can be determined. An excellent paper comparing several different methods for doing this can be found in reference [44]. Also the paper by Saxton and Jacobson[45] provides details on several different types of motion vesicles experience inside of cells.

Before moving to the next section it should be mentioned that in the various works discussed above no attempt was made to examine the size of vesicles using single angle TIRFM. However reference [46] attempts to determine vesicle size through images obtained using multi-angle TIRFM. The work presented in this research takes a different approach and attempts to determine average vesicle sizes based on the movement of the vesicles themselves.

1.2.3 Nonsteroidal anti-inflammatory drug-Activated Gene-1 (NAG-1)

As previously mentioned it is believed that the nonsteroidal anti-inflammatory drug-Activated Gene-1 (NAG-1) may play an important role in the growth of healthy

cells while inhibiting the growth of cancerous cells. Several different cancer cell lines, including colon [47] and prostate [48], have been examined to see what role NAG-1 plays with regards to the spread of the disease. Dr. Seung Joon Baek's Pathobiology lab, located in the College of Veterinary Medicine at the University of Tennessee, has examined the response of NAG-1 in colorectal cancer cell lines when treated with natural plant extracts such as horehound leaf and wild cherry [49]. Okazaki et al. completed similar studies but used ionizing radiation treatments rather than plant extracts [50]. For a review of the effect of NAG-1 and its role in modulating tumorigenesis reference [51] is recommended.

CHAPTER II NANOPARTICLE TRACKING

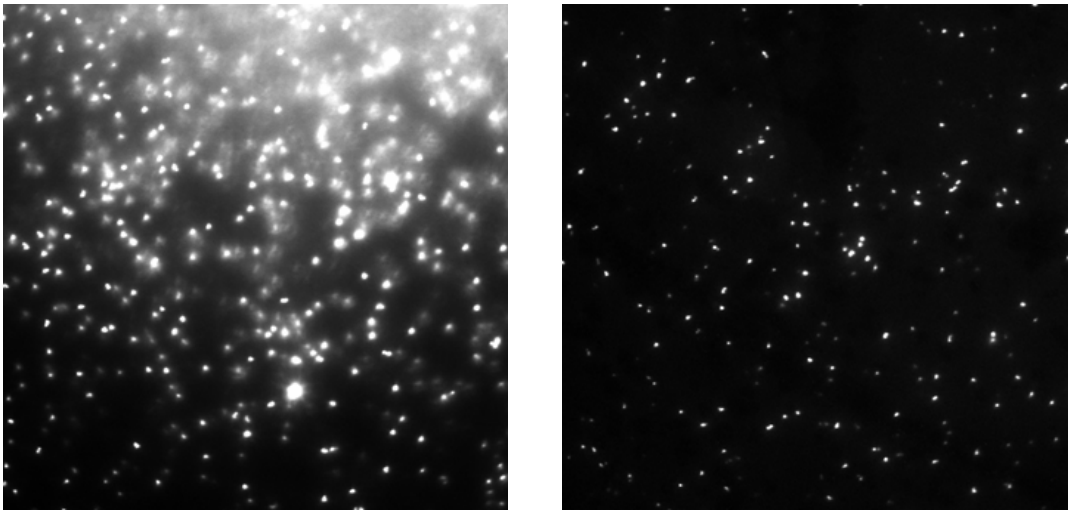
The goal of this research was to compare the motion of nanoparticles, undergoing Hindered Brownian Diffusion near a surface, with existing theories. An objective based Total Internal Reflection Fluorescence Microscopy (TIRFM) system was used to track the particles three-dimensionally under varying concentrations of salinity. Software was developed to track the particles automatically, in order to insure enough data was captured to provide reliable statistics. The following sections provide details on the microscopy approach, relevant theory, and the developed image processing codes. Also included are descriptions of the experimental methods and procedures used for data acquisition. Finally the results and conclusions are provided in detail.

2.1 Total Internal Reflection Fluorescent Microscopy (TIRFM)

Total internal reflection fluorescence microscopy is a technique used to examine the near wall region of a specimen. This region is limited to a distance of approximately 1 μm or less from the coverslip. One of the largest benefits to TIRFM microscopy is its ability to examine fluorescent material close to the surface, while eliminating fluorescent signals located further into a sample. Figure 2.1 provides both TIRF and non-TIRF images of 100 nm radii polystyrene spheres. The next three sections will discuss the theory behind total internal reflection fluorescence microscopy, the ability to make measurements normal to a surface using this technique, and two different experimental methods common to this application.

2.1.1 Total Internal Reflection and Evanescent Waves

When light travels between two mediums of differing indices of refraction, part will be reflected into the incident medium while the remaining part is refracted into the transmitted medium. For case 1, if the transmitted medium has a higher refractive index than the incident medium, and the incident light is traveling at some angle, θ_i, with respect to the normal of the intersection of the two mediums, then the transmitted angle,



(a) (b)

Figure 2.1 100 nm radii polystyrene beads in water. (a) Non-TIRFM conditions. (b) TIRFM conditions

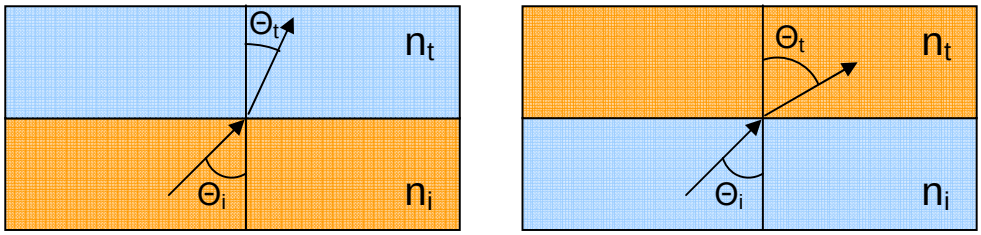


Figure 2.2 Comparison of incident angle to refracted angle. (a) When n_t is greater than n_i, θ_t is less than θ_i. (b) When n_t is less than n_i, θ_t is greater than θ_i. The magnitude of the transmitted wave can be calculated using Snell's law (Equation (1)).

θ_t, will be smaller than the incident angle θ_i. Figure 2.2(a) provides a schematic of this situation. In order to determine θ_t, Snell's Law, shown in equation (1), may be used:

$$n_i \sin\theta_i = n_t \sin\theta_t \qquad (1)$$

where n_i and n_t are the indices of refraction for the incident and transmitted mediums respectively [52]. For case 2, where the index of refraction of the transmitted medium is lower than that in the incident medium, Snell's law shows that the transmitted angle is greater than the incident angle. This case may be observed in Figure 2.2(b).

For case 2, as the incident angle is continually increased, a point is reached where the transmitted angle equals ninety degrees, i.e. no light is transmitted. This angle is termed the critical angle and represents the point of total internal reflection. Although it appears no light passes into the transmitted medium from a macroscopic point of view, closer observation reveals a light wave traveling tangentially to the surface and decreasing exponentially in intensity normal to the surface. This wave is termed an evanescent wave and its intensity may be calculated using equation (2):

$$I_{ev}(z) = I_o e^{\left(-z_p z\right)} \qquad (2)$$

where I_o is the intensity at the interface, and z is the distance above the interface [22]. z_p is the penetration depth given by equation (3):

$$z_p = \frac{4\pi}{\lambda_0}\left[(n_i)^2 \sin^2\theta_i - (n_t)^2\right]^5 \qquad (3)$$

where λ_o is the wavelength of incident light in a vacuum [22]. Typically the penetration depth is on the order of a few hundred nanometers and detectability of the evanescent wave is less than 1μm.

2.1.2 Fluorescence and the Ratiometric Method

As previously stated the goal of this work was to examine the motion of fluorescent nanoparticles very near the surface. The details of these experiments will be discussed in subsequent sections but it is important at this point to introduce the ratiometric method used to track particle motion normal to the surface. This method is based on relating changes in emission intensity of a fluorescent particle to changes in elevation above the surface. It should be noted that a similar method can be used with total internal reflection microscopy, where a relationship is observed between the amount of the evanescent wave that is scattered by a particle and its location above a surface [53].

In general the emission wavelength of a fluorescent material is longer than the wavelength of the excitation source and the difference between the two is known as the Stokes shift. The power of the emission is also directly related to the power of the excitation, therefore it is possible to increase or decrease emission intensity by adjusting the power of the excitation source. As shown in section 2.1.1 the intensity of an evanescent wave, which for our experiments is used as the excitation source, decreases exponentially with distance from the surface. Therefore as fluorescent particles move in and out of this wave their emission intensities change due to their dependence on excitation power. The emission intensity captured by a Charge Coupled Device (CCD) is given by equation (4):

$$I(x,y;z_p) = \phi \cdot I_o(x,y;z_p) \cdot \int_0^\infty [Q(z) \cdot PSF(z)] \cdot c(x,y,z) \cdot e^{-z \cdot z_p} dz \qquad (4)$$

where φ is the quantum efficiency of a fluorophore and CCD array, I_o is the incident intensity of the excitation source at the surface, Q is the collection efficiency of the system, PSF is the point spread function of the microscope, c is the fluorophore

distribution on the nanoparticle, and z is the elevation of a particle above the surface [46]. While it is possible to calculate or measure most of these values, with the exception of the fluorophore distribution, it is not necessary. Because the desire is to measure z displacements based on intensity changes, by taking a ratio of intensities at two different elevations it is possible to calculate the z displacement, as all other variables are assumed to remain approximately constant for similar sized particles. If a measurement can be taken of a particle attached to the surface then comparing all other measurements to this value will provide absolute elevations above the surface. Equation 5 is derived from equation 4 by taking a ratio between a particle at a given elevation and a particle attached to the surface, and then solving for the elevation:

$$z = -z_p \ln\left(\frac{I_z}{I_0}\right) \qquad (5)$$

Figure 2.3 provides a schematic of the ratiometric method for particles located within an evanescent wave.

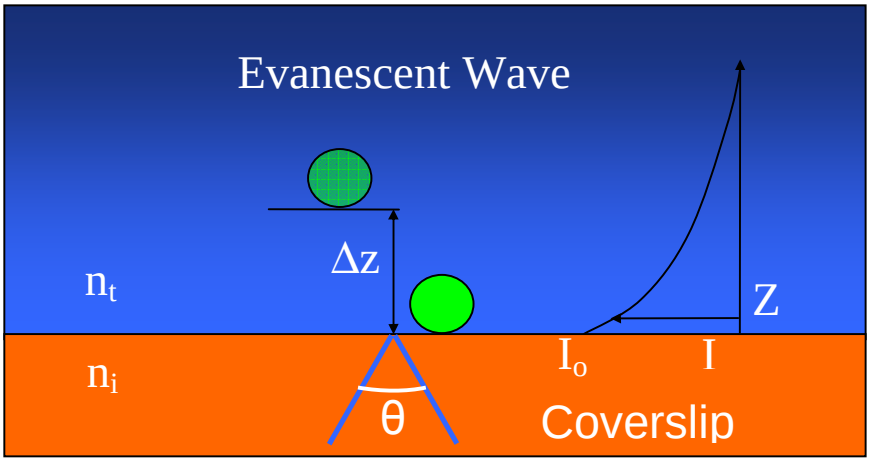


Figure 2.3 Particles closer to the surface appear brighter due to the higher intensity of the evanescent wave near the surface. By taking a ratio of the intensity of a moving particle and an attached particle, equation 5 can be used to measure absolute elevations.

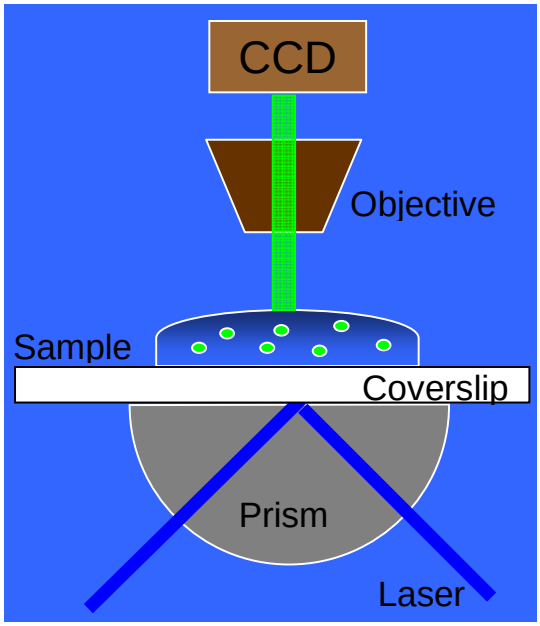
2.1.3 Objective vs. Prism Based Systems

There are essentially two different experimental approaches taken for conducting TIRFM measurements. The first technique is referred to as the prism based method and the second is the objective based method. Figure 2.4 provides a schematic of these two different approaches.

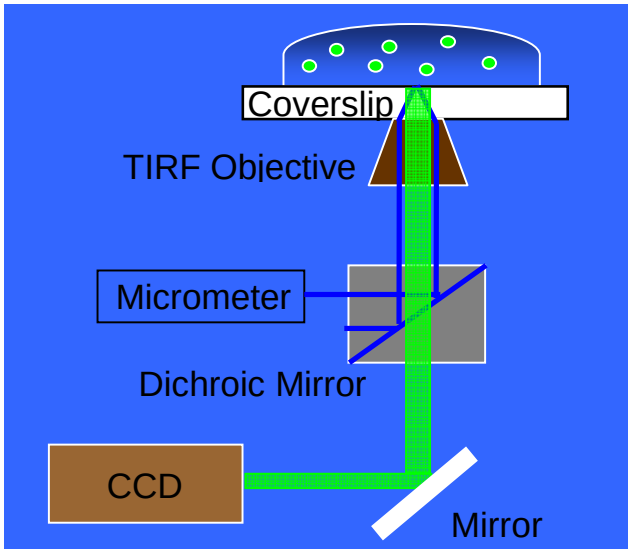
The prism based approach directs the light into the sample area through a prism that is optically coupled to the coverslip. The incident angle is adjusted by rotating a mirror upstream of the prism. An upright microscope is used to capture the emission from the fluorescent particles from above the sample and then it is passed to a CCD camera.

For the objective based approach, the laser is passed through a micrometer, which can be used to adjust the incident angle to a very high precision. The laser is then reflected off a dichroic mirror and through a special TIRFM objective before being passed into the sample. The dichroic mirror is a special optical device that allows a specific range of wavelengths of light to be reflected, while remaining wavelengths are allowed to pass. The emitted light from the particles are passed back through the objective and dichroic mirror where the image is captured by a CCD camera. Because an inverted microscope is used for this set-up no optical devices are placed above the sample.

Due to the fact that fewer optical components are needed for the prism based approached and no special objectives are required, this system is substantially cheaper. However the ability to set the incident angle accurately is sufficiently reduced for this approach and accessing the sample from above is difficult due to the need for an upright microscope. For the objective based approach it is possible to combine other types of transmitted microscopy, such as Differential Interference Contrast Microscopy (DICM), as there are no restrictions to access of the sample from above. Because of the benefits provided by the objective based method, it was chosen for all TIRFM studies presented in this work.



(a)



(b)

Figure 2.4 (a) Schematic of objective based TIRFM system. Angle of the wave is adjusted by changing the location it enters the prism. Imaging is completed from above the sample. (b) Objective based TIRFM system. Incident angle is set by the micrometer and the dichroic mirror is used to split the excitation and emission signals allowing imaging to occur below the sample.

2.2 Theory

This section will begin with a discussion of Einstein's theory for free Brownian Diffusion. Next separate theories related to the tangential and normal hindered Brownian diffusion of particles very near a wall will be explained in detail. Finally electrostatic forces, which may cause derivations from the proposed theories for hindered Brownian motion, will be discussed as well as important parameters affecting this force.

2.2.1 Free and Hindered Brownian Diffusion

When micron sized particles are suspended in a quiescent fluid, close observation reveals random movement of these small particles. This movement, referred to as Brownian Diffusion or Brownian Motion, is the result of the large number of collisions occurring between the particle and the surrounding fluid molecules. If a particle's density is close to that of the fluid, and it is located far from any boundaries, the motion will be isotropic in nature. In 1905 Einstein showed that the movement of these particles could be calculated based on the surrounding fluid environment, the size of the particle, and the Boltzmann constant. He defined this value as the free diffusion coefficient which is shown in equation 6:

$$D_o = \frac{kT}{3\pi\mu d_p} M \; ; \quad M = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{6}$$

where k is the Boltzmann's constant, T is the absolute temperature, μ is the dynamic viscosity of the fluid, d_p is the diameter of the particle, and M is the free diffusion tensor [4]. The ones, located on the diagonal of the tensor matrix, indicate the isotropic motion of the particles.

As the location of a particle with respect to a solid surface becomes very small its diffusion is reduced due to the no-slip boundary condition created by the surface. This motion is referred to as Hindered Brownian Diffusion, and while it is still random in nature it is no longer isotropic. Experiments have shown that the motion of the particle in the direction normal to the surface is significantly more hindered than the motion tangential to the surface.

In 1961 H. Brenner developed a theoretical model, based on an infinite series, to predict the normal hindered effect of a wall on the motion of a particle. In this work he proposed an elevation dependant correction factor for the free diffusion coefficient. This correction factor is presented in equation (7a) with the parameter α provided in equation (7b):

$$\lambda_{zz} = \lambda_{\perp} = \frac{4}{3}\sinh\alpha \cdot \sum_{n=1}^{\infty} \frac{n(n+1)}{(2n-1)(2n+3)} \left[\frac{2\sinh(2n+1)\alpha + (2n+1)\sinh 2\alpha}{4\sinh^2(n+\frac{1}{2})\alpha - (2n+1)^2 \sinh^2\alpha} - 1 \right] \quad (7a)$$

$$\alpha = \cosh^{-1}\left(2z/d_p + 1\right) \quad (7b)$$

While this equation is quite lengthy it should be noted that it is only a function of the particles size and elevation above the surface [6].

In 1967 A.J. Goldman and his fellow researchers created a second correction factor for motion tangential to the surface which is presented in equation 8:

$$\lambda_{xx} = \lambda_{yy} = \lambda_{\parallel} = \left[1 - \frac{9}{16}\left(\frac{d_p}{d_p + 2h}\right) + \frac{1}{8}\left(\frac{d_p}{d_p + 2h}\right)^3 - \frac{45}{256}\left(\frac{d_p}{d_p + 2h}\right)^4 - \frac{1}{16}\left(\frac{d_p}{d_p + 2h}\right)^5 \right]^{-1} \quad (8)$$

Just as in equation (7) this correction factor is only a function of the particle size and elevation above the surface [7]. Figure 2.5 provides an example of these correction factors for particle radii of 100 nm and 500 nm. This figure shows that the motion of larger particles is hindered at greater distances from the wall than smaller particles. Also it should be noticed that both tangential and normal values move toward an asymptotic solution of one. In other words at large distances from the wall the solution approaches free diffusion values.

2.2.2 Electric Double Layer Forces

In the previous section hindered diffusion was discussed with regards to the effects of a nearby wall on the motion of a particle. However this type of hindrance, generally referred to as a hydrodynamic effect, is not the only phenomenon that can alter the diffusion of a particle near a surface. Anther hindered effect may arise from the interaction of electric fields that are created by charged surfaces.

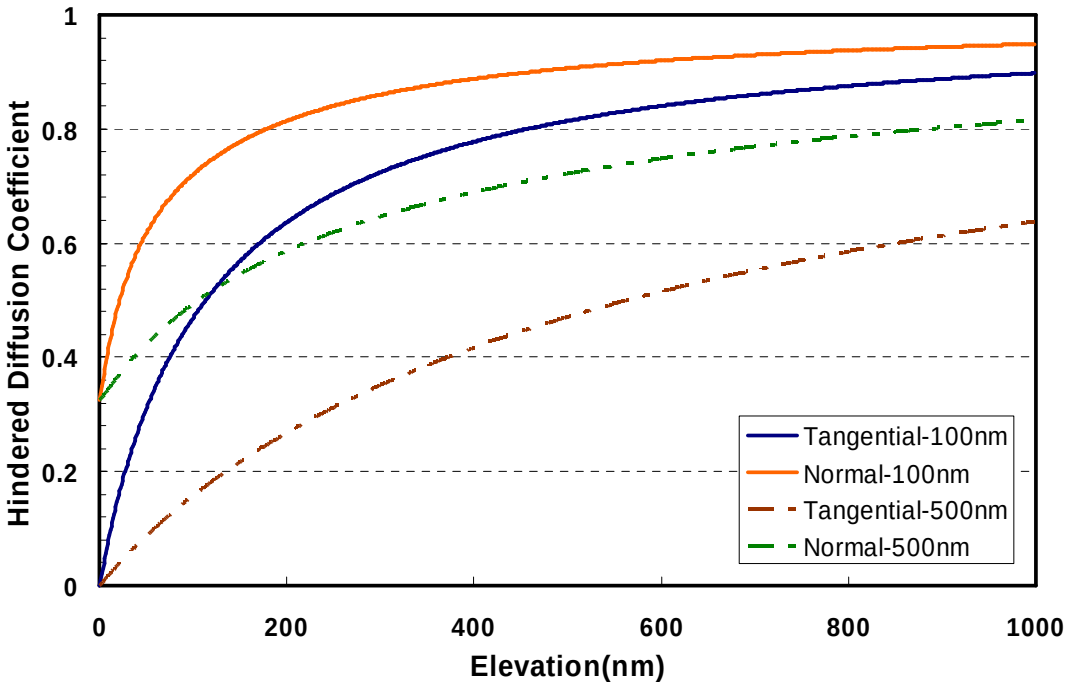


Figure 2.5 Normal and tangential hindered diffusion coefficients for 100 nm and 500 nm radii particles. The hindered effects for larger particles are experienced much further from the wall than for smaller particles.

An electric double layer, as schematically illustrated in Figure 2.6, is formed when a charged particle is placed in a solution containing ions of opposite charge. These oppositely charged ions form a very tightly packed layer directly next to the charged particle called the Stern layer, and outside of the Stern layer is the loosely bound diffuse layer. It is these two regions that give rise to the name electric double layer [54]. The thickness of the electric double layer is given by the Debye length which is defined by equation (9):

$$\kappa = \sqrt{\frac{\varepsilon RT}{2q^2F^2C}} \qquad (9)$$

where ε is the permittivity, R is the gas constant, T is the absolute temperature, q is the protonic charge, F is Farraday's constant, and C is the concentration of ions [55].

For the case of a glass-water interface a negative surface charge is formed through the dissociation of the silanol group of the glass surface [56]. If glass is placed in a solution containing Sodium Chloride (NaCl), and assuming there are no other ions present, the positive sodium ions will form a double layer at the glass surface. For experiments conducted in this work negatively charged particles were used, therefore as they approached the glass surface a repulsive force was present as the double layers of the surface and particles began to overlap. By increasing the concentration of ions, in this case by adding NaCl to the solution, the double layers were reduced and the electrostatic force, at a given elevation, was also reduced. Thus at lower sodium concentration greater hindrance should be expected at a given elevation.

2.3 Experimental Methods

This section will provide details of the imaging system, how samples were prepared, and the parameters used for data acquisition.

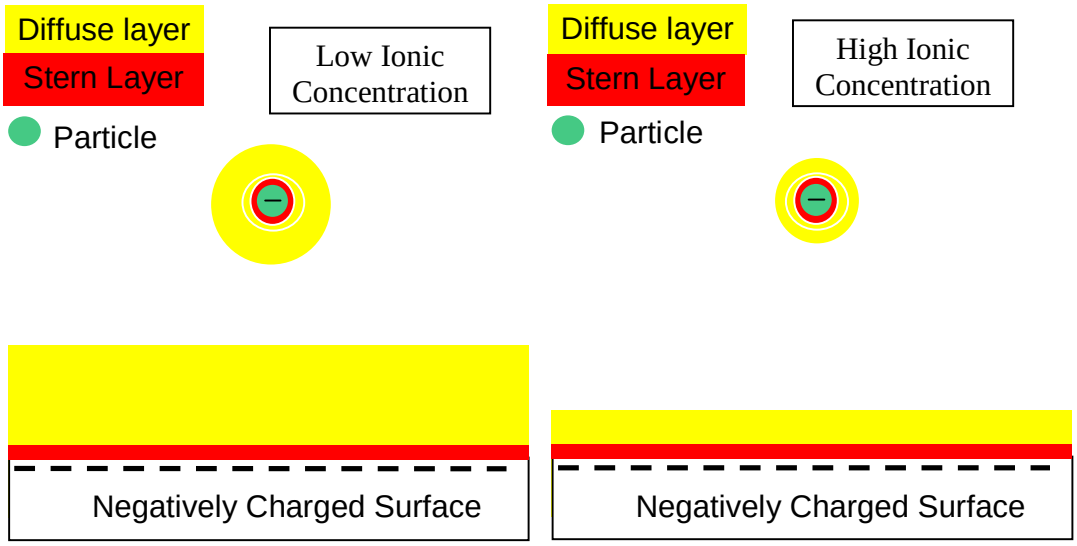


Figure 2.6 Effect of adding ions to the electric double layer. As the ionic concentration is increased, the Debye length is reduced, which in turn reduces the elevations over which the double layer forces are felt. This infers there should be less hindrance to the motion of a particle, at a given elevation, for higher ionic concentrations.

2.3.1 Imaging System

As stated previously, an objective based TIRFM system was used in all experiments. This system consisted of an Olympus IX-50 microscope equipped with a specially designed 1.45 numerical aperture 60x TIRF objective. A Melles Griot 10 mW Argon-ion laser provided 488 nm wavelength (blue) light to the system through a fiber optic cable. The IX-50 was equipped with an illumination kit, specifically design to precisely control the incident angle of the laser. A dichroic mirror was inserted prior to the objective that reflected the laser but allowed the 505-515 nm wavelength (green) emitted light from the fluorescent particles to pass to the CCD camera. A 14-bit electron multiplier (EM) Hamamatsu camera, having both features of a cooled and intensified-CCD, was used for recording all images. Also included in the emission light path were a 0.3x lens and a 5x PE lens providing a total magnification of 90x. Figure 2.7 shows an image taken of this system.

Based on the refractive indices of the glass coverslip and water (1.515 and 1.33), the critical angle for total internal reflection was calculated to be 61.38^{o}. An angle of 62.5^{o}, which provided a penetration depth of 202 nm, was chosen for tests examining changes in MSD values for different particle sizes at constant NaCl concentrations. An angle of 62.0^{o}, which provided a penetration depth of 272 nm, was chosen for tests examining changes in MSD values for a single particle size and differing NaCl concentrations. This relatively deep penetration depth was chosen so that particles could be tracked as far from the glass surface as possible. It should be noted that the penetration depth is simply a reference value and does not prevent measurements further into the sample. Because of the high sensitivity of our CCD camera measurements were made, for the larger penetration depth, up to approximately 750 nm from the surface.

2.3.2 Sample Preparation and Data Acquisition

100 nm, 250 nm, and 500 nm radii fluorescent polystyrene spheres were purchased from Molecular Probes Inc. and placed in deionized water containing 10 mM of NaCl. The 250 nm particles were also tested in deionized water containing NaCl concentrations of .5 mM and 1 mM in order to examine the effect of varying salinities.

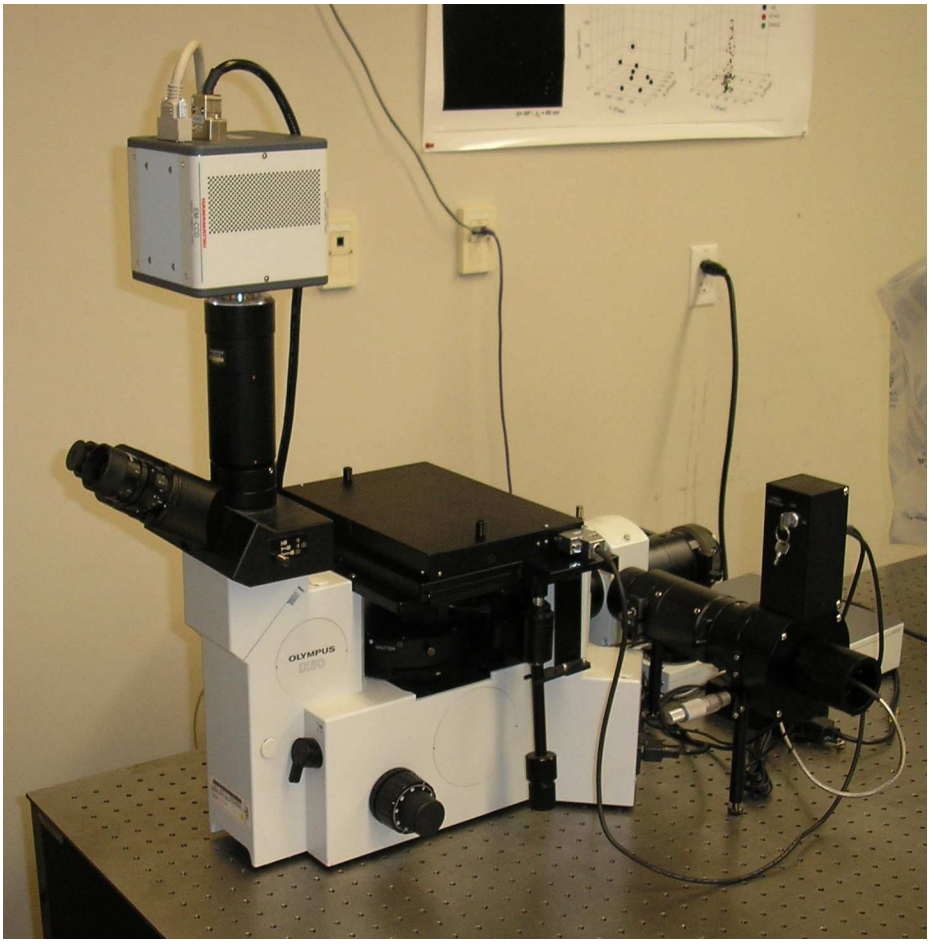


Figure 2.7 Picture of TIRFM imaging system.

Table 2.1 Sample Preparation Conditions for Hindered Brownian Motion Experiments

Particle radius (nm)	100	250	500
Volume concentration (%)	0.00016	0.0019	0.0016
Number of particle /μl	362059	275165	28965
Average distance (μm)	14.0	15. 4	32.6
Surface Charge Density (mC/cm^2)	0.02655	0.12447	0.03247

Before adding the particles the solution was filtered through a 200 nm pore syringe-driven filter to remove any possible impurities. The 100 nm, 250 nm, and 500 nm particles were diluted such that the average distance between particles was 140-, 61-, and 65-times their radius, respectively. This distance insured that possible particle-particle interactions would remain negligible during the experiments [54]. Table 2.1 provides details of the sample preparation including the charge densities of the particle provided by the supplier.

No 1.5 glass coverslips and plastic wells were rinsed with deionized water and allowed to air dry. It was necessary to use 1.5 coverslips for TIRFM due to the small working distance of the objective. Three wells were then epoxied onto the coverslips and allowed to dry for 15 minutes. During testing approximately 2 mL of the desired particle solution were placed inside the well and visual checks were made to insure there was no leakage before testing began.

Moving particles were imaged for a total of 1 minute. The exposure time of the camera was set at 2 ms for 100 nm and 250 nm particles and 20 ms for 500 nm particles, which provided frame rates of 55.8 and 47.7 frames per second (fps), respectively. Shorter expose times were needed for smaller particles to prevent streaking as their movements were significantly faster than for larger particles. In order to acquire a stronger signal the sensitivity of the camera was increased for smaller expose times. This

tended to add more fluctuation in the background noise but did not change the average background values. The handling of noise in these experiments will be discussed in more detail in section 2.4.2.

2.4 Digital Image Processing

In order to get reliable statistics for diffusion coefficients it was necessary to take large numbers of images for each particle size. As stated before the maximum frame rate was set at 55.8 fps, and the total imaging time was set at 60 seconds yielding a total of 3348 frames per test. Furthermore up to 15 tests were conducted on a single particle size so it was necessary to create image processing software capable of dealing with the large amount of images collected.

Along with tracking software, secondary codes were written to deal with background issues and uneven illumination created by the laser. Further codes calculated the average value of stuck particles to use for normal motion calculations, as well as examine the effects of photobleaching on the fluorescent material. A copy of these codes is contained in Appendix A. The next several sections will discuss all of the work completed to insure that the most accurate measurements and calculations possible were obtained.

2.4.1 Photobleaching

Photobleaching is a phenomenon experienced by fluorescent materials in which the emission intensity decreases with continuous excitation [57]. The rate at which a specific material photobleaches, is directly related to the excitation power, and varies substantially between different fluorescent materials. Because the ratiometric method discussed in section 2.1.2, relates a particle's intensity to its elevation above a surface it is imperative to reduce photobleaching in order to obtain accurate tracking data.

Experiments were conducted using both the smallest (100 nm) and largest (500 nm) radii particles to examine the rate of photobleaching under four different excitation powers. Particles were placed in deionized water containing 100 mM of NaCl. A large concentration of NaCl was used in order to substantially reduce the double layer effect

which allowed the particles to become attached to the surface. Next the particles were illuminated and imaged for ninety seconds at the highest laser setting. The stage was then moved and the experiment repeated until data for at least forty particles had been collected. This process was then repeated using neutral density filters that reduced excitation powers to 25%, 5%, and 3% of the maximum excitation power.

Figure 2.8 provides the results for both particle sizes and all excitation powers. It is obvious that the lower the excitation power the slower the rate of photobleaching. For the highest excitation power both particles sizes considered lost almost 60% of their original intensity over the ninety seconds they were imaged. At the lowest setting this change was reduced to less than ten percent. It should be noted that these particles were illuminated under non-TIRF conditions; therefore the excitation power presented here represents a worst case scenario. Under TIRF conditions the illumination power decreases exponentially away from the surface, thus only the very bottom of a stuck particle would experience this relatively high excitation power.

Based on these results the total time selected for moving particle imaging was set at sixty seconds and a neutral density filter, which reduced the power to 3% of the total possible power, was used. Again it is worth noting that the moving particles were rarely exposed to even modest durations of high excitation power over the sixty seconds of testing, as they tended to move in and out of the evanescent wave periodically.

2.4.2 Intensity Variations and Dark Noise Corrections

As stated in the previous section, due to the fact that the ratiometric method was used to track particles normal to the surface it was essential to obtain the most accurate intensity data possible. Two areas that could cause bias to the measured intensities were uneven illumination caused by the laser, and dark noise from the camera. Corrections for both of these situations, which will be discussed in detail in the following paragraphs, were implemented during testing.

In order to observe the unevenness of the illuminated area, a fluorescent die, that matched the emission wavelength of the fluorescent particles, was added to deionized water. Approximately 2 mL of the solution was then place in a well on a glass coverslip

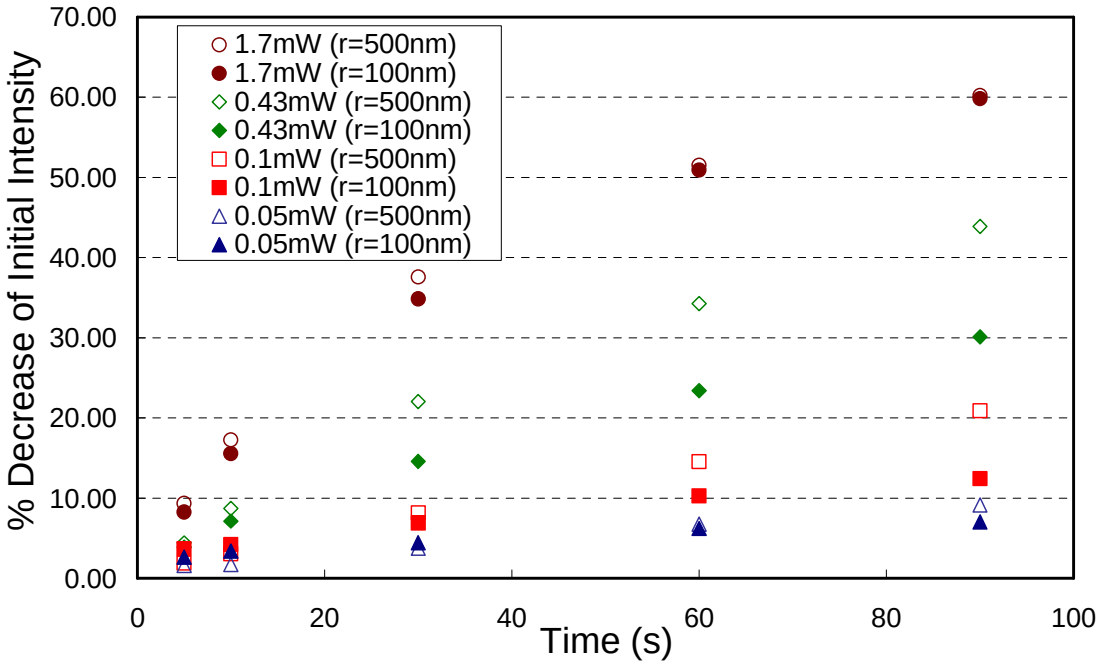
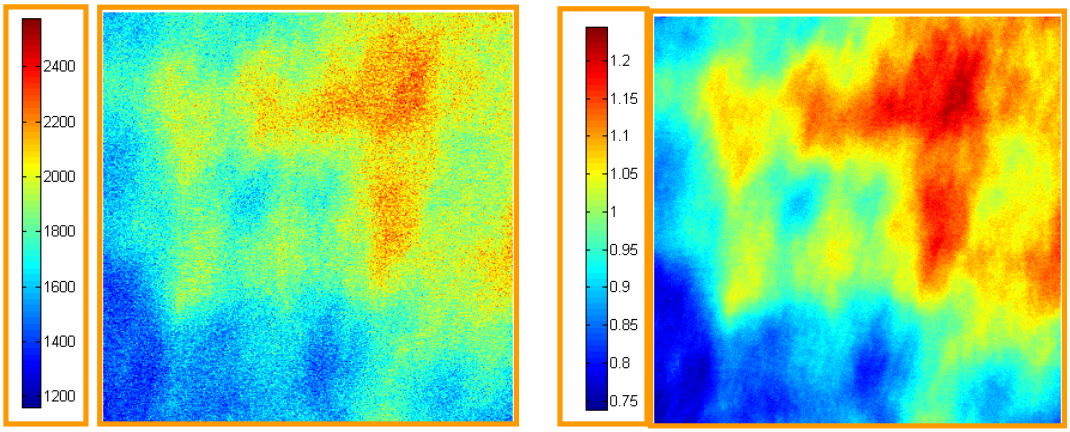


Figure 2.8 Photobleaching rates for various excitation intensities for 100 nm (solid symbols) and 500nm (regular symbols) radius fluorescent particles. The inset caption illustrates the illumination powers.

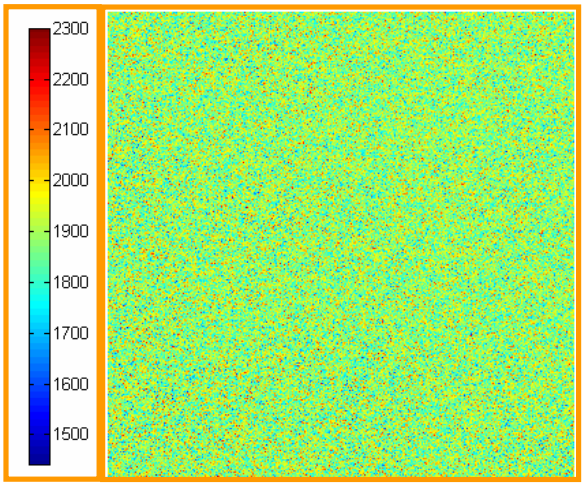
and the illumination conditions were set to match those in the experiment for moving particles. The camera settings were adjusted to prevent saturation as the amount of die used provided a substantially higher signal than the nanoparticles. Movies were then taken for approximately 1 second at a frame rate of 58.5 frames per second. The stage was then moved and the process was repeated two times in order to help ensure that there was no local biasing. It should be noted that this process was repeated for each experiment in order to insure any changes in the illumination zone were accounted for.

A simple digital image processing algorithm was created to combine all images taken, of the fluorescent die, into a single normalized map that could be applied to moving particle images. Figure 2.9 provides the results from this work. Figure 2.9 (a) shows a single image taken of the fluorescent die, while figure 2.9 (b) shows the map created from hundreds of images. Figure 2.9(c) shows the effect of dividing a single image by the map. While it is appears that this image has been substantially corrected, close observation reveals that there is still some variation in intensity throughout. It is believed that these variations are temporal and are caused by the random movement of the fluorescent molecules themselves. Because hundreds of images are averaged these random fluctuation should be accounted for. For nanoparticles the fluorescent material is attached to the particles and thus this type of temporal variation should not be expected.

Dark noise, or electrical noise, is produced when transmitting a signal through an electrical device. For images taken in these experiments, dark noise appeared as low grey scale values. In order to account for this noise, the camera was first shuttered so that no external light reached the CCD, and then images over 5 seconds were taken. These images were averaged, pixel by pixel, to create a map that could be subtracted from images of moving particles. It was determined that the average noise produced was equal to approximately 450 grey scale value, while the maximum grey scale value for the 14 bit camera was equal to 16384. Thus the dark noise is less than 3% of the total possible value.



(a) (b)



(c)

Figure 2.9 Intensity corrections for uneven illumination. (a) A single image of fluorescent die dissolved in water under the same testing conditions used for moving particle measurements. (b) Average of 100 single images, normalized by the average intensity of all images. (c) Effect of dividing (a) by (b). Note no pattern is observed and illumination is almost completely even.

2.4.3 Reference Particle Intensity Determination

In order to determine absolute elevations of moving particles with respect to the coverslip, it was necessary to determine the intensity of the moving particles at the surface (See section 2.1.2). Ideally a moving particle would be brought into contact with the surface in order to compare all intensity changes of a particle with itself. However for nanoparticles this idea is not feasible.

In order to determine an average intensity value for particles at the surface, the particles were placed in deionized water containing 100 mM of NaCl. Again this high concentration of NaCl reduced the electric double layer forces and allowed the particles to become attached to the surface as discussed in section 2.4.1. The laser was positioned to provide total internal reflection conditions that matched the moving particle measurements. Movies were then taken for three seconds, at which point the stage was repositioned to an area that had not been illuminated, and the process was repeated until at least one hundred stuck particles were captured.

An algorithm was created to make adjustments to the measured stuck particles intensity based on illumination unevenness and background noise. An average stuck particle intensity was then calculated based on all of the images taken. This provided a single stuck particle intensity that was used to calculate absolute elevations of all moving particles of a given size.

2.4.4 Particle Tracking

In order to compare the theoretical values for hindered Brownian Diffusion with measured data it was necessary to calculate the average Mean Square Displacement (MSD) of the particles. The relationship between the three dimensional diffusion of a particle and the mean square displacement is provided in equation 10:

$$MSD \equiv \left\langle \left(X(t) - X(0)\right)^2 \right\rangle + \left\langle \left(Y(t) - Y(0)\right)^2 \right\rangle + \left\langle \left(Z(t) - Z(0)\right)^2 \right\rangle = 6 D_H \Delta t \qquad (10)$$

where X, Y, and Z are the position of a particle, the brackets < > represent an average of the square of the displacement, D_H is the free diffusion coefficient which must be adjusted by the appropriate hindered coefficient, and delta t is the time step between

successive measurements [22]. For one and two dimensional values the constant 6 is replaced by 2 and 4 respectively.

As mentioned previously it was necessary to obtain large data sets to insure that the statistical methods were valid. Therefore a particle tracking algorithm, named the Brownian Motion Code (BMC), was created using Matlab software, to measure particle displacements and calculate the MSD values. A flow chart of this code is provided in figure 2.10.

Moving particle images were captured using Wasabi software and stored in a proprietary movie format. These movies were then segmented into individual Tagged Image Format Files (TIFF) that could be read into Matlab. In order to track individual particles small windows were selected that isolated individual particles from the rest of the image as shown in figure 2.11. Each window was examined for all frames to insure that only a single particle was being tracked at a given time. This prevented the possibility of measuring displacements between two different particles over consecutive frames.

The brightest pixel of a particle was used to track its motion both tangentially and normally to the surface. Tangential distances were based on the number of pixels traveled between consecutive images. A single pixel size can be calculated based on the magnification and the size of the CCD chip using equation 11:

$$pixsize = Chip_Size / M \qquad (11)$$

where the chip size for the camera is equal to 8 μm and M is the total magnification (90 for these experiments).

The normal motion was calculated, as discussed previously, using the ratiometric method with section 2.4.3 providing the details as to how a single reference intensity was calculated experimentally. This value was then multiplied by the intensity variation map and the dark noise was subtracted to provide a local reference intensity at every pixel. No adjustments were then needed for the moving particle intensities other than a subtraction of the dark noise.

Because the hindered diffusion coefficient is elevation dependant it was necessary to divide the displacement data into appropriate elevations. 25 nm elevations bins were

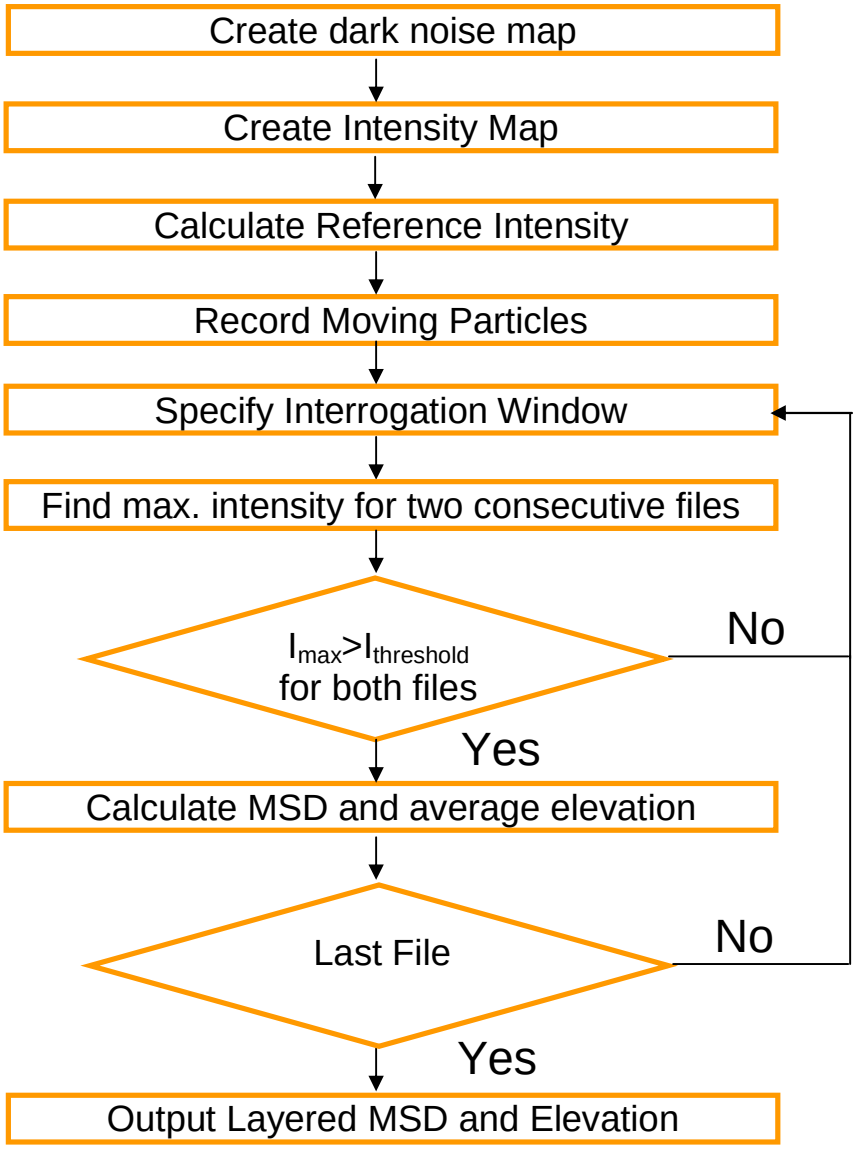
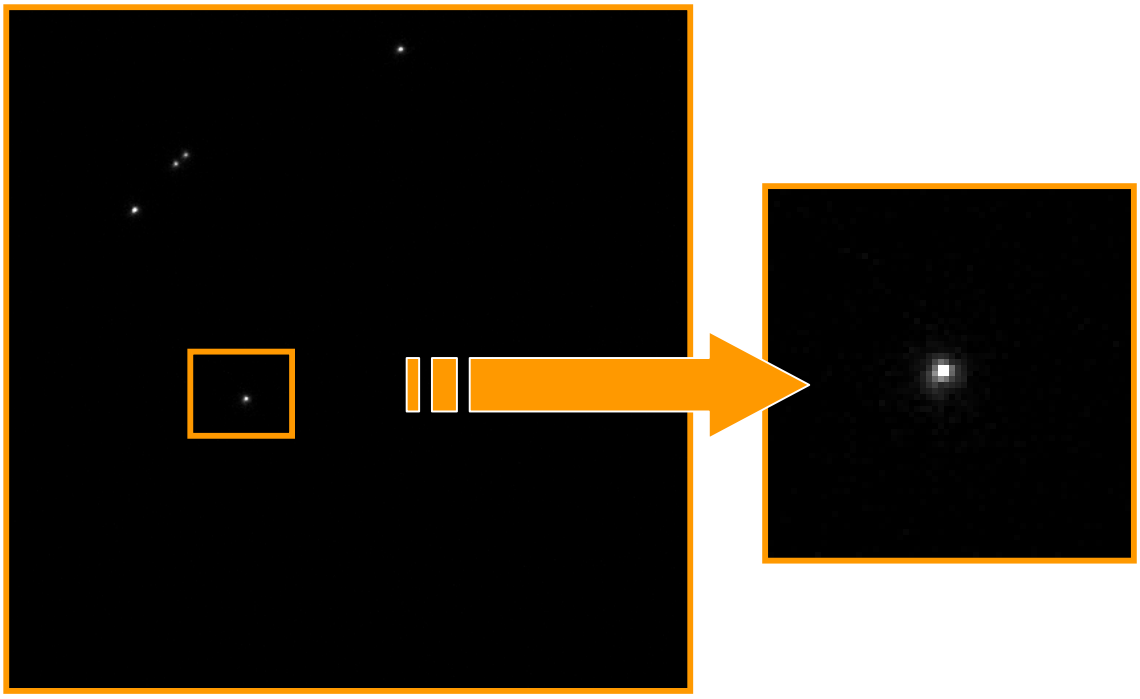


Figure 2.10 Flow chart for the Brownian Motion Code (BMC). This algorithm was developed to track particle motion and calculate MSD values both normally and tangentially to the surface.



(a) (b)

Figure 2.11 (a) Initial frame. (b) Selected window. Individual windows were chosen from the initial images in order to track single particles. Each window was examined over all frames to insure only one particle was being tracked at a given

chosen up to a maximum of 500 nm from the surface. The average elevation, calculated between two consecutive measurements, was used to place the normal and tangential displacement values into the appropriate bins. After all particles had been tracked an average MSD value, both normal and tangential to the surface, was calculated for each elevation bin.

2.5 Results

The following two sections provide detailed results for hindered motion both normal and tangential to the surface. Three particle sizes were examined, and for the 250 nm particles, three concentrations of NaCl were tested. These results were then compared with existing theories by Goldman and Brenner, as discussed in section 2.2.1. time. This prevented the possibility of mistaking displacement measurements with distance between particles.

2.5.1 Hindered Tangential Diffusion

Figure 2.12 shows both experimental and theoretical tangential MSD values for three different particle sizes with each data point representing an average of at least 80 MSD values. It is obvious that the experimental values follow the theoretical predictions very closely for all three cases. For these results the concentration of NaCl was held constant at a value of 10 mM.

Figure 2.13 provides MSD values for a single particle size, 250 nm, at three concentrations of NaCl. The reason for testing at different concentrations is to observe the effect of electrostatic forces, which are directly related to the amount of ions in the solution (see section 2.2.2). Two observations can be made from figure 2.13. First, for the highest concentration of NaCl (10 mM) enough measurements were made to obtain average values within 25 nm from the surface. For the 1 mM and .5 mM cases the lowest measured values were between 75-100 nm and 225-250 nm respectively. As the electrostatic forces increase, with a decrease in concentration of ions, the particles are not allowed to move as close to the surface as at higher concentration and thus fewer measurements are possible.

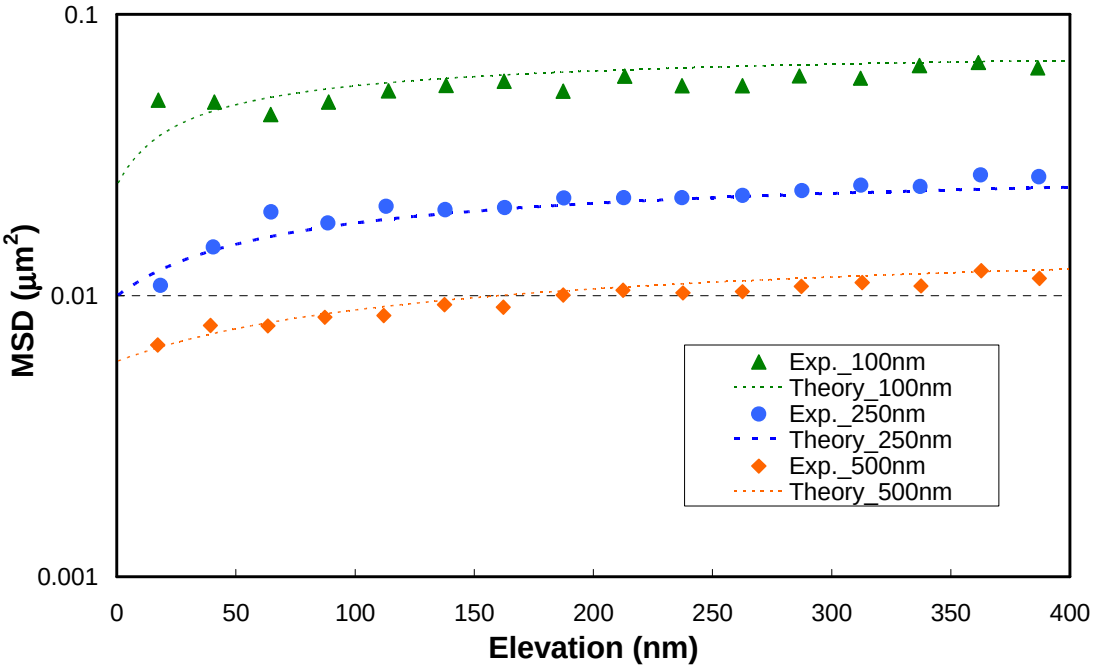


Figure 2.12 Comparison of theoretical and experimental tangential MSD values for three particle sizes. All three particle sizes follow the theoretical values very closely.

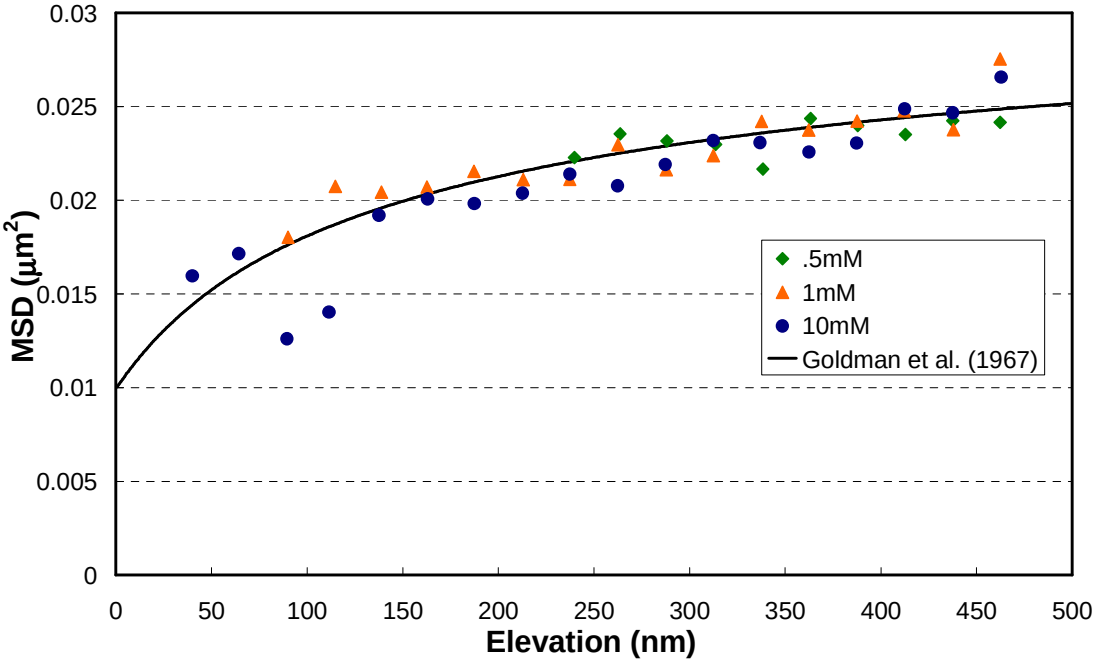


Figure 2.13 Comparison of theoretical and experimental tangential MSD values for 250 nm radii particles at three different concentrations of NaCl. Little change is observed for MSD values at different NaCl concentrations.

The second observation that can be made from Figure 2.13 is that all data, regardless of the concentration of NaCl, follow the theoretical predictions very closely. While the lower concentrations of NaCl may prevent the particles from approaching the surface as closely as higher concentrations it does not appear to change its tangential motion at a given elevation. This should be expected as the electrostatic forces act normally to the surface. However if a gradient in the charge concentration on the surface of the coverslip existed then changes in the tangential motion would be expected. This however did not appear to be the case in these experiments.

It should be noted that two different maximum elevations are presented on the independent axis in Figures 2.12 and 2.13. For Figure 2.12 the relatively high concentration of NaCl, used for all particle sizes, allowed the particles to approach more closely and thus sufficient data was collected very near the surface. Therefore it was possible to use a higher incident angle, providing a shallower penetration depth, but also providing more accurate measurements as will be discussed shortly. For the lower values of NaCl the particles tended to stay further from the surface and thus a smaller incident angle was used to capture data further from the surface. While this added some amount of uncertainty to the measurements, it was necessary to examine the effects on the changing concentrations.

Before discussing the normal hindered diffusion it is important to cover one more topic with regards to the tangential values. In order to insure that the motion is indeed Brownian it is important to examine the MSD values for multiple time steps. True Brownian motion will appear linear with respect to time step as observed in Figure 2.14 where a single elevation, near the penetration depth, was chosen for examination. It should be noted that other elevations also showed linearity with respect to the time step used to calculate the MSD values.

2.5.1 Hindered Normal Diffusion

Figure 2.15 provides normal MSD values for the three different particle sizes examined at a NaCl concentration of 10 mM. Unlike the tangential case where all particle sizes followed the theory very closely, the normal case shows a significant

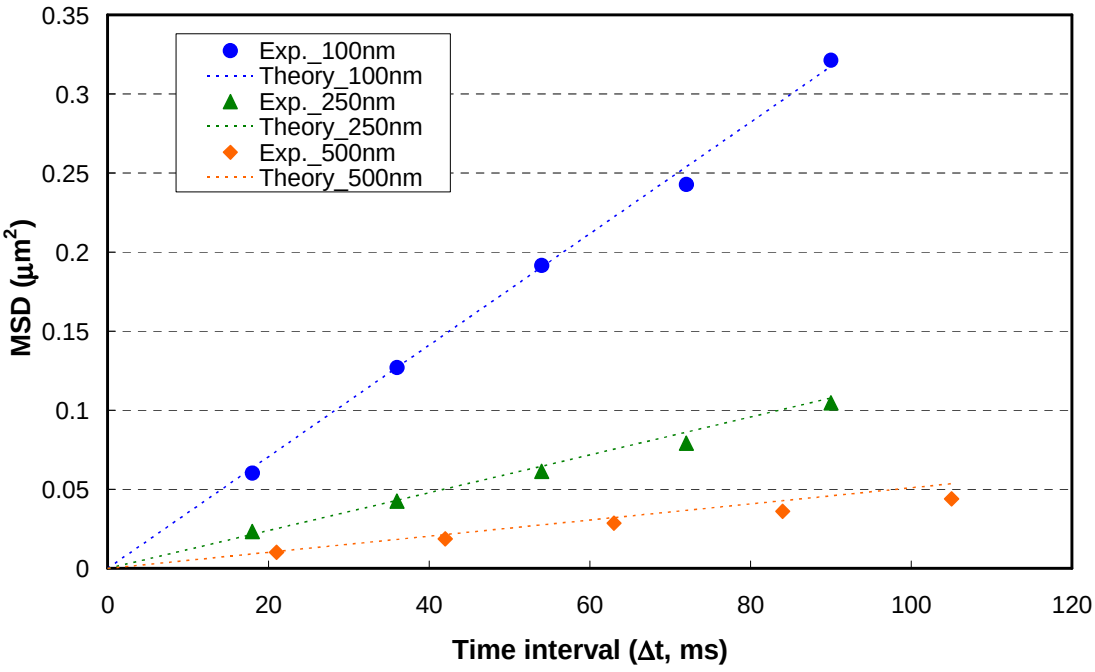


Figure 2.14 Tangential MSD values versus time step. Each data point represents measurements made for a given particle size at an elevation near the penetration depth. The linearity of the MSD values versus time step indicate the Brownian nature of the movement.

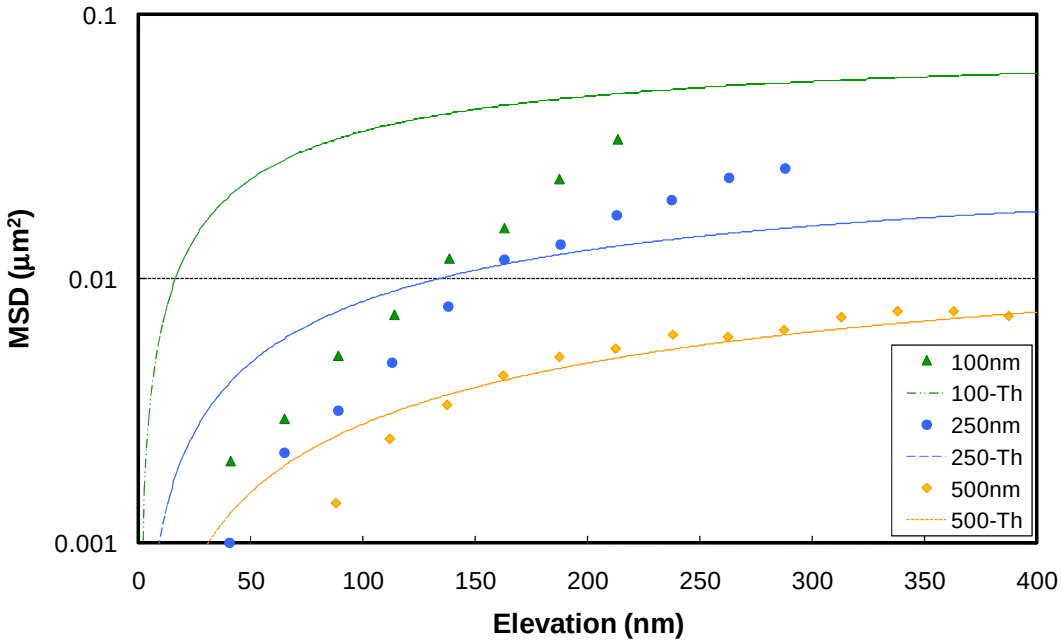


Figure 2.15 Comparison of theoretical and experimental normal MSD values for three particle sizes. Larger particles tend to follow the theoretical values more closely, particularly further from the surface. NaCl concentration is at 10 mM for all particle sizes.

departure from theoretical values. It also appears that deviation from this theory increases with decreasing particle size. The 500 nm particles tend to approach the theory more closely at larger distances from the surface but are still significantly different within approximately 100 nm. It is believed that these deviations occur due to the electrostatic forces and therefore a second series of tests was completed to examine a single particle size under multiple NaCl concentrations.

Figure 2.16 shows normal MSD values for 250 nm particles under NaCl concentrations of .5 mM, 1 mM, and 10 mM. It is observed that as the amount of NaCl decreases, which in turn increases the electrostatic repulsive force, the hindered effect increases close to the surface. It is also observed that the measured values significantly increase beyond the predicted levels at higher elevations. This result was unexpected and will be discussed more in the conclusions section. Figure 2.17 provides normal MSD values versus time step, at several elevations, for 250 nm radii particles at a NaCl concentration of 10mM. Unlike the tangential case the motion is not linear with respect to time step indicating the motion is not truly Brownian.

2.5.3 Measurement Uncertainty

As with all experimental work a certain amount of measurement uncertainty must be considered. For these measurements uncertainty was divided into tangential and normal components. For the tangential motion of the particles the amount of uncertainty was determined to be one half of a pixel or 44.5 nm. For the normal motion the uncertainty is more complicated to predict due to the large number of factors that can alter the measurement.

The three critical factors affecting the uncertainty of a given measurement are accuracy of the micrometer for setting the appropriate incident angle, variation of the intensity of a stuck particle, and variation of intensity of a moving particle. The accuracy of the micrometer causes an uncertainty of $\pm 0.16^{o}$ for the incident angle. It is assumed that the fluctuation of the particle intensity at the surface is approximately the same for moving particles. When checking for stuck particle intensity an average standard

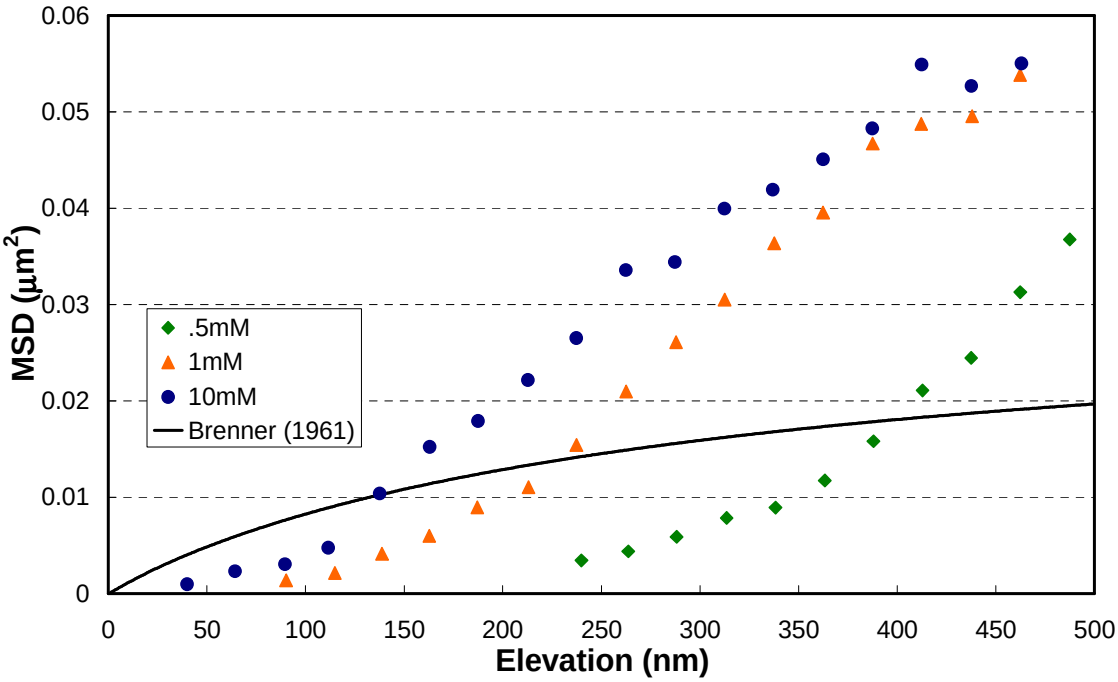


Figure 2.16 Comparison of theoretical and experimental normal MSD values for 250 nm particles at NaCl concentrations of .5, 1, and 10 mM. As the amount of NaCl decreases, the hindered effect very near the surface increases.

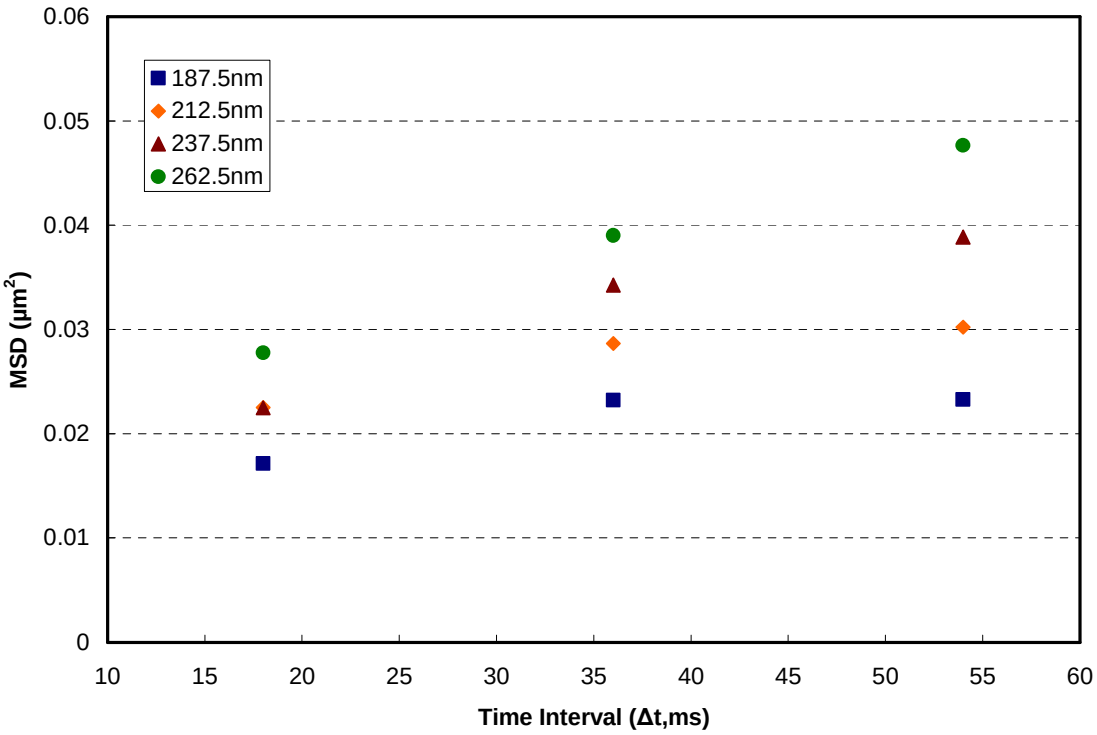


Figure 2.17 250 nm radii particle motion is not linear with respect to time step, evidence that the motion is not Brownian. NaCl concentration is at 10 mM.

deviation was calculated and this value was used to calculate the uncertainty for both moving and stuck particles.

In order to calculate the total uncertainty the single point detection analysis developed by Kline and McClintock was used on equation 5 to produce equation 12:

$$w_{\Delta z} = \pm\left[\left(\frac{\partial \Delta z}{\partial z_p} w_{z_p}\right)^2 + \left(\frac{\partial \Delta z}{\partial I_{stuck}} w_{I_{stuck}}\right)^2 + \left(\frac{\partial \Delta z}{\partial I_{moving}} w_{I_{moving}}\right)^2\right]^{\frac{1}{2}} \qquad (12)$$

where w_{z_p} is ±14.2 nm, $w_{I_{stuck}}$ and $w_{I_{moving}}$ are 1441, 1305, and 927 for 100 nm, 250 nm, and 500 nm radii particles, respectively [58].

Figure 2.18 provides the calculated uncertainty versus elevation for all three particles sizes. In order to calculate elevations a natural log of the ratio of the moving particle to the stuck particle intensity is multiplied by the penetration depth, z_p. Therefore any uncertainty in z_p is magnified as the elevation increases. This is the cause for the increase in uncertainty with increase in elevation. The larger particles have a smaller amount of uncertainty at the surface than the smaller particles because of the smaller deviation in intensity. This is most likely due to the fact that the gain is decreased for the larger particles which in turn decreased the standard deviation of the background noise. It should be noted that Figure 2.18 is for a penetration depth of 202 nm. Obviously for the multiple ionic concentration studies, where the penetration depth was increased, the uncertainty will also increase.

2.6 Conclusions and Future Work

The tangential and normal mean square displacement of 100 nm, 250 nm, and 500 nm fluorescent particles were examined using TIRF microscopy at a NaCl concentration of 10 mM. Following this work the 250 nm particles were selected for examination at .5 mM, 1 mM, and 10 mM to determine the effect of the electrostatic forces on the particles motion.

As expected the addition of NaCl allowed particles to approach the surface more closely due to the reduction in electrostatic forces caused by a screening of the double

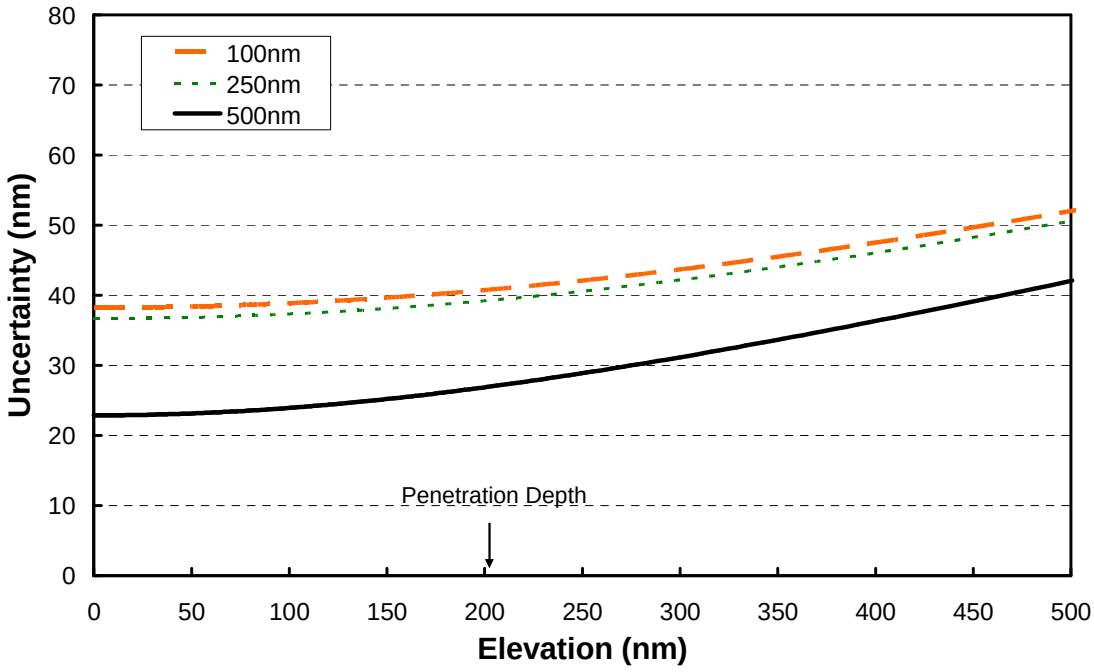


Figure 2.18 Uncertainty in elevation measurements due to micrometer accuracy and intensity fluctuations for moving and stuck particles.

layer. Tangential mean square displacements for all particle sizes showed excellent correlation with the theory predicted by Goldman. While this theory has been examined and correlated with larger particles, it is believed that this work represents the first time it has been examined at different elevations for nanoparticles.

Normal MSD values showed substantial deviation from theoretical values, particularly for the smaller particles. However, as noted in the discussion on Brenner's theory for normal hindered diffusion, only hydrodynamic effects were considered. This is important as it should not be inferred that the theory itself is incomplete in any way. Rather an additional term may need to be added to the hindered coefficient to correctly predict the electrostatic forces that are present. It should also be noted that inertial effects may negate some of the electrostatic forces as the larger particles followed the theory more closely.

Proposed future work would be to attempt to match the existing theory for normal hydrodynamically hindered diffusion with non-charged nanoparticles where electrostatic forces are no longer present. This could also be conducted by eliminating the charge on the coverslip, however completely eliminating the charge from either surface is not a trivial matter.

Further proposed work would be to develop theory capable of predicting the normal hindered diffusion of charged nanoparticles. It would also be important to conduct more experiments for different particle sizes to correlate the theory appropriately. One difficulty with this is the rapid movement of smaller particles. For this work the 250 nm particles were chosen because they tended to deviate from the theory more than the larger particles while at the same time remaining in the viewing zone long enough to get appropriate statistics. In order to measure the 100 nm particles a faster camera would be required or a more viscous fluid could be used. However viscosity changes increase the inertial terms in which case the motion might appear similar to that observed for the larger particles, where the motion tended to deviate from the theory by smaller amounts.

CHAPTER III DIGITAL IMAGE PROCESSING OF LIVE CELLS

Before beginning work on intracellular vesicle trafficking, it was necessary to become familiar with cell handling techniques. This included learning sterile practices to prevent contamination, changing cellular medium during cell growth, splitting and transferring cells after they had reached confluence, and several other techniques involved with maintaining healthy cell lines. During this transition period, digital image processing (DIP) software was developed as part of two separate cellular studies being conducted in the Micro Nano Scale Fluids and Energy Transport (MiNSFET) Lab, by Dr Changkyoung Choi. This chapter will provide a brief background on the goal of these two experiments with the main focus on the DIP codes and the metrics they provided for analysis of the experimental data.

3.1 Differential Interference Contrast Microscopy Image Analysis

Differential Interference Contrast Microscopy (DICM) is an effective technique which provides a pseudo three dimensional profile of a transparent surface without the help of a marking agent such as a fluorescent die. The goal of this work was to create software capable of automatically calculating the coverage area of cells over time by processing images created using DICM. The purpose of these experiments, the details of a DICM system, and an explanation of the DIP algorithm will be covered in the next several sections. Finally the results and conclusions made from this work are provided.

3.1.1 Purpose of Experiment

In order to study the cellular barrier function electrical impedance experiments, developed by Giaver and Keese [59,60], have been conducted to study the change in impedance as cells are grown on an electrically conductive material. However because impedance changes are a function of cell coverage area, cell-substrate adhesion, and cell-cell junctions it is impossible to use electrical measurements alone to isolate what is creating a change in the impedance values. Because most conductive materials are highly

reflective it has been difficult to couple these experiments with microscopy techniques capable of examining changes in cell morphology, coverage area, and cell-cell contact. Therefore it was important to find a material capable of both electrical conduction and optical transmission to gain a better understanding of the nature of impedance changes.

Indium-Tin Oxide (ITO) was chosen as a material capable of meeting both of these requirements [61]. A very thin patterned layer of ITO was coated on a glass coverslip as shown in figure 3.1. A well was then placed around the anode and cells were allowed to grow over time while both electrical impedance measurements and DICM images were recorded. The images were then analyzed to help determine the causes for changes in impedance over time [62].

3.1.2 Microscopy Technique

DICM is a relatively complicated microscopy technique which translates optical path differences between two adjacent rays of light into intensity variations that provide a pseudo three dimensional effect [63]. Figure 3.2 provides a detailed schematic of the light path and optical components necessary for DICM imaging.

Unpolarized white light, provided by a halogen lamp, is passed through a 45^{o} polarizer. The light is then passed through a Nomarski prism, which separates the beam into two components that are normal with respect to each other. Next the two beams



Figure 3.1 Indium-Tin Oxide coated coverglass. ITO is both electrically conductive and optically transparent. These properties make it an excellent choice for conducting optical and electrical measurements simultaneously.

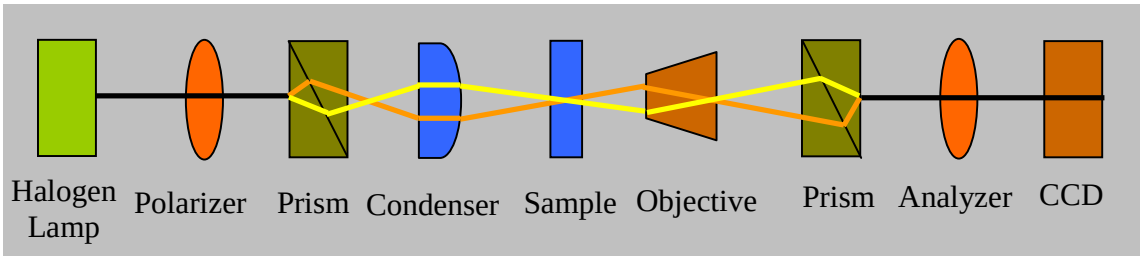


Figure 3.2 Schematic of light path and optical components necessary for DICM imaging. White light from a halogen lamp is polarized before entering the Nomarski prism. It is then split into two components, normal to each other, and is passed through the condenser and into the sample. The two beams pass through the sample approximately 200 nm apart from each other. The beams then travel through the objective and are recombined into a single beam as they pass through the second Nomarski prism. They are then passed through the analyzer to remove any extraneous light and finally inter the CCD for image capturing.

proceed through a condenser and are directed through the sample at a distance of approximately 200 nm apart from each other. The beam then passes through the objective and a second Nomarski prism where they are recombined into a single polarized wave. Finally the beam passes through an analyzer that removes any extraneous light before entering the CCD camera.

Because the two beams of light do not pass through the exact same location of the sample there is a high probability that they will not traverse the same optical path length. The optical path length is defined in Hecht as "the distance in vacuum equivalent to the distance traversed in the medium of index n" and is given in equation 1:

$$OPL = \int_{s_i}^{s_0} n(s)ds \qquad (1)$$

where OPL stands for the optical path length, s is the physical distance traveled by the beam through medium s, and n is the index of refraction of the medium which may vary with location [52]. If the sample does not have a constant thickness, or there are local variations in the index of refraction, then the two beams will have different optical path lengths. When the two beams are recombined in the second Nomarski prism the change in OPL will create interference as the two beams will no longer be in phase. This change in phase is observed by a change in the intensity of the light throughout the cell. Figure 3.3 is a DICM image of an endothelial cell grown on a glass coverslip.

Changes in intensity are most apparent along the optical axis which runs at an angle of 45^o to the image shown in figure 3.3. Moving from the bottom right to the top left, an increase in the thickness of the cell provides an increase in intensity, assuming little or no change in the index of refraction, while a decrease in cell thickness is apparent by a decrease in intensity. This is a general property of DICM microscopy. It is possible to enhance the intensity variations by changing the optics such that the adjacent beams of light proceed through the sample at further distances from each other. However this method decreases the resolution of the sample and will provide less detail about changes in the cell.

It is important to note at this point that the intensity variations cannot be used to quantify the thickness or morphology of the cell. The local index of refraction within a

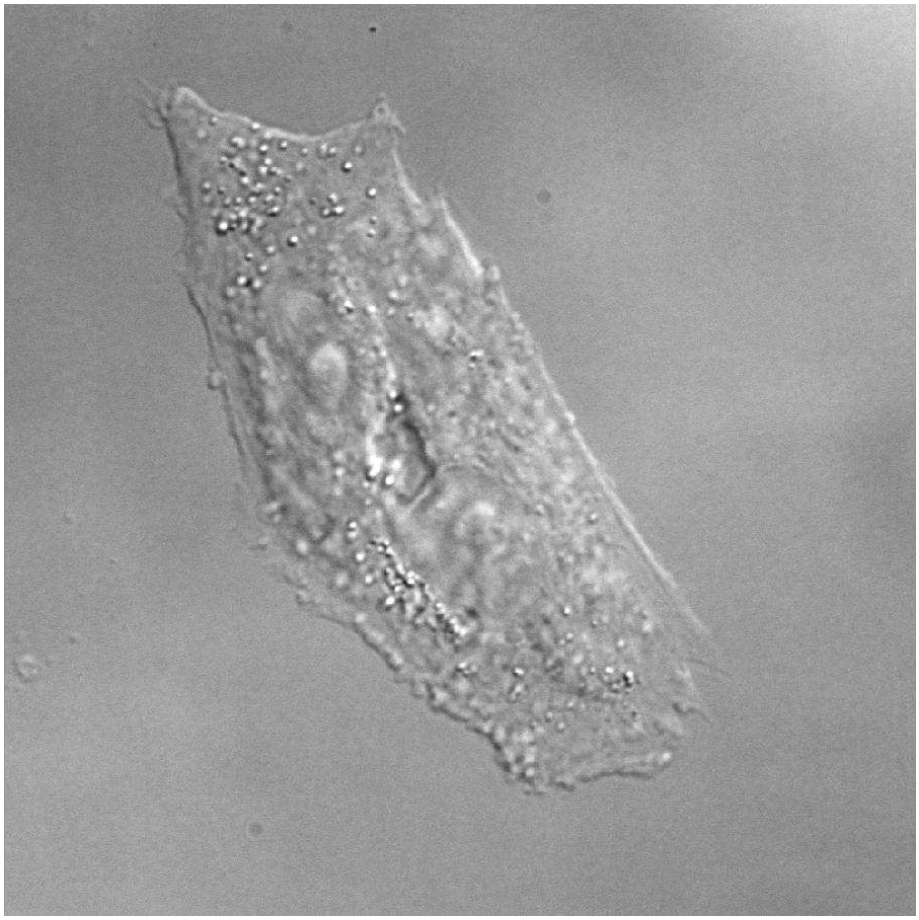


Figure 3.3 DICM image of an endothelial cell. Note variations in the intensity are created by the change in phase between the two beams of light passing through adjacent sections of the sample.

cell can be changed by cellular structures such as the nucleus and intracellular vesicles thus the information contained in these variations are not a function of the cell thickness alone. However it is possible to quantify the coverage area of the cell as the outline is very distinguishable due to the large change in OPL from the surrounding medium to the cell itself. The following section will provide the details of software created to examine the change in coverage area over time.

3.1.3 Algorithm

Up to one hundred and forty images were collected for experiments conducted to examine the spreading of endothelial cells on an ITO electrode. Therefore it was desirable to develop software capable of automatically calculating their coverage. Figure 3.4 provides a flow chart for the algorithm developed for these experiments using Matlab software. Individual TIFF files were read into Matlab and deconvolved using a Lucy-Richardson algorithm with a Gaussian point spread function which provided a sharper edge to the cell boundary [64]. Next six separate filters, which will be discussed in detail below, were applied to the image to either remove the background or reinsert parts of the cell that had been removed from a previous filter. Finally an overlay image was created to quickly observe the effectiveness of the procedure on separating the cell covered area from the background.

After deconvolving an image, which can be observed in figure 3.5(b), it was copied so that two filters could be conducted on the same image independently. One image was passed through an edge detection filter, using either the Canny or Sobel method, which provided a binary output of the initial image [65]. These two methods examine the intensity gradient, or derivative, at every pixel to determine if an edge exists. The Sobel method uses the following approximation to determine the derivative in the x and y directions at every pixel.

$$S_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad S_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} \qquad (2)$$

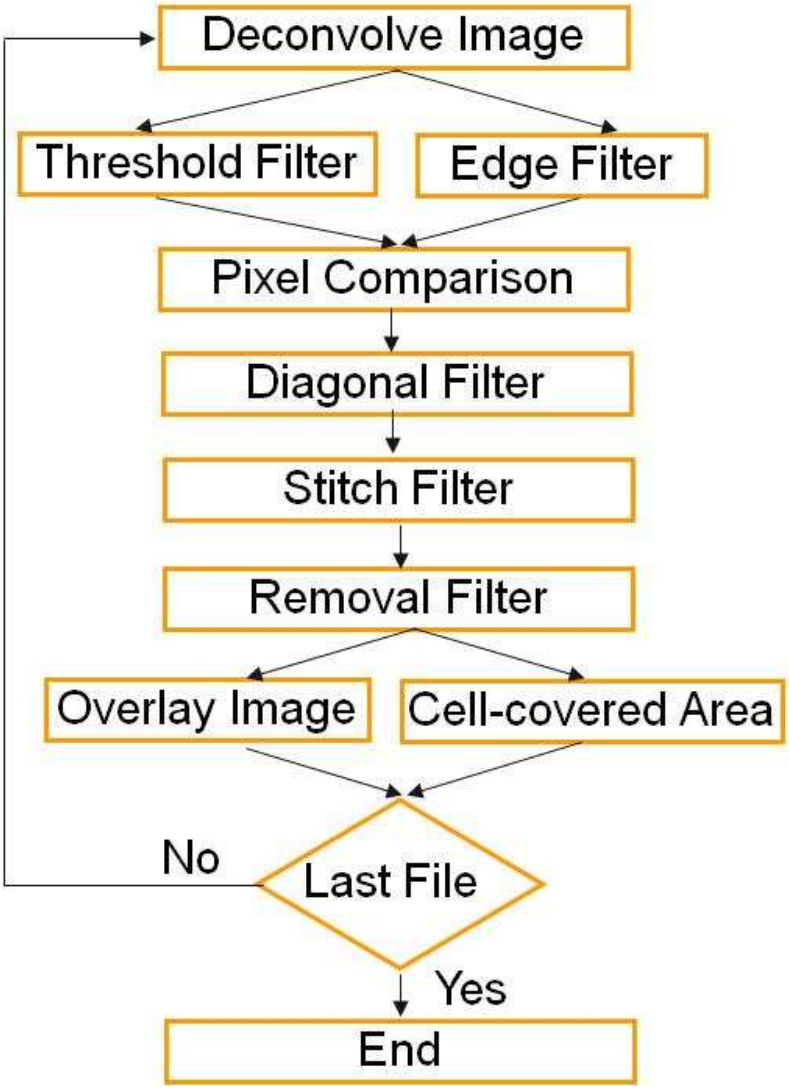


Figure 3.4 DIP algorithm developed to calculate the coverage area of endothelial cells growing on an ITO electrode.

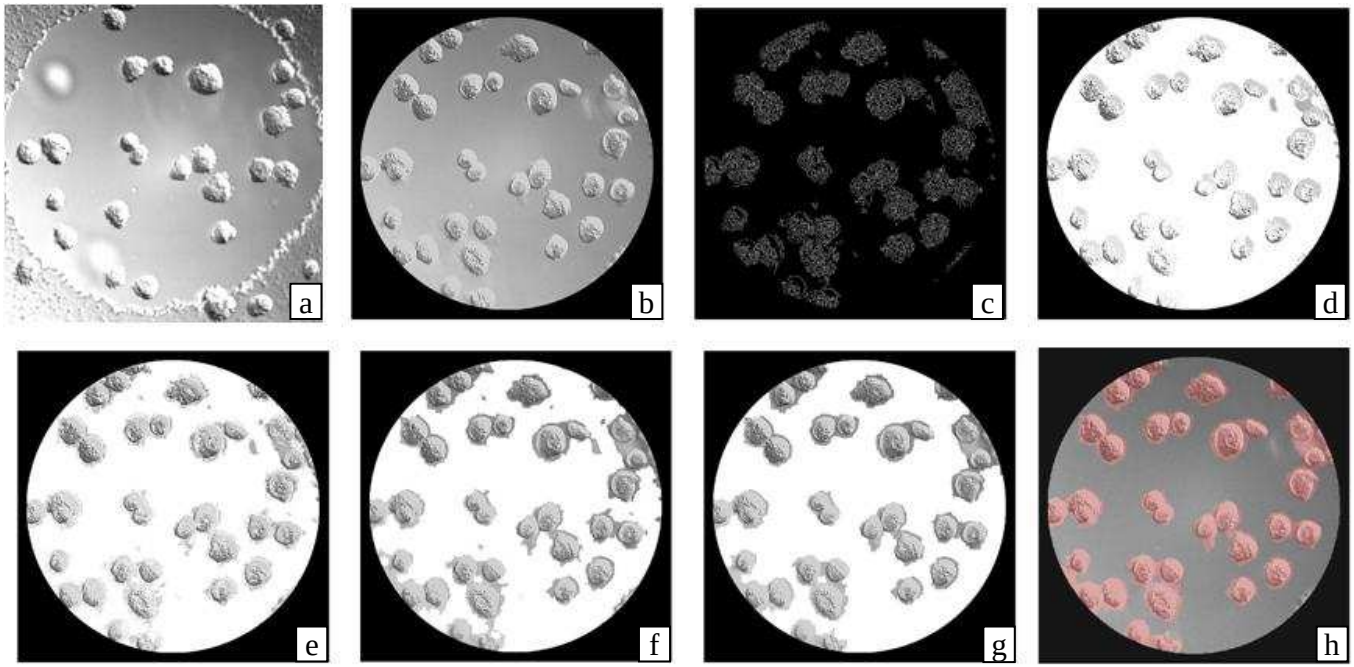


Figure 3.5 Digital image processing results. (a) Original image. (b) Deconvoluted image. (c) Edge detection image. (d) Image after thresholding filter. (e) Image after diagonal filter. (f) Image after stitch filter. (g) Image after removal filter. (h) Overlay image. This displays the pixels determined to be part of the cell coverage area (red) on the original image (grey).

These three by three kernels are multiplied with the pixel being examined and the surrounding pixels, to create a value for the derivative at that point. After applying this method throughout the image (convolution), with adjustments made at the corners, any pixel value above a certain threshold will be considered an edge.

The Canny method is similar to the Sobel filter but takes into account noise using a Gaussian filter. After calculating the derivatives at each pixel two thresholds are used to determine if an edge exists. If the value of the pixel is above the higher threshold then an edge is marked. If the pixel is not above the higher threshold but is above the lower threshold, and is adjacent to a pixel that is marked as an edge, then it is also marked. For some instances these thresholds were calculated by Matlabs preset standards, while in others they were adjusted, by the user, for more accurate edge resolution. It should be noted that due to the complexity of the cells, both internally and at their boundaries, the edge detection methods were helpful but not sufficient to completely determine cell coverage area. Figure 3.5(c) shows the effect of this filter on the original image.

The second filter used on the initial image was a threshold filter. Because some illumination unevenness is present in all experiments the image was divided into nine sections and an average local intensity was calculated. It was assumed that this intensity equaled the intensity of the medium and was used as a course threshold. The image was then passed through the threshold filter where any pixel grey level that was within a percentage above or below the threshold value was removed. The percentage was selected by the user and varied slightly within each region. As mentioned before a general property of DICM is that increases in cell thickness, along the optical axis, produce increases in intensity, while decreases in cell thickness produce decreases in intensity. Therefore moving from one side of the cell to the other, along this axis, will produce an increase and then decrease in intensity. At some point this intensity will be approximately equal to the medium intensity and will therefore be removed by the threshold filter. Thus some of the cell will have to be reinserted for a correct area calculation. Figure 3.5(d) shows the effect of the threshold filter on the original image.

The results from the threshold and edge filters were then combined into a single image, and pixels that were determined not to be part of the cell were given a value above

the maximum grey level of the camera so that they could be easily identified. Next two new filters were employed consecutively in order to reinsert areas of the cell that had been removed during the threshold filter. The first filter was termed the diagonal filter. The image was scanned to locate pixels that were not marked as part of the cell. A search routine was then employed to examine pixels diagonally, in both directions along the optical axis, for a user defined distance. If pixels representing a cell were found in both directions, and the pixel located to the upper right had a lower intensity value than the one located to the bottom left, it was assumed that this area was interior to a cell and the pixel was changed back to its original value along with the pixels scanned along the optical axis. However if the intensity of the pixel to the upper right was greater than the one to the lower left it was assumed that this pixel was located between two adjacent cells and was thus not changed. This is due to the fact that the cell located to the lower left would be dark as the change in cellular thickness along the optical axis was decreasing while the cell located to the upper right would be bright as the change in cellular thickness along the optical axis would be increasing. Figure 3.5(e) shows the effect of this filter.

The second filter used to reinsert cellular area was termed the stitch filter. Again the image was scanned to located pixels that were not marked as cells. Another searching routine was conducted in which pixels above, below, to the left, and to the right were examined to determine if there was a cell within a user defined distance from the given pixel. If cells were found on all four sides then the pixel was returned to its original value along with the pixels in each direction. This routine was often run for multiple cycles through a single image. Figure 3.5(f) shows the effect of this filter.

The final filter examined the entire image and removed small areas that were marked as a cell but appeared to be cellular debris in the medium. This was completed by scanning the image and locating pixels marked as cells. Similar to the stitch filter, locations were checked within a user defined distance up, down, to the left, and to the right of the pixel and if in all directions a non cell pixel was located then the pixel was changed to a non-cell value as well all the pixels along the search paths. Figure 3.5(g) shows the effect of this filter leading to the final processed image.

A final image was created to visually examine the effectiveness of the algorithm on a single image. Pixels that had been marked as part of a cell were changed to red and then overlaid onto the original image. Figure 3.5 (h) show an example of these images. A copy of the codes developed for this work is contained in Appendix B.

3.1.4 Results

In the previous section it was explained that several of the filters required user defined parameters. Many of these parameters had to be varied throughout the analysis as the cell morphology changed and the coverage area increased. In order to insure the highest accuracy possible it was necessary to calibrate these parameters by manually checking several images using Wasabi software.

At least nine images were selected for manual analysis and a greater number were chosen at the beginning of the experiment as this is where the most change in coverage area occurred. Regions Of Interests (ROI) were selected by manually tracing the outline of a cell, or multiple cells, when they were connected. The total area was then calculated by the software based on the number of pixels contained within an ROI. This value was used as the target for the automatic measurements.

Single images were processed using the automated code and then compared to the manual value. Parameters were then adjusted to match the manual value as close as possible. After all parameters had been calculated adjustments were made to the code to change the selected parameters at the appropriate time. Figure 3.6 shows a comparison of the manual and automatic area calculations where area is given as a percentage of the total electrode area. The total average difference between the manual and automatic coverage area values was less than 5% in all experiments.

3.1.5 Experimental Conclusions and Future Work

While this algorithm required quite of bit of time for initial calibration, it proved very capable of examining large sets of images automatically. Figure 3.7 shows normalized electrical resistance measurements versus normalized cellular coverage area

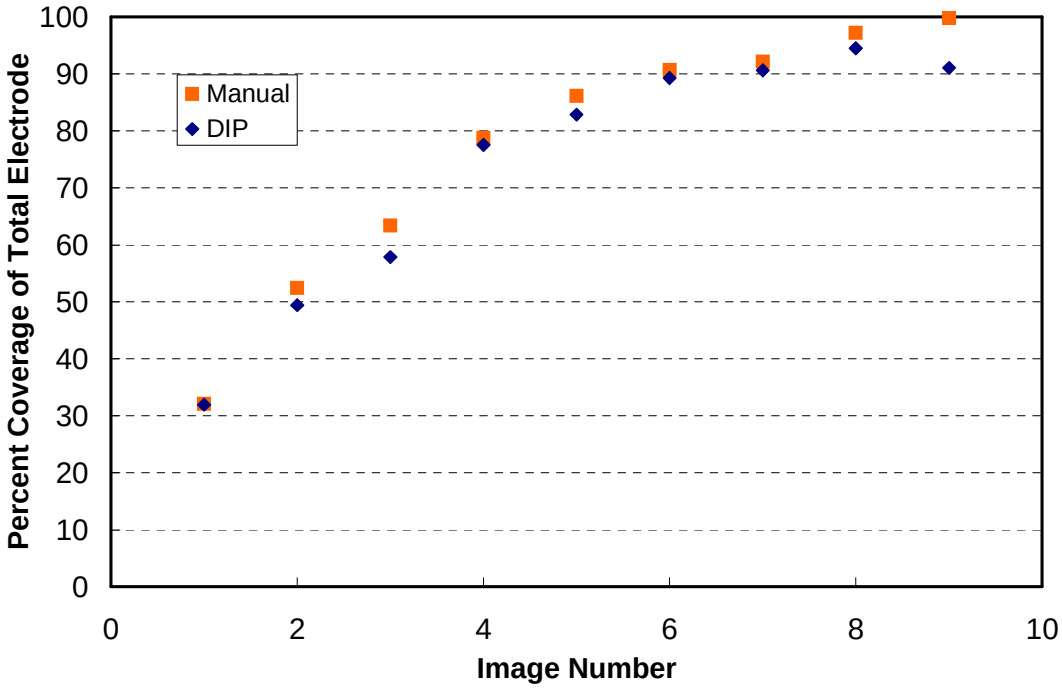


Figure 3.6 Comparison of manual and automatic area calculations. Manual calculations were made by tracing the boundary of cells using Wasabi imaging software.

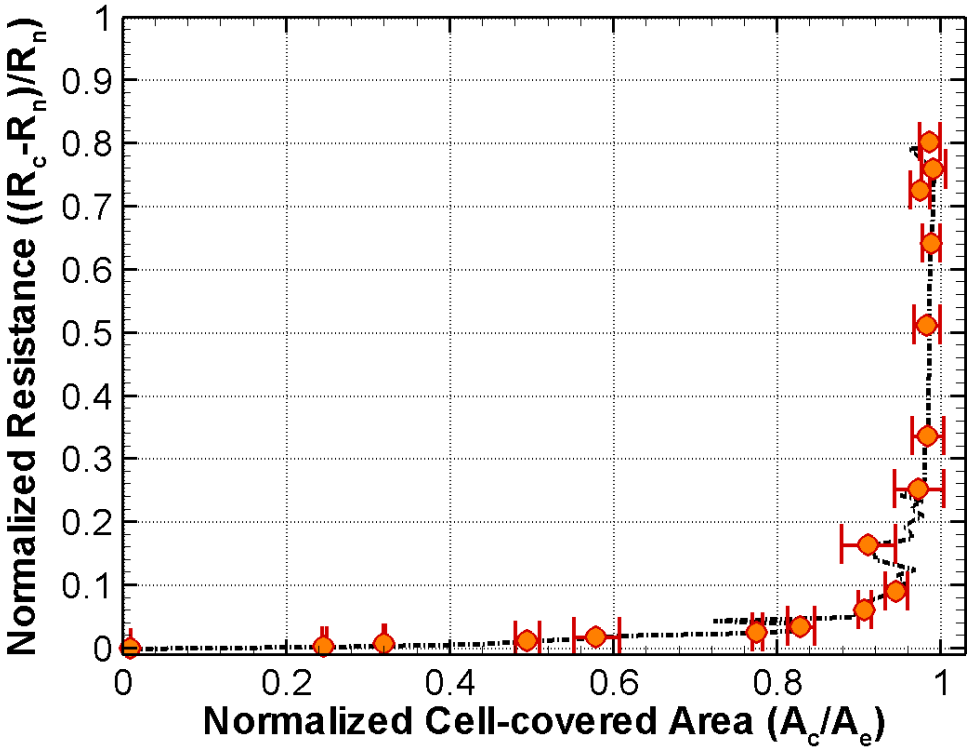


Figure 3.7 Comparison of normalized resistance vs. normalized coverage area. Resistance continues to increase even after the electrode is completely covered.

calculated using this software. The resistance is normalized by dividing the measured impedance by the impedance of an electrode that was not covered with any cells. It quickly became obvious that the change in impedance was not only a function of coverage area, as the impedance continued to increase after the cells completely covered the electrode. Therefore as previously mentioned this proved that impedance is not only a function coverage area but is also a function of cell-cell and cell-matrix contract.

One modification that could be implemented to improve the code is an optimization routine capable of selecting parameters to most closely match the manually calculated data. It would also be desirable to determine the number of cells located on an electrode rather than simply output a total coverage area. Future work on these suggestions may be considered if and when they become necessary.

3.2 Interference Reflection Contrast Microscopy Image Analysis

Interference Reflection Contrast Microscopy (IRCM) is an effective imaging technique for examining the bottom surface of a transparent material. Like DICM, IRCM does not require any marker and thus is an excellent tool for examining cellular contacts without the introduction of a foreign agent such as a fluorescent die. The goal of this work was to create software capable of automatically calculating the coverage area of focal and/or close contacts over time by processing images created using IRCM. The purpose of these experiments, the details of an IRCM system, and an explanation of the DIP algorithm will be covered in the next several sections. Finally the results and conclusions made from this work are provided.

3.2.1 Purpose of Experiment

IRCM has been used extensively in the past to examine the contact regions of a cell with a substrate [66-70]. The purpose of these experiments was to determine how cellular contacts changed when a cytotoxic agent was introduced to the cellular medium. The reaction of two cells to a 6 μM concentration of the cytotoxic agent Cytochalasin D was observed using both DICM and IRCM techniques sequentially. IRCM images were

used to examine the contact areas while DICM was used to observe how the overall morphology of the cell changed with time. After approximately one hour under cytotoxic conditions the medium was replaced with fresh nontoxic medium and observations were continued to determine if the cell would recover. All IRCM images were then analyzed to examine how the cellular contacts changed throughout the experiment. The total cellular coverage area was also calculated using the DICM images. For these experiments approximately one hundred IRCM images and twenty DICM images were taken. Therefore the coverage area, using DICM images, was calculated manually while an automated algorithm was developed to calculate the contact zones using the IRCM images.

3.2.2 Microscopy Technique

Figure 3.8 (a) provides a detailed schematic of the light path and optical components necessary for IRCM imaging [67]. Unlike DICM, IRCM is relatively simple and requires very few optical components making it one of the more inexpensive optical techniques. Because of its ability to examine the near-field region of a specimen IRCM is comparable to TIRFM which is discussed in detail in Chapter 2.

To produce IRCM images for these experiments a Xenon lamp, which produces white light, was passed through a filter system capable of providing red (635nm), green (535nm), or blue (465nm) light. The monchromatic light was then passed into an Olympus IX-71 inverted microscope containing a bright field cube. The bright field cube is an optical device, similar to a beam splitter, which allowed part of the incident light to pass, while the remaining light was reflected through a 100X oil immersion objective and into the sample. A portion of the light entering the sample was then reflected at multiple interfaces back through the objective. At this point the light was again split at the bright field cube, with a portion passed to the CCD camera for imaging. Each time the light encountered the bright field cube a significant portion of its intensity was lost. Also due to the high transparency of the cells the reflections from the different surfaces are very weak. Therefore it is necessary to use a relatively high intensity light source to acquire clear IRCM images.

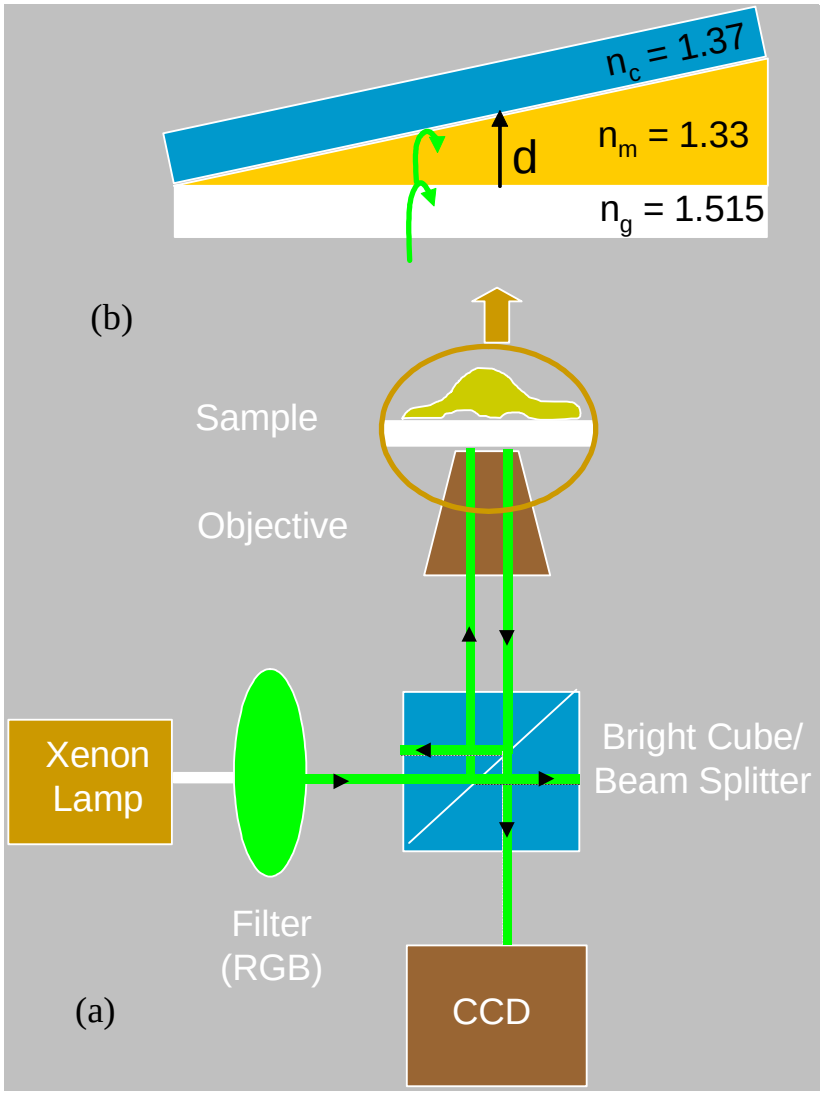


Figure 3.8(a) Schematic of light path and optical components necessary for IRCM. (b) Schematic of reflection of incident light at the glass medium interface and the medium cell interface.

Figure 3.8 (b) shows a schematic of two reflections occurring in a sample, the first at the glass-medium interface and the second at the medium-cell interface. The percentage of light reflected at these interfaces is known as the reflectance, and can be calculated using equation (3)

$$R = \left(\frac{n_t - n_i}{n_t + n_i} \right)^2 \qquad (3)$$

where n_t and n_i are the index of refraction of the transmitted and incident mediums respectively [52]. The index or refraction for glass is 1.515, for the cellular medium 1.33, and for the cell 1.37. Table 3.1 provides details on the amount of light reflected (I_r) and transmitted (I_t), as well as possible phase changes (φ_t, φ_r), at each interface with respect to the initial incident light. At the glass-medium interface approximately 99.6% of the light is transmitted, leaving only .4% reflected back into the objective. Because the index of refraction of the incident medium, glass in this case, is higher than that of the transmitted medium there is no change in the phase of the reflected light [52]. At the medium-cell interface approximately .02% is reflected back toward the medium and there is a 180^o phase change as the index of refraction of the transmitted medium is larger than that of the incident medium. Finally following the beam of light reflected at the medium-cell interface, almost all of the light is transmitted through the medium-glass interface to the objective with no phase change occurring.

Table 3.1 Effect of multiple reflections on the intensity and phase of an incident light source.

I-T	R	I_r	I_t	φ_t	Φ_r
Glass-Medium	**4.23e-3**	**.423**	**99.577**	**0**	**0**
Medium-Cell	**2.19e-4**	**.0218**	**99.555**	**0**	**180**
Medium-Glass	**4.23e-3**	**.0001**	**.0217**	**180**	**0**

The relationship of the phase between the two reflected beams of light will be determined by the mediums being considered and the distance the second reflected beam travels with respect to the first. It has already been mentioned that the second beam is 180^o out of phase with the first at the medium-cell interface. Therefore the phase between the two waves can be determined by examining the distance traveled between the glass surface and cell membrane. In order to determine if the two lights are in phase the optical path difference, given in equation 4, must be calculated:

$$\Delta = 2d\cos(\theta) \qquad (4)$$

where Δ is the optical path difference, d is the distance between the cell and glass substrate as shown in figure 3.8(b) and θ is the angle of the reflected beam with respect to the normal, which is considered equal to zero for this case [52]. From this equation it can be shown that a bright fringe results when the distance d, between the two beams, is equal to a multiple of one quarter of the wavelength of light assuming there is no phase change. However due to the 180 degree phase change at the medium-cell interface in these experiments, the two beams would actually be in phase at this distance. Figure 3.9(a) shows that if the two beams are in phase, then their intensities will be added creating the brightest possible intensity. If the distance (d) is equal to one half of the wavelength then the two beams will be 180^o out of phase thus creating a dark fringe. The effect of this is observed in figure 3.9(b). It should be noted that for this case the intensity of the returning light will not be zero, as the two reflected beams do not have the same intensities which is shown in Table 3.1.

At this point the possibility reflected waves from the tops surface of the cell or the cell nucleus should be considered. Of primary importance in examining cells using IRCM is the Numerical Aperture (NA) of the objective. For low NA (NA < 1) the small cone angle produced, as observed in figure 3.10, provides a relatively large depth of field which can be calculated using equation 5:

$$\mathrm{DOF} = \frac{\lambda}{4 \cdot \mathrm{NA}^2} + \frac{n}{M \cdot \mathrm{NA}} e \qquad (5)$$

where λ is the wavelength of the light, n is the index of refraction of the medium, M is the magnification, and e is the smallest distance that can be resolved by a detector in the

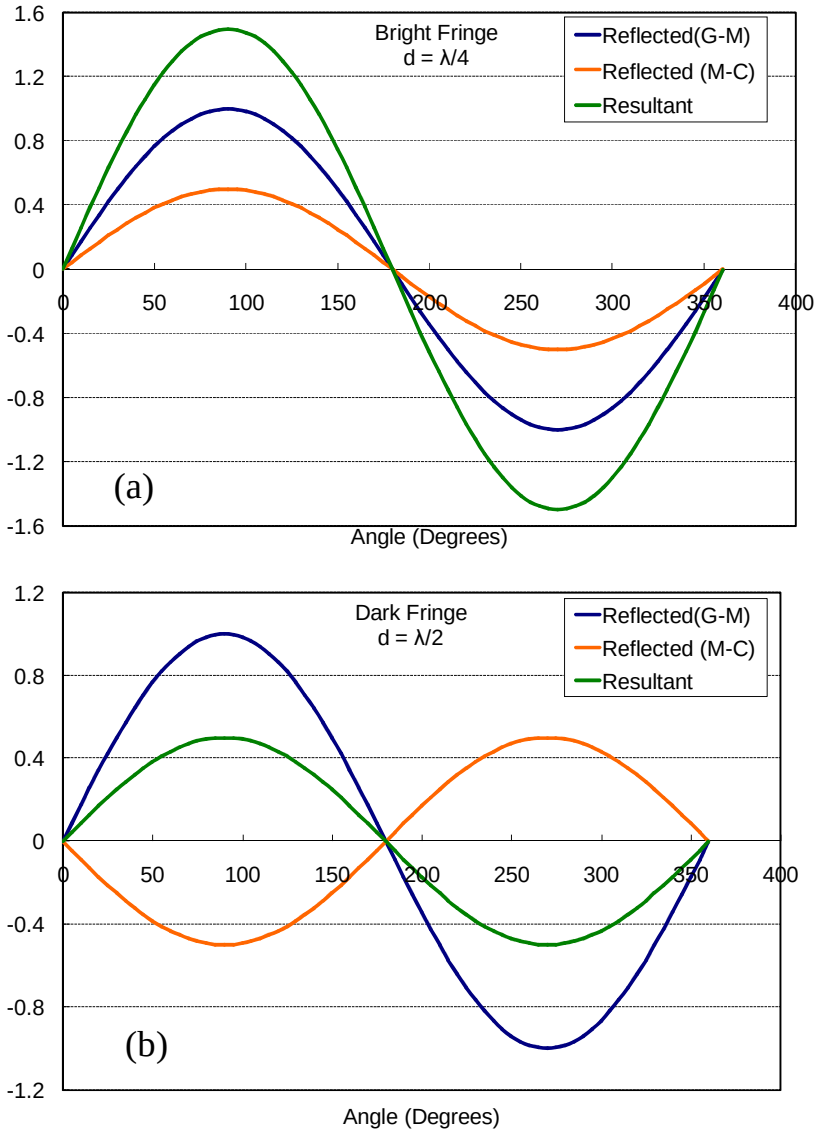


Figure 3.9(a) A bright fringe is created when the two reflected waves are in phase, resulting from an optical path difference equal to one quarter of the incident wavelength in this case. (b) A dark fringe is created when the two reflected beams are out of phase, resulting from an optical path difference equal to one half of the incident wavelength in this case.

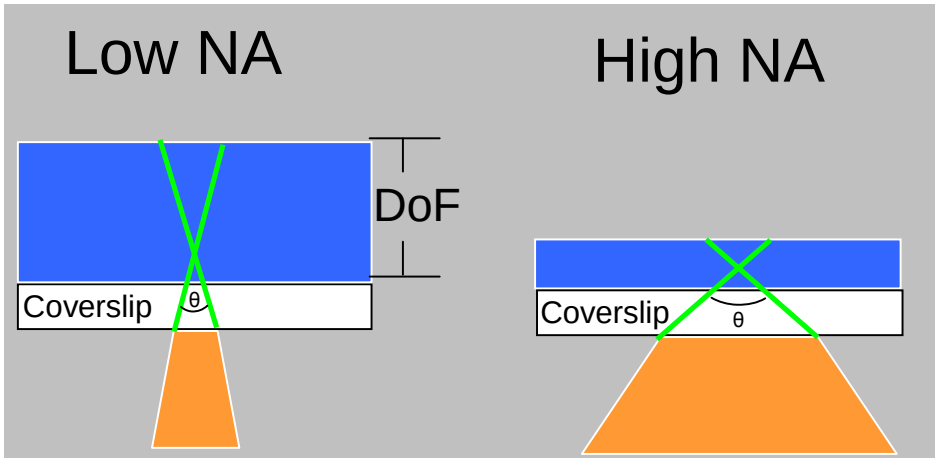


Figure 3.10 For low NA values the depth of field is relatively deep due to the small conical angle θ of the incoming light, while for large NA values the depth of field is relatively shallow due to the high conical angle of the incoming light.

image plane of the objective [63]. The depth of field represents the axial range for which an object is considered in focus. For high NA values (NA > 1) the cone angle is much larger and a smaller DOF is created. For smaller NA there is a greater chance that the reflections from the upper surface of the cell will be captured by the objective due to the large DOF. It has also been suggested that the larger cone angles, produced from a large NA, produces a condition where all interior angles cancel each other out leaving only the zero order fringe [72].

For these experiments an NA value of 1.15 was used to reduce the chance of capturing secondary reflections from the top surface of the cell. By only observing the zero order fringe it is possible to correlate a single grey scale value to an elevation. When multiple fringes are present a single grey scale value could represent multiple elevations, thus inhibiting the quantitative analysis of the bottom morphology using a monochromatic light source.

3.2.3 Algorithm

As mentioned previously the goal of this study was to examine the change in focal and/or close contacts during the administration of the drug Cytochalasin D, and after replacement with fresh medium. Because it was difficult to manually determine the coverage area of these contacts, and due to the fact that up to one hundred images were taken during an experiment, it was important to automate this process as much as possible. A copy of the codes developed for this work is contained in Appendix C.

Before creating the algorithm it was necessary to correlate the pixel grey levels of the CCD camera to cell-substrate gap distances. This was done using equations 6-8:

$$I_{Grey} = I_{Background} + \left(\frac{N}{N_{\max}}\right)\left(I_{\max} - I_{Background}\right) \tag{6}$$

$$N = \frac{\left(\frac{2r}{\left(1-r^2\right)}\right)^2 \sin^2\left(\frac{\delta}{2}\right)}{1+\left(\frac{2r}{\left(1-r^2\right)}\right)^2 \sin^2\left(\frac{\delta}{2}\right)} \tag{7}$$

$$\delta = 4\pi n_{medium} d / \lambda_0 \qquad (8)$$

where I is the pixel grey level, N is the ratio of the intensity of the reflected light to the incident light, r^2 is the reflectance, n is the refractive index, λ is the wavelength in vacuum, and d is the cell-substrate gap distance [71]. It should be noted that this analysis was completed by Dr. C.K. Choi.

Figure 3.11 provides the change in pixel grey level with respect to elevation. For these experiments I_{max}, which represents the largest grey level observed in all images, was equal to 14700, and $I_{background}$, which represents the approximate shot noise of the camera, was set at 450. Because only the zero order fringe was examined it was assumed that the brightest pixel was equal to a distance of 90 nm which is one quarter of the wavelength of the light being used (488 nm), or the maximum possible distance that could be measured [71]. Based on this analysis a pixel grey level of 6500, which correlated to a gap distance of 40 nm, was chosen to represent areas of focal and/or close contacts between the cell and coverslip. While there is still some discussions as to the exact distances for these contact points, 40 nm is generally accepted as a good approximation.

IRCM images were taken at varying intervals throughout the experiment and recorded in TIFF format. Also several images were taken of the cellular medium alone and an intensity map, similar to the one created for TIRF imaging, was used to correct for uneven illumination. After reading in an IRCM image this map was applied to the image before any processing was completed (see Chapter 2 section 2.4.2). Next the location and total number of all pixel locations less than 6500, which were assumed to be focal contacts, were recorded. A new image was then created showing focal contacts in blue on a white background, so that changes throughout the experiment could be easily observed.

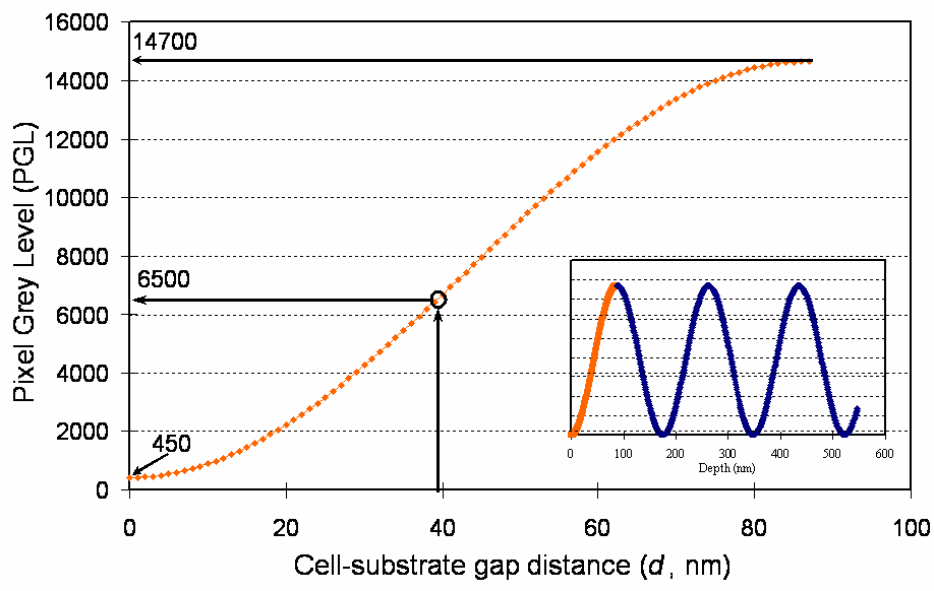


Figure 3.11 Relationship between pixel grey level and cell-substrate gap distance as calculated by equations 6-8.

3.2.4 Results

As mentioned in section 3.2.1 both DICM and IRCM images were taken throughout the course of the experiment. Figure 3.12 (a) and (b) provides DICM and IRCM images before the cytotoxic agent was introduced, 10 minutes and 1 hour after the agent was introduced, and 20 minutes 1 hour 50 minutes and 5 hours after the toxic agent had been replaced with fresh medium. Figure 3.12 (c) shows an overlay of the two images with the DICM images presented in red and the IRCM in green. Figure 3.12 (d) shows the IRCM images with the pixel grey levels placed in bins of 1000 represented by different colors with dark blue representing the lowest values and dark red the largest. Finally 3.12 (e) shows the pixels with values below 6500 which are believed to represent focal and/or close contacts.

Examination of the DICM images show that after the introduction of the cytotoxic agent the cell begins to shrink and pull away from the surface taking on a much more rounded shape. Before this agent is introduced the cell shape appears relatively flat and well spread out. After replacement with fresh medium the cell eventually returns to its pretoxic condition.

IRCM images show relatively large dark areas throughout the cell before the addition of the drug. However after the addition of Cytochalasin D most of the remaining contact areas tend to be on the periphery of the cell, which can be most easily observed in figure 3.12 (e). Finally after the replacement of the toxic medium with fresh medium the cell appears similar to its pretoxic condition with the exception of the focal contacts which do not quite reach their pretoxic levels.

Figure 3.13 provides quantitative values for changes in the cell over time. Figure 3.13 (a) shows the focal contact area calculated using the IRCM images and the DIP algorithm. Figure 3.13 (b) shows how the normalized cell coverage area (actual covered area divided by initial coverage area) changes during the different stages of the experiment. These values were calculated by manually tracing the area of the cell using DICM images and Wasabi software. Figure 3.13 (c) provides information regarding the ratio of focal contact areas of the cell to the total cell coverage area over time. It should be noted that significantly fewer DICM images were taken than IRCM images due to the

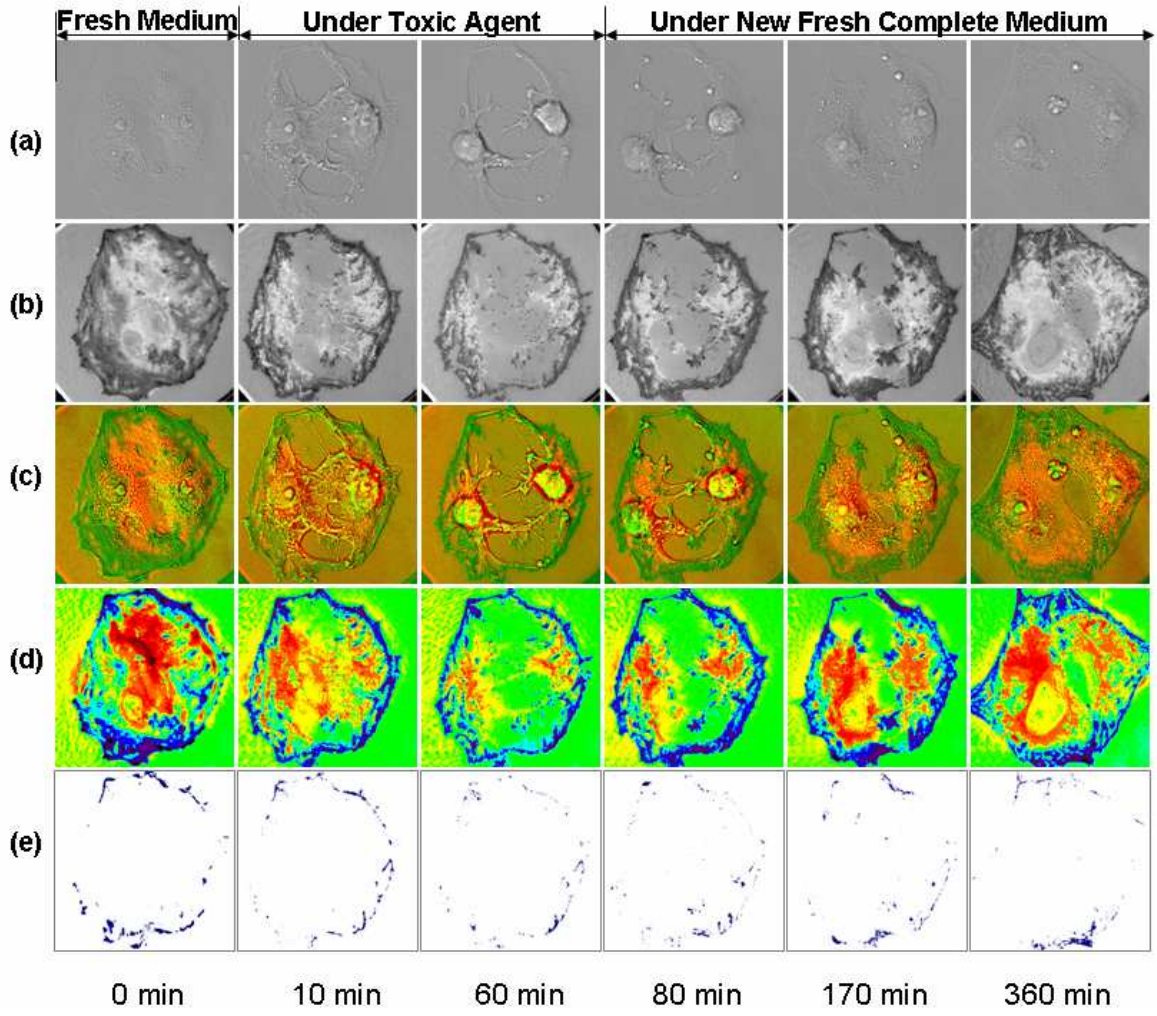


Figure 3.12 (a) DICM images. (b) IRCM images. (c) Overlay of DICM (Red) and IRCM (Green) images. (d) IRCM images with colors representing pixle grey levels of 1000. Dark blue represents the smallest values with dark red the largest. (e) Blue values represent focal contacts determined from IRCM images. The first column shows the pretoxic state of the cells. The second and third column show changes in the cell 10 minutes and 1 hour after the addition of the toxic agent. The last three columns show cell recovery 20 minutes, 1 hour 50 minutes, and five hours after replacement with new fresh medium.

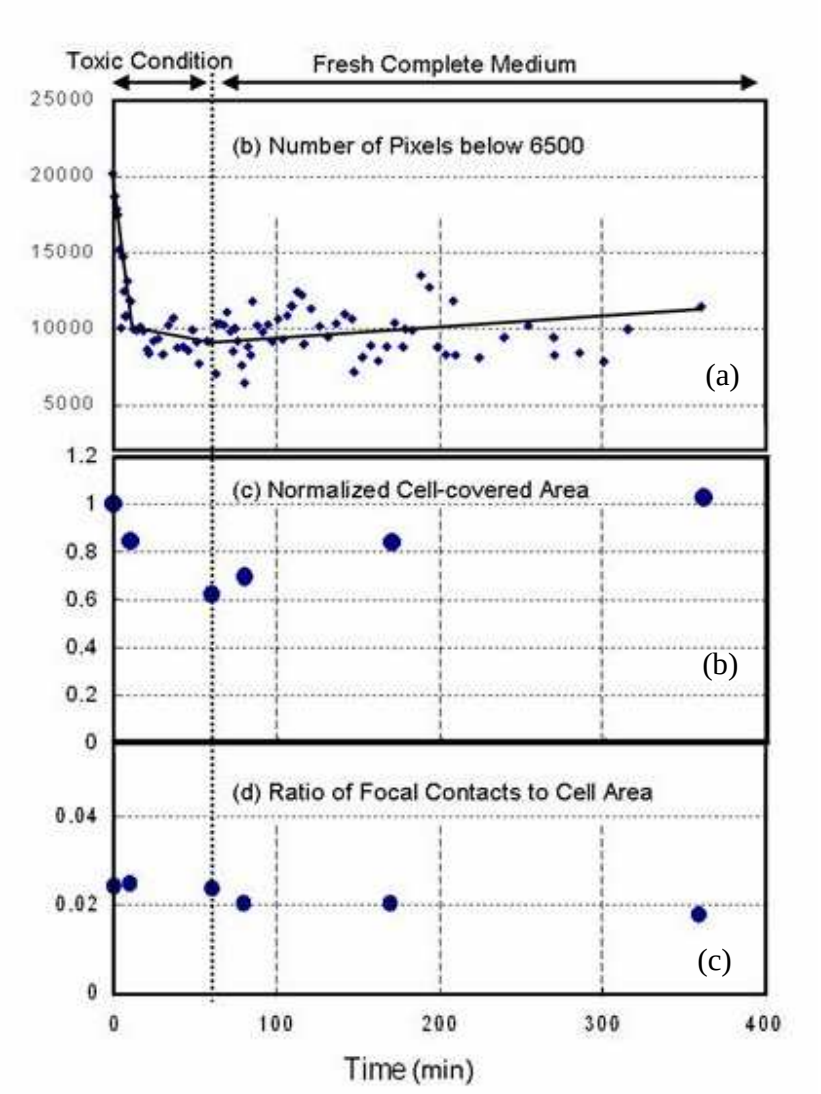


Figure 3.13 (a) Changes in the focal contact area over time. (b) Changes in the normalized cell-covered area over time. (c) Ratio of the focal contact area to the total cell covered area over time.

fact that several changes to the microscope were needed to switch between techniques. Because the main interest of this work was to examine cell adhesion, IRCM images were given a higher priority.

The results provided in figure 3.13 show that there is a rapid reduction in cellular coverage area after the addition of the drug followed by an almost complete recovery approximately five hours after the toxic medium is replaced with fresh medium. The focal contacts also experience a significant reduction following exposure to Cytochalasin D. However these areas remain almost constant throughout the rest of the experiment, even after the toxic medium was replaced with fresh medium.

3.2.4 Experimental Conclusions and Future Work

This experiment showed that there is a group of tightly adhered focal and/or close contacts that are resistant to high doses of the cytotoxic agent Cytochalasin D. While the total focal contact area did decrease following initial exposure to the agent, a large portion remained throughout the entire experiment. These contacts also appeared to provide a framework for cellular recovery after the toxic agent had been replaced with new fresh medium.

It should be noted that examining focal contacts alone is not enough information to evaluate the health of the cell. While the adhesion sites remained relatively constant, DICM images proved that the cell was significantly stressed while under cytotoxic conditions. Therefore the use of complimentary imaging techniques provides a much more comprehensive examination of cellular response.

The coupling of DICM and IRCM techniques could prove very valuable in several areas of cellular research. Introduction of a fluorescent die could also enable a third microscopy technique, such as TIRFM or epi-fluorescence, to be added to the DICM/IRCM combination. It is possible that a combination of TIRFM and IRCM could be used to obtain more reliable information on the bottom cellular morphology and how it changes over time. It is believed that the use of multiple microscopy techniques may provide significant improvement to the world of biological sciences in the future.

CHAPTER IV INTRACELLULAR VESICLE FORMATION AND TRACKING

Cellular vesicles are small structures used to move proteins within a cell body. An important function of these vesicles is their involvement in exocytosis, a mechanism for transferring proteins through the plasma membrane and to the extracellular matrix. This process is important for disposing of cellular waste, signaling between other cells, and several other biological functions.

The process of vesicle docking and fusion has been examined by several researchers using TIRFM. The goal of these previous experiments was to examine how vesicles move towards, and transfer their contents into, the plasma membrane. However because TIRFM only examines the very near wall location, the effect of the coverslip on the movement of the vesicles may play an important, and unphysiological, role in this process. As shown in Chapter II, the Brownian Motion of nanoparticles very near a surface is significantly hindered due to the no slip boundary condition imposed by the surface. This study attempts to examine the hindered motion caused by the glass surface on the movement of the intracellular vesicles.

4.1 Purpose

Brain cancer is one of the top ten causes of cancer related deaths in women. It is believed that the Non-steroidal anti-inflammatory drug-Activated Gene-1 (NAG-1) protein may play an important role in promoting the growth of healthy brain cells while retarding the growth of cancerous brain cells. Because the movement of these proteins is carried out in vesicles, examination of vesicle movement and size may play an important role in understanding what function this protein plays in retarding the increase of cancerous cells.

Before attempting to examine the movement of these vesicles a series of experiments were conducted to determine what part, if any, of the NAG-1 DNA chain was responsible for the formation of vesicles. Several variations of the naturally

occurring NAG-1 DNA chain were transfected into brain cancer cells to determine if vesicle formation was disrupted. Section 4.3 provides details on this study.

4.2 Cell Handling

Cell handling and transfection was completed by members of the Pathobiology Department in the College of Veterinary Medicine at the University of Tennessee. The next two sections will review the protocol provided by these researchers in preparing the cells for examination using the MiNSFET Lab microscopy equipment.

4.2.1 Cell Culturing and Constructs

The human T98G glioblastoma cell line was purchased from American Type Culture Collection (Manassas, VA) and maintained in DMEM medium supplemented with 10% fetal bovine serum (FBS), 100 U/mL of penicillin, and 100 μg/mL of streptomycin. Cells were incubated at 37°C under a humidified atmosphere of 5% CO_2.

Full length NAG-1 cDNA was amplified using ReadyMix Taq polymerase (Sigma, St. Louis, MO) with the following primers: forward 5'-GCCATGCCCGGGCAAGAACTC-3',and reverse 5'-ATATGCAGTGGCAGTCTTTGGC-3'. PCR was performed for 30 cycles at 94°C for 1 min, 55°C for 1 min, and 72°C for 2 min. The amplified products were subcloned into pcDNA3.1/CT-GFP (Invitrogen, Carlsbad, CA) to generate GFP-tagged proteins (pcDNA3.1/NAG-1/GFP). For ΔN deletion form of NAG-1 construct, pcDNA3.1/NAG-1/GFP was amplified using forward 5'-GCCATGCTGTCTCTGGCCGAGGCGAGC-3, and reverse 5'-ATATGCAGTGGCAGTCTTTGGC-3'. For active form of NAG-1 construct, pcDNA3.1/NAG-1/GFP was amplified using forward 5'-GCCATGGCGCGCAACGGGGACCAC-3, and reverse 5'-ATATGCAGTGGCAGTCTTTGGC-3'. The point mutant clones for the RXXR site of NAG-1 were generated with pcDNA3.1/NAG-1/GFP using the QuickChange II site-directed Mutagenesis Kit (Stratagene, La Jolla, CA) according to the manufacturer's protocols.

4.2.2 Transfection

Transient transfections were performed using Lipofectamine (Invitrogen) according to the manufacturer's instructions. The cells were plated in MatTek glass bottom culture dishes at a concentration of 2×10^5 cells/well and left overnight. The cells were then transfected with the wild type or mutant pcDNA3.1/NAG-1/GFP expression vector (1 μg DNA) for 5 h and fed fresh medium (DMEM with 10% FBS) overnight.

4.3 Vesicle Formation

As mentioned in section 4.1 several experiments were conducted to determine what part of the NAG-1 DNA chain played a role in the formation of vesicles. The following sections provide detailed information on the methods used to examine if the disruption of vesicle formation could be observed. Also included is a description of the controls used for comparison with modified DNA results, as well as conclusions drawn from this research.

4.3.1 Controls

Two control groups were created to determine if the formation of vesicles had been disrupted by modifying the DNA code of NAG-1. For the first control, brain cancer cells were transfected with wild-type NAG-1 that had been tagged with a green fluorescent protein. This procedure, also used for tracking purposes, produced very clearly defined vesicles that could be imaged using TIRFM. For the second control, brain cancer cells were transfected with only GFP and a very cloudy image was observed using TIRFM. It is expected that if a disruption in vesicle formation occurs the images should appear more like the second control. Figure 4.1 shows images of the two control groups. TIRFM images, DICM images, and an overlay of the two, with TIRFM in green and DICM in red, are provided so that the location of vesicles within the cell can be observed.

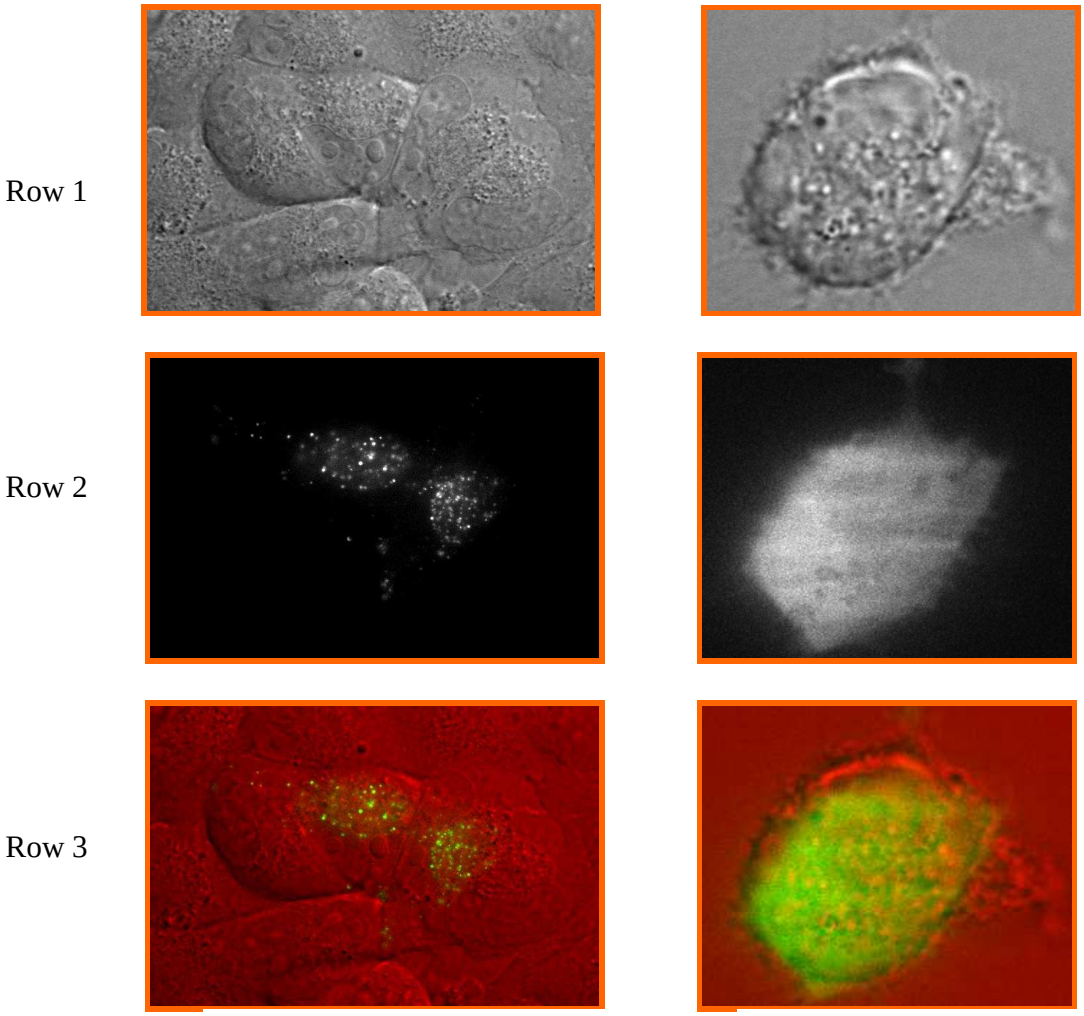


Figure 4.1 Row 1 DICM images, Row 2 TIRFM images, Row 3 overlay DICM (Red) and TIRFM (Green). (a) Control 1 with wild type NAG-1 tagged with GFP. Vesicle formation is present. (b) Control 2 with GFP directly transfected into the cell. No vesicles present with a cloudy formation distributed throughout the cell.

4.3.2 NAG-1 DNA Modifications and Mutants

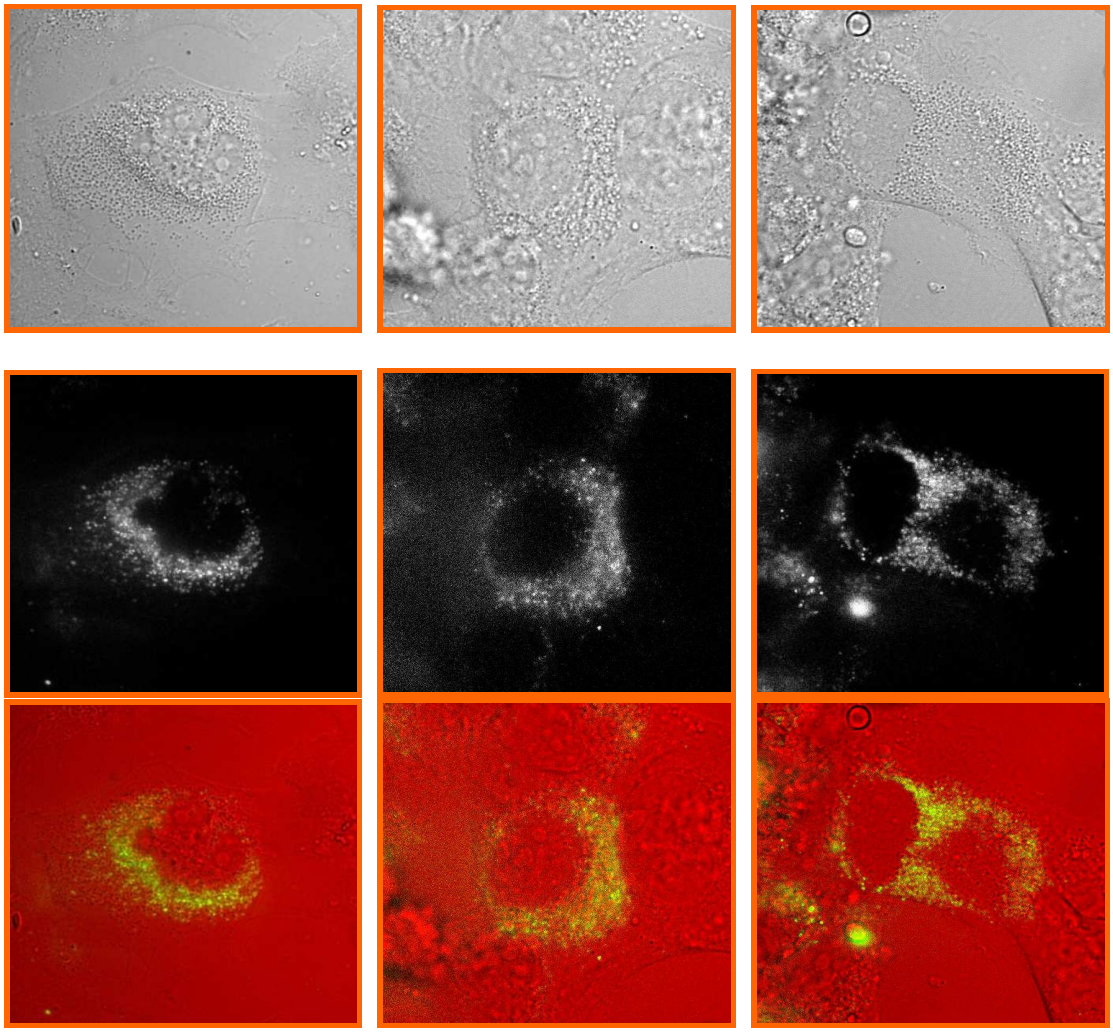
Three different forms of the NAG-1 DNA chain were tested to determine if vesicle formation could be disrupted. The initial experiment consisted of removing the first 29 amino acids (referred to as Δ29) from the DNA chain. It is believed that this sequence of the code is responsible for translocation of NAG-1 to different areas of the cell. Therefore if vesicle formation occurs at a specific region within the cell, removal of these amino acids could prevent translocation of NAG-1 to that region, thereby eliminating vesicle formation. Figure 4.2(a) shows results from this experiment.

The next alteration to the DNA sequence consisted of mutating the RXXR site. It is known that this location is the point where the NAG-1 DNA is cleaved to form the so called "active form" which is responsible for allowing it to perform its various biological functions. It is believed that prevention of the active form may lead to the prevention of vesicle formation. Figure 4.2(b) shows images taken of the RXXR mutant experiment.

Finally the third modification tested was transfection of only the "active form" of the NAG-1 into the cell. In this case the DNA was cleaved at the RXXR site prior to transfection. This process is the opposite of the second modification where the DNA was prevented from cleaving. Figure 4.2(c) shows images taken during this section of the experiment.

4.3.3 Results and Conclusions

None of the modifications made to the NAG-1 DNA chain resulted in the disruption of vesicle formation. However a few observations can be made that show a change from the wild-type NAG-1 control. First there appeared to be a significantly higher number of vesicles inside the cell for all modifications. Second it was apparent that these vesicles tended to form large concentrated clusters surrounding what appears to be the nuclear region of the cell. This is apparent in the TIRFM images where a large dark spot is surrounded by a very bright perimeter. Finally it appeared that there was very little movement of these vesicles when compared to the control.



(a) (b) (c)

Figure 4.2 Row 1 DICM images, Row 2 TIRFM images, Row 3 overlay DICM (Red) and TIRFM (Green). (a) Δ29 modification. (b) RXXR mutation. (c) Active form of NAG-1 only.

It was concluded that the mechanism for creation of vesicles containing NAG-1 was not disrupted by any of these changes to the DNA sequence. However there does appear to be some change in the cell with regards to the movement of the vesicles that may be worth examining in more detail in future studies. Further the significance of clustering around the nuclear region may also yield important information.

4.4 Intracellular Vesicle Tracking

As discussed in Chapter I a good deal of work has been conducted using TIRFM in examining the movement of vesicles and granules inside of various types of cells. Much of this work has considered how vesicles dock to the cellular membrane in order to transfer their contents into the membrane itself. However little consideration has been given to what effect the glass coverslip plays in hindering the motion of these vesicles as they approach the wall. It was observed in chapter II that the Brownian motion of nanoparticles is severely restricted in both the normal and tangential directions due to the no slip boundary condition found at the wall. It is believed that the size of the vesicles located inside the brain cancer cells are on the same order of magnitude as the nanoparticles and thus may be susceptible to the same hindrance effects. This condition may therefore create a non-physiological environment and provide docking conditions that are unrealistic. The goal of this study is to determine if the motion of vesicles in the near wall region experience the same hindered motion as observed in nanoparticles.

4.4.1 Types of motion

Previous works examining the motion of vesicles have divided their movements into either three or four categories. These consist of Brownian Motion, directional motion, caged motion, and anomalous diffusion [45]. For this work the three model approach was chosen in which anomalous diffusion is not considered.

As discussed in detail in Chapter II Brownian motion is random in nature and can be determined by examining the change in mean square displacement with increasing time steps. If this change is linear then the motion is considered Brownian. Equation (1)

shows the relationship between Brownian motion and the time step, Δt, for two dimensional tracking:

$$MSD(\Delta t) = 4D\Delta t \qquad (1)$$

where D is the free diffusion coefficient, which may be adjusted in the normal and tangential direction as a particle approaches a surface [45].

Directional motion, as implied by the name, is the movement of a vesicle from one location to another in a directed fashion. It is possible that along with directional motion there may still exist a random component as well. Equation (2) provides the relationship of mean square displacement with Δt for directional motion:

$$MSD(\Delta t) = 4D\Delta t + V^2\Delta t^2 \qquad (2)$$

where V is the velocity of the vesicle moving from one location to another. Note the first term is the same as for equation (1) and takes into account the possibility of any random component of the mean square displacement [45].

Caged motion is perhaps the most obscure type of motion of the three being considered. This type of motion appears random in nature but is confined to a specified area. Put simply it appears that a vesicle is moving randomly while trapped inside a cage. Physiollogically it is possible that the vesicle is tethered to a microtubule or to the plasma membrane and thus cannot move further than its tether. The relationship between the mean square displacement and Δt for this type of motion is given in equation (3):

$$MSD(\Delta t) = R^2\left[1 - A_1 e^{-\frac{4A_2 D\Delta t}{R^2}}\right] \qquad (3)$$

where R is the radius of the cage in which the particle is confined, and A1 and A2 are equal to .99 and .85 respectively, as determined by Li et al. who completed three-dimensional tracking of secretory vesicles inside of PC12 cells [35]. It is assumed that for two dimensional tracking A_1 and A_2 do not change whereas the coefficient 4 becomes 6 for the three-dimensional case.

Each vesicle tracked was placed into one of these three types of motion. The method for determining which type of motion was prevalent will be discussed in section 4.4.3. It should be noted that other possible type of motion exist, such as directed motion

inside of a moving cage, but only the three types listed were considered for this study. Future work may be considered in determining if better models may approximate the actual vesicle motion more appropriately.

4.4.2 Calculation of Vesicle Center

In order to track the movement of nanoparticles (Chapter II), the brightest pixel from a single particle was chosen as the center so that the movement of this spot between two frames could be used to calculate distances. By using this method the uncertainty between two locations is equal to one half of a single pixel size. For experiments involving vesicle movement a single pixel was equal to 133 nm yielding a measurement uncertainty of 66.5 nm. In order to improve the accuracy a sub-pixel resolution method was employed.

There are several different methods used to determine the center of a vesicle with sub-pixel accuracy. An in depth look at these methods can be found in the study conducted by Cheezum et al. [44]. For these experiments the Gaussian based method was considered to be the most accurate for determination of vesicle center and subsequently for vesicle tracking. Figure 4.3(a) shows a TIRFM image of a single vesicle inside a brain cancer cell that has been transfected with the NAG-1 wild type protein tagged with a GFP. Figure 4.3(b) shows the intensity variation of a single vesicle. In order to show the Gaussian nature of the intensity distribution clearly, figure 4.3(c) plots a one dimensional intensity distribution using the center column of pixels, which passes through the maximum pixel intensity. Note the typical bell-shaped curve produced by these points.

To determine the sub-pixel location of the center of the vesicle it is necessary to fit a Gaussian curve to the collected data. The general form of a Gaussian function is given by equation (4):

$$I(x,y) = Ae^{\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma^2}} + I_b \qquad (4)$$

where A is the maximum pixel intensity, x_o and y_o are the locations of the vesicle center, x and y are the pixel locations for the distribution, σ is the shape of the Gaussian curve, and I_b is the local background intensity. A nonlinear regression algorithm was created to

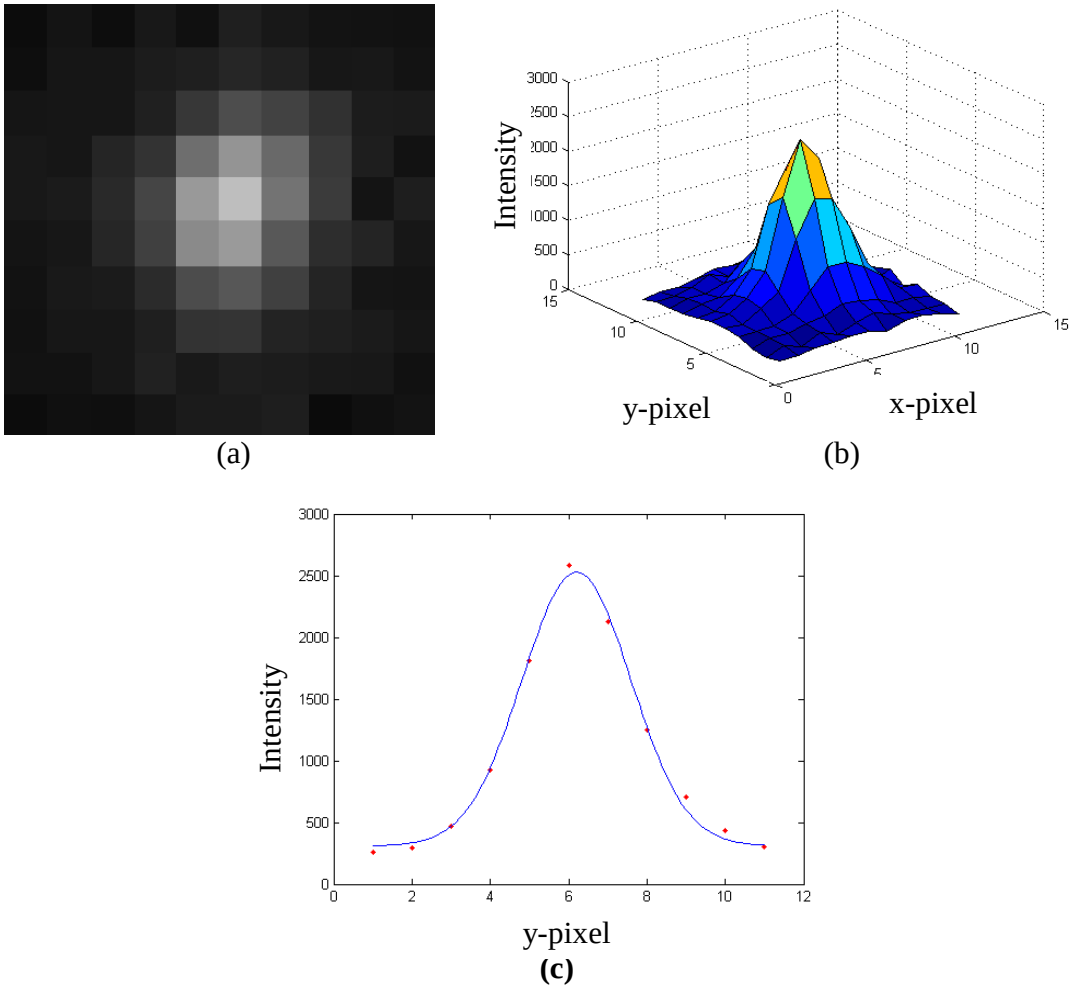


Figure 4.3 (a) Digital image of a single vesicle located inside a brain cancer cell. The cell was transfected with the wild-type NAG-1 tagged with a GFP. (b) Two dimensional plot of the intensity distribution versus pixel for a single vesicle. (c) One dimensional curve fit of the particle intensity to a Gaussian Curve.

solve for the variables, A, x_o, y_o, σ, and I_b for each image. An example of a one dimensional curve fit is shown in figure 4.3(c). Once the vesicle center was determined it was possible to then track individual vesicles, between frames, in order to calculate the mean square displacement.

4.4.3 Particle Tracking Algorithm

When tracking nanoparticles a Eulerian approach [73] was used in which a specific area of the total image captured was examined. Particles located within this area were then tracked until they moved outside of the evanescent wave and were no longer visible. Because all particles were approximately the same size and undergoing the same type of motion their results were combined to calculate the average motion versus elevation above the surface. However as mentioned in section 4.4.1 vesicle motion is divided into three categories and therefore it is important to combine similar types of motion to obtain average values that are relevant. Because of this it was important to switch to a Lagrangian approach [73] in which individual vesicles were tracked rather than examining only certain areas of the image.

Figure 4.4 provides a flow chart for the algorithm developed to track individual vesicles. Initially a dark noise map was created by taking images while the CCD camera was shuttered. Next movies, lasting one minute, were taken of moving vesicles inside brain cancer cells. At the conclusion of the experiment these movies were examined in order to mark the initial location of vesicles to be tracked. The maximum pixel location of the vesicles, as well as the initial and final frame in which tracking was to take place, were placed in an input file used for automated tracking. The movie files were then segmented into individual TIFF files so that they could be read into Matlab software.

Using the input file the initial image was imported into the software and the background was subtracted. The vesicle center was then determined using the Gaussian method as described above. The next image was then imported and a searching algorithm was employed to locate the new position of the vesicle. This algorithm searched a radius, centered about the maximum vesicle pixel intensity from the previous image, of less than four pixels. The maximum pixel value was chosen as the new

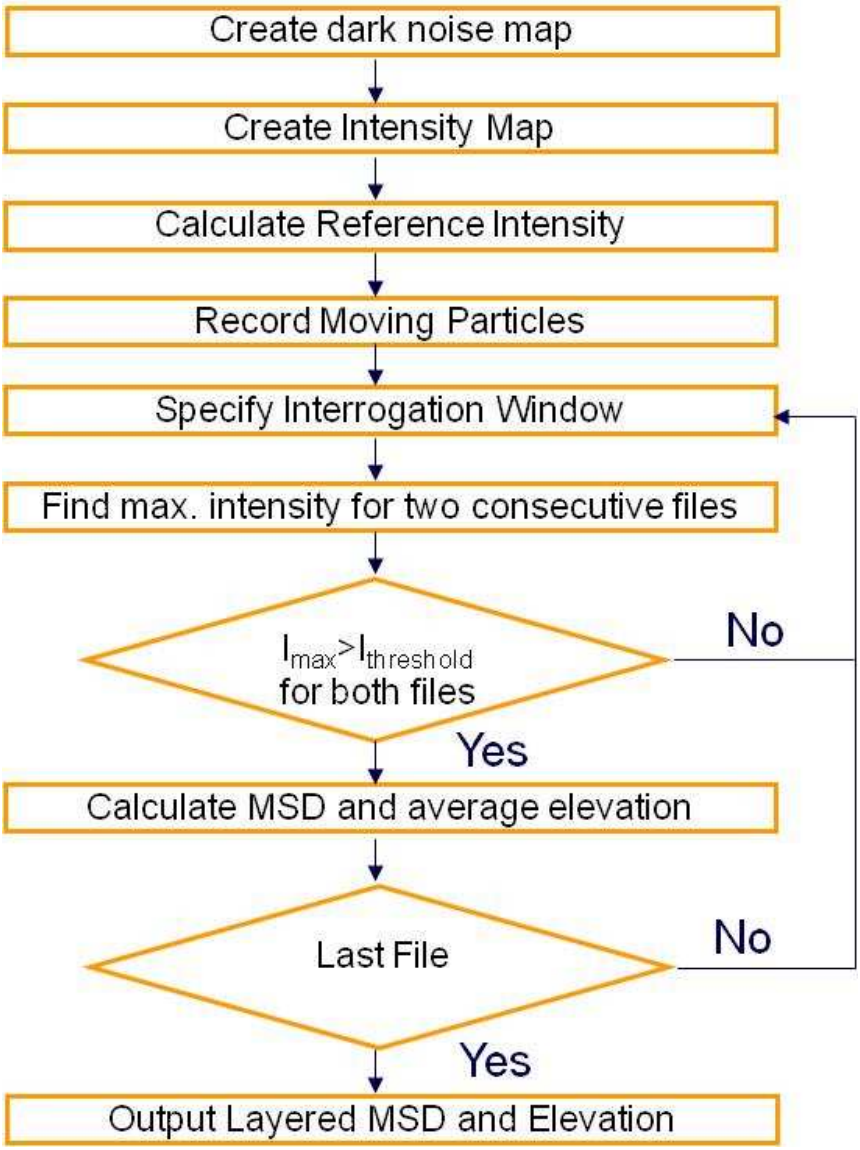


Figure 4.4 Flow Chart for vesicle tracking algorithm.

location and the center was recalculated using the Gaussian filter. To limit the possibility of tracking distances between two separate vesicles the sample rate of image acquisition was set at 8.3 frames per second, reducing the distance a vesicle could travel between frames. Also each vesicle was visually tracked and the position and intensity of the vesicle in the final frame was recorded. This value was then compared to the last position obtained from the automatic routine to insure that the vesicle being tracked had not changed. A separate output file containing the vesicle position and intensity was created for each individual vesicle. A copy of the codes developed for this work is contained in Appendix D.

4.5 Analysis of Vesicle Motion

Three separate analyses were completed regarding the motion of vesicles. The first part analyzed individual vesicle motion to divide vesicles into Brownian, directed, or caged types of motion. The second analysis averaged the motion of all vesicles, within their particular type, at varying elevations to determine if a hindered effect could be observed as a particle approach the slide glass. The final analysis attempted to determine an average vesicle size as well as separate hydrodynamic hindrance effects from other hindrance effects.

4.5.1 Division of Vesicle Motion

As discussed in section 4.4.1 determination of which type of motion a vesicle was undergoing was achieved by examining the change of the MSD values with increasing time steps. Individual vesicle location files were read in and an average MSD value was calculated for a single time step. Next the time step was doubled such that the distances between files one and three, two and four, three and five, and so on were calculated. Ideally values for one and three, three and five, five and seven, etc. would be used to achieve independent results. However in order to prevent significant photobleaching, as well as to maintain cell health, it was not viable to conduct long term experiments where large data sets could be gathered. Therefore to get statistically relevant data it was necessary to use overlapping time steps. This procedure was repeated for time steps up to five seconds.

Following this procedure measured MSD values were fit to equations (1), (2), and (3) using the same nonlinear regression technique applied to calculate vesicle centers. In order to determine the best fit the root mean square method, as shown below, was used:

$$RMS = \sqrt{\frac{\sum_{n=1}^{n_{max}} MSD_{calculated} - MSD_{measured}}{n_{max}}} \qquad (5)$$

Where $MSD_{calculated}$ is taken from parameters solved for in the curve fitting process, $MSD_{measured}$ is taken from the actual data, and n_{max} is the total number of time steps evaluated. The equation yielding the lowest RMS value was chosen as the type of motion the vesicle experienced during testing. Figure 4.5 shows a representative case for each type of motion analyzed. It should be noted that three of the fifty-three vesicles tested were placed into the Brownian type motion, although the RMS value for either caged or directional were slightly lower. This was due to the fact that both caged and directional motion incorporates curvature into the solution so it is possible at times to more closely approximate a linear curve that may have scatter in the data. For the three vesicles changed the difference in the RMS values between Brownian and the other type of motion was less than five percent.

The next step in this process was to determine average values for the diffusion coefficients in all three models, the velocity in directional type motion, and the radius in caged motion. Table 4.1 provides these values along with the number of vesicles represented in each type of motion while figure 4.6 shows the maximum, minimum, and average values for the different cases. From these results it was shown that the Brownian type of motion resulted in the lowest diffusion coefficient and the directional with the highest. The average cage size was equal to732 nm and the average velocity of the vesicles moving in a directed fashion was equal to 174 nm/sec.

Table 4.1 Average parameters for varying types of vesicle motion.

	No. of Ves./ Total No.	Diffusion Coefficient (μm^2/sec)	Radius (Caged) Velocity (Directional)
Brownian	9/53	.025	NA
Caged	19/53	.037	732 nm
Directional	25/53	.053	174 nm/sec

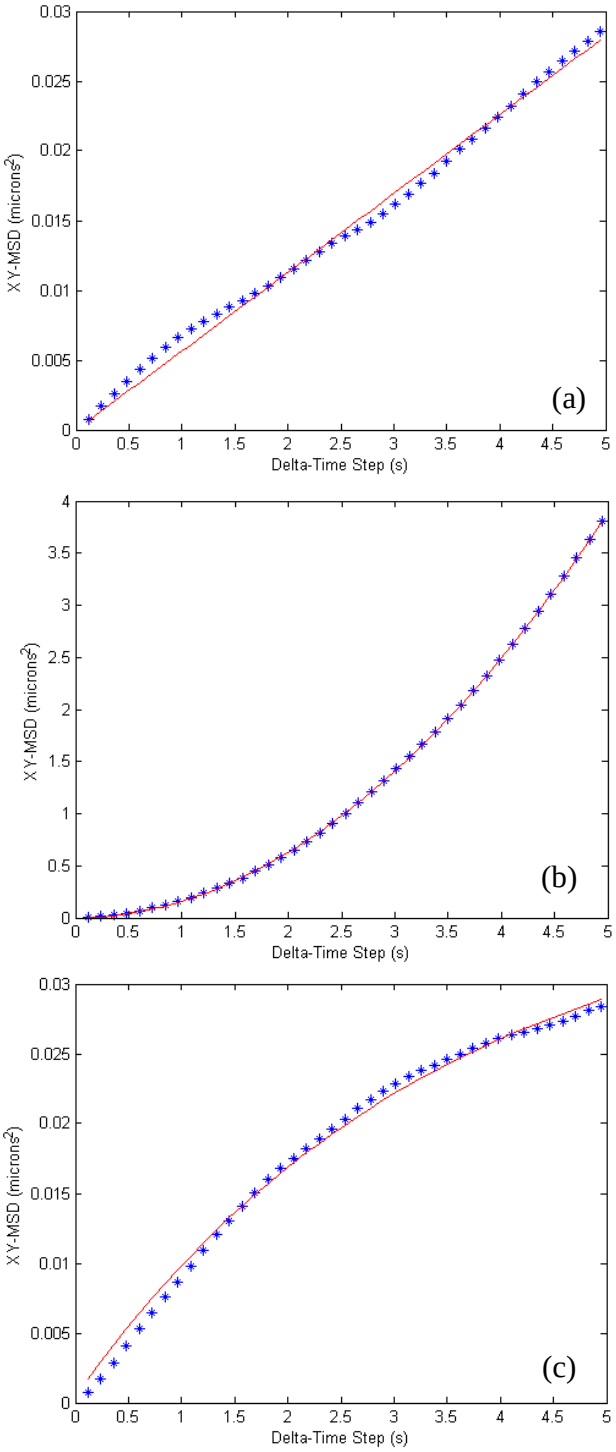


Figure 4.5 Tangential MSD values vs. varying time steps. (a) Brownian motion. (b) Directional motion. (c) Caged motion.

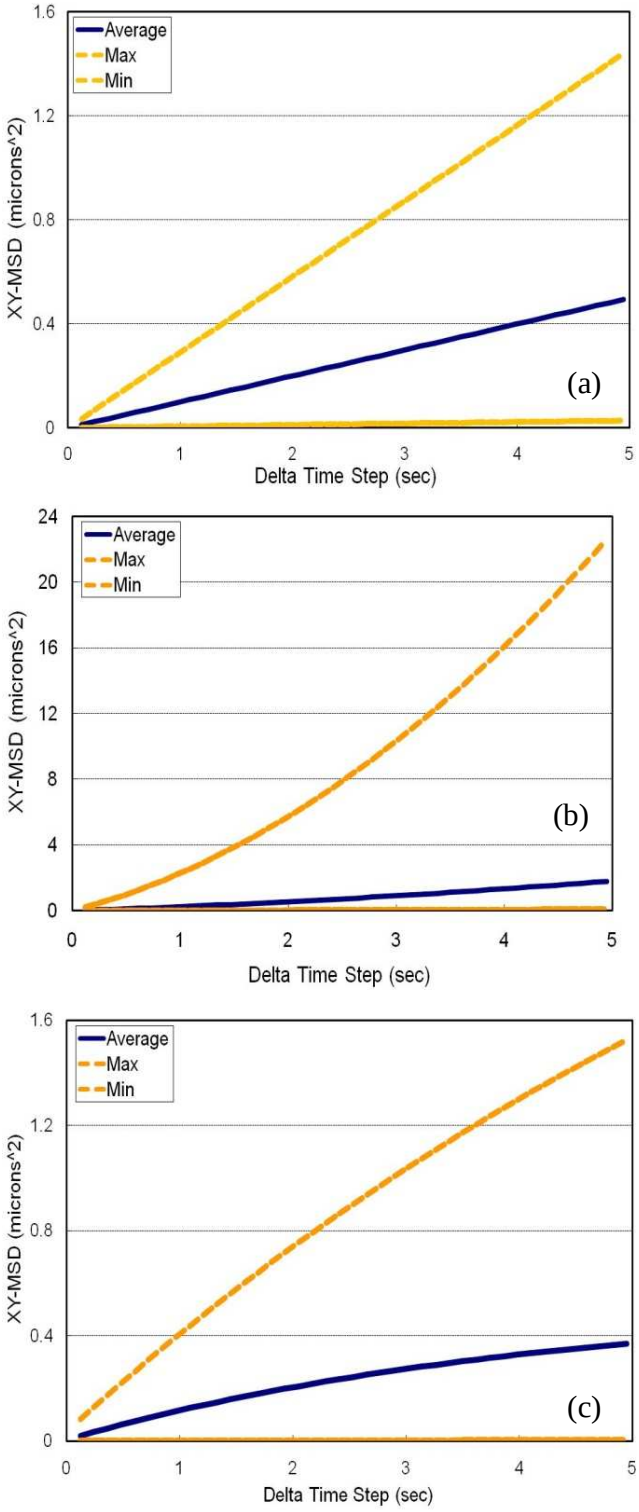


Figure 4.6 Maximum, minimum, and average tangential MSD values vs. varying time steps. (a) Brownian motion. (b) Directional motion. (c) Caged motion.

4.5.2 Examination of Near Wall Hindrance

The next analysis conducted was to determine what effect the wall played on inhibiting the motion of the particles as they approached the surface. First the vesicles were separated into their individual types of motion. Next the displacement of a single vesicle, both tangential to the surface as well as normal to the surface, was calculated. Normal displacements were determined using the ratiometric method similar to that discussed in Chapter II. However unlike in Chapter II, where an average stuck particle intensity was calculated so that absolute elevation above the surface could be determined, these elevation changes are calculated by taking a ratio between two frames of a single vesicle. This prevents errors caused in displacements from vesicles which may have significantly different intensities at a given height due to a change in size or the amount of GFP contained in a single vesicle. The initial elevation of the vesicle being tracked was set at zero and all displacements were measured from this reference point. After vesicle tracking was completed, the minimum elevation was determined, and this value was set as the new zero such that all values were shifted up accordingly. This procedure was repeated for all vesicles within a given type of motion.

In order to examine changes in motion as vesicles approached the surface the tangential and normal mean square displacements were divided into fifty nanometer elevation bins, starting from the surface and extending to three hundred nanometers above the surface. The movement of all vesicles, for a given type of motion (i.e. Brownian, directional, or caged), were averaged within these bins. This required making two very limiting assumptions. First that all particles minimum elevations were the same, and second, that this elevation was at or very near the coverslip. At present there is no way to obtain absolute elevations and thus these assumptions were necessary.

Figure 4.7 shows tangential results from this analysis. Because of the fewer vesicles observed showing Brownian type motion it was not possible to get enough data to be statistically relevant past two hundred nanometers. For caged motion enough data was taken so that measurements could be made up to two hundred and fifty nanometers bove the surface and for directional motion elevations up to three hundred nanometers above the surface were possible. It should be noted that each point in figure 4.7

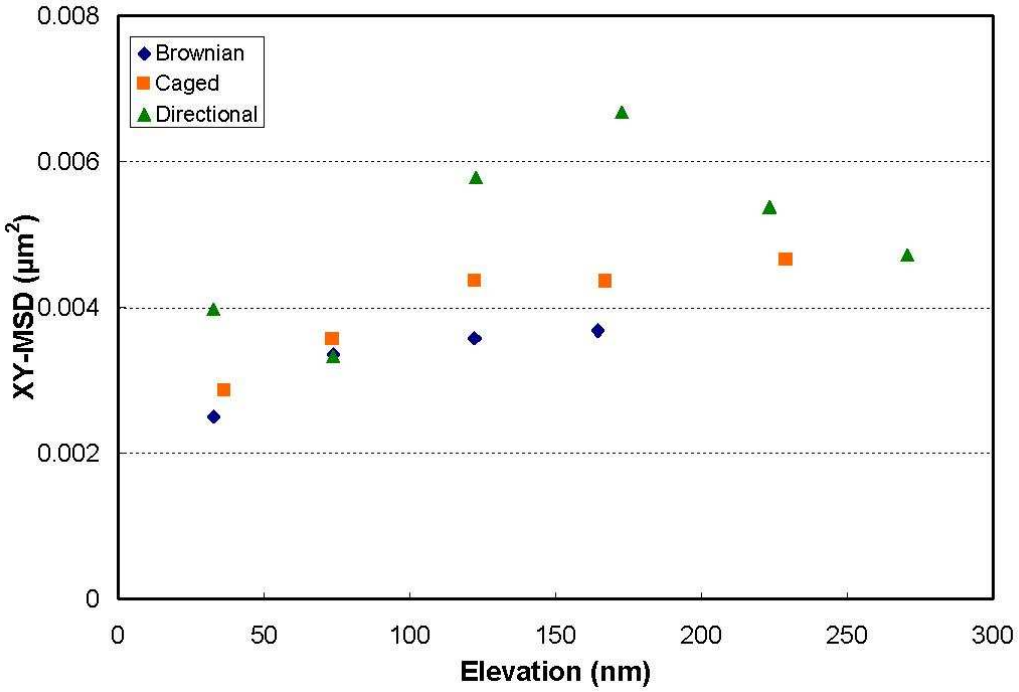


Figure 4.7 Examination of tangential hindrance effect as vesicles approach a glass coverslip.

represents an average of at least one hundred and forty data points. An increase in tangential mean square displacement was observed for both caged and Brownian type motion with increasing elevation. For directional motion an increase was initially observed, up to two hundred nanometers, followed by a decrease. Reasons for this will be discussed in detail in section 4.6.

Figure 4.8 shows the normal mean square displacement results from this analysis. While there is an initial increase in motion with increasing elevation for all three cases, the caged motion tends to asymptote while the directional motion is somewhat more erratic. Because of the few data points collected for the Brownian case it is difficult to draw definite conclusions as to its behavior. Again more discussion of these results will be included in section 4.6.

4.5.3 Examination of Particle Size and Hindrance Forces

The final goal of this work was to attempt to estimate the average vesicle size, for a given type of motion, and to examine if the hindrance of the vesicle motion was due only to hydrodynamic forces or if there were other hindering factors. As mentioned in Chapter I section 2.2 an attempt was made to estimate vesicle size using a multi-angle TIRFM approach. In this work Bovine chromatin cells were examined and average vesicle sizes ranged from 70 – 300 nm in diameter. This approach required that the incident angle be adjusted between images thus slowing down the rate of acquisition. Yoneda and Doering examined vesicle sizes inside of Cryptococcus Neoformans, a fungal organism responsible for causing fatal infections in people who are immunocompormised [74]. Using Electron Microscopy (EM) techniques vesicle sizes were measured ranging in size from 20 – 130 nm. Kelly et al. also used EM, as well as Atomic Force Microscopy (AFM), to examine Zymogen granule swelling in Pancreatic acinar cells [75]. These vesicles were significantly larger ranging in size from 200 – 1300 nm. While both EM and AFM are excellent techniques to measure vesicle sizes, they require that cells be fixed and sectioned before examination. The benefits of the method presented in this research over previous work, is the ability to maintain high image acquisition rates as well as conducting live cell experiments.

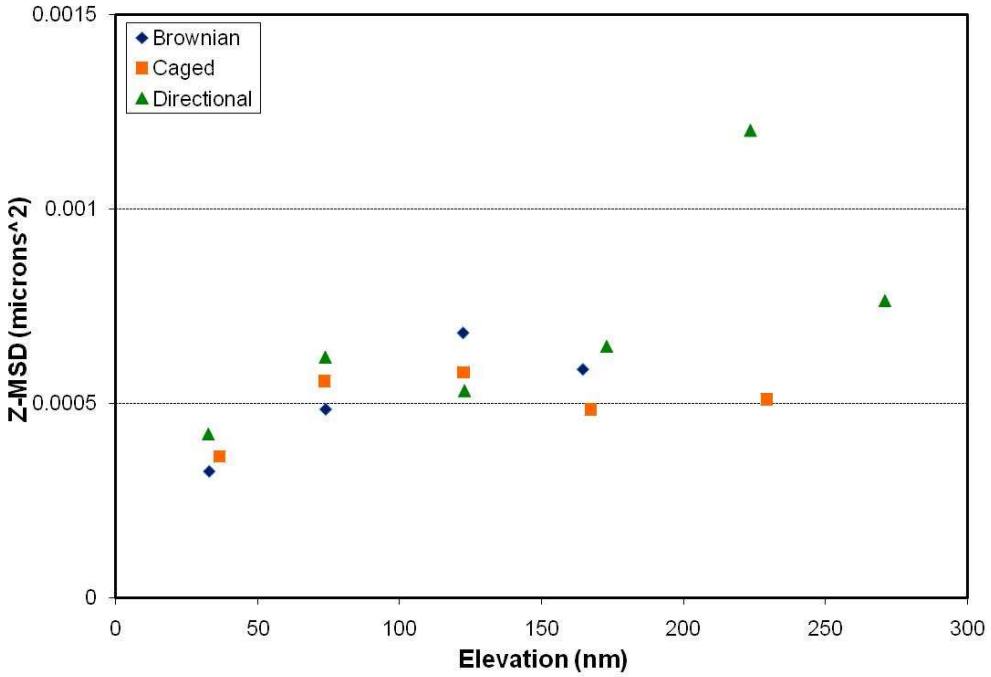


Figure 4.8 Examination of normal hindrance effect as vesicles approach a glass coverslip.

In order to estimate the particle size, based on particle motion, it was necessary to choose an appropriate theory. Form previous work with nanoparticles, discussed in Chapter II, the tangential hindrance theory proposed by Goldman was deemed the most appropriate. Because this theory is based on Brownian type motion, both the Brownian and caged particles were examined. While caged motion is not truly Brownian, it is random in nature and thus it is believed it should follow the same hindered values as purely Brownian motion.

The normal motion of the particles was not compared to the hindered theory proposed by Brenner for several reasons. First the motion of the particles shown in figure 4.8, did not appear to follow this theory. Second, the density of the vesicles is not known and therefore the gravitational force may be large enough to prevent true Brownian behavior. In other word neutral buoyancy may not be a valid assumption. Finally, if there is any significant charge to the vesicles this could introduce electrostatic forces making the theory invalid. This was observed with the charged nanoparticles discussed in Chapter II.

Figure 4.9 shows both the Brownian and caged mean square displacements along with the theoretical values obtained by fitting Goldman's theory using a nonlinear regression technique. Equation (6) provides the relationship between MSD values and the proposed theory:

$$MSD = \frac{4KT}{3\pi\mu d}\lambda_{xy}\Delta t \qquad (6)$$

where k is the Boltzmann constant, T is the absolute temperature, μ is the viscosity of the fluid, d is the diameter of the vesicle, and λ_{xy} is the hindered coefficient discussed in Chapter II. For this work all vesicles were considered to be spherical in nature. Because of the Gaussian distribution of their intensities it is believed that this assumption is reasonable, although future work should consider other possible shapes.

For this part of the analysis the only variable considered for curve fitting was the vesicle diameter. It was assumed that the viscosity of the fluid was close to that of water.

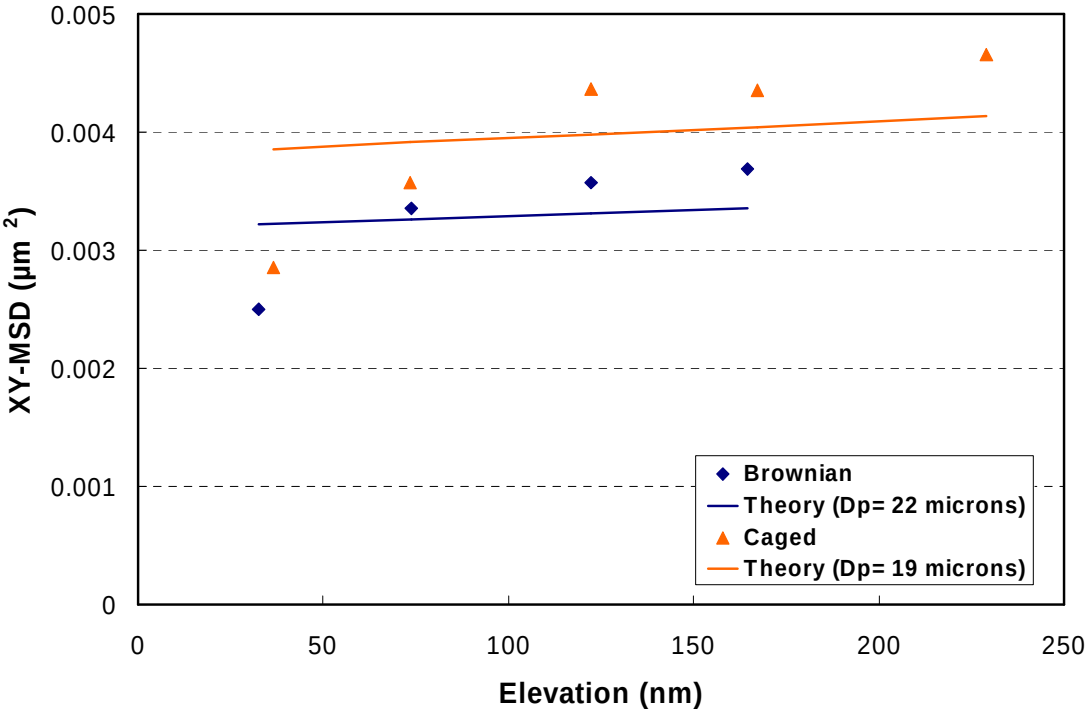


Figure 4.9 Curve fit of Goldman's hindered tangential diffusion coefficient to experimental data, with particle diameter as the only variable.

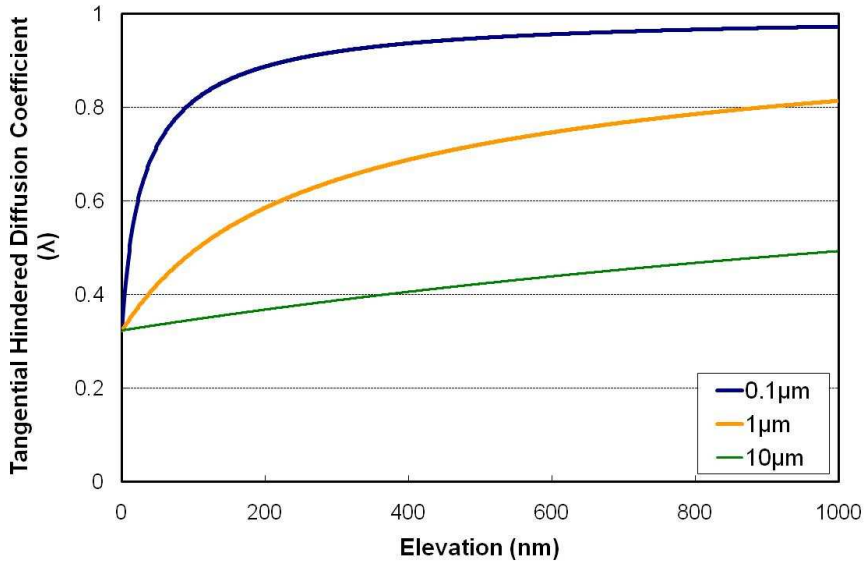
Also an incubator attached to the IX-71 microscope kept the temperature constant at 37^{o} C while maintaining a carbon dioxide concentration at 5%. This was important so that the cells were kept in a condition similar to their natural environment.

Using this method a cell diameter of twenty-two microns for Brownian type motion and nineteen microns for caged diffusion were estimated. Based on DICM images, which provided the coverage area of a single cell, a single vesicle would be equal to approximately one quarter of the total area of the cell. Images clearly show that as many as fifty vesicles can be observed within a single cell, which indicates that the calculated vesicle-size must be incorrect.

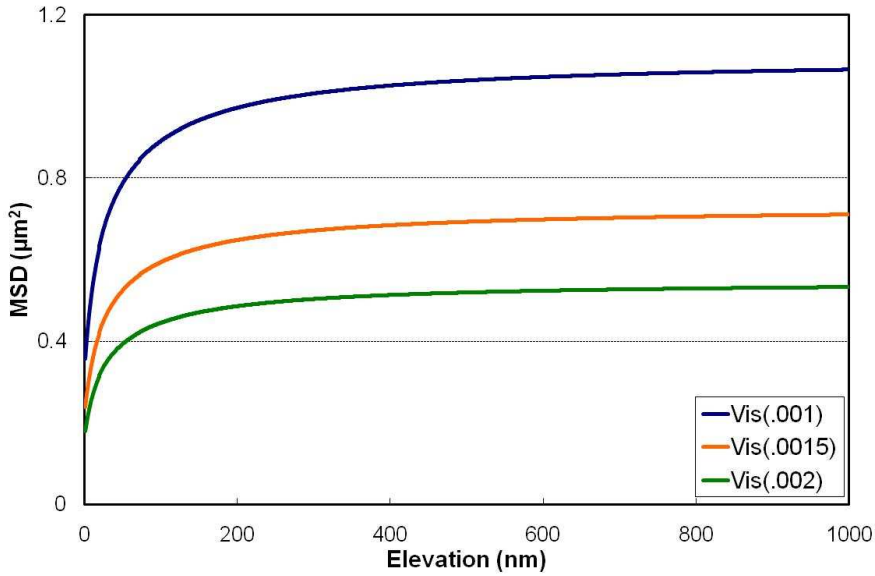
Two possibilities were considered to account for these unrealistic vesicle sizes. The first, and most obvious, is that the theory in not appropriate for this analysis. In order to determine if this were so a second possibility, that there were other hindrance effects, was examined. The next several paragraphs will attempt to determine if this hypothesis is plausible.

Figure 4.10(a) shows Goldman's tangential hindered coefficients for three different particle sizes. Figure 4.10(b) shows theoretical MSD values, based on Goldman's hindered diffusion coefficients, for a single particle size in three fluids of differing viscosities. Several important observations can be made from these curves. First the shape of the hindered coefficient curve is only a function of the particle size. Also a larger particle, at a given elevation, will have much smaller mean square displacements due to its smaller hindered coefficient. Figure 4.10(b) shows that the value of the mean square displacement for a single particle size is only a function of the fluid viscosity given that the temperature remains constant.

In order to examine the importance of these observations, with regards to determining vesicle size, consider a small spherical particle moving in a fluid with viscosity much higher than water. In this situation a lower mean square displacements would be observed, when compared to the same particle moving in water, due to the larger viscous forces. If an analysis is now conducted to determine the particle size, using the viscosity of water, then a significantly larger particle size would be predicted



(a)



(b)

Figure 4.10 (a) Tangential hindered coefficient for .1,1, and 10 µm sized particles. (b) Mean square displacement values for a .1µm sized particle at viscosities of .001, .0015, and .002 N*s/m^2 .

due to the low mean square displacements. It is believed that this is the same situation occurring when attempts were made to estimate vesicle sizes.

Therefore in order to account for these additional hindrance forces a second analysis was conducted where the viscosity of the fluid was left as a variable. Figure 4.11 shows the results of this analysis for particles exhibiting both caged and Brownian type motion. Obviously this analysis provides a much better regression than when the particle diameter is the only variable allowed to change. Discussion of these findings will be provided in the next section.

4.6 Results

Three analyses were completed involving the movement of vesicles inside of brain cancer cells. The first goal was to divide the tracked vesicles into the type of motion they exhibited during testing. It was observed that the majority of vesicles exhibited either caged or directional type motion with less than twenty percent experiencing Brownian type motion. A note of caution should be mentioned regarding the Brownian motion. Because the evidence of caged motion is directly related to the time step used, it is possible that even the vesicles showing Brownian motion could prove to be caged when examined with larger time steps. Obviously taken to the extreme all motion would be considered caged as they cannot move outside of the cellular membrane.

The next goal was to determine if evidence of hindered motion, created by the coverslip, could be observed. For vesicles experiencing either Brownian or caged motion an increase in the tangential mean square displacement was observed with increasing elevation from the surface. Directional motion initially increased with increasing elevation but then decreased. This should not be unexpected as the motion of these vesicles is being governed by non random forces. Therefore there is no reason to expect these forces would necessarily cause an increase in motion with an increase in elevation. For normal mean square displacements there was an initial increase in vesicle motion with increase in elevation for all three cases. However for caged motion it appeared that the normal mean square displacement reached a constant value, after this initial increase,

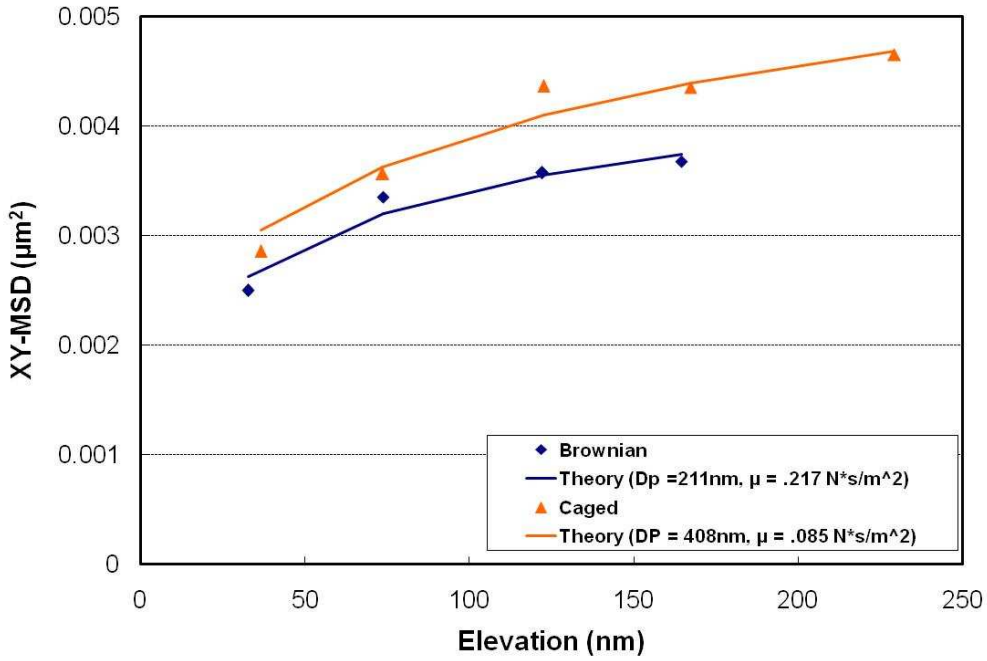


Figure 4.11 Curve fit of Goldman's hindered tangential diffusion coefficient to experimental data, with both particle diameter and viscosity left as variable.

Table 4.2 Estimated vesicle size and effective fluid viscosity for vesicles experiencing Brownian or caged motion.

	Vesicle Diameter (nm)	Effective Viscosity ($N*sec/ m^2$)	$\mu_{effective}/\mu_{water}$
Brownian	211	.217	217
Caged	408	.085	85

while the directional motion appeared to be a bit erratic. As stated previously there was not enough data for the Brownian case to reach any definite conclusions. Because of the possibility of several different forces occurring in the normal direction, including gravitational and electrostatic, which could be attractive or repulsive, it is not surprising that the hindered values were different than those predicted by purely hydrodynamic theory.

The final goal of this work was to estimate the average vesicle size as well as quantify other possible hindrance forces. By curve fitting the data taken for the tangential mean square displacement of vesicles experiencing both Brownian and caged type motion to the hydrodynamic theory developed by Goldman, and allowing both the particle diameter and fluid viscosity to be variable, excellent agreement was observed. Figure 4.11 provides both the data and curve fit while Table 4.2 provides the numerical results from this analysis.

The average size of Brownian and caged vesicles was approximately 200 nm and 400 nm respectively which agrees well with the previous work completed regarding vesicle size. The effective viscosity for the two cases was 217 and 85 times that of water. This should not suggest that the viscosity of the cellular fluid is equal to this value but that there are other forces limiting the motion of the vesicles. These forces could be collisions with other vesicles in the case where vesicle density is high. It is also possible that vesicles are tethered to the cell membrane or actin filament, which would in turn reduce their motion.

A final word of caution must be added before listing the conclusions made from this analysis. In order to curve fit this data only four points were used for the Brownian vesicles and five for the caged vesicles. Figure 4.10(a) showed that for small particles the change in the hindered coefficient with respect to elevation is significant over the distances being considered for this analysis. Thus trying to capture this curve accurately with a small number of data points leaves a great deal of room for uncertainty. Also because of the assumptions listed earlier regarding the uncertainty of the absolute elevations of these vesicles there is much room for improvement in this analysis which will be discussed in the final section.

4.6 Conclusions and Future Research

Because a significant amount of biological research has been conducted using TIRFM regarding the movement of vesicles inside of a cell, it was important to examine whether the coverslip provided a large hindrance effect on vesicle movement creating an unphysiological environment. This study showed that while evidence of hindered motion created by the surface was indeed present it appears to be orders of magnitude smaller than other hindrance forces. This is not to suggest that the coverslip may not create other unphysiological conditions, it is only to state that the hindered tangential motion does not appear to be primarily a function of hydrodynamic effects.

As stated at the end of section 4.5 there were several assumptions needed to complete this analysis that severely limits the accuracy of this method. Therefore future work should be considered to validate this method. The first step would be to examine nanoparticles of a given size in a variety of fluids of differing viscosity. Using the methods presented here it would be possible to determine if reasonable results could be found when compared with these known values. Obviously due to the large change in the hindered diffusion coefficient close to the surface, particularly for small particles, it would be important to obtain enough data to create very small elevation bins. This would also help determine what bin size is appropriate for accurate results when the particle size and viscosity are not known.

After completion of these initial experiments there are several areas where this method could be employed as a measurement device. Perhaps the easiest would be to use this method as a type of viscometer. By placing particles of a known size in an unknown fluid it would be possible to determine the viscosity of the fluid based on the particle movement. It should be noted however that the index of refraction of the fluid would need to be measured in order to get accurate elevation estimates as the two are directly related.

Another possible use of this method would be to determine the size of nanoparticles by placing them in a fluid of known viscosity. For manufacturers creating nanometer sized probes this could prove to be an effective quality control method. It could also prove to be significantly less expensive than using such devices as scanning electron microscopy or atomic force microscopy.

Finally with regards to vesicle size determination inside of a cell one of the most challenging obstacles would be to determine absolute elevations. Unlike with nanoparticles where they can be brought close to the surface by changing the ion concentration, this is not viable for cellular work. Therefore this appears to be a very challenging yet possibly beneficial area that should be considered in the future.

LIST OF REFERENCES

[1] Gray S, “Several microscopical observations and experiments, made by Mr. Stehpen Gray,” Phil Trans 19:280-287 (1695).

[2] Brown R, “A brief account of Microscopical Observations made in the months of June, July and August, 1827, on the particles contained in the pollen of plants; and on the General Existence of Active Molecules in Organic and Inorganic bodies,” Phil Mag 4: 161-173 (1828).

[3] Renn J, “Einstein’s invention of Brownian motion,” Ann. Phys. 14: supplement 23-37 (2005).

[4] Einstein A, “On the motion, required by the molecular-kinetic theory of heat, of particles suspended in a fluid at rest,” Ann. Phys. (Leipzig) 17:549-560 (1905).

[5] Perrin JB, *Atoms*, Ox Bow Press, New York (1990).

[6] Brenner H, “The slow motion of a sphere through a viscous fluid towards a plane surface,” Chem. Eng. Sci. 16: 242-25 (1961).

[7] Goldman AJ, Cox RG, and Brenner H, “Slow viscous motion of a sphere parallel to a plane-I: Motion through a quiescent fluid,” Chem. Eng. Sci. 22: 637-651 (1967).

[8] Burghardt, T., and Axelrod, D., “Total internal reflection/fluorescence photobleaching recovery study of serum-ablumin absorption dynamics,” Biophys. J.: 33, 455-567 (1981).

[9] Axelrod, D., “Cell-substrate contacts illuminated by total internal reflection fluorescence,” J. of Cell Biology: 89, 141-145 (1981).

[10] Tohmpson, N., and Axelrod, D., “Immunoglobulin surface-binding kinetics studied by total internal reflection with fluorescence correlations spectroscopy,” Biophys. J.: 43, 103-114 (1983).

[11] Prieve, D.C., Luo, F., and Lanni, F., “Brownian motion of a hydrosol particle in a colloidal force field,” Faraday Disuss. Chem. Soc.: 83, 297-307 (1987).

[12] Bike, S.G., and Prieve, D.C., “Measurements of double-layer repulsion for slightly overlapping counterion clouds,” J. Multiphase Flow: 16(4), 727-740 (1990).

[13] Prieve, D.C., and Frej, N.A., “Total internal reflection microscopy: a quantitative tool for the measurement of colloidal forces,” Langmuir: 6, 396-403 (1990).

[14] Frej, N.A., and Prieve, D.C., “Hindered diffusion of a single sphere very near a wall in a nonuniform force field,” J. of Chem. Phys.: 98, 7552-7563 (1993).

[15] Pagac, E.S., Tilton, R.D., and Prieve, D.C., “Hindered mobility of a rigid sphere near a wall,” Chem. Eng. Comm.: Vols. 148-150, 105-1122 (1996).

[16] Bevan, M.A., and Prieve, D.C., “Hindered diffusion of colloidal particles very near to a wall: revisited,” J. of chemical physics: 114, 1228-1236 (2000).

[17] Prieve, D.C., “Measurement of colloidal forces with TIRM,” Advances in Colloid and Interface Science: 82, 93-125 (1999).

[18] Schumacher, G.A., and van de Ven, T.G.M., “Evanescent wave scattering studies on latex-glass interactions,” Langmuir: 7, 2028-2033 (1991).

[19] Polverari, M., and van de Ven, T.G.M., “Effect of flow on the distrivution of colloidal particles near surfaces studied by evanescent wave light scattering,” Langmuir: 11, 1870-1876 (1995).

[20] Kihm, K.D., Banerjee, A. Choi, C.K., and Takagi, T., “Near-wall hindered Brownian diffusion of nanoparticles examined by three-dimensional ratiometric total internal reflection fluorescence microscopy (3-D R-TIRFM),” Exp. in Fluids: 37, 811-824 (2004).

[21] Margraves CH, Choi CK, and Kihm KD, “Measurements of the minimum elevation of nanoparticles by 3-D nanoscale tracking using ratiometric evanescent wave imaging,” Exp. In Fluids: 41, 173-183 (2006).

[22] Choi, C.K., Margraves, C.H., and Kihm, K.D., “Examination of near-wall hindered Brownian diffusion of nanoparticles: Experimental comparison to theories by Brenner (1961) and Goldman et al. (1967),” Physics of Fluids: 19 (2007).

[23] Jin, S., Huang, P., Park, J., Yoo, J.Y., and Breuer, K.S., “Near-surface velocimetry using evanescent wave illumination,” Exp. In Fluids: 37, 825-833 (2004).

[24] Huang, P., Guasto, J.S., and Breuer, K.S., “Direct measurement of slip velocities using three-dimensional total internal reflection velocimetry,” J. Fluid Mech.: 566, 447-464 (2006).

[25] Li, H., Sadr, R., and Yoda, M., “Multilayer nano-partilce image velocimetry,” Exp. In Fluids: 41, 185-194 (2006).

[26] Toomre, D., Keller, P., White, J., Olivo, J., and Simons, K., “Dual-color visualization of trans-Golgi network to plasma membrane traffic along microtubules in living cells,” J. of Cell Science: 112, 21-33 (1999).

[27] Toomre, D., Steyer, J.A., Keller, P., Almer, W., and Simons, K., “Fusion of constitutive membrane traffic with the cell surface observed by evanescent wave microscopy,” J. of Cell Biology: 149(1), 33-40 (2000).

[28] Schmoranzer, J., Goulian, M., Axelrod, D., and Simon, S.M., “Imaging constitutive exocytosis with total internal reflection fluorescence microscopy,” J. of Cell Biology, 149(1), 23-31 (2000).

[29] Toonen, R.F., Kochubey, O., de Wit, H., Gulyas-Kovacs, A., Konijnenburg, B., Sorensen, J.B., Klingauf, J., and Verhage, M., “Dissecting docking and tethering of secretory vesicles at the target membrane,” EMBO J.: 25, 3725-3737 (2006).

[30] Nofal, S., Becherer, U., Hof, D., Matti, U., and Rettig, J., “Primed vesicles can be distinguished from docked vesicles by analyzing their mobility,” J. of Neuroscience: 27(6), 1386-1395 (2007).

[31] Oheim, M., and Stuhmer W., “Interaction of secretory organelles with the membrane,” J. Membrane Biol.: 178, 163-173 (2000).

[32] Gerhard, J.S., Axmann, M., Freudenthaler, S., Schindler, H., Kandror, K., Roder, J.C., and Jeromin, A., “Visualization of vesicle transport along and between distinct pathways in neuritis of living cells,” Microscopy Research and Technique: 63, 159-167 (2004).

[33] Lemke, E.A., and Klingauf, J., “Single synaptic vesicle tracking in individual Hippocampal Boutons at rest during synaptic activity,” j. of Neuroscience: 25(47), 11034-11044 (2005).

[34] Yoshina-Ishii, C., Chan, Y.M., Johnson, J.M., Kung, L.A., Lenz, P., and Boxer, S.G., “Diffusive dynamics of vesicles tethered to a fluid supported bilayer by single-particle tracking,” Langmuir: 22, 5682-5689 (2006).

[35] Li, D., Xiong, A., and Xu, T., “Three-dimensional tracking of single secretory granules in live PC12 cells,” Biophysical J.: 87, 1991-2001 (2004).

[36] Li, C.H., Bai, L., Li, D.D., Xia, S., and Xu, T., “Dynamic tracking and mobility analysis of single GLUT4 storage vesicle in live 3T3-L1 cells,” Cell Research: 14(6), 480-486 (2004).

[37] Konopka, M.C., and Weisshaar, J.C., “Heterogeneous motion of secretory vesicles in the actin cortex of live cells: 3D tracking to 5-nm accuracy,” J. Phys. Chem. A: 108, 9814-9826 (2004).

[38] Wang, X., Teng, Y., Wang, Q., Li, X., Sheng, X., Zheng, M., Samaj, J., Baluska, F., and Lin, J., “Imaging of dynamic secretory vesicles in living pollen tubes of Picea meyeri using evanescent wave microcopy,” Plant Physiology: 141, 1591-1603 (2006).

[39] Medintz, I.L., Uyeda, H.T., Goldman, E.R., and Mattoussi, H., “Quantum dot biocunjugates for imaging, labeling and sensing,” Nature Materials: 4, 435-446 (2005).

[40] Ruan, G., Agrawal, A., Marcus, A.I., and Nie, S., “Imaging and tracking of Tat Peptide-conjugated quantum dots in living cells: new insights into nanopartilce uptake, intracellular transport and vesicle shedding,” J. Am. Chem. Soc.: 129, 14759-14766 (2007).

[41] Watanabe, T.M., and Higuchi, H., “Stepwise movements in vesicle transport of HER2 by motor proteins in living cells,” Biophysical J.: 92, 4109-4120 (2007).

[42] Ku, T.C., Huang, Y.N., Huang, C.C., Yang, D.M., Kao, L.S., Chiu, T.Y., Hsieh, C.F., Wu, P.Y., Tsai, and Y.S., Lin, C.C., “An automated tracking system to measure the dynamic properties of vesicles in living cells,” Microscopy Research and Technique: 70, 119-134 (2007).

[43] Singh, S., Bhaskar, H., Tavare, J., and Welsh, G., “Multiple particle tracking for live cell imaging with green fluorescent protein (GFP) tagged videos,” Pattern

Recognition and Image Analysis, Volume 3687, Springer Berlin / Heidelberg, 792-803 (2005).

[44] Cheezum, M.K., Walker, W.F., and Guilford, W.H., "Quantitative comparison of algorithms for tracking single fluorescent particles," Biophysical J.: 81, 2378-2388 (2001).

[45] Saxton, M.J., and Jacobson, K., "Single-particle tracking: applications to membrane dynamics," Annu. Rev. Biophys. Ciomol. Struct.: 26, 373-399 (1997).

[46] Rohrback, A., "Observing secretory granules with a multiangle evanescent wave microscope," Biophysical J.: 78, 2641-2654 (2000).

[47] Baek, S.J., Kim, J.S., Nixon, J.B., DiAugustine, R.P., and Eling, T.E., "Expression of NAG-1, a transforming growth factor-β superfamily member, by Troglitazone requires the Early Growth Response Gene EGR-1," J. Biological Chemistry: 279(8), 6883-6892 (2004).

[48] Shim, M.S., and Eling, T.E., "Protein Kinase C-dependant regulation of NAG-1/Placental Bone Borphogenic Protein/MIC-1 expression in LNCaP Prostate Carcinoma Cells," J. Biological Chemistry: 280(19), 18636-18642 (2005).

[49] Yamaguchi, K., Liggett, J.L., Kim, N.C., and Baek, S.J., "Anto-proliferative effect of horehound leaf and wild cherry bark extracts on human colorectal cancer cells," Oncology Reports: 15, 275-281 (2006).

[50] Okazaki, R., Moon, Y., Norimura, T., and Eling, T., "Ionizing radiation enhances the expression of the Nonsteroidal Anti-inflammatory Drug-Activate Gene (NAG1) by incrasing the expression of TP53 in human colon cancer cells," Radiation Research: 165, 125-130 (2006).

[51] Eling, T.E., Baek, S.J., Shim, M., and Lee, C.H., "NSAID Activated Gene (NAG-1), a modulator of tumorigensis," J. of Biochemistry and Molecular Biology: 39(6), 649-655 (2006).

[52] Hecht E , *Optics, 4th ed.*, Addison Wesley, San Francisco, 2002.

[53] Wu HJ, and Bevan MA, "Direct Measurement of Single and Ensemble Average Particle-Surface Potential Energy Profiles," Langmuir: 21, 1244-1254 (2005).

[54] Russel W.B., Saville DA, and Schowalter, W.R., *Colloidal dispersions*, Cambridge University Press, New York, New York (1999)

[55] Weiss, T.F., *Cellular Biophysics Transport, Vol. 1*, The MIT Press, Massachusetts, 1996.

[56] Behrens SH, and Grier DG, "The Charge of Glass and Silica Surfaces," J. Chem. Phys.: 115(14), 6716-6721 (2001).

[57] Song, L., Hennink, E.J., Young, I.T., and Tanke, H.J., "Photobleaching kinetics of fluorescein in quantitative fluorescence microscopy," Biophysical J.: 68, 2588-2600 (1995).

[58] Kline SJ, and McClintock FA, "Describing uncertainties in single sample experiments," Mech Eng: 75(1), 3–9 (1953).

[59] Giaever, I., and Keese, C. R., "Monitoring fibroblast behavior in tissue culture with an applied electric field," Proc. Natl. Acad. Sci. USA: 81(12), 3761-3764 (1984).

[60] Giaever, I., and Keese, C. R., "Micromotion of mammalian cells measured electrically," Proc. Natl. Acad. Sci. USA: 88(17), 7896-7900 (1991).

[61] Penza, M., Cozzi, S., Tagliente, M.A., Mirenghi, L., Martucci, C., and Quirini, A., "Characterization of transparent and conductive electrodes of indium tin oxide thin films by sequential reactive evaporation," Thin Solid Films: 349(1), 71-77 (1999).

[62] Choi, C.K., English, A.E., Kihm, K.D., and Margraves, C.H., "Simultaneous dynamic optical and electrical properties of endothelial cell attachment on indium tin oxide bioelectrodes," J. Biomed. Opt.: 12(6), 064028 (2007).

[63] Inoue, S., Spring, K.R., *Video Microscopy The Fundamentals*, 2nd ed., Plenum Press, New York, (1997).

[64] Lucy, L.B., "An iterative technique for the rectification of observed distributions," The Astronomical J.: 79(6), 745-754 (1974).

[65] Parker, J.R., *Algorithms For Image Processing and Computer Vision*, John Wiley and Sons, Inc., New York (1997).

[66] Curtis, A.S.G., “The mechanism of adhesion of cells to glass,” J. Cell Biol.: 20, 199-215 (1964).

[67] Izzard, C.S., and Lochner, L.R., “Cell-to-substrate contacts in living fibroblasts: an interference reflexion study with an evaluation of the technique,” J. Cell Sci.: 21, 129-159 (1976).

[68] Gingell, D., and Todd, I., “Interference reflection microscopy a quantitative theory for image interpretation and its application to cell-substratum separation measurement,” Biophys. J.: 26, 507-526 (1979).

[69] Schindl, M., Deubzer, E. B., Witke, W., Gerisch, G., and Sackmann, E., “Cell-substrate interactions and locomotion of Dictyostelium wild-type and mutants defective in three cytoskeletal proteins: a study using quantitative reflection interference contrast microscopy,” Biophys. J.: 68, 1177-1190 (1995).

[70] Richter, E., Hitzler, H., Zimmermann, H., Hagedorn, R., and Fuhr, G., “Trace formation during locomotion of L929 mouse fibroblasts continuously recorded by interference reflection microscopy (IRM),” Cell Motility and the Cytoskeleton: 47, 38-47 (2000).

[71] Choi, C.K., Margraves, C.M., English, A.E., and Kihm, K.D., “A multi-contrast microscopy technique to dynamically fingerprint live cell focal contacts during exposure and replacement of a cytotoxic medium,” J. of Biomedical Optics, [Accepted].

[72] Vershueren, H., “Interference reflection microscopy in cell biology: methodology and applications,” J. Cell Sci.: 75, 279-301 (1985).

[73] Munson, B.R., Young, D.F., and Okiishi, T.H., *Fundamentals of Fluid Mechanics*, 2nd ed., John Wiley and Sons, Inc., New York (1990).

[74] Yoneda, A., and Doering, T.L., “A Eukaryotic capsular Polysaccharide is synthesized intracellularly and secreted via exocytosis,” Molecular Biology of the Cell: 17, 5131-5140 (2006).

[75] Kelly, M.L., Cho, W.J., Jeremic, A., Abu-Hamdah, R., and Jena, B.P., “Vesicle swelling regulates content expulsion during secretion,” Cell Biology International: 28, 709-716 (2004).

APPENDIX A

Brownian Motion Code

```
%Program:  BMC
%Objective:  Using TIF images particle motion is tracked from one frame to the next.
%The brigtest pixel of a particle is used to track it in both the X-Y plane
%as well as the Z-direction.  MSD values are calculated experimentally
%through the measured movement as well as theoretically.  To get
%theoretical predictions hindered diffusion coefficients are obtained at
%each measured height.

%Clear all memory
clear all; clc;

%Open files for output - Sort_out.txt (Data sorted into bins determined by
%user.  All_out.txt (All data output, sorted by elevation)

%Block 1
dia_entries = {'Test Number','End File Number'};
test_num = inputdlg(dia_entries,'Input Desired Test Number',1);
first_test = str2num(char(test_num(1)));
last_test = str2num(char(test_num(2)));
%End Block 1

%Block 2
for set = first_test:last_test
   so = ['Sort_out' int2str(set) '.txt'];
   ao = ['All_out' int2str(set) '.txt'];
   aod = ['All_out_data' int2str(set) '.txt'];
   sd = ['Sort_data' int2str(set) '.txt'];
   dis = ['Displacements' int2str(set) '.txt'];
   fid = fopen(so,'w');
   fid1 = fopen(aod,'w');
   fid2 = fopen(ao,'w');
   fid4 = fopen(sd,'w');
   fid5 = fopen(dis,'w');
   fprintf(fid,'  Elevation   X-MSD     Y-MSD     Z-MSD     3D-MSD    2D-Th    1D-Th     3D-Th
Data\n');
   fprintf(fid2,'  Elevation   X-MSD     Y-MSD     Z-MSD     3D-MSD    2D-Th    1D-Th     3D-Th
\n');
   fprintf(fid1,'  Elevation   X-MSD     Y-MSD     Z-MSD     3D-MSD    2D-Th    1D-Th     3D-Th
\n');
   fprintf(fid5,'  Elevation1   Elevation2    DeltaH    AverageH     DeltaX     DeltaY     Window
File\n');
%End Block 2

%Block 3
   %System Variables read in from file parameters.txt
   fid3 = fopen('parameters.txt');
   parameters = textscan(fid3,'%12s %10.5f');
   fclose(fid3);
   ni = parameters{1,2}(1);            %Index of Refraction(glass)
   nt = parameters{1,2}(2);            %Index of Refraction(water)
   theta = parameters{1,2}(3);         %Incident angle of incoming light
```

```
   lambda = parameters{1,2}(4);         %Wavelength of laser
   maxI = parameters{1,2}(5);          %Intensity at Surface
   pix_size = parameters{1,2}(6);       %CCD Pixel Size
   diam = parameters{1,2}(7);          %Particle Diameter
   delta_time = parameters{1,2}(8);     %Time Between image recording
   thresh = parameters{1,2}(9);        %Intensity threshold (determines if particle is in view)
   BK = parameters{1,2}(10);           %Coefficient for diffusion calc.
   Temp = parameters{1,2}(11);          %Ambient Temperature
   mu = parameters{1,2}(12);           %Fluid Viscocity
   max_h = parameters{1,2}(13);         %Maximum height to be examined for output
   h_incr = parameters{1,2}(14);       %Number of height increments

   %Read in background noise and intensity map
   back_temp = dlmread('back.txt');
   map_temp = dlmread('map.txt');
   for ii = 1:512
      for jj = 1:512
         back(ii,jj) = back_temp(ii,jj);
         map(ii,jj) = map_temp(ii,jj);
      end
   end
%End Block 3

%Block 4
   %Set intensity profile for glass surface.  This will be used to measure
   %actual heights of data. Also calculate zp.
   I_O = maxI*map;
   Izp = I_O*exp(-1);
   zp = (lambda/(4*pi))*((ni^2*sind(theta)^2-nt^2)^(-.5)); %Penetration Depth
%End Block 4

%Block 5
   %Interrogation window input: includes x and y locations of top left corner
   %and bottom right corner, beginning file, ending file, and # of file sets.
   %File name - windows.txt
   win_temp = ['..\test',int2str(set),'\windows' int2str(set) '.txt'];
   window = dlmread(win_temp);
   [win_r win_c] = size(window);
   fprintf(fid4,'%5.0f %5.0f \n',h_incr,win_r);
   [inrow incol] = size(window);


   for in = 1:inrow
      xpixi = window(in,1);
      ypixi = window(in,2);
      xpixf = window(in,3);
      ypixf = window(in,4);
      start_file = window(in,5);
      end_file = window(in,6);
      %set = window(in,7);
      max_files = end_file - start_file;
      col = (xpixf - xpixi)+1;
      row = (ypixf - ypixi)+1;
      step = 1;
```

```
      count = 1;
%End Block 5
%Block 6
      %Read in entire image(B) and select range to be looked at(A).  Repeat for
      %all desired images.
      %for set = 1:max_set
      for zz = start_file:end_file
         B = im_read(zz,set) - back;
         for ii = 1:row
            for jj = 1:col
               A(ii,jj) = B(ii-1+ypixi,jj-1+xpixi);
            end
         end

%End Block 6

%Block 7
         %Determine the peak value for each frame.  If it is between the
         %preset limits also record its location.
         [peak_val x_val y_val] = maxn(A);
         [check1 dum] = size(x_val);
         if peak_val > thresh & peak_val < I_O(x_val,y_val) & check1 ==1
            h(count) = -zp*log(peak_val/I_O(x_val,y_val));
            peak_data(count) = peak_val;
            r_loc(count) = x_val;
            c_loc(count) = y_val;
            count = count+1;
         else
            h(count) = -1;
            peak_data(count) = -1;
            r_loc(count) = -1;
            c_loc(count) = -1;
            count = count+1;
         end
         step = step + 1;
      end
      %end
      count = count - 1;
%End Block 7

%Block 8
      %Calculate Experimental Displacements and Root Mean Square Values.  A
      %check is also performed to insure measurements are made between two
      %consecutive files.  This ensures that if a particle goes out of the
      %viewing zone and then back in later, measurements are not made between
      %these two positions.
      pos_loc = 1;
      for ii = 1:count-1
         if r_loc(ii) == -1 | r_loc(ii+1) == -1 | h(ii) >max_h | h(ii+1) > max_h
            dum = -1;
         else
            delx(pos_loc) = (c_loc(ii+1) - c_loc(ii)).*pix_size;
            if delx(pos_loc) == 0
               delx(pos_loc) = .5.*pix_size;
```

```
            end
            x_sq(pos_loc) = delx(pos_loc).^2;
            dely(pos_loc) = (r_loc(ii+1) - r_loc(ii)).*pix_size;
            if dely(pos_loc) == 0
               dely(pos_loc) = .5.*pix_size;
            end
            delz(pos_loc) = h(ii+1) - h(ii);
            y_sq(pos_loc) = dely(pos_loc).^2;
            z_sq(pos_loc) = delz(pos_loc).^2;
            r_sq(pos_loc) =x_sq(pos_loc)+y_sq(pos_loc)+z_sq(pos_loc);
            havgtemp = (h(ii)+h(ii+1))/2;
            fprintf(fid5,'  %7.2f      %7.2f       %7.2f    %7.2f      %7.2f     %7.2f   %7.2f   %5.0f
\n',h(ii),h(ii+1),delz(pos_loc),havgtemp,delx(pos_loc),dely(pos_loc),in,start_file+ii);
            pos_loc = pos_loc + 1;
         end
      end

      delx_avg = mean(delx);
      dely_avg = mean(dely);
      delz_avg = mean(delz);
      x_avg = (mean(x_sq))./(1000^2);
      y_avg = (mean(y_sq))./(1000^2);
      z_avg = (mean(z_sq))./(1000^2);
      r_avg = (mean(r_sq))./(1000^2);
%End Block 8

%Block 9
      %Calculate Theoretical Diffusion Coefficients based on measured heights
      %Elevation matrix containing only measured heights is created
      data_points = count;
      jj = 1;
      for ii = 1:length(h)-1
         if h(ii)~=-1 & h(ii+1)~=-1
            if h(ii) <= max_h & h(ii+1) <= max_h
               elevation(jj) = (h(ii)+h(ii+1))/2;
               beta(jj) = 1 - (9/16)*(diam/(diam+2*elevation(jj))) +
(1/8)*(diam/(diam+2*elevation(jj)))^3 ...
                  -(45/256)*(diam/(diam+2*elevation(jj)))^4 - (1/16)*(diam/(diam+2*elevation(jj)))^5;
               alpha(jj) = acosh((2*elevation(jj)+diam)/diam);
               for n = 1:5
                  term1 = (n*(n+1))/((2*n-1)*(2*n+3));
                  num = (2*sinh((2*n+1)*alpha(jj)))+((2*n+1)*sinh(2*alpha(jj)));
                  den = (4*((sinh((n+.5)*alpha(jj))).^2)) - (((2*n+1).^2)*(sinh(alpha(jj)).^2));
                  term2 = (num/den)-1;
                  sum_temp(n) = term1*term2;
               end
               zeta(jj) = 1/((4/3)*(sinh(alpha(jj))*sum(sum_temp)));
               jj = jj+1;
            end
         end
      end

      beta_avg = mean(beta);
      zeta_avg = mean(zeta);
```

```
    %Calculate Theoretical MSD values and Diffusion Coefficients
    free_diff = (BK*Temp)/(3*mu*pi*(diam*1e-9))*1000000^2;
    norm_diff = zeta*free_diff;
    tang_diff = beta*free_diff;
    psi = (2*beta+zeta)/3;
    hind_diff = psi*free_diff;
    x_msd_theory = 2*tang_diff*delta_time;
    z_msd_theory = 2*norm_diff*delta_time;
    msd_theory = hind_diff*6*delta_time;
    min_elevation = min(elevation);
    elevation_average = mean(elevation);
    max_elevation = max(elevation);
    data_points_last = size(delx);

%End Block 9

%Block 10
    %Calculate Frequency vs. Height for histograms
    delta_h = max_h/h_incr;
    h_bins = zeros(h_incr,1);
    for ii = 1:h_incr
       bin_plot(ii) = (ii*delta_h + (ii-1)*delta_h)/2;
       limits (ii) = ii*delta_h;
    end

    for ii = 1:length(elevation)
       jj = 1;
       while jj <= h_incr
          if elevation(ii) < jj*delta_h & elevation(ii)>= (jj-1)*delta_h;
             h_bins(jj) = h_bins(jj) + 1;
             jj = h_incr+1;
          else
             jj = jj+1;
          end
       end
    end
%End Block 10

%Block 11
    %Set up data stores for msd values at given height
    for ii = 1:h_incr
       if h_bins(ii) > 0
          msd_store{ii} = ones(h_bins(ii),8);
       elseif h_bins(ii)==0
          msd_store{ii} = zeros(1,8);
       end
    end

    %Create matrix containing elevation and all calculated and theoretical
    %values.  Then sort the matrix, by elevation, from least to
    %greatest

    msd_matrix = [elevation' x_sq'  y_sq' z_sq' r_sq' x_msd_theory' z_msd_theory' msd_theory'];
```

```
      msd_sort = sortrows(msd_matrix);
      msd_sort(:,2) = msd_sort(:,2)./1000^2;
      msd_sort(:,3) = msd_sort(:,3)./1000^2;
      msd_sort(:,4) = msd_sort(:,4)./1000^2;
      msd_sort(:,5) = msd_sort(:,5)./1000^2;

      % Sort Data into stores
      dum = 0;
      for ii = 1: h_incr
         if msd_store{ii}~=0
            for jj = 1:h_bins(ii)
               for kk = 1:8
                  msd_store{ii}(jj,kk) = msd_sort(jj+dum,kk);
               end
            end
         end
         dum = dum + h_bins(ii);
      end

      %Take average values for data in each bin
      for ii = 1:h_incr
         if h_bins(ii) > 0
            for jj = 1:8
               ele_output(ii,jj) = mean(msd_store{ii}(:,jj));
            end
            ele_output(ii,9) = h_bins(ii);
         else
            for jj = 1:8
               ele_output(ii,jj) = 0;
               ele_output(ii,9) = 0;
            end
         end
      end


      %Create Output files
      fprintf(fid,'%10.3f %10.7f %10.7f %10.7f %10.7f %10.7f %10.7f %10.7f %4.0f\n',ele_output');
      fprintf(fid4,'%10.3f %10.7f %10.7f %10.7f %10.7f %10.7f %10.7f %10.7f
%4.0f\n',ele_output');
      if in < inrow
         fprintf(fid,'Track %3.0f\n',in+1);
      end
      fprintf(fid2,'%10.3f %10.7f %10.7f %10.7f %10.7f %10.7f %10.7f %10.7f \n',msd_sort');
      fprintf(fid1,'%10.3f %10.7f %10.7f %10.7f %10.7f %10.7f %10.7f %10.7f \n',msd_sort');
      if in < inrow
         fprintf(fid2,'Track %3.0f\n',in+1);
      end

      %Clear Variables for next run
      clear ele_output h peak_data r_loc c_loc delx dely delz x_sq y_sq z_sq r_sq
      clear delx_avg dely_avg delz_avg x_avg y_avg z_avg elevation beta alpha
      clear zeta msd_store msd_matrix msd_sort A
   end
%End Block 11
```

```
%Block 12
   fclose(fid);
   fclose(fid2);
   fclose(fid1);
   fclose(fid4);
   fclose(fid5);
end
%End Block 12
```

Illumination Unevenness/Camera Noise Code

```
%Determines the spatial variation in intensity due to unevenness of the illumination
% and creates a map in order to correct for variation in TIRFM measurements
%Similar method is used to create background map.  However there is no normalization for this
%process.  Only average values are created at each pixel and then output to a file.

clear all; clc;
start_file = 1;
end_file = 168;
set=1;
count = 1;
track = 512;
dum = 0;

%Read in background file (Camera Noise)
back_temp = dlmread('back.txt');
for ii = 1:512
   for jj = 1:512
      back(ii,jj) = back_temp(ii,jj);
   end
end

%Begin reading image files
int = zeros(512,512);  %512 by 512 image
for zz = start_file:end_file
   sum = 0;
   std_sum = 0;
   n = 0;
   b= im_read(zz,set);
   b = b-back;

   %sum files for average at each pixel
   int = int+b;
end
%Average intensity for all files at each pixel and then output to map file.
int = int/(end_file);
int_total_avg = mean(int_avg);
map = int/int_total_avg;
fid = fopen('Map.txt','w');
for ii = 1:512
   for jj = 1:512
      if jj ~= 512
      fprintf(fid,'%7.5f  ',map(ii,jj));
      else
         fprintf(fid,'%7.5f  ',map(ii,jj));
         fprintf(fid,'\n');
      end
   end
end
fclose(fid);
```

Stuck Particle Intensity

```
%Program:  Stuck Particle
%Objective:  Examine intensity variation of stuck particles to determine
%average stuck particle intensity.

clear all; clc;

dia_entries = {'Test Number','End File Number'};
test_num = inputdlg(dia_entries,'Input Desired Test Number',1);
first_test = str2num(char(test_num(1)));
last_test = str2num(char(test_num(2)));
fid2 = fopen('Stuck_Average','w');
for set = first_test:last_test
   %Input for file size counters and threshold
   xpixi = 1;
   ypixi = 1;
   xpixf = 512;
   ypixf = 512;
   max_files = 167;
   count = 1;
   step = 1;
   thresh = 4000;
   max_I = 16383;
   map_temp = dlmread('map.txt');
   back_temp = dlmread('back.txt');
   for ii = 1:512
      for jj = 1:512
         map(ii,jj) = map_temp(ii,jj);
         back(ii,jj) = back_temp(ii,jj);
      end
   end
   clear back_temp map_temp;

   s = ['..\test',int2str(set),'\',int2str(set),'MyStream0001.tif'];
   A = double(imread(s,'tif'));
   %A = (A-back)./map;

   %Find all of the stuck particles from the initial file
   for ii = 5:xpixf-4
      for jj = 5:ypixf-4
         if A(ii,jj) > thresh & A(ii,jj)<max_I
            if A(ii,jj)>A(ii-1,jj+1)&...
                  A(ii,jj)>A(ii,jj+1)&...
                  A(ii,jj)>A(ii+1,jj+1)&...
                  A(ii,jj)>A(ii-1,jj)&...
                  A(ii,jj)>A(ii+1,jj)&...
                  A(ii,jj)>A(ii-1,jj-1)&...
                  A(ii,jj)>A(ii,jj-1)&...
                  A(ii,jj)>A(ii+1,jj-1);
               xpeak(count) = ii;
               ypeak(count) = jj;
               count = count+1;
```

```
            end
        end
    end
end

%Using stuck particle location measure intensity for each particle over all
%files and calculate statistics for all particles.
peak_data = peak_vector(xpeak,ypeak,max_files,set);
peak_max = max(peak_data);
peak_min = min(peak_data);
peak_avg = mean(peak_data);
peak_std = std(peak_data);
[m n] = size(peak_data);
plot(peak_data);
peak_avg = peak_avg';

%Output Data to file
[m n] = size(peak_data);
st = ['Stuck' int2str(set) '.txt'];
fid = fopen(st,'w');
for ii = 1:m
    for jj = 1:n
        fprintf(fid,'%5.0f  ',peak_data(ii,jj));
    end
    fprintf(fid,'\n');
end
fclose(fid);

%     sta = ['Stuck_Average' int2str(set) '.txt'];
%     fid2 = fopen('Stuck_Average','w');
for ii = 1:length(peak_avg);
    fprintf(fid2,'%7.2f \n',peak_avg(ii));
end

end
fclose(fid2);
```

Maximum Intensity Finder

```
function[x_loc y_loc] = max_loc_finder(peak_val,A,row,col)

%This function determines the pixel location for the maximum value
%location/locations.  If there is more than one location it averages
%them and returns this value

dum = 1;
for ii = 1:row
   for jj = 1:col
      if A(ii,jj) == peak_val;
         xx(dum) = ii;
         yy(dum) = jj;
         dum = dum+1;
      end
   end
end
x_loc = mean(xx);
y_loc = mean(yy);
return;
```

Data Analysis

```
%Code:  Data_Analysis
%This code reads input files created by the Brownian motion code and
%analyzes the data.  MSD values can then be plotted versus elevation.
clear all; clc;

%Input Parameters
max_h = 500;
h_incr = 20;
part_size = 500;
size1 = char(num2str(part_size));

%User enters the number of files to be examined
dia_entries = {'Test Number','End File Number'};
test_num = inputdlg(dia_entries,'Input Desired Test Number',1);
first_test = str2num(char(test_num(1)));
last_test = str2num(char(test_num(2)));

%Set Loop to make one long vector for each measurement of interest
for ii = first_test:last_test
   s = ['Displacements' int2str(ii) '.txt'];
   t = ['All_out_data' int2str(ii) '.txt'];
   [EL1 EL2 delz avgh delx dely win file] = textread(s,'%f %f %f %f %f %f %f %f ','headerlines',1);
   [ELAVG XMSD YMSD ZMSD MSD3 XTH ZTH TH3D] = textread(t,'%f %f %f %f %f %f %f %f
','headerlines',1);
   if ii == first_test
      elevation = EL1;
      el_avg = ELAVG;
      x_msd = XMSD;
      y_msd = YMSD;
      z_msd = ZMSD;
      count(1) = length(EL1);
   else
      sum_count = sum(count);
      count(ii) = length(EL1);
      for jj = 1:length(EL1)
         elevation(sum_count+jj) = EL1(jj);
         el_avg(sum_count+jj) = ELAVG(jj);
         x_msd(sum_count+jj) = XMSD(jj);
         y_msd(sum_count+jj) = YMSD(jj);
         z_msd(sum_count+jj) = ZMSD(jj);
      end
   end
end

xy_msd = (x_msd+y_msd)./2;

%Calculate Frequency vs. Height for histograms
delta_h = max_h/h_incr;
h_bins = zeros(h_incr,1);
for ii = 1:h_incr
```

```
   bin_plot(ii) = (ii*delta_h + (ii-1)*delta_h)/2;
   limits (ii) = ii*delta_h;
end

for ii = 1:length(elevation)
   jj = 1;
   while jj <= h_incr
      if elevation(ii) < jj*delta_h & elevation(ii)>= (jj-1)*delta_h;
         h_bins(jj) = h_bins(jj) + 1;
         jj = h_incr+1;
      else
         jj = jj+1;
      end
   end
end

%Normalized h_bins for histograms
h_bins_norm = h_bins./sum(h_bins);

%Plot Histogram
hist(elevation,bin_plot);
title('Realization vs. Elevation');
t2 = [size1 'nm'];
text(425,1400,t2);
ylim([0 1800]);
ylabel('Realizations');
xlabel('Elevation');

%*************************************************************************%
%Begin examinination of MSD values

%Sort msd values by average elevation (least to greatest) in order to store in
%appropriate bins
msd_matrix = [el_avg xy_msd z_msd];
msd_sort = sortrows(msd_matrix);

%Create Bins for average elevation
h_bins_avg = zeros(h_incr,1);
for ii = 1:length(el_avg)
   jj = 1;
   while jj <= h_incr
      if el_avg(ii) < jj*delta_h & el_avg(ii)>= (jj-1)*delta_h;
         h_bins_avg(jj) = h_bins_avg(jj) + 1;
         jj = h_incr+1;
      else
         jj = jj+1;
      end
   end
end

%Set up data stores for msd values at given height
for ii = 1:h_incr
   if h_bins_avg(ii) > 0
      msd_store{ii} = ones(h_bins_avg(ii),3);
```

```
    elseif h_bins(ii)==0
        msd_store{ii} = zeros(1,8);
    end
end

%Sort Data into stores for statistical analysis
count = 1;
for ii = 1:h_incr
    for jj = 1:h_bins_avg(ii)
        msd_store{ii}(jj,1) = msd_sort(count,1);
        msd_store{ii}(jj,2) = msd_sort(count,2);
        msd_store{ii}(jj,3) = msd_sort(count,3);
        count = count+1;
    end
end

%Determine Average values and standard deviations for each bin
for ii = 1:h_incr
    if h_bins_avg(ii) > 0
        for jj = 1:3
            avg_output(ii,jj) = mean(msd_store{ii}(:,jj));
            avg_output(ii,3+jj) = std(msd_store{ii}(:,jj));
        end
        avg_output(ii,7) = h_bins_avg(ii);
    else
        for jj = 1:3
            avg_output(ii,jj) = 0;
            avg_output(ii,3+jj) = 0;
            avg_output(ii,7) = 0;
        end
    end
end
```

APPENDIX B

Cell Coverage Area Calculations

```
%Program: Cell_Electrode
%This code uses a series of filters to seperate the cell covered area from
%the background area.  DICM images are used as input.

%clear all variables
clear all; clc;
pixel_area = .00025^2;  %Area in mm^2

%Cacluclate initial area of the examined region
tol = 470;  %maximum number of pixels from the center to be include
Initial_Area = pi*(.00025*tol)^2

%Set up output file
fid = fopen('Area.txt','w');
fprintf(fid,'Total Area = %10.3f\n',Initial_Area);
%Enter input for the files desired
dia_entries = {'start file','end file'};
%def = {'270','338','1','1'};
dum = inputdlg(dia_entries,'Input Pixel Data and # of Files to be Examined',1);
start_file = str2num(char(dum(1)));
end_file = str2num(char(dum(2)));

%Read in initial image file
PSF = fspecial('gaussian',7,10);
for zz = start_file:end_file
   %Deconvolve image using Lucy-Richardson algorithm with a Gaussian PSF.
   Initial = double(im_read(zz,1));
   Initial = deconvlucy(Initial,PSF);
   [row col] = size(Initial)
   row_cent = row/2;
   col_cent = col/2;

   %Determine area to be examined for cell coverage
   for ii = 1:row
      for jj = 1:col
         if (ii-row_cent)^2+(jj-col_cent)^2 > tol^2
            Initial(ii,jj) = 0;
         end
      end
   end

   %Divide Image into 9 sections
   init = mat2cell(Initial,[333 333 334],[333 333 334]);
   [cell_row cell_col] = size(init);
   for ii = 1:cell_row
      for jj = 1:cell_col
         [r1(ii,jj) c1(ii,jj)] = size(init{ii,jj});
      end
   end

   %Calculate average background for the sections to use for thresholding
```

```
for ii = 1:cell_row
   for jj = 1:cell_col
      sec_avg(ii,jj) = avg_calc(ii,jj,r1,c1,init);
   end
end

%Make two copies of image for examination using threshold and edge
%filters.
im_temp = Initial;
im1 = init;

%Use canny or sobelfilter to determine edge and then set pixels outside of the
%boundary to zero
if zz < 7
[im_edge,im_thresh] = edge(im_temp,'sobel');
else
   [im_edge,im_thresh] = edge(im_temp,'canny');
end
for ii = 1:row
   for jj = 1:col
      if (ii-row_cent)^2+(jj-col_cent)^2 > (tol-1)^2
         im_edge(ii,jj) = 0;
      end
   end
end

%Set area outside the electrode to zero
for ii = 1:row
   for jj = 1:col
      if Initial(ii,jj) == 0
         im_edge(ii,jj) = 0;
      end
   end
end

%Use thresholding technique
for ii = 1:cell_row
   for jj = 1:cell_col
      im1 = sort_cells(ii,jj,r1,c1,init,im1,sec_avg);
   end
end
im_1 = cell2mat(im1);
im_1_temp = im_1;
%
%
% %Compare sobel image and thresholding image to attempt to close cell
% %boundary
for ii = 1:row
   for jj = 1:col
      if im_edge(ii,jj) == 1
         im_1(ii,jj) = im_temp(ii,jj);
      end
   end
end
```

```
im_edge_thresh = im_1;


%Fill in open spaces on the diagonal
for ii = 2:row-1
   for jj = 2:col-1
      if im_1(ii-1,jj+1)==20000 & im_1(ii,jj) ~= 0
         count = 1;
         while im_1(ii-count,jj+count) == 20000
            count = count+1;
         end
         if count < 15 & im_1(ii,jj) < im_1(ii-count,jj+count)% & im_1(ii-count,jj+count)>9000
            for ll = 1:count
               im_1(ii-ll,jj+ll) = im_temp(ii-ll,jj+ll);
            end
         end
      end
   end
end
im_diag = im_1;

%Fill in open spaces inside cells
[im_1] = stitch(im_1,im_temp,row,col);
im_stitch = im_1;
%Eliminate non-cell pixels
[im_1] = remove_extra(im_1,row,col);

%Calculate Cell Coverage Area
count = 0;
for ii = 1:row
   for jj = 1:col
      if im_1(ii,jj) ~=0 & im_1(ii,jj) ~= 20000
         count = count + 1;
      end
   end
end
area = count*pixel_area

%Calculate Background Area
count_back = 0;
for ii = 1:row
   for jj = 1:col
      if im_1(ii,jj) == 20000
         count_back = count_back + 1;
      end
   end
end
area_back = count_back*pixel_area;

%Display all images
figure(1)
imshow(im_temp,[0,20000]);
out1 = ['aoriginal' int2str(zz) '.jpg'];
out1b = ['aoriginal' int2str(zz) '.tif']
```

```
    saveas(gcf,out1,'jpg')
    imwrite(uint16(im_temp),out1b);
    figure(2)
    imshow(im_1_temp,[0,20000])
    out2 = ['bthresh' int2str(zz) '.jpg'];
    out2b = ['bthresh' int2str(zz) '.tif'];
    saveas(gcf,out2,'jpg')
    imwrite(uint16(im_1_temp),out2b);
    figure(3)
    imshow(im_edge,[0,1])
    out3 = ['cedge' int2str(zz) '.jpg'];
    out3b = ['cedge' int2str(zz) '.tif'];
    saveas(gcf,out3,'jpg')
    imwrite(uint16(im_edge),out3b);
    figure(4)
    imshow(im_edge_thresh,[0,20000]);
    out4 = ['dedge_thresh' int2str(zz) '.jpg'];
    out4b = ['dedge_thresh' int2str(zz) '.tif'];
    saveas(gcf,out4,'jpg')
    imwrite(uint16(im_edge_thresh),out4b);
    figure(5)
    imshow(im_diag,[0,20000]);
    out5 = ['ediag' int2str(zz) '.jpg'];
    out5b = ['ediag' int2str(zz) '.tif'];
    saveas(gcf,out5,'jpg')
    imwrite(uint16(im_diag),out5b);
    figure(6)
    imshow(im_stitch,[0,20000]);
    out6 = ['fstitch' int2str(zz) '.jpg'];
    out6b = ['fstitch' int2str(zz) '.tif'];
    saveas(gcf,out6,'jpg')
    imwrite(uint16(im_stitch),out6b);
    figure(7)
    imshow(im_1,[0,20000])
    out7 = ['gfinal' int2str(zz) '.jpg'];
    out7b = ['gfinal' int2str(zz) '.tif'];
    imwrite(uint16(im_1),out7b);
    saveas(gcf,out7,'jpg')
    fprintf(fid,'Area %3.1f =  ',zz);
    fprintf(fid,'%9.7f,  %7.0f, %9.7f,  %7.0f\n',area,count,area_back,count_back);
    test = uint16(Initial);
    xx = ['wasabi' int2str(zz) '.tif'];
    imwrite(test,xx);
end
fclose(fid);
```

Threshold Filter

```
function[im1] = sort_cells(ii,jj,r1,c1,init,im1,sec_avg);
%This function elimanates cells within a specified range of the overall  local
%background as well as places zeros at the boundaries
if ii == 1 & jj == 1
   for kk = 1:r1(ii,jj)
      for ll = 1:c1(ii,jj)
         if init{ii,jj}(kk,ll)==0
            im1{ii,jj}(kk,ll) = 0;
         elseif (im1{ii,jj}(kk,ll) > (sec_avg(ii,jj)*.80) & im1{ii,jj}(kk,ll)<...
               (sec_avg(ii,jj)*1.2)) %| abs((im1{ii,jj}(kk,ll) - init{ii,jj}(kk,ll))) < 200;
            im1{ii,jj}(kk,ll) = 20000;
         end
      end
   end
elseif ii ==1 & jj == 2
   for kk = 1:r1(ii,jj)
      for ll = 1:c1(ii,jj)
         if init{ii,jj}(kk,ll)==0
            im1{ii,jj}(kk,ll) = 0;
         elseif (im1{ii,jj}(kk,ll) > (sec_avg(ii,jj)*.75) & im1{ii,jj}(kk,ll)<...
               (sec_avg(ii,jj)*1.25)) %| abs((im1{ii,jj}(kk,ll) - init{ii,jj}(kk,ll))) < 200;
            im1{ii,jj}(kk,ll) = 20000;
         end
      end
   end
elseif ii == 1 & jj == 3
   for kk = 1:r1(ii,jj)
      for ll = 1:c1(ii,jj)
         if init{ii,jj}(kk,ll)==0
            im1{ii,jj}(kk,ll) = 0;
         elseif (im1{ii,jj}(kk,ll) > (sec_avg(ii,jj)*.80) & im1{ii,jj}(kk,ll)<...
               (sec_avg(ii,jj)*1.2)) %| abs((im1{ii,jj}(kk,ll) - init{ii,jj}(kk,ll))) < 100;
            im1{ii,jj}(kk,ll) = 20000;
         end
      end
   end
elseif ii == 2 & jj == 1
   for kk = 1:r1(ii,jj)
      for ll = 1:c1(ii,jj)
         if init{ii,jj}(kk,ll)==0
            im1{ii,jj}(kk,ll) = 0;
         elseif (im1{ii,jj}(kk,ll) > (sec_avg(ii,jj)*.80) & im1{ii,jj}(kk,ll)<...
               (sec_avg(ii,jj)*1.2)) %| abs((im1{ii,jj}(kk,ll) - init{ii,jj}(kk,ll))) < 100;
            im1{ii,jj}(kk,ll) = 20000;
         end
      end
   end
elseif ii == 2 & jj == 2
   for kk = 1:r1(ii,jj)
      for ll = 1:c1(ii,jj)
         if init{ii,jj}(kk,ll)==0
```

```
            im1{ii,jj}(kk,ll) = 0;
        elseif (im1{ii,jj}(kk,ll) > (sec_avg(ii,jj)*.80) & im1{ii,jj}(kk,ll)<...
                (sec_avg(ii,jj)*1.2)) %| abs((im1{ii,jj}(kk,ll) - init{ii,jj}(kk,ll))) < 100;
            im1{ii,jj}(kk,ll) = 20000;
        end
    end
end
elseif ii == 2 & jj == 3
    for kk = 1:r1(ii,jj)
        for ll = 1:c1(ii,jj)
            if init{ii,jj}(kk,ll)==0
                im1{ii,jj}(kk,ll) = 0;
            elseif (im1{ii,jj}(kk,ll) > (sec_avg(ii,jj)*.85) & im1{ii,jj}(kk,ll)<...
                    (sec_avg(ii,jj)*1.15)) %| abs((im1{ii,jj}(kk,ll) - init{ii,jj}(kk,ll))) < 100;
                im1{ii,jj}(kk,ll) = 20000;
            end
        end
    end
elseif ii == 3 & jj == 1
    for kk = 1:r1(ii,jj)
        for ll = 1:c1(ii,jj)
            if init{ii,jj}(kk,ll)==0
                im1{ii,jj}(kk,ll) = 0;
            elseif (im1{ii,jj}(kk,ll) > (sec_avg(ii,jj)*.80) & im1{ii,jj}(kk,ll)<...
                    (sec_avg(ii,jj)*1.1)) %| abs((im1{ii,jj}(kk,ll) - init{ii,jj}(kk,ll))) < 100;
                im1{ii,jj}(kk,ll) = 20000;
            end
        end
    end
elseif ii == 3 & jj == 2
    for kk = 1:r1(ii,jj)
        for ll = 1:c1(ii,jj)
            if init{ii,jj}(kk,ll)==0
                im1{ii,jj}(kk,ll) = 0;
            elseif (im1{ii,jj}(kk,ll) > (sec_avg(ii,jj)*.80) & im1{ii,jj}(kk,ll)<...
                    (sec_avg(ii,jj)*1.2)) %| abs((im1{ii,jj}(kk,ll) - init{ii,jj}(kk,ll))) < 100;
                im1{ii,jj}(kk,ll) = 20000;
            end
        end
    end
elseif ii == 3 & jj == 3
    for kk = 1:r1(ii,jj)
        for ll = 1:c1(ii,jj)
            if init{ii,jj}(kk,ll)==0
                im1{ii,jj}(kk,ll) = 0;
            elseif (im1{ii,jj}(kk,ll) > (sec_avg(ii,jj)*.80) & im1{ii,jj}(kk,ll)<...
                    (sec_avg(ii,jj)*1.2)) %| abs((im1{ii,jj}(kk,ll) - init{ii,jj}(kk,ll))) < 200;
                im1{ii,jj}(kk,ll) = 20000;
            end
        end
    end
end
```

Stitch and Fill Filters

```
function[im_1] = stitch(im_1,im_temp,row,col)
%This function is used to reconstruct cells from main program
%Cell_Electrode  Spaces that have been removed by the threshold filter are
%reinserted using this code.
kk_max = 12
iterations = 2
for ll = 1:iterations
   ll
   for ii = 1:row
      for jj = 1:col
         right = 0;
         left = 0;
         top = 0;
         bottom = 0;
         check1 = 0;
         check2 = 0;
         check3 = 0;
         check4 = 0;
         if im_1(ii,jj) == 20000;
            kk = 1;
            while kk <=kk_max
               if im_1(ii+kk,jj)~=20000 %& im_1(ii+kk,jj) ~= 0
                  right = kk;
                  kk = kk_max+1;
                  check1 = 1;
               end
               kk = kk+1;
            end
            kk = 1;
            if check1 == 1;
               while kk <=kk_max
                  if im_1(ii-kk,jj)~=20000 %& im_1(ii-kk,jj) ~= 0
                     left = kk;
                     kk = kk_max+1;
                     check2 =1 ;
                  end
                  kk = kk+1;
               end
               kk = 1;
               if check2 == 1;
                  while kk <=kk_max
                     if im_1(ii,jj+kk)~=20000 %& im_1(ii,jj+kk) ~= 0
                        top = kk;
                        kk = kk_max+1;
                        check3 = 1;
                     end
                     kk = kk+1;
                  end
                  kk = 1;
                  if check3 == 1;
                     while kk <=kk_max
```

```
                    if im_1(ii,jj-kk)~=20000 %& im_1(ii,jj-kk) ~= 0
                        bottom = kk;
                        kk = kk_max+1;
                        check4 = 1;
                    end
                    kk = kk+1;
                end
                if (check1 + check2 + check3 + check4)>3
                    im_1(ii,jj) = im_temp(ii,jj);
                    im_1 = fill(im_1,im_temp,right,left,top,bottom,ii,jj);
                end
            end
        end
    end
  end
end



function[im_1] = fill(im_1,im_temp,right,left,top,bottom,ii,jj)
%This code fills in blocks determined in stitch

%fill right
for kk = 1:right
   im_1(ii+kk,jj) = im_temp(ii+kk,jj);
end

%fill left
for kk = 1:left
   im_1(ii-kk,jj) = im_temp(ii-kk,jj);
end

%fill above
for kk = 1:top
   im_1(ii,jj+kk) = im_temp(ii,jj+kk);
end

%fill below
for kk = 1:bottom
   im_1(ii,jj-kk) = im_temp(ii,jj-kk);
end
```

Remove/Clear Filter

```
function[im_1] = remove_extra(im_1,row,col)
%Removes spurious areas that are not covered by cells

kk_max = 20;
iterations = 2;
for ll = 1:iterations
   for ii = 1:row
      for jj = 1:col
         right = 0;
         left = 0;
         top = 0;
         bottom = 0;
         check1 = 0;
         check2 = 0;
         check3 = 0;
         check4 = 0;
         if im_1(ii,jj) ~= 20000 & im_1(ii,jj) ~= 0;
            kk = 1;
            while kk <=kk_max
               if im_1(ii+kk,jj)==20000
                  right = kk;
                  kk = kk_max+1;
                  check1 = 1;
               elseif im_1(ii+kk,jj) == 0
                  kk = kk_max+1;
                  check1 = 1;
               end
               kk = kk+1;
            end
            kk = 1;
            if check1 == 1;
               while kk <=kk_max
                  if im_1(ii-kk,jj)==20000
                     left = kk;
                     kk = kk_max+1;
                     check2 =1 ;
                  elseif im_1(ii-kk,jj) == 0
                     kk = kk_max+1;
                     check2 =1 ;
                  end
                  kk = kk+1;
               end
               kk = 1;
               if check2 == 1;
                  while kk <=kk_max
                     if im_1(ii,jj+kk)==20000
                        top = kk;
                        kk = kk_max+1;
                        check3 = 1;
                     elseif im_1(ii,jj+kk) == 0
                        kk = kk_max+1;
```

```
                    check3 = 1;
                end
                kk = kk+1;
            end
            kk = 1;
            if check3 == 1;
                while kk <=kk_max
                    if im_1(ii,jj-kk)==20000
                        bottom = kk;
                        kk = kk_max+1;
                        check4 = 1;
                    elseif im_1(ii,jj-kk) == 0
                        kk = kk_max+1;
                        check4 = 1;
                    end
                    kk = kk+1;
                end
                if (check1 + check2 + check3 + check4)>3
                    im_1(ii,jj) = 20000;
                    im_1 = clear_cell(im_1,right,left,top,bottom,ii,jj);
                end
            end
        end
    end
  end
end


function[im_1] = clear_cell(im_1,right,left,top,bottom,ii,jj)
%This code removes area determined in remove_extra

%fill right
for kk = 1:right
  im_1(ii+kk,jj) = 20000;
end

%fill left
for kk = 1:left
  im_1(ii-kk,jj) = 20000;
end

%fill above
for kk = 1:top
  im_1(ii,jj+kk) = 20000;
end

%fill below
for kk = 1:bottom
  im_1(ii,jj-kk) = 20000;
end
```

APPENDIX C

Focal Adhesion Quantification/Histogram Maker

```
%Program: Focal Adhesion quantification
%This program reads in previously processed images (intensity corrected)
%and creates intensity histograms
clear all; clc;

iptsetpref('ImshowBorder','tight')
iptsetpref('ImshowAxesVisible','off')
section = ['AR'];
time = ['200'];

%Read in pre-processed file
oo = ['..\Images_BlackBG\Processed_'  section time 'min.tif'];
A = imread(oo,'tif');
[row_mask col_mask] = size(A);
pp = ['..\Final\Red_Final_'  section time 'min.tif'];
Red = imread(pp,'tif');
qq = ['..\Final\Green_Final_'  section time 'min.tif'];
Green = imread(qq,'tif');
rr = ['..\Final\Blue_Final_'  section time 'min.tif'];
Blue = imread(rr,'tif');
[row col] = size(Blue);

%Show pre-prossed image
% figure(1)
% imshow(A,[0 max(max(A))]);


%
for ii = 1:row
   for jj = 1:col
      if A(ii,jj) == 0
         Red(ii,jj) =Red(1,1);
         Green(ii,jj) = Green(1,1);
         Blue(ii,jj) = Blue(1,1);
      end
   end
end


%Divide pixels into intensity bins
hcount = 1;
[hist_red(hcount,:) hist_green(hcount,:) hist_blue(hcount,:)...
   red_dist(hcount,:) green_dist(hcount,:) blue_dist(hcount,:) ...
   bin_red(hcount,:) bin_green(hcount,:) bin_blue(hcount,:) mx_red...
   mx_green mx_blue mn_red mn_green mn_blue ]...
   = Hist_bins(A,row,col,Red,Green,Blue);

max_red(hcount) = mx_red;
max_green(hcount) = mx_green;
max_blue(hcount) = mx_blue;
min_red(hcount) = mn_red;
```

```
min_green(hcount) = mn_green;
min_blue(hcount) = mn_blue;
out = [red_dist' green_dist' blue_dist'];
out2 = [max_red max_green max_blue min_red min_green min_blue];



function[hist_red hist_green hist_blue red_dist green_dist blue_dist bin_red bin_green bin_blue ...
    max_Red max_Green max_Blue min_Red min_Green min_Blue] =...
    Hist_bins(B,row,col,Red,Green,Blue)
%Program - Hist_bins
%This program takes input from the focal_adhesion_quantification code
%containing cell information and distributes
%the intensity into a histogram for plotting.
max_Red = double(max(max(Red)));
max_Green = double(max(max(Green)));
max_Blue = double(max(max(Blue)));
max_B = double(max(max(B)));
for ii = 1:row
    for jj = 1:col
        if B(ii,jj) == 0
            B(ii,jj) = 100000;
            Red(ii,jj) = 100000;
            Green(ii,jj) = 100000;
            Blue(ii,jj) = 100000;
        end
    end
end

%Calculate the minimum value for each color for plotting histogram
min_B = double(min(min(B)));
min_Red = double(min(min(Red)));
min_Green = double(min(min(Green)));
min_Blue = double(min(min(Blue)));

%Set the range of the histogram
bin_red = [5000:500:16000];
bin_green = [5000:500:16000];
bin_blue = [5000:500:16000];

%Set the median of each section of the histogram for plotting
for ii = 1:length(bin_red)-1
    hist_red(ii) = (bin_red(ii)+bin_red(ii+1))/2;
    hist_green(ii) = (bin_green(ii)+bin_green(ii+1))/2;
    hist_blue(ii) = (bin_blue(ii)+bin_blue(ii+1))/2;
end

%Divide red data into bins
red_dist = zeros(length(bin_red)-1,1);
nn_max = length(red_dist);
for ii = 1:row
    for jj = 1:col
        if Red(ii,jj)~= 100000;
            nn = 1;
```

```
            while nn < nn_max+1
                if Red(ii,jj) >= bin_red(nn) & Red(ii,jj) < bin_red(nn+1)
                    red_dist(nn) = red_dist(nn) + 1;
                    nn = 100;
                else
                    nn = nn+1;
                end
            end
        end
    end
end

%Divide green data into bins
green_dist = zeros(length(bin_green)-1,1);
nn_max = length(green_dist);
for ii = 1:row
    for jj = 1:col
        if Green(ii,jj)~= 100000;
            nn = 1;
            while nn < nn_max+1
                if Green(ii,jj) >= bin_green(nn) & Green(ii,jj) < bin_green(nn+1)
                    green_dist(nn) = green_dist(nn) + 1;
                    nn = 100;
                else
                    nn = nn+1;
                end
            end
        end
    end
end

%Divide red data into bins
blue_dist = zeros(length(bin_blue)-1,1);
nn_max = length(blue_dist);
for ii = 1:row
    for jj = 1:col
        if Blue(ii,jj)~= 100000;
            nn = 1;
            while nn < nn_max+1
                if Blue(ii,jj) >= bin_blue(nn) & Blue(ii,jj) < bin_blue(nn+1)
                    blue_dist(nn) = blue_dist(nn) + 1;
                    nn = 100;
                else
                    nn = nn+1;
                end
            end
        end
    end
end
```

APPENDIX D

Vesicle Tracker

```
%Vesicle_Tracker - This code calculates the mean square dispacement of
%vesicles inside cancer cells.
clear all; clc;

%Physical Parameters
lambda = 488;
ni = 1.515;
nt = 1.33;
theta = 61.98;
zp = (lambda/(4*pi))*((ni^2*sind(theta)^2-nt^2)^(-.5)); %Penetration Depth
pix_size = 133.33;   %Pixel Size in nanometers

%Read in initial vesicle locations and the time at which they are observed
ves_initial = dlmread('Particles2.txt');
[ves_num dum] = size(ves_initial);

%Read in background
back = dlmread('back.txt');
back(:,687) = [];

%Distribute initial values into appropriate variables
start_file = ves_initial(:,1);
ves_start = 1;

%MSD Values
msd_file = fopen('Output2\MSD_Values.txt','w');
fprintf(msd_file,'MSD \n');
fprintf(msd_file,' Data     X-MSD      Y-MSD      Z-MSD     2D-MSD    3D_MSD \n');

%Read in initial file and select vesicle for tracking
for ves = ves_start:ves_num
%Set up output files
   %Vesicle Positions
   file_name = ['Output2\Vesicle_Position' int2str(ves) '.txt'];
   fid = fopen(file_name,'w');
   fprintf(fid,' File Pixel Number  X-Y Pairs Intensity \n');

   %Vesicle Displacements
   file_name2 = ['Output2\Vesicle_Displacement' int2str(ves) '.txt'];
   fid2 = fopen(file_name2,'w');
   fprintf(fid2,'    X-Dis.      Y-Dis.     Z-Dis.    2D-Dis.    3D-Dis. \n');

   %Read in initial file and select vesicle fro tracking
   xx_max(1,ves) = ves_initial(ves,3)+1;
   yy_max(1,ves) = ves_initial(ves,2)+1;
   file_out(1,ves) = ves_initial(ves,1);
   count = 1;
   im1 = im_read(start_file(ves))-back;
   [xx_c(count,ves) yy_c(count,ves) x_temp{ves,count} y_temp{ves,count} i_temp{ves,count}
num_pix(count,ves) c_thresh(count,ves)]=...
      ves_center(im1,xx_max(count,ves),yy_max(count,ves));
```

```
   max_int(count,ves) = im1(xx_max(count,ves),yy_max(count,ves));
   count = count + 1;

   %Track individual vesicle chosen from above
   file=start_file(ves)+1;
   while file <= 497
      im1 = im_read(file) - back;
      [xx_max_t(count,ves) yy_max_t(count,ves)] = ves_max(im1,xx_max(count-
1,ves),yy_max(count-1,ves));
      [xx_c_t(count,ves) yy_c_t(count,ves) x_temp_t{ves,count} y_temp_t{ves,count}
i_temp_t{ves,count} num_pix_t(count,ves) c_thresh_t(count,ves)] =...
         ves_center(im1,xx_max_t(count,ves),yy_max_t(count,ves));
      max_int_t(count,ves) = im1(xx_max_t(count,ves),yy_max_t(count,ves));
      file_out_t(count,ves) = file;
      if max_int_t(count,ves) < 2.*c_thresh_t(count,ves)
         file = 500;
      else
         xx_max(count,ves) = xx_max_t(count,ves); yy_max(count,ves) = yy_max_t(count,ves);
         xx_c(count,ves) = xx_c_t(count,ves); yy_c(count,ves) = yy_c_t(count,ves);
x_temp{ves,count}=x_temp_t{ves,count};
         y_temp{ves,count}= y_temp_t{ves,count}; i_temp{ves,count} = i_temp_t{ves,count};
num_pix(count,ves) = num_pix_t(count,ves);
         c_thresh(count,ves) = c_thresh_t(count,ves); max_int(count,ves) = max_int_t(count,ves);
file_out(count,ves) = file_out_t(count,ves);
         file = file + 1;
         count = count + 1;
      end
   end

   %Calculate Displacements
   for dis = 1:count-2
      del_x(dis,ves) = (xx_c(dis+1,ves) - xx_c(dis,ves))*pix_size;
      x_sq(dis,ves) = del_x(dis,ves).^2;
      del_y(dis,ves) = (yy_c(dis+1,ves) - yy_c(dis,ves))*pix_size;
      y_sq(dis,ves) = del_y(dis,ves).^2;
      del_z(dis,ves) = -zp*log(max_int(dis+1,ves)/max_int(dis,ves));
      z_sq(dis,ves) = del_z(dis,ves).^2;
      r_2d(dis,ves) = (x_sq(dis,ves) + y_sq(dis,ves))^.5;
      r_3d(dis,ves) = (x_sq(dis,ves) + y_sq(dis,ves) + z_sq(dis,ves))^.5;
   end

   %Caclulate MSD values
   x_msd_temp = 0;
   y_msd_temp = 0;
   z_msd_temp = 0;
   xy_msd_temp = 0;
   xyz_msd_temp = 0;
   for ii = 1:count-2
      x_msd_temp = x_sq(ii,ves) + x_msd_temp;
      y_msd_temp = y_sq(ii,ves) + y_msd_temp;
      z_msd_temp = z_sq(ii,ves) + z_msd_temp;
      xy_msd_temp = r_2d(ii,ves) + xy_msd_temp;
      xyz_msd_temp = r_3d(ii,ves) + xyz_msd_temp;
   end
```

```
    x_msd(ves) = (x_msd_temp./(count-2));%./(1000^2);
    y_msd(ves) = (y_msd_temp./(count-2));%./(1000^2);
    z_msd(ves) = (z_msd_temp./(count-2));%./(1000^2);
    xy_msd(ves) = x_msd(ves)+y_msd(ves);%./(1000^2);
    xyz_msd(ves) = x_msd(ves)+y_msd(ves)+z_msd(ves);%./(1000^2);

    %Print Vesicle Location
    fprintf(fid,'%5.0f \n',count-1);
    for ii = 1:count-1
        %fprintf(fid,'%5.0f  %5.0f    %5.0f ',file_out(ii,ves),xx_max(ii,ves),yy_max(ii,ves));
        fprintf(fid,'%5.0f  %5.0f  ',file_out(ii,ves),num_pix(ii,ves));
        for jj = 1:length(x_temp{ves,ii})
            fprintf(fid,' %4.0f %4.0f %5.0f   ',x_temp{ves,ii}(jj),y_temp{ves,ii}(jj),i_temp{ves,ii}(jj));
            if jj == length(x_temp{ves,ii})
                fprintf(fid,'\n');
            end
        end
    end
    %Print Vesicle Displacement
    for ii = 1:count-2
        fprintf(fid2,'%10.3f  %10.3f %10.3f %10.3f %10.3f
\n',del_x(ii,ves),del_y(ii,ves),del_z(ii,ves),r_2d(ii,ves),r_3d(ii,ves));
    end
    data(ves) = count-2;
    %Close files
    fclose(fid);
    fclose(fid2);
end

%Print MSD for each vesicle
msd_mat = [data' x_msd' y_msd' z_msd' xy_msd' xyz_msd'];
fprintf(msd_file,'%5.0f %10.2f %10.2f %10.2f %10.2f %10.2f \n',msd_mat');
fclose(msd_file);

% %Output final file for vesicle visualization
% fidxx = fopen('Output\All_out_XX.txt','w');
% fprintf(fidxx,'%5.0f ',xx_max);
% fidyy = fopen('Output\All_out_YY.txt','w');
% fprintf(fidyy,'%5.0f ',yy_max);
% fclose(fidxx);
% fclose(fidyy);

xx_max = int16(xx_max);
yy_max = int16(yy_max);
[row col] = size(xx_max);
for ii = 1:row
    for jj = 1:col
        ll = xx_max(ii,jj);
        kk = yy_max(ii,jj);
        if ll~=0
            intensity(ii,jj) = max_int(ii,jj) + back(ll,kk);
        end
    end
end
```

Vesicle Center

```
function[xx_c yy_c thresh] = ves_center(im1,x_max,y_max)
%This code calcuates the center of a vesicle based on a Guassian curve fit
%model using a nonlinear regression technique.

count = 2;
dist = 5;
thresh_dist = 5;
thresh_temp = 0;
t_count = 0;
pix = 3;

%Determine threshold level
for kk = x_max-thresh_dist:x_max+thresh_dist
   for ll = y_max-thresh_dist:y_max+thresh_dist
      if ((kk-x_max)^2+(ll-y_max)^2)^.5 > (thresh_dist-1)  & ((kk-x_max)^2+(ll-y_max)^2)^.5 <
(thresh_dist+1)
         thresh_temp = thresh_temp + im1(kk,ll);
         t_count = t_count+1;
      end
   end
end
thresh = 2*(thresh_temp/t_count);

%Set evaluation area
for ii = x_max-pix:x_max+pix
   for jj = y_max-pix:y_max+pix
      B(ii-(x_max-pix-1),jj-(y_max-pix-1))= im1(ii,jj);
   end
end
B = double(B);
[row col] = size(B);

%Initial Guess
Axy = B(pix+1,pix+1);
xo2 = pix+1;
yo2 = pix+1;
sigmaxy = 1.;
Ioxy = 400;
G_init_xy = B;

%2D Nonlinear Solver
zz = 1;
while zz < 100
   for ii = 1:row
      for jj = 1:col
         G_sol_xy(ii,jj) = Axy*exp(-((ii-xo2)^2+(jj-yo2)^2)/(2*sigmaxy^2))+Ioxy;
      end
   end
   count = 1;
   for ii = 1:row
      for jj = 1:col
```

```
            d_betaxy(count) = (G_init_xy(ii,jj) - G_sol_xy(ii,jj));
            count = count+1;
        end
    end
    count = 1;
    for ii = 1:row
        for jj = 1:col
            dAxy(count,1) = exp(-((ii-xo2)^2+(jj-yo2)^2)/(2*sigmaxy^2));
            dAxy(count,2) = ((-Axy*(-2*ii+2*xo2))/(2*sigmaxy^2))*exp(-((ii-xo2)^2+(jj-
yo2)^2)/(2*sigmaxy^2));
            dAxy(count,3) = ((-Axy*(-2*jj+2*yo2))/(2*sigmaxy^2))*exp(-((ii-xo2)^2+(jj-
yo2)^2)/(2*sigmaxy^2));
            dAxy(count,4) = ((Axy*((ii-xo2)^2+(jj-yo2)^2))/sigmaxy^3)*exp(-((ii-xo2)^2+(jj-
yo2)^2)/(2*sigmaxy^2));
            dAxy(count,5) = 1;
            count = count +1;
        end
    end
    aaxy = dAxy'*dAxy;
    bbxy = dAxy'*d_betaxy';
    d_lamxy = aaxy\bbxy;
    Axy = d_lamxy(1)+Axy;
    xo2 = d_lamxy(2) + xo2;
    yo2 = d_lamxy(3) + yo2;
    sigmaxy = d_lamxy(4) + sigmaxy;
    Ioxy = d_lamxy(5) + Ioxy;
    zz = zz+1;
    if abs(max(d_lamxy)) < 1e-5
        zz = 101;
    end
end


xx_c = x_max-1-pix+xo2;
yy_c = y_max-1-pix+yo2;

clear x_temp y_temp i_temp im_temp
```

Vesicle Movement

```
%Vesicle_Analyzer
%This code is reads in an output file created in Vesicle Tracker and
%analyze the vesicle movement.
clear all; clc;
pix_size = 133.33;

%Read in and distribute data to appropriate variables
ves_start = 1;
ves_end = 6;
avgz_out = zeros(500:ves_end)
for ves = ves_start:ves_end
    %Open Input and Output Files
    t = ['Output\Vesicle_Center' int2str(ves) '.txt'];
    fid1 = fopen(t);
    s = ['All_Data\Cell1_Ves' int2str(ves) '.txt']
    all_data = fopen(s,'w');
    fprintf(all_data,'MSD \n');
    fprintf(all_data,'    Z_Avg      X-SQ      Y-SQ      Z-SQ      2D-SQ     \n');
   %Read in input data
    dum = textscan(fid1,'%12s',2);
    files = cell2mat(textscan(fid1,'%5.0f',1));
    data = cell2mat(textscan(fid1,'%5.0f %5.0f %5.0f %8.3f %8.3f %8.3f',files-1));
    data2 = cell2mat(textscan(fid1,'%5.0f %5.0f %5.0f %8.3f %8.3f',1));
    xxc = data(:,4); xxc(files) = data2(1,4);
    yyc = data(:,5); yyc(files) = data2(1,5);
    fclose(fid1);

    %Z Distances
    delz_org = data(:,6);
    delz_org = delz_org';
    zzc(1) = 0;
    for ii = 1:length(delz_org)
        zzc(ii+1) = zzc(ii)+delz_org(ii);
    end
    min_el = abs(min(zzc));
    for ii = 1:length(zzc)
        zzc(ii) = zzc(ii)+min_el;
    end
    z_max(ves) = max(zzc);

    %Calculate Displacments and Displacement Squared Values.
    delt = 1;
    for dis = 1:length(xxc)-delt
        delx(dis) = (xxc(dis+delt)-xxc(dis)).*pix_size;
        dely(dis) = (yyc(dis+delt)-yyc(dis)).*pix_size;
        delz(dis) = (zzc(dis+delt)-zzc(dis));
        delx_sq(dis) = (delx(dis).^2)./(1000^2);
        dely_sq(dis) = (dely(dis).^2)./(1000^2);
        delz_sq(dis) = delz(dis).^2./(1000^2);
        delxy_sq(dis) = delx_sq(dis) + dely_sq(dis);
        avgz(dis) = (zzc(dis+delt)+zzc(dis))/2;
```

```
    end
    for ii = 1:length(avgz)
        avgz_out(ii,ves) = avgz(ii);
    end

    %Sort data from smallest elevation to the largest
    output = [avgz' delx_sq' dely_sq' delz_sq' delxy_sq'];
    output2 = sortrows(output);
    %[row col] = size output2
    %for
    fprintf(all_data,'%10.3f %10.7f %10.7f %10.7f %10.7f \n ',output2');
    clear delx dely delz delx_sq dely_sq delz_sq delxy_sq avgz data data2 xxc yyc zzc
    clear delz_org
     fclose(all_data);
end
% xy_msd = xy_msd';
% plot(xy_msd)
```

Type of Vesicle Motion

```
%Vesicle_Analyzer
%This code is made to read in an output file created in Vesicle Tracker and
%analyze the vesicle movement to determine if the motion is Brownian, Directional
%or Caged.  RMS values are used to classify the movement
clear all; clc;
format long g
pix_size = 133.33;
init_dt = 0.120724346;
gf_caged =1;
%Read in and distribute data to appropriate variables

ves_start = 1;
delt_num = 41;
ves_end = 1;
t = ['Output\Vesicle_Center' int2str(ves_start) '.txt'];
fid1 = fopen(t);
dum = textscan(fid1,'%12s',2);
files = cell2mat(textscan(fid1,'%5.0f',1));
data = cell2mat(textscan(fid1,'%5.0f %5.0f %5.0f %8.3f %8.3f %8.3f',files-1));
data2 = cell2mat(textscan(fid1,'%5.0f %5.0f %5.0f %8.3f %8.3f',1));
xxc = data(:,4); xxc(files) = data2(1,4);
yyc = data(:,5); yyc(files) = data2(1,5);

%Z Distances
delz_org = data(:,6);
delz_org = delz_org';
zzc(1) = 0;
for ii = 1:length(delz_org)
   zzc(ii+1) = zzc(ii)+delz_org(ii);
end

%Calculate Displacments for different delta_time steps.

for delt = 1:delt_num;
   for dis = 1:files-delt
      delx{delt}(dis) = (xxc(dis+delt)-xxc(dis)).*pix_size;
      dely{delt}(dis) = (yyc(dis+delt)-yyc(dis)).*pix_size;
      delz{delt}(dis) = (zzc(dis+delt)-zzc(dis));
      delx_sq{delt}(dis) = delx{delt}(dis).^2;
      dely_sq{delt}(dis) = dely{delt}(dis).^2;
      delz_sq{delt}(dis) = delz{delt}(dis).^2;
      r_dis{delt}(dis) = (delx_sq{delt}(dis)+dely_sq{delt}(dis)).^(.5);
   end
   dt_pts(delt) = dis;
end
dt_pts;
%Calculate MSD Values for each delta_time step.
for delt = 1:delt_num
   x_msd(delt) = (mean(delx_sq{delt}))./(1000^2);
   y_msd(delt) = (mean(dely_sq{delt}))./(1000^2);
   z_msd(delt) = mean(delz_sq{delt})./(1000^2);
```

```
   xy_msd(delt) = (x_msd(delt)+y_msd(delt));
end

output = [dt_pts' x_msd' y_msd' z_msd' xy_msd'];

%Begin Curve fit
%Linear
[xysq_lin rms_lin D_lin] = linear_fit(xy_msd,init_dt);
lin_out = D_lin;
%Directional
[xysq_dir rms_dir D_dir V] = directional_fit(xy_msd,init_dt);
dir_out = [D_dir V]';
 %Caged
 [xysq_caged rms_cage D_cage R] = caged_fit(xy_msd,init_dt);
caged_out = [D_cage R]';
rms_out = [rms_lin rms_dir rms_cage]'

param_out = [D_lin D_dir V D_cage R]';
```

Linear, Directional, and Caged Regression Algorithms

Linear

```
function [xy_msd_sol rms_lin D] = linear_fit(xy_msd,init_dt)
%This code uses a nonlinear regresion technique to fit a linear curve to a
%given data set

%Initial Guess
D = 1;
dt(1) = init_dt;
for ii = 2:length(xy_msd)
   dt(ii) = dt(ii-1)+init_dt;
end
nn = length(xy_msd);

%Nonlinear Solver Y-Direction
zzmax = 10;
zz = 1;
while zz < zzmax
   for jj = 1:length(xy_msd)
      xy_msd_sol(jj) = 4*D*dt(jj);
   end
   d_betay = (xy_msd - xy_msd_sol);
   %Output Residual
   debetay_sq = d_betay.^2;
   dir_res = sum(debetay_sq);

   for ii = 1:length(xy_msd)
      dAy(ii,1) = 4*dt(ii);
   end

   aay = dAy'*dAy;
   bby = dAy'*d_betay';
   d_lamy = aay\bby;
   D = d_lamy(1)+D;
   figure(1)
   plot(dt,xy_msd,'b*');
   hold on
   plot(dt,xy_msd_sol,'r-');
   xlabel('Delta-Time Step (s)')
   ylabel('XY-MSD (microns^2)')
   hold off
%    if abs(max(d_lamy)) < 1e-7 & zz > 5;
%       zz_lin = zz
%       zz = zzmax-1;
%    end
   zz = zz+1;
   %pause
end
rms_lin = (((sum(d_betay.^2))/nn)).^.5;
```

Directional

```
function [xy_msd_sol rms_dir D V] = directional_fit(xy_msd,init_dt)
%This code uses a nonlinear regresion technique to fit a directional curve
%to a given data set

%Initial Guess
ao = 1;
a1 = 1;
dt(1) = init_dt;
for ii = 2:length(xy_msd)
   dt(ii) = dt(ii-1)+init_dt;
end
nn = length(xy_msd);

%Nonlinear Solver Y-Direction
zzmax = 200;
zz = 1;
while zz < zzmax
   for jj = 1:length(xy_msd)
      xy_msd_sol(jj) = exp(ao)*dt(jj) + exp(a1)*dt(jj).^2;
   end
   d_betay = (xy_msd - xy_msd_sol);

   for ii = 1:length(xy_msd)
      dAy(ii,1) = exp(ao).*dt(ii);
      dAy(ii,2) = exp(a1).*(dt(ii).^2);
   end
   aay = dAy'*dAy;
   bby = dAy'*d_betay';
   d_lamy = aay\bby;
   ao = 1*d_lamy(1)+ao;
   %    a1 = 1*d_lamy(2)+a1;
   if a1 < -31000
      a1 = a1;
   else
      a1 = 1*d_lamy(2)+a1;
   end
   figure(2)
   plot(dt,xy_msd,'b*');
   hold on
   plot(dt,xy_msd_sol,'r-');
   xlabel('Delta-Time Step (s)')
   ylabel('XY-MSD (microns^2)')
   hold off

   zz = zz+1;

end

D = exp(ao)/4;
V = exp(a1).^.5;
rms_dir = (((sum(d_betay.^2))/nn)).^.5;
```

Caged

```
function [xy_msd_sol rms_cage D R] = caged_fit(xy_msd,init_dt)
%This code uses a nonlinear regresion technique to fit a caged curve to a
%given data set
%Initial Guess
D = .1;
R = 2;
A1 = .99;
A2 = .85;
dt(1) = init_dt;
for ii = 2:length(xy_msd)
   dt(ii) = dt(ii-1)+init_dt;
end
nn = length(xy_msd);

%Nonlinear Solver Y-Direction
zzmax = 1000;
zz = 1;
while zz < zzmax
   for jj = 1:length(xy_msd)
      xy_msd_sol(jj) = R^2*(1-A1*exp(-4*A2*D*dt(jj)/R^2));
   end
   d_betay = (xy_msd - xy_msd_sol);
   %Output Residual
   debetay_sq = d_betay.^2;
   dir_res = sum(debetay_sq);
   for ii = 1:length(xy_msd)
      dAy(ii,1) = 4*A1*A2*dt(ii)*exp(-4*A2*D*dt(ii)/R^2);
      dAy(ii,2) = (2*R)-(2*R*A1*exp(-4*A2*D*dt(ii)/R^2))-(8*A1*A2*D*dt(ii)*exp(-
4*A2*D*dt(ii)/R^2)/R);
   end
   aay = dAy'*dAy;
   bby = dAy'*d_betay';
   d_lamy = aay\bby;
   D = .02*d_lamy(1)+D;
   R = .02*d_lamy(2)+R;
   figure(3)
   plot(dt,xy_msd,'b*');
   hold on
   plot(dt,xy_msd_sol,'r-');
   xlabel('Delta-Time Step (s)')
   ylabel('XY-MSD (microns^2)')
   hold off
   %pause
%    if abs(max(d_lamy)) < 1e-7 & zz > 5;
%       zz_cage = zz
%        zz = zzmax-1;
%    end
   zz = zz+1;
end

rms_cage = (((sum(d_betay.^2))/nn)).^.5;
```

Vesicle Size and Viscosity Calculation Algorithm

```
%This program determines the partilce size and viscosity by curve fitting MSD
%data to hindered diffusion values

clear all; clc;
%%Caged

% %Brownian
% h = [32.7       73.745  122.18  164.4];
% h = h./1e9;
% msd = [0.0025005     0.0033557       0.0035765       0.0036828]./(1e6)^2;

%Caged
h = [36.618       73.589  122.5    167.2    229.18];
h = h./1e9;
msd = [0.0028588        0.0035689       0.0043692       0.0043575       0.0046572]./(1e6)^2;

% %L-C
% h = [34.911   73.629  122.41  166.64  228.92];
% h = h./1e9;
% msd = [0.0027027     0.0035146       0.0041457       0.0042231       0.004706]./(1e6)^2;


%Inputs
T = 310;
k = 1.3805e-23;
delt = 0.120724346;
aa = 4*k*T*delt/(3*pi);

%Initial Guess
dp = 1050e-9;
mu = .001;

zzmax = 500;
zz = 1;
while zz < zzmax
   for ii = 1:length(h)
      mm = dp/(dp+2*h(ii));
      msd_sol(ii) = (aa/(dp*mu))*(1 - (9/16)*mm + (1/8)*mm^3-(45/256)*mm^4 - (1/16)*mm^5);
      msd_sol(ii) = msd_sol(ii);
   end
   d_betay = (msd - msd_sol);
   for ii = 1:length(h)
      mm = dp/(dp+2*h(ii));
      nn = dp+2*h(ii);
      dAy(ii,1) =(-aa/(mu*dp^2))*(1-(9/16)*mm + (1/8)*mm^3-(45/256)*mm^4 - (1/16)*mm^5)...
         +((aa/(dp*mu))*((-9/16)*(1/nn)+(9/16)*(dp/nn^2)+(3/8)*(dp^2/nn^3)...
         -(69/64)*(dp^3/nn^4)+(25/64)*(dp^4/nn^5)+(5/16)*(dp^5/nn^6)));
      dAy(ii,2) = (-aa/(dp*mu^2))*(1 - (9/16)*mm + (1/8)*mm^3-(45/256)*mm^4 - (1/16)*mm^5);
   end
   aay = dAy'*dAy;
   bby = dAy'*d_betay';
```

```
    d_lamy = (aay\bby);
    dp = .1*d_lamy(1)+dp;
    mu = .1*d_lamy(2) + mu;
    zz = zz+1;
    plot(h,msd,'r*')
    hold on
    plot(h,msd_sol,'b')
    hold off
    %pause
end
msd = msd*1e6^2
msd_sol = msd_sol*1e6^2

for ii = 1:length(h)
tang_coeff(ii) = 1 - (9/16)*(dp/(dp+2*h(ii))) + (1/8)*(dp/(dp+2*h(ii)))^3 ...
          -(45/256)*(dp/(dp+2*h(ii)))^4 - (1/16)*(dp/(dp+2*h(ii)))^5;
end
h = h*1e9;
figure(1)
plot(h,msd,'r*')
hold on
plot(h,msd_sol,'b-')
hold off

dc = (k*T/(3*pi*mu*dp))*(1e6)^2;
dhind = dc.*tang_coeff;
mean(dhind)
nn = length(d_betay);
rms_val = (((sum(d_betay.^2))/nn)).^.5

msd_sol=msd_sol'
```

VITA

Charles H. Margraves was born in Hartford Connecticut, on September 8, 1973. He grew up in Athens Tennessee and graduated from McMinn County High School in 1991. He then received his Bachelors of Science in Engineering Degree (BSE) from the University of Tennessee at Chattanooga in 1996.

Following graduation Charles worked for a year and half at Denso Manufacturing as a production engineer in Maryville Tennessee. In 1998 he accepted a position with Stone Engineering Company in Huntsville Alabama as a Computational Fluid Dynamics (CFD) analyst. While working at Stone, Charles completed his Master of Science in Mechanical Engineering (MSME) degree as a distance learning student through the Georgia Institute of Technology. He then resigned his position at Stone to begin his Doctoral work at the University of Tennessee at Knoxville (UTK) in the fall of 2003.

Charles spent two years as a teaching assistant with the Freshmen Engineering Fundamentals Division at UTK. In the fall of 2004, Charles joined his major professor's, Dr. Kenneth Kihm, Micro Nano Scale Fluids and Enegry Transport (MiNSFET) lab as a research assistant. In the summers of 2007 and 2008 Charles also worked as an instructor for the Mechanical, Aerospace, and Biomedical Engineering Department teaching a junior level Fluid Mechanics class.

Charles completed his PhD in mechanical engineering in the summer of 2008 and accepted a Lecturer position at Michigan Technological University. He and his wife, Ivette, have one child, Christina, and are expecting a second in the winter of 2009.