

APPLICATION OF OPTICAL ANALYSES TO CANCER THERAPEUTICS AND
DIAGNOSTICS

A Thesis

by

ANDREW PEYTON FISHER

Submitted to Swansea University in fulfilment of the requirements for the Degree of

DOCTOR OF PHILOSOPHY

Primary Advisors, Kenith E. Meissner & Haifa Shen

Secondary Advisors, Huw Summers & Mauro Ferrari

Head of College, Huw Summers

August 2019

ABSTRACT

The extreme diversity and heterogeneity of cancer cells creates a need for equally varying analysis methods to diagnose and treat the disease. Many methods have been developed for targeting and combatting a specific cancer. The oestrogen receptor, HER2, is overexpressed in certain varieties of breast cancer and subsequently used as a target for chemotherapy antibody treatments.¹ Unfortunately, not all cancers overexpress an easily identifiable protein and are therefore limited to systemic, non-targeted therapies such as resection, radiation, and chemotherapy.² Here, three novel techniques are investigated for analysing cancer therapeutic and diagnostic techniques.

Porous silicon microparticle (PSM) delivery methods take advantage of tumour vessel tortuosity and fenestrations to selectively deliver cancer therapeutics to tumours. In chapter 2, the previously uncharacterized rolling mechanics of PSM on endothelial cells are examined at physiologically relevant shear rates. Custom analysis software allows for the identification, tracking, and characterization of particles in flow.

Tertiary lymphoid structures (TLS) in and around tumours have been prominently associated with positive patient prognoses but are difficult and time consuming to identify. In chapter 3, optical tissue clearing and 3D imaging are utilized to investigate the ability of dendritic cell (DC) vaccines and checkpoint blockade therapies to affect tumour growth in association with TLS presence in the tumour. When treated with the DC vaccine or a combination with anti-PD1 treatments, tumour growth is severely inhibited. The 3D imaging and analysis was used to correlate the tumour growth inhibition with an increase in the concentration of intratumoural T cells and of TLS present in the tumour.

Optical spectroscopic imaging of biological systems has important applications in medical diagnosis and biochemistry. However, the extrinsic fluorescence of staining molecules often masks the intrinsic vibrational signals of biomolecules. In chapter 4, simultaneous spectroscopic bioimaging and photostability analysis of rhodamine 6G stained red blood cells using both fluorescence and resonance Raman imaging in a single laser excitation experiment were performed. A corresponding data processing algorithm was developed to separate the two previously indistinguishable spectroscopic signals.

DECLARATIONS AND STATEMENTS

DECLARATION

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed 

Date July 14th, 2020

STATEMENT 1

This thesis is the result of my own investigations, except where otherwise stated.

Other sources are acknowledged by footnotes giving explicit references.

A bibliography is appended.

Signed 

Date July 14th, 2020

STATEMENT 2

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed 

Date July 14th, 2020

DEDICATION

I would like to dedicate this work to my wife, Carolanne, who continuously encourages me to better myself in work and in life. She pushes me to greater heights than I could have ever accomplished on my own. I will forever be grateful for her willingness to accompany me around the globe, supporting me throughout, while we were living together and apart. Our many weekend trips throughout Europe were an unimaginable treasure and I would not have survived these four years without you and your wanderlust. Thank you for keeping life exciting and always loving me.

ACKNOWLEDGEMENTS

I would first like to thank my primary advisors, Dr. Kenith Meissner and Dr. Haifa Shen, for their support and guidance throughout my studies. Dr. Shen, thank you for taking a chance on an engineer in a biology laboratory. The experience I gained working in your lab will open my future to many opportunities that would not have been possible without your guidance and leadership. Dr. Meissner, thank you for my first chance in an undergraduate laboratory and for encouraging me to pursue my Ph.D. Thank you for bringing CC and I to Wales for the most exciting adventure of our lives. You have been a remarkable mentor, friend, and Closer. To my secondary advisors, Dr. Mauro Ferrari and Dr. Huw Summers, thank you for support and advice over these last four years.

I would also like to thank the members of both of my labs for their continuous support over my entire Ph.D. journey. With a special thanks to Dr. Sandra Bustamante Lopez, Dr. Junhua Mai, Dr. Qing Shi, Jun Li, and Louis Hinkle for being overwhelmingly helpful in teaching me the art of biological research.

To my entire family, parents, grandparents, in-laws, brother, and sister, thank you for your love and support and for always showing interest in my work. To my mother and father especially, thank you for raising me into the man I am and installing a love of science in me from an early age. To my grandfather, Jack Buttram, you inspire me to be a better scientist and a better husband every day. Thank you for being an outstanding role model for me to aspire to.

And lastly, to my friends, new and old, from Texas, Wales, Italy, China, Columbia, Mexico, France, and many more backgrounds, thank you for keeping me sane inside and outside of the lab. Meeting people from so many different countries and cultures has been a blessing to no end and I wouldn't trade your friendships or our experiences together for anything.

TABLE OF CONTENTS

ABSTRACT.....	ii
DECLARATIONS AND STATEMENTS	iii
DEDICATION	iv
ACKNOWLEDGEMENTS	v
TABLE OF CONTENTS.....	vi
LIST OF FIGURES	x
LIST OF TABLES	xiv
LIST OF ABBREVIATIONS	xv
1. INTRODUCTION.....	1
1.1. Drug Delivery.....	1
1.1.1. Systemic Delivery.....	1
1.1.2. Targeted Delivery	2
1.1.3. Multistage Vector Delivery	3
1.2. Cancer	3
1.2.1. Prevalence.....	4
1.2.2. Causes	4
1.2.3. Metastasis	4
1.2.4. Cancer strengths.....	5
1.2.5. Treatment.....	5
1.3. Tools for Visualization	6
1.3.1. Microscopy	6
1.3.2. Raman spectroscopy	10
1.3.3. Ultraviolet/visible absorption spectroscopy	11
1.3.4. Light scattering	12
1.3.5. Flow chamber	12
1.3.6. Immunohistochemistry.....	13
1.3.7. Inductively coupled plasma mass spectroscopy	13
1.4. Tools for Analysis	14
1.4.1. 3D Visualization Software	14
1.4.2. Cell tracking Software.....	15
1.4.3. Raman Peak Location Software.....	16
1.5. Outline of Thesis	16

2. DISCOIDAL POROUS SILICON MICROPARTICLE ADHESION AND ROLLING KINETICS	17
2.1. Introduction	17
2.2. Materials and Methods.....	20
2.2.1. Cell lines.....	20
2.2.2. PSM Preparation	21
2.2.3. Flow Cytometry	23
2.2.4. Static Chamber Setup.....	24
2.2.5. Flow Chamber Setup.....	25
2.2.6. Flow Chamber Sample Preparation	29
2.2.7. Flow Chamber Imaging.....	29
2.2.8. Static Chamber Analysis	31
2.2.9. Flow Chamber Analysis	32
2.2.10. Statistical Analyses	37
2.3. Results.....	38
2.3.1. PSM modification has little effect on Brownian motion step sizes	38
2.3.2. The addition of a protein interface increases PSM Brownian motility.....	40
2.3.3. E-PSM have the overall least displacement	43
2.3.4. ESTA and SCR PSM modification reduce rolling velocity on HUVECs	45
2.3.5. The addition of aptamers interferes with cellular interactions	46
2.3.6. A-PSM have a high affinity for endothelial cells	49
2.4. Discussion	50
3. DENDRITIC CELL VACCINES IN COMBINATION WITH CHECKPOINT BLOCKADE THERAPY LEAD TO AN INCREASE IN INTRATUMOURAL TERTIARY LYMPHOID STRUCTURES.....	54
3.1. Introduction	54
3.2. Materials and Methods.....	57
3.2.1. Mice	57
3.2.2. Cell lines.....	58
3.2.3. Preparation of Antigen-Encapsulated Liposomes and Loading into PSM.....	58
3.2.4. Bone Marrow Derived Dendritic Cells (BMDC)	58
3.2.5. Live animal treatment protocol.....	59
3.2.6. Flow Cytometry	59
3.2.7. Sample Preparation	60
3.2.8. Whole Tissue Immunolabelling.....	61
3.2.9. Tissue Clearing	62
3.2.10. Confocal Microscopy	62
3.2.11. Imaging Analysis	63

3.2.12. Error Analyses	69
3.2.13. Statistical Analyses	75
3.3. Preliminary Results.....	75
3.3.1. Combination treatment has a positive effect on tumour growth .	75
3.3.2. All treatments show an ability to influence T cell recruitment ...	76
3.4. Results.....	77
3.4.1. Deep tissue staining and imaging	77
3.4.2. Vaccine and combination treatments initiate an increase in the concentration of tumour infiltrating CD3+ T Cells.....	81
3.4.3. Combination treatment increases the concentration of TLS in the tumour.....	83
3.5. Discussion	86
4. ANALYSIS OF FLUORESCENCE IMAGING OF STAINED RED BLOOD CELLS WITH SIMULTANEOUS RESONANCE RAMAN PHOTOSTABILITY ANALYSIS	92
4.1. Introduction	92
4.2. Materials and methods	3
4.2.1. Cell Staining	3
4.2.2. Raman Spectroscopy	4
4.2.3. Data processing algorithm.....	5
4.3. Results.....	11
4.3.1. Filter optimization.....	11
4.3.2. Photobleaching unveils Raman signal	16
4.3.3. R6G is successfully attached to RBCs	20
4.3.4. Aging may reduce the intensity of the Raman signal	21
4.3.5. R6G Raman signal shows signs of photodamage at high laser intensities	23
4.4. Discussion	24
5. SUMMARY OF CONCLUSIONS	27
5.1. PSM Adhesion Kinetics	27
5.2. Tumour Vaccine TLS Development.....	28
5.3. Simultaneous Collection of Overlapping Raman and Fluorescent Signals	30
6. REFERENCES.....	31
APPENDIX A	48

APPENDIX B	55
APPENDIX C	71
APPENDIX D	91

LIST OF FIGURES

Figure 1. Jablonski diagram.....	7
Figure 2. Comparison of optical microscopy depth of field.....	9
Figure 3. Confocal microscopy pinhole diagram.....	10
Figure 4. Fabrication of discoidal porous silicon microparticles.....	21
Figure 5. Chemical Modification of PSM.	22
Figure 6. FTIR Absorbance Spectrum of the PSM.	23
Figure 7. TNF- α Activates HUVECs to Overexpress E-Selectin.	24
Figure 8. Fluorescent layer of HUVECs for the static chamber studies.	25
Figure 9. Flow Chamber Experimental Setup.....	26
Figure 10. Forces on an Attached PSM.....	28
Figure 11. Static chamber analysis process diagram.....	30
Figure 12. Static chamber HUVEC layer boundary detection.....	31
Figure 13. Static chamber PSM identification and tracking.....	32
Figure 14. Flow chamber analysis process diagram.	33
Figure 15. Identifying confluency of the HUVECs in the flow chamber.	34
Figure 16. Flow chamber identification of PSM on HUVECs.	35
Figure 17. Flow chamber tracking PSM through consecutive frames.	37
Figure 18. Brownian motion step size comparing PSM modifications on different substrates.....	39
Figure 19. Mean square displacement (MSD) of PSM in static conditions. .	40

Figure 20. Brownian motion step size comparing different substrates on each PSM modification.	42
Figure 21. Individual PSM movement paths recorded across the substrates.	43
Figure 22. PSM displacement from origin after 1 minute.	45
Figure 23. Tracking the rolling velocity of PSM.	46
Figure 24. Tracking the number of attached PSM after 22 minutes.	47
Figure 25. Adhesion kinetics of modified PSM on multiple substrates.	48
Figure 26. A-PSM attached to different substrates.	49
Figure 27. Representative diagram of the DC vaccination.	55
Figure 28. Tumour associated TLS.	56
Figure 29. Tumour preparation.	60
Figure 30. Setup for 3D imaging on the confocal microscope.	63
Figure 31. Subtract out autofluorescence.	65
Figure 32. Overview of the data processing algorithm.	66
Figure 33. Cell identification in the tissue samples.	67
Figure 34. Using background autofluorescence to capture sample volume. .	68
Figure 35. Cell and vessel segmentation.	69
Figure 36. Low intensity threshold allows more cells to be identified.	71
Figure 37. Examples of different sized scanning boxes.	72
Figure 38. Background noise filtering and cell segmentation.	74
Figure 39. Treatment efficacy against TUBO solid tumours.	76

Figure 40. Flow cytometry shows a rise in T cells with treatment.	76
Figure 41. iDisco setup and depth visualization.	78
Figure 42. Lymph node stain confirmation.	80
Figure 43. Comparison of the T cell penetration.	82
Figure 44. B cell and HEV concentrations inside the tissue samples.	83
Figure 45. TLS identification and concentration inside tumour samples.....	85
Figure 46. TLS meta-analysis.	86
Figure 47. TLS precursors.	89
Figure 48. Jablonski diagram.	2
Figure 49. RBC collection and staining.	4
Figure 50. Overview of RBC staining and imaging.....	5
Figure 51. Data analysis process diagram.	6
Figure 52. Graphical user interface (GUI) for the analysis software.	7
Figure 53. Raman signal extraction and fitting functions.	9
Figure 54. Comparison of R^2 values for Raman and fluorescence signals....	11
Figure 55. Comparison of spectral filters.	12
Figure 56. Comparison of Savitzky-Golay filter parameters.....	14
Figure 57. Comparison of spatial filters.	16
Figure 58. Photostability of the R6G fluorescence.	18
Figure 59. RBCs with and without R6G.....	21

Figure 60. Fresh and 1-week old, R6G stained RBCs compared to unstained control. 23

Figure 61. 1-week old, R6G-stained RBCs at varying power levels. 24

LIST OF TABLES

Table 1. Summary of Visualization Tools Utilized.....	14
Table 2. PSM Characteristics Post-Modification.....	23
Table 3. Flow Chamber Specifications.	28
Table 4. Rate of PSM Attachment and Detachment per square millimetre. .	50
Table 5. Antibodies used for staining tumour samples.	61
Table 6. Cumulative dose at each power level.	19
Table 7. Raman shifts and band assignments.	20

LIST OF ABBREVIATIONS

μm	Micrometre
μM	Micromolar
μW	Microwatt
AF	Attachment Factor
aN	attonewton
APC	Antigen Presenting Cell
A-PSM	APTES modified PSM
APTES	(3-Aminopropyl)triethoxysilane
BMDC	Bone Marrow derived Dendritic Cell
CA	California
CARS	Coherent Anti-Stokes Raman Scattering
CTLA4	Cytotoxic T-Lymphocyte-Associated protein 4
DBE	Dibenzyl Ether
DC	Dendritic Cell
DCM	Dichloromethane
DiO	3,3'-Dioctadecyloxacarbocyanine Perchlorate
DIPEA	Diisopropylethylamine
DLS	Dynamic Light Scattering
DMF	Dimethylformamide
DMSO	Dimethyl Sulfoxide

DNA	Deoxyribonucleic Acid
DOF	Depth of Field
DOPC	1,2-dioleoyl-sn-glycero-3-phosphocholine
ECM	Extracellular Matrix
EDTA	Ethylenediaminetetraacetic Acid
E-PSM	ESTA modified PSM
ESTA	E-Selectin Thioaptamer
FBS	Foetal Bovine Serum
FITC	Fluorescein Isothiocyanate
FL	Florida
fs	Femtosecond
FTIR	Fourier Transform Infrared
GUI	Graphical User Interface
Hb	Haemoglobin
HER2	Human Epidermal Growth Factor 2
HEV	High Endothelial Venule
HSC	Hematopoietic Stem Cells
HUVEC	Human Umbilical Vein Endothelial Cell
i.p.	Intraperitoneal
i.v.	Intravenous
IA	Iowa

IACUC	Institutional Animal Care and Use Committee
ICP	Inductively Coupled Plasma
iDISCO	Immunolabeling-enabled three-Dimensional Imaging of Solvent-Cleared Organs
LSGS	Low Serum Growth Supplement
M200	Media 200
MA	Massachusetts
MD	Maryland
min	Minutes
mm	Millimetre
NA	Numerical Aperture
NHS	N-hydroxysuccinimide
nm	Nanometre
PBS	Phosphate Buffered Saline
PD-L1	Programmed Death-Ligand 1
PMT	Photomultiplier Tube
PNAd	Peripheral Node Addressin
POI	Peak of Interest
PS	Penicillin-Streptomycin
PSM	Porous Silicon Microparticle
PtWH	PBS and 0.2% Tween-20 containing 10 mg ml ⁻¹ Heparin

R6G	Rhodamine 6G
RANKL	Receptor Activator of Nuclear Factor Kappa-B Ligand
RBC	Red Blood Cell
RBCFL	Red Blood Cell Free Layer
s	Seconds
s.c.	Subcutaneous
SCR	Scrambled Thioaptamer
SEF	Surface Enhanced Fluorescence
SEM	Scanning Electron Microscope
SERS	Surface Enhanced Raman Scattering
SMCC	Succinimidyl-4-(N-maleimidomethyl)cyclohexane-1-carboxylate
SNR	Signal to Noise Ratio
S-PSM	SCR modified PSM
STAT-3	Signal Transducer and Activator of Transcription 3
str	Stretching
SU	Swansea University
TAMU	Texas Agricultural and Mechanical University
TCEP	Tris (2-carboxyethyl)phosphine
TLS	Tertiary Lymphoid Structure
TNF	Tumour Necrosis Factor
UK	United Kingdom

USA	United States of America
UV	Ultraviolet
VA	Virginia
VEGF	Vascular Endothelial cell Growth Factor
WA	Washington
v, ip	In Plane Mode
Y, op	Out of plane mode

1. INTRODUCTION

Cancer is a very broad, catch-all term for a disease that takes on many different forms in varying areas of the body. Due to the vast diversity and heterogeneity of cancer cells, many methods have been, and more still need to be, developed for targeting and combatting all the different forms of cancer. One of the first steps to targeting a disease is diagnosis and analysis. Here, we primarily focus on TUBO, a HER2 positive breast cancer. In TUBO, the oestrogen receptor, HER2, is overexpressed and can be used as a target for chemotherapy antibody treatment.¹ Not all cancers have a protein that is overexpressed and easily targetable and therefore, treatments must rely on systemic, non-targeted therapies such as resection, radiation, and chemotherapy.² In this thesis, three novel visual analysis methods are developed to investigate the mechanisms of up and coming therapeutic and diagnostic techniques.

1.1. Drug Delivery

Delivering drugs to cancerous cells is a difficult task due to their similarity to their healthy progenitor cells. It is further complicated by the diversity of cancer cell phenotypes person to person and nodule to nodule.³ Systemic chemotherapy has advanced to include nano-sized drugs that have an enhanced retention around cancer cells to increase the selectivity of the therapy.⁴ While some cancers are still only treatable by systemic chemotherapy or radiation, others can be targeted with therapies that selectively weaken and/or kill cancer cells based on their extracellular protein makeup.⁵ Also, multistage vectors have been developed to prevent drugs from interacting with healthy tissues while in route to their target destination. Once trapped in the tumour vasculature or engulfed by cells, the vector begins to degrade and release the drug directly at the disease site at a high potency.⁶

1.1.1. Systemic Delivery

Traditional chemotherapy drugs attack cells at different points along the cell cycle to interrupt the replication process and prevent tumour growth.⁷ These are typically macromolecular substances that, once intravenously injected, circulate around the entire body and attack every available cell that is currently dividing, including healthy cells.⁷ This can cause side effects to other tissues with rapidly

dividing cells besides the cancer, such as cells in the bone marrow, stomach lining, and skin. This has led to a push to increase the selectivity of cancer therapeutic agents to reduce side effects and improve effectiveness.

Formulations of some chemotherapeutics have been reduced in size down to a nano scale (<100 nm).⁸ By reducing their size, drugs can penetrate through the large fenestrations in the tumour vasculature to directly interact with the cancerous cells. These fenestrations create an enhanced permeability and retention (EPR) effect inside the tumour space whereas the rest of the body has tight junctions between their vascular endothelial cells.^{9,10} This creates a tendency for nano-formulations to preferentially accumulate in the tumour space, allowing for higher intratumoural drug concentrations and lower concentrations in healthy tissue.

Although spherical micelles are the dominate shape for nanocarriers due to the ease of production, shape and size characteristics can be fine-tuned to improve the cellular uptake and biodistribution of the drug.¹¹ Long and short cylindrical shapes have been shown to increase circulation time and accumulation in the tumour tissue.^{12,13} These shapes are able to pass between vascular endothelial cells to reach the tumour space while also avoiding uptake by the circulating immune cells in the bloodstream, leading to a larger concentration of drug in contact with the cancer.

1.1.2. Targeted Delivery

In approximately 15-20% of breast cancers, the protein human epidermal growth factor 2 (HER2) is overexpressed. When HER2 interacts with the human epidermal growth factor (HER) protein, cancer cells are stimulated to grow and divide. Targeted antibodies, such as pertuzumab and trastuzumab, for breast cancer use monoclonal antibodies to attach to and block the functionality of HER2, preventing the interaction between HER and HER2.^{5,18} This treatment is used in combination with chemotherapy to retard the growth rate of new cancer cells. Another way of blocking the HER/HER2 interaction is with tyrosine kinase inhibitors of HER2, such as lapatinib and neratinib.^{14,15} These inhibit the function of HER2 and, when used in combination with chemotherapy drugs, create the same desired outcome of targeted antibodies, inhibiting the growth rate of the cancerous cells. The chemotherapy drug can also be conjugated directly to an antibody, such as Ado-Trastuzumab Emtansine

and Fam-Trastuzumab Deruxtecan.^{16,17} These serve as a mechanism for delivering and connecting drugs selectively to HER2 positive cancerous cells.

1.1.3. Multistage Vector Delivery

Chemotherapy drugs can be loaded into a nano- or micro-sized particles and then transported to a cancerous region of the body while minimizing the amount of interaction with healthy cells that encounter a drug that is freely circulating around the body. One example of a nanocarrier is a liposome. Liposomes are spherical vesicles composed of a lipid bilayer membrane and can transport hydrophilic and hydrophobic agents within its aqueous core or in between its lamellae, respectively.¹⁹ By encapsulating a drug inside the liposome, it can achieve higher drug concentrations in the tumour space as compared to healthy tissue.²⁰ The liposomes are then taken up by cellular phagocytosis and the high pH of the endosomes releases the drug into the cell.

Porous silicon microparticles (PSM) are disk shaped particles on the micron scale and are used to deliver drugs or antigens to cells in the body. PSM protect nanoparticles from degradation in the bloodstream and uptake by circulating immune cells, ensuring that the main effect of the drugs is focused on the cancerous region. Microparticles are usually too large to be phagocytosed by cells or to fit through the fenestrations in tumour vasculature. However, due to their size and shape, they stick to vessel walls in areas of low shear stress, such as tortuous tumour vasculature, where they begin to degrade and release their cargo.^{21,160} The cargo then enters the extracellular space through the large vasculature fenestrations around the tumour and are phagocytosed by the surrounding cells. PSM can be filled with a variety of drugs to fight diseases or antigens to stimulate the immune system against diseases.⁶

1.2. Cancer

Cancer is broadly defined as the uncontrolled replication of cells at an unnaturally fast rate. If undetected by the immune system, these abnormal cells can grow into a primary tumour mass and eventually inhibit the essential processes of the organ it inhabits.¹⁸ In some types of cancer, cancerous cells can break apart from the primary mass and infiltrate the blood and lymph vessels.²² This allows the cancer to spread to, and continue growing in, many other organs throughout the body. These

metastases are commonly associated with poor prognoses and patient death.²³ This thesis primarily focuses on breast cancer and its metastatic lesions.

1.2.1. Prevalence

Breast cancer is the most prevalent type of cancer in the United Kingdom (UK), accounting for 31% of all new female cancer cases and 7% of all cancer deaths in 2016. The prognosis for breast cancer is relatively positive, with 65% of women diagnosed surviving for twenty years or more. 1 in 7 women in the UK will develop breast cancer at some point in their lives.¹⁸

1.2.2. Causes

Cancer can be naturally caused by DNA mutations inside a cell but there are certain risk factors that have been shown to increase a person's chances of developing cancer. In breast cancer, some causation has been demonstrated with post-menopausal hormones, obesity, alcohol, and not breastfeeding, making up approximately 23% of cases in the UK.¹⁸ On the individual cellular level, accidental DNA mutations are a common occurrence inside every cell and happen during the transcription and translation phases of cell division. While most mutations either have no effect on the cell or are detrimental to its survival, a mutation to the small portion of genes that control cell division rates, cell cycle arrest, and programmed cell death could lead to an uncontrolled cell growth.²⁵ Any of the previously mentioned risk factors, as well as exposure to radiation and carcinogens, can increase the chance of DNA mutations.²⁶

1.2.3. Metastasis

As tumours continue their unrestricted growth, they can become malignant and begin to invade the surrounding tissue. This includes the bloodstream and the lymphatic system, giving the cancerous cells access to the entire body.²⁶ Breast cancers are most likely to metastasize in the lungs, liver, bone, and brain.²⁷ The current hypothesis around these specific metastasis locations is a combination of the "seed and soil hypothesis" and the "anatomical/mechanical hypothesis". The seed and soil hypothesis, which was suggested by Stephen Paget in 1889, states that the tumour

cells, or seeds, could only grow in the microenvironment of specific organs, or soils.²⁸ The anatomical/mechanical hypothesis states that metastases are most likely to form in the first organ that cancerous cells come across once they have entered the bloodstream or the lymphatic system.²⁷ Metastases can be difficult to treat due to their small size and their ability to be almost anywhere in the body makes them difficult to detect. In Chapter 2 of this Thesis, a treatment method is investigated for breast cancer metastases in bone.

1.2.4. Cancer strengths

Cancerous cells originate from host cells and are therefore very difficult for the immune system to identify as a threat because they present all the necessary identification proteins. In addition to that, the structure of tumour blood vessels prevents immune cells from having easy access to tumours,³² nutrient deficient microenvironments caused by the large energy consumption of rapidly dividing cancer cells discourage immune cell survival once they reach the tumour,³³ and tumours have the ability to recruit immunosuppressive cells to inhibit immune cell hostilities once they have arrived.³⁴

1.2.5. Treatment

In breast cancer, if the size of the tumour is small and it is at an early stage, many patients begin their treatment with surgery to remove the cancerous tissue or the entire breast. This is usually followed by radiation therapy to eradicate any cancerous cells that were left behind after the surgery. If the tumour has spread into the surrounding tissue or metastasized, a more systemic approach is required. Chemotherapy drugs are intended to destroy quickly dividing cells or prevent cell division. These drugs are not specifically targeted to cancer cells and therefore affect quickly dividing noncancerous cells in the body as well, such as skin, intestinal linings, and hair.¹⁸

Immunotherapy takes advantage of the body's own immune system to attack the cancerous cells. Cancerous cells present "self" identifying proteins, such as PDL1 and CTLA4, to immune cells to prevent being attacked. Checkpoint blockage therapy uses targeted antibodies to block these proteins from interacting and causing

immunosuppression.²⁹⁻³¹ This allows the immune system to identify and destroy cancerous cells.

Although cancerous cells are cleverly disguised from the immune system, there are still exploitable characteristics that are being explored to target and eliminate them. As cancerous cells divide and spread quickly, they release copious amounts of VEGF to stimulate blood vessel growth to deliver nutrients to the tumour mass. This expansion has a difficult time keeping up with the tumour growth rate.³⁵ Due to this, the construction of these vessels is uncharacteristically sloppy for the body. There are vessel regions with fenestrations,³⁶ wildly tortuous sections,³⁷ and some vessels lead to a dead end with no connection back to the venous system.³² Another characteristic of a rapidly growing tumour is inflammation. While inflammation has been shown to support tumour growth,³⁸ it also stimulates the expression of E-selectin on endothelial cells.³⁹ All of these vessel attributes create an environment that is unique to the rest of the body and are therefore targetable.⁴⁰⁻⁴¹

1.3. Tools for Visualization

The term cancer encompasses a family of diseases that originate from a broad array of cells that attack a variety of organs.¹⁸ While some varieties of cancer have a well-known disease progression profile and treatment strategy, others require more research to fully understand and treat them. Due to the size of the cancerous cells and their treatments, the ability to visualize, identify, and characterize micro and nano-scale objects is vital for gathering an understanding of therapy mechanics and cellular processes. Many different techniques have been developed to capture this information, each with its own advantages and disadvantages. The following techniques were all used in this work.

1.3.1. Microscopy

Brightfield microscopy is one of the most basic functions performed on a microscope. A sample is placed on the stage of a microscope and illuminated by a white light. The sample will then attenuate the amount of light passing through via scattering or absorption, which allows the user to distinguish topographical features.

⁴² The addition of fluorescence, either through autofluorescence or artificial staining, substantially increases the amount of observable information.

In fluorescence microscopy, a fluorophore absorbs photons from a laser light source at a specific wavelength which forces it into an excited electronic state (Figure 1D). ⁴³ This excited state is not energetically favourable for the fluorophore and so the absorbed energy is released by radiative and non-radiative means. It is possible for energy to be non-radiatively dissipated as heat. The radiative energy is released as a photon of lower energy (or longer wavelength) than the incident laser wavelength due to the non-radiative energy dissipation as heat. The difference between the excitation and emission wavelength is called the Stokes shift. The emitted photons are then captured by a detector to identify the location and concentration of the fluorophore. ⁴⁴ Fluorophores attached to antibodies can be used to identify cells expressing a protein of interest. Using different fluorophores, excitation wavelengths, and optical filters, multiple proteins can be identified on the same sample at the same time. ⁴⁵

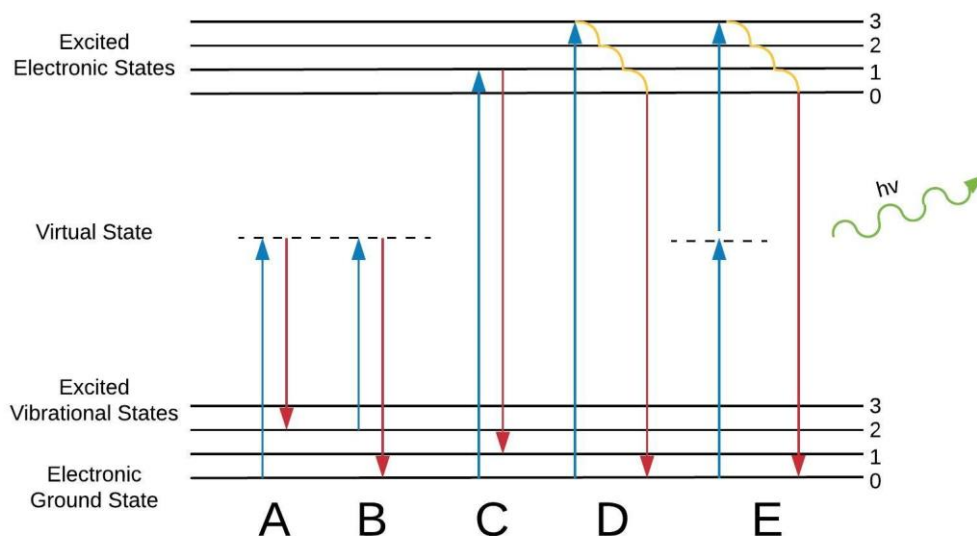


Figure 1. Jablonski diagram.

Excitation (blue) and emission/relaxation (red) for Stokes Raman scattering (A), anti-Stokes Raman scattering (B), resonance Raman scattering (C), fluorescence (D), and two-photon excitation fluorescence (E). The numbers represent the vibrational levels within the electronic states.

Single-photon excitation fluorescent microscopy is depth limited to less than 100 μm in biological tissues due to light scattering at the many interfaces between cells and the extracellular matrix (ECM). ⁴⁶ Each component in biological tissue has a

unique index of refraction. As light penetrates the tissue, it passes through many different indices of refraction along its travel path. At each interface, a portion of the light will be transmitted into the new medium while the rest will be reflected (or scattered) and continue along a new orientation in the original medium, therefore decreasing the intensity of the transmitted light. Biological tissues can also non-radiatively absorb photons. In the visible spectrum, shorter wavelengths are heavily absorbed by the skin and the blood while longer wavelengths are absorbed by the water content in the body.⁴⁷ The amount of absorption grows exponentially with an increase in the path length of the light through the tissue. The combination of these limitations prevents to use of linear microscopy on whole tissue and live imaging techniques more than surface deep.

Limiting the depth resolution of a microscope is important for identifying details throughout the depth of a cell. Traditional, widefield optical microscopy techniques end up exciting more molecules than those on the focal plane alone (Figure 2A). The entire excited volume then has the potential to emit detectable light that will be collected by the camera. The computer then takes the signal acquired from a volume and projects it onto a 2D image. The excited particles above and below the focal point will distort the features in the focal plane on the acquired image because they sum together to brighten a signal or to create a false signal.⁴⁸ This is especially important when trying to determine if a signal is coming from the interior or the exterior of the cell.⁴⁹ Another limiting factor affecting the depth resolution is the scattering of emitted light. After the target molecules have been excited and emitted photons, those photons still must travel through the same scattering material that created a problematic pathway for the excitation light. This scattered light is then recorded by the camera alongside the non-scattered light, contributing to the added distortion of the acquired image.⁴⁸

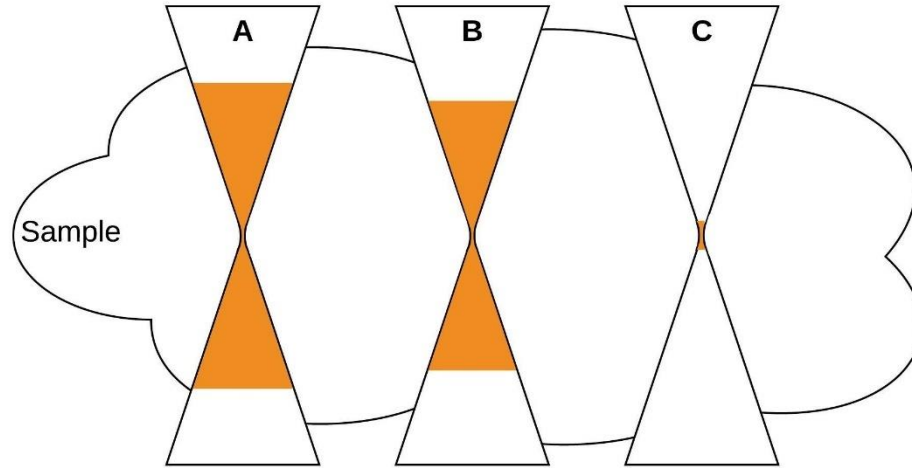


Figure 2. Comparison of optical microscopy depth of field.

A general representation of the excitable volume (represented in orange) captured during widefield microscopy (A), confocal microscopy (B), and two-photon excitation microscopy (C).

$$r_{axial} = \frac{2\lambda}{NA^2} \quad \text{Equation 1}$$

$$r_{axial} = \frac{1.4\lambda\eta}{NA^2} \quad \text{Equation 2}$$

Confocal microscopy places a micro-sized pinhole at the conjugate focal plane of the excitation source to ensure the focused light is diffraction limited and another pinhole at the conjugate focal plane of the detector to prevent out-of-focus and scattered light from being recorded (Figure 3). While this only contributes a 30% reduction on the excitation depth of field (Figure 2B; Equation 1 and 2, where r is the radius of the Airy disk, λ is the wavelength, η is the index of refraction, and NA is the numerical aperture),⁵⁰⁻⁵¹ any scattered light or emitted light from outside of the focal plane will not focus through the pinhole and will be blocked from reaching the detector. The size of the pinhole can be adjusted to decrease the amount of light allowed through to the detector and increase the depth resolution and contrast. This greatly reduces the amount of incoming light and so a photomultiplier tube (PMT) is employed to collect the signal rather than a standard camera detector. PMTs use a photocathode to absorb photons, thereby ejecting electrons that will multiply as they rebound off dynodes until they are absorbed by an anode.⁵² This creates a large gain up to 10^6 X that is directly proportional to the influx of photons. In order to capture a larger FOV, the excitation laser is scanned over the sample in a raster pattern using two mirror galvanometers, one to control each lateral axis.

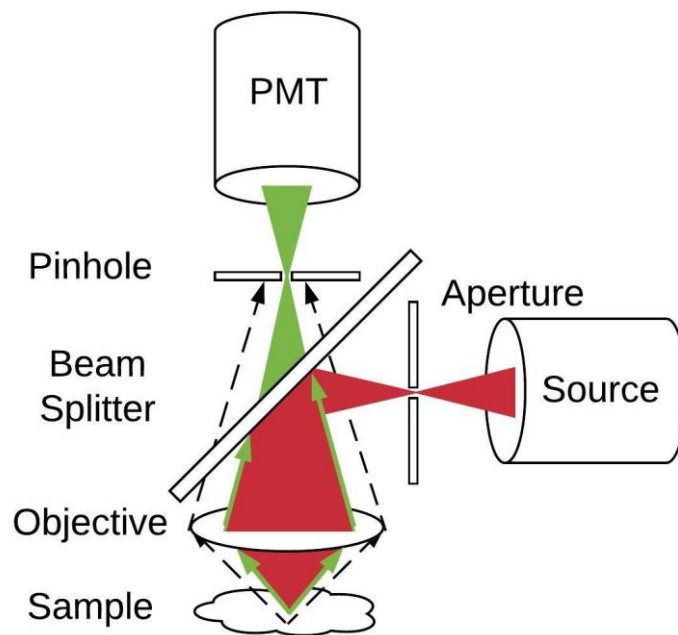


Figure 3. Confocal microscopy pinhole diagram.

Emitted light coming from the focal plane of the sample (green) passes through the pinhole whereas out-of-focus light (dashed lines) is blocked by the pinhole.

Two-photon microscopy increases the imaging depth approximately ten-fold to a maximum of 1 mm by exciting with near infrared light.⁴⁶ This technique takes advantage of nonlinear excitation by adopting two photons of lower energy (near-infrared) to simultaneously (within 0.5 fs) stimulate the fluorophore (Figure 1E). The energy of the two, low energy photons together is enough to excite the molecule to fluoresce. Even with a femto-second excitation laser source, the chances of exciting a molecule is slim and only happen in the vicinity of the focal spot. This provides z-resolution without the need of a pinhole to block out nonspecific excitations, allowing more emitted photons to be collected by the PMT. Near-infrared light is not readily absorbed nor scattered by biological tissue like much of the visible spectrum is, so it can penetrate deeper into tissues while also causing fewer stray excitations and less photodamage along the way.

1.3.2. Raman spectroscopy

During Raman scattering, an excitation source pumps light into a sample where photons inelastically scatter off molecules, causing a temporary polarization of the chemical bonds.⁵³ During this collision between the molecule and the photon, it is

possible for energy to transfer from one to the other through a change in the vibrational level of the molecule. When a molecule starts in the 0 vibrational level, an incident photon has the potential to force a molecule into a virtual state between the electronic ground state and the excited electronic state. As the molecule relaxes back down to its ground state, it may land in an excited vibrational state, maintaining a portion of energy from the photon (Figure 1A).⁵⁴ This is referred to as Stokes scattering. The molecule also has the potential to start in an excited vibrational state and then relax completely back to the ground state, leaving the scattered photon with more energy than it arrived with (Figure 1B).⁵⁴ This is referred to as anti-Stokes scattering. As the scattered photon continues along its new trajectory, it has either gained or lost energy to the molecular vibrations and is now traveling at a shorter or longer wavelength, respectively. The difference between the incident and scattered wavelengths is again called the Raman shift. It is therefore possible to identify the chemical composition of a sample by comparing the scattered Raman spectra with known shift values of simple chemical compounds.⁵³

Raman spectroscopy is useful in many biological applications due to its non-destructive ability to identify chemical structures. As it pertains to this work, Raman is employed to diagnose and estimate the prognosis of cancer in human patients.^{55,56} Specifically, Raman has been shown to identify neoplasia in many different tissues, including brain,⁵⁷⁻⁶⁰ breast,⁶¹ bladder,⁶² colorectal,^{63,64} larynx,⁶⁵ lung,⁶⁶ lymph nodes,^{67,68} oesophageal,⁶⁹⁻⁷¹ prostate,⁷²⁻⁷⁵ uterine, and cervical.⁷⁶⁻⁷⁹ Raman is also useful in aqueous environments and can therefore be utilized for 2D and 3D characterization of cellular models *in vitro*.⁸⁰⁻⁸³

1.3.3. Ultraviolet/visible absorption spectroscopy

Ultraviolet (UV) / visible spectroscopy scans a range of wavelengths from ultraviolet through the visible spectrum and measures the amount of light that is transmitted through a sample.⁸⁴ Molecules that absorb light in that range will create dips in the transmittance measurement which are directly proportional to the concentration of that molecule. This is particularly useful for DNA quantification, which has an absorption maximum at 260 nm. UV/visible absorption spectra can be used to identify and quantify DNA and other molecules in a sample.

1.3.4. Light scattering

Dynamic light scattering (DLS) measures fluctuating scattered light interference (called a speckle pattern) to determine the size of a particle. Particles evenly distributed in a fluid are in constant Brownian motion due to collisions with the liquid molecules they are suspended in. Due to conservation of momentum, the speed of these particles depends on the size of the particle, as defined by the Stokes-Einstein equation. This speed can be measured by observing the rate of fluctuation of the intensities in the speckle pattern. Since the particles are constantly moving, the speckle pattern is also continuously changing as the light scattering paths fluctuate. The size of the particles is determined by comparing the rate of the intensity fluctuation in the speckle pattern to the Stokes-Einstein equation.⁸⁵

The zeta potential of a particle is the net charge just outside the region in which ions are strongly affixed to the particle surface. To measure this potential, an electric field is applied across a particle suspension. Particles will begin to move toward the electrode of opposite charge to their zeta potential and eventually an equilibrium traveling speed will be reached. This speed depends on the strength of the electric field, the dielectric constant of the medium, the viscosity of the medium, and the zeta potential of the particles. This speed can be measured using laser doppler velocimetry. Yet again, the rate of intensity fluctuation of the scattered light is proportional to the speed of the particles and can be used to extract the exact speed value. Since the strength of the electric field, the dielectric constant of the medium, and the viscosity of the medium are known, the Henry equation can be used to determine the zeta potential of the particles.⁸⁶

1.3.5. Flow chamber

Flow chambers can be used to study cell-cell interactions and therapy-cell interactions under physiological flow conditions.⁸⁷ Using brightfield or fluorescence microscopy, objects can be monitored and tracked at shear stresses and flow speeds relevant to different sections of the cardiovascular system. It has been shown that endothelial cells will adapt their shape and alignment to the state of their media.⁸⁸ If the media is static, cells will grow in various shapes in varying directions but when the media flows laminaarly over the cells, they stretch and align in the direction of the flow.

These chambers are useful in studying the rolling and adhesion kinetics of leukocytes and therapies injected intravenously.⁸⁷ Flow chambers have also been employed to study cancer cell rolling and adhesion in the vasculature, highlighting the abilities of cancer to metastasize and spread.^{89,149} By providing physiologically relevant conditions inside the flow chamber, the behaviour of cells and particles in flow are able to be observed and then calibrated to desired interaction characteristics. This is apparent in studies intended to determine the ideal size and shape of a drug delivery particle into torturous vasculature regions such as those in the tumour space.^{90,91}

1.3.6. Immunohistochemistry

Histochemistry is a common method for staining cells in a tissue sample to observe biological structures. Tissue samples are collected from patients, fixed, and either preserved in paraffin or cryogenically frozen. The samples are sliced into <20 µm thick sections and stained with non-specific or antibody specific fluorophores. The thickness of the samples helps create an approximate monolayer of cells that allows every cell to be stained and imaged without worrying about fluorophore penetration depth or light scattering. These thin slices are imaged on a fluorescence microscope to identify cells and their biological layout.⁹²

1.3.7. Inductively coupled plasma mass spectroscopy

Inductively coupled plasma mass spectroscopy (ICP) is a quick and accurate method for measuring ion concentration in a biological sample. This is an important quantification method for biodistribution and conjugation efficiency studies. Samples are first blended or dissolved down to a slurry or a solution, respectively, and then run through an analytical nebulizer to convert the liquids into a fine mist. This mist is then passed through the inductively coupled argon plasma torch to ionize the passing molecules. Mass spectroscopy is then used to separate and quantify the different ions. These values are compared to known concentration controls to determine the concentration of an ion in a sample.⁹³

Table 1. Summary of Visualization Tools Utilized.

A comparison of the visualization tools utilized in this work, the underlying physical principle for collecting data, the end goal of the technique, and the usefulness of the tool in cancer medicine.

Visualization Tool	Underlying Physical Principle	Measurement Objective	Use for Cancer	References
Confocal Microscopy	Light excitation/emission	High-resolution imaging	3D analysis of tumour tissues	37,203,204
Raman Spectroscopy	Inelastic light scattering	Molecular composition	Cancer identification	94,95,96,97
UV Spectroscopy	Light absorption	Molecular composition	Cancer identification	98,99
DLS	Light scattering angles	Particle size and zeta-potential	Characterizing drug delivery vehicles	100,101,102
Flow Chamber	Shear stress	Adhesion dynamics	Customizing drug delivery vehicles	103,149,150
Immunohistochemistry	Antibody affinity	Cell expression differentiation	Cancer identification	105,106,107,108

1.4. Tools for Analysis

Analysing large data sets can be time consuming and sometimes impossible to do by hand alone. Some programs provide basic functionality, but each experiment should be designed to test a specific hypothesis, not to fit a predefined analysis procedure. This Thesis focuses on experiments without widely available analysis methodologies. While ImageJ could have been utilized to provide the base analysis framework for this thesis, MATLAB was used to program the analyses for this work due to the author's familiarity with the language. The analysis methods listed here are the main commercial options for similar experiments.

1.4.1. 3D Visualization Software

Current commercial softwares have done remarkable work in the field of 3D visualization of biological fluorescent samples. Cell profiler (The Carpenter Lab, The Broad Institute, Cambridge, MA, USA) is a cell image analysis software that recently added the ability to visualize and identify objects in 3D but is still relatively new to the area of 3D analysis. Amira (ThermoFisher Scientific, Waltham, MA, USA) is a 3D analysis software for biological applications. This software has been shown to be effective at vessel segmentation and single cell identification, including measurements for each.³⁷ Imaris software (Bitplane, Zurich, Switzerland) has the most functionality out of all 3D analysis softwares currently available. Not only does it identify vessels and cells, but it can also quantify cell colocalization, track cell movements, and work

in conjunction with outside programming languages. This gives the user the flexibility to customize their analysis to their specific experimental goals.

Some scientists have created algorithms as plugins for ImageJ to investigate very specific 3D organizations and structures. TANGO is a software developed to visualize and characterize the nucleus of a cell in 3D.¹⁰⁹ This software enables life scientist to extract quantitative data from fluorescent imaging to classify the functionality of different nuclear compartments. A separate lab created an ImageJ plugin, BoneJ, which allows a user to reduce large vessels, bones, and other long objects into pixel sized strings that are more readily visualized and interpreted.¹¹⁰ Also, a software called DiAna has the functionality to segment 3D images and provide colocalization and distance measurements.¹¹¹ Each of these softwares utilized the ImageJ framework and expanded it to create their own specific analysis methods.

1.4.2. Cell tracking Software

Most cell tracking softwares are developed with the intension of tracking cells migrating on a petri dish over long periods of time. Ibidi (Ibidi, Gräfelfing, Germany) provides multiple video analysis softwares that track cells during migration and wound healing which has the possibility of translating to flow chambers if the frame rate is high enough. Nikon provides its own tracking software with NIS Elements (Nikon, Tokyo, Japan) for videos taken on a Nikon microscope which can be used to identify objects in consecutive frames and determine object speed through the field of view. The open source software ImageJ (National Institutes of Health, Bethesda, MD) has methods that can track objects in up to 5 dimensions. ImageJ also allows functionality with exterior programming languages such as MATLAB and Python for improved customization.

Scientists have taken advantage of the ability to write plugins for ImageJ for their own scientific purposes. Most recently, Trackmate is a plugin that was developed for ImageJ that allows the user to identify and track individual particles or small spots that also provides a user interface, visualization software, and the ability to write extendable code to enhance and customize the experience.¹¹² This and other programs like it have been developed for the field of life sciences with the goal of providing a framework for life scientists to utilize or build from.¹¹³⁻¹²¹

1.4.3. Raman Peak Location Software

The main segment of Raman peak identification softwares require that the signal be acquired either without fluorescence or sufficiently far away from a fluorescence signal to have relatively low background. Origin (OriginLab, Northampton, MA, USA) is a general data analysis and graphing software that has peak analysis functionality that can be used to identify individual Raman peaks. It is also compatible with external programming languages to allow the implementation of more custom protocols. Other softwares, such as Knowitall ID Expert (Malvern Instruments, Malvern, UK) and WIRE (Renishaw, Wotton-under-Edge, UK) are specific to their Raman spectrometers from Malvern and Renishaw, respectively. These softwares have large databases from which to pull Raman peak information and match it to an acquired, unknown spectrum.

1.5. Outline of Thesis

This thesis investigates using multiple imaging modalities and analysis methodologies to assess the mechanisms controlling cancer diagnosis and treatment. A study into the rolling speed and adhesion kinetics of a discoidal porous silicon microparticle (PSM) drug delivery vehicle is discussed in Chapter 2. By employing a parallel plate flow chamber to mimic the conditions of tumour vasculature and custom analysis software, PSM adhesion to vascular endothelial cells are discovered to be negatively affected by the addition of targeting surface aptamers. It is hypothesized that this effect is due to a change in particle size and surface charge. Chapter 3 examines the use of optical tissue clearing and deep tissue staining to explore the 3D tumour microenvironment. Specifically, looking at the development of tertiary lymphoid structures (TLS) in association with a cancer vaccine treatment shows an increase in the amount of TLS in the tumour space. This increase in TLS also correlates with a decrease in the tumour size development over time. Chapter 4 is devoted to developing a methodology for extracting both Raman and fluorescent signals from overlapping spectrums. These electromagnetic signals can then be employed in discovering and identifying cancer in human patients. Each of these methodologies is brought together in Chapter 5 to discuss and summarize their conclusions and future uses in the field of cancer diagnosis and treatment.

2. DISCOIDAL POROUS SILICON MICROPARTICLE ADHESION AND ROLLING KINETICS

2.1. Introduction

Late stage breast cancer has the potential to develop solid tumour metastases in the bone, with over 60% of oestrogen receptor-positive breast cancer and about 10% of triple negative breast cancer patients eventually developing bone metastases.¹⁰⁴⁻¹²² Although bone metastases are not usually the cause of death in these patients, the symptoms normally associated with bone metastasis, including pathological fractures, chronic bone pain, spinal cord compression, and life-threatening hypercalcemia, can have harsh consequences on the quality of life of the patient.¹²³⁻¹²⁵ While several treatment strategies have demonstrated the ability to manage the tumour progression using bisphosphonates¹²⁶ or neutralizing RANKL through antibody targeting¹²⁷⁻¹²⁸, only a small population of patients experienced improved survival.¹²⁹⁻¹³⁰

To treat metastatic cancers, therapeutics must be efficiently delivered to the tumour lesion in potent quantities. This is especially difficult in bone tumour metastases due to the bone metastasis microvessel density in half of breast cancer patients being significantly reduced as seen in histological analyses.¹³¹ This leads to a significant disadvantage when delivering intravenous therapeutics to metastatic tumours compared to primary tumours. This disadvantage is further multiplied by the perivascular region of bone marrow providing metastatic niches for cancer cells to grow in, as well as shielding them from circulating therapeutic agents.¹³² Circulating tumour cells are attracted to the bone marrow by chemo-attractants and pro-growth factors, such as E-selectin,¹³³ produced by the perivascular stromal cells and the endothelium.

E-selectin is a leukocyte adhesion molecule that is expressed by bone marrow endothelial cells¹³⁴ and endothelial cells in areas of high inflammation.³⁹ When expressed on the vessel walls, E-selectin interacts with the ligands on leukocytes to slow their rolling speed and subsequently attach them to the vessel wall, allowing them the opportunity to exit the blood stream and penetrate the extravascular space.¹³³ E-selectin also plays a large role in the ability of chemotherapy agents and radiation to affect cells in these vascular niches. Hematopoietic stem cells (HSC) reside inside the

vascular niches in the bone marrow surrounded by endothelial cells expressing E-selectin. When the E-selectin is blocked, the rate of HSC proliferation is retarded, creating less opportunities for chemotherapy and radiation to affect the cell during division. When E-selectin knockout mice were treated with chemotherapy and radiation, the survival rate of HSCs increased by 3- to 6-fold.¹³⁴ Although blocking E-selectin can prevent unwanted side effects, bone marrow has a natural tendency to produce E-selectin and inflammation caused by the uncontrolled growth of a tumour lesion increases the amount of E-selectin in the tumour microenvironment, therefore E-selectin has also been demonstrated to be a promising target for the delivery of cancer therapeutics.¹³⁵⁻¹³⁶

Mann et. al. has previously developed a discoidal porous silicon microparticle (PSM) delivery method that takes advantage of tumour vessel tortuosity and fluid dynamic properties of a disk to selectively deliver cancer therapeutics to tumour masses.⁴⁰ These PSM are chemically conjugated with 3-aminopropyltriethoxysilane (APTES) to obtain a positive surface charge to provide a favourable environment for loading negatively charged therapeutics.⁴⁰ The Poiseuille flow inside the blood vessels continuously pushes the PSM toward the vessel walls and into the red blood cell-free plasma layer (RBCFL) where the particles become trapped.¹³⁸ Their discoidal shape allows for a more economical attachment to the vessel walls when compared to microspheres because they have a larger surface area to bind to the vessel wall and additionally, a small surface area exposed to the forces caused by the continuing flow.¹³⁹ Once attached, these PSM slowly degrade into bioinert orthosilicic acid, safely releasing their therapeutics around the tumour mass.¹⁴⁰ This delivery method has been successful at delivering siRNA oligos to primary breast cancer¹⁴⁰ and metastatic ovarian cancer.¹⁴¹⁻¹⁴² Mann et. al. has also developed a thioaptamer (ESTA) that specifically binds to E-selectin.¹⁴³ By chemically conjugating the ESTA to the PSM (E-PSM), an effect similar to leukocyte rolling is hypothesized to be achieved in the vasculature. Using E-PSM, the accumulation of PSM in the bone marrow increased dramatically⁴⁰ and when loaded with STAT3 siRNA, they demonstrated an effective inhibition of metastatic MDA-MB-231 tumour growth.¹⁴⁴ Although the use of E-PSM has been demonstrated to be therapeutically effective, understanding and improving the adhesion kinetics behind the increased E-PSM accumulation is an important

component in the process of improving the efficacy of this treatment. In this study, two separate techniques are used to identify the accumulation mechanism, diffusing colloidal probes and flow chamber assays.

When PSM are mixed into a liquid via sonication, they form a metastable colloid suspension. If left alone, gravity will eventually cause the PSM to fall out of suspension. This effect can be taken advantage of by allowing the PSM to slowly collect on the surface of a substrate and then observing their lateral movements, or Brownian motion, produced by collisions with vibrating molecules. By measuring the distance travelled by the PSM, it is possible to characterize potential interactions between the PSM and the substrate. Conservative forces (such as van der Waals, electrostatic, and steric interactions) and nonconservative forces (such as dissipative hydrodynamic interactions) are ever present between the PSM and the substrate beneath it and can have a substantial influence on the lateral motion of the PSM.¹⁴⁵ From this, it can be inferred that PSM that have stopped moving are impeded by an external force and weak interactions would slow down the Brownian motion, decreasing the distance travelled. It is also possible that particles moving unencumbered for large distances have steric interference between them and the surface to prevent other attractive forces from hindering their movement.¹⁴⁵ This technique has been used to study the strand length of DNA by attaching one end of DNA to a surface, the other end to a nanobead, and tracking the Brownian motion of the nanobead.¹⁴⁶ Here, diffusing colloid probes are used to corroborate the adhesion characteristics identified using the flow chamber.

Flow chambers are a common instrument for studying leukocyte rolling, slowing, and adhesion on endothelial cells.¹⁴⁷⁻¹⁴⁸ They have also been applied to successfully study similar adhesion of spherical microparticles¹⁴⁹ and even PSM.¹⁵⁰ By providing a consistent, laminar, Poiseuille flow profile, flow chambers imitate microvessels by providing physiological flow speeds¹⁴⁹ and/or shear stresses.^{138,151} In this study, it was decided to emulate the shear stresses on the vessel walls to replicate the forces that the particles must withstand to attach and remain attached. Shear stress is a measure of force on the chamber wall caused by the friction of the flow moving past it and is dependent on the viscosity of the fluid. Shear rate is a flow characteristic that is dependent on the flow rate and the geometry of the flow chamber yet is

independent of viscosity.¹⁵² Therefore, shear rate is matched as it governs PSM-cell collisions and PSM-cell contact time.

The average shear rates in healthy human microvascular have been reported¹⁵³⁻¹⁵⁴ to be $632 \pm 73 \text{ s}^{-1}$ whereas the average shear rates in human tumour microvascular is around $77 \pm 6 \text{ s}^{-1}$.¹⁵⁵ These numbers differ from mice where the average shear rates in healthy mouse microvascular has been reported¹⁵⁶ to be $275 \pm 20 \text{ s}^{-1}$ whereas the average shear rates in mouse tumour microvascular is around $188 \pm 50 \text{ s}^{-1}$.¹⁵⁵ The attachment ability of the discoidal PSM have previously been studied using flow chambers at shear rates well below physiological levels ($<50 \text{ s}^{-1}$) and only on extracellular matrix (ECM) substrates.¹⁵⁰ The conclusions from these experiments have showcased the previously mentioned benefits of the discoidal shape in a laminar flow. When flow chamber studies have been performed over cells, again at lower than physiological shear rates ($<10 \text{ s}^{-1}$), spherical microparticles adhered to the exposed collagen coated plate at a higher rate than they adhered to the endothelial cells.¹⁴⁹ At the time of writing this dissertation, there have been no know studies on discoidal microparticles flowing over endothelial cells. Previous in vivo studies with the E-PSM were performed in mice and therefore, in the current in vitro study, the rolling mechanics of PSM on HUVECs are investigated at the physiological levels of shear rate in the mouse tumour microvasculature.

2.2. Materials and Methods

2.2.1. Cell lines

Human umbilical vein endothelial cells (HUVECs) were purchased from ATCC (Manassas, VA, USA). Cells were maintained at 37°C with 5% CO_2 in medium 200 (M200, ThermoFisher Scientific, Waltham, MA, USA) supplemented with 2% low serum growth supplement (LSGS, ThermoFisher Scientific, Waltham, MA, USA) and 2% antibiotic-antimycotic (AA, ThermoFisher Scientific, Waltham, MA, USA). Cells were plated on plastic cell culture plates coated with gelatine attachment factor protein (AF; ThermoFisher Scientific, Waltham, MA, USA). Cells were split at 80% confluency. All experiments were performed on 100% confluent cells of passage 5-8.

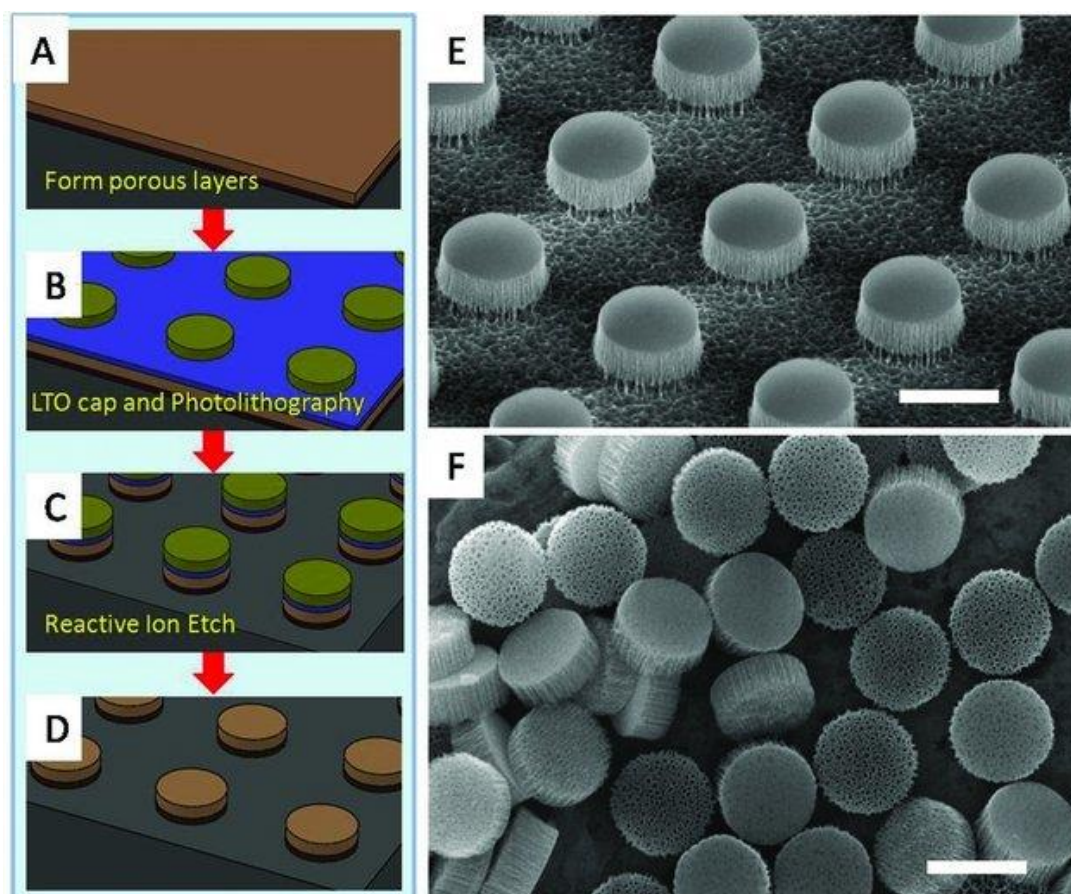


Figure 4. Fabrication of discoidal porous silicon microparticles.

A) A layer of porous silicon film is laid down. B) The porous silicon was then capped with a low temperature silicon oxide via low pressure chemical deposition and photoresistant caps were patterned in the discoidal shape via photolithography. C) Ion etching was used to remove excess material. D) Hydrofluoric acid is used to strip the silicon oxide and photoresistant caps. E) SEM image of 1000 nm \times 400 nm discoidal porous silicon particle array retained on a wafer. F) Released monodispersed 1000 nm \times 400 nm discoidal porous silicon particle. Scale bars are 1 μ m. Figure reproduced with permission from the rights holder, Wiley Library.¹⁵⁹

2.2.2. PSM Preparation

Discoidal PSM were fabricated by electrochemical etching of silicon wafers by Dr. Xuewu Liu's lab at Houston Methodist Research Institute following previously described methods (Figure 4).¹⁵⁹ These PSM were 2.6 μ m in diameter and 0.7 μ m tall. The surface of the PSM was oxidized using H₂O₂ and functionalized by chemical conjugation to APTES (Figure 5).¹⁴² The E-selectin thioaptamers (ESTA) or scrambled thioaptamers (SCR), developed by Mann et al¹⁴³ and produced by Integrated DNA Technologies (Coralville, IA, USA), were chemically conjugated to the APTES using succinimidyl 4-(N-maleimidomethyl)cyclohexane-1-carboxylate (SMCC) as a polylinker.⁴⁰ Briefly, 10 billion APTES-modified PSM (A-PSM) were

dispersed in 0.1 mL of dimethylformamide (DMF), gently vortexed, and briefly sonicated. 0.5 mL of 10 mM SMCC in DMF and 20 μ l of diisopropylethylamine (DIPEA) were then added and the reaction system was shook, end-to-end, for 4 hours. The PSM were then centrifuged for 9 minutes at 11,000 RPM and washed in DMF, 3 times. The resulting maleimide-PSM were dried in vacuum at room temperature overnight. 5 nmol of each aptamer (SCR and ESTA) were separately activated using 100 μ l of 10 mM immobilized TCEP disulphide reducing resin in phosphate buffered saline (PBS) to ensure any oxidized sulphur-sulphur bonds returned to sulphur-hydrogen bonds. Activated aptamers were then subjected to purification by PD-10 columns. The 10 billion maleimide-PSM were well dispersed in 100 μ l of PBS buffer (pH 7.2, 0.1 M phosphate salts, 0.15 M NaCl, 5mM EDTA). The maleimide-PSM and the activated aptamer solutions were mixed together and kept on a shaker for 4 hours at room temperature and then overnight at 4°C. The solution was washed with PBS buffer and centrifuged until there was no aptamer detectable in the supernatant by Ultraviolet (UV) spectroscopy. Aptamer conjugated PSM were dried in vacuum at room temperature for 12 hours and then stored in -20°C for the following experiments.

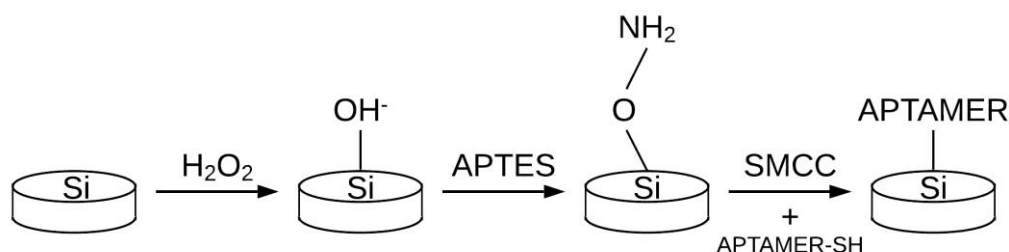


Figure 5. Chemical Modification of PSM.

A PSM (left) is oxidised using H_2O_2 to open an OH^- binding site (left-centre), conjugated to APTES to create an amine group (right-centre) which conjugates to the NHS ester on the SMCC linker, opening a maleimide group to conjugate to the thiol group on the aptamer (right).

Size and zeta potential of the A-PSM, E-PSM, and scramble (SCR) aptamer-modified PSM (S-PSM) were detected with a Zetasizer (Table 2; Malvern Panalytical, Malvern, UK). The size value is in between the diameter of the flat surface and the height of the particles because the Zetasizer calculates size assuming particles are spherical. Therefore, the reported diameter sizes are smaller than the actual values. Fourier transform infrared (FT-IR) spectroscopy was performed on the PSM using a Nicolet 6700 FT-IR spectrometer (ThermoFisher Scientific, Waltham, MA, USA)

with smart diamond crystal attenuated total reflection to ensure the thioaptamers had successfully grafted onto the PSM (Figure 6). To determine grafting density of ESTA on PSM, 3.75 billion E-PSM were dissolved in 3 mL 1 M NaOH solution overnight. Phosphorus concentration was detected using a Varian 720-ES inductively coupled plasma optical emission spectrometer (ICP, Varian, USA). Yttrium was used as internal control (Table 2).

Table 2. PSM Characteristics Post-Modification.

The average diameter, average zeta potential, and the average number of aptamers per particle for each PSM functionalization group were measured and calculated using DLS and ICP.

PSM Modification	Average Diameter (nm)	Average Zeta Potential (mV)	Aptamers/Particle
APTES	989.9	11.55	Below std Range
SCR	1247	-27.63	7.8×10^5
ESTA	1303	-23.87	7.6×10^5

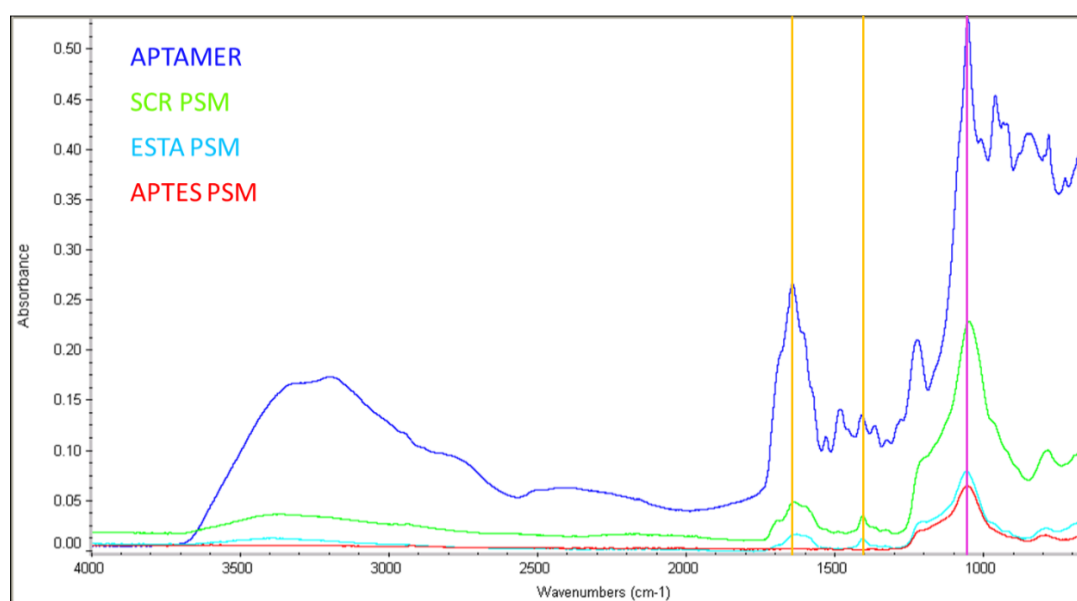


Figure 6. FTIR Absorbance Spectrum of the PSM.

FTIR was performed on dried PSM and the resulting spectra were compared to know chemical bond peak values. Gold lines indicate peaks corresponding to the thioaptamer phosphate backbone (1650 , 1400 cm^{-1}) and the pink line indicates a shared peak between the phosphate backbone (1060 cm^{-1}) and the oxidized silicon (1050 cm^{-1}).¹⁶⁰

2.2.3. Flow Cytometry

To measure E-selectin overexpression in HUVECs, cells were treated with the positive control TNF- α (20 ng/mL) in M200 for 6 hours at 37°C with $5\% \text{ CO}_2$. Cells

were incubated in 20mM Ethylenediaminetetraacetic acid (EDTA) for 3 min at 37°C with 5% CO₂ to detach the cells from the bottom of the plate. They were then stained with 1:20 diluted PE conjugated mouse anti-human E-selectin antibody (CD62E) and 1:200 diluted FITC conjugated mouse anti-human endoglin antibody (CD105, BD Biosciences, San Jose, CA) in 2% FBS, and analysed with a BD Accuri C6 Plus flow cytometer (Figure 7; BD Biosciences, San Jose, CA, USA).

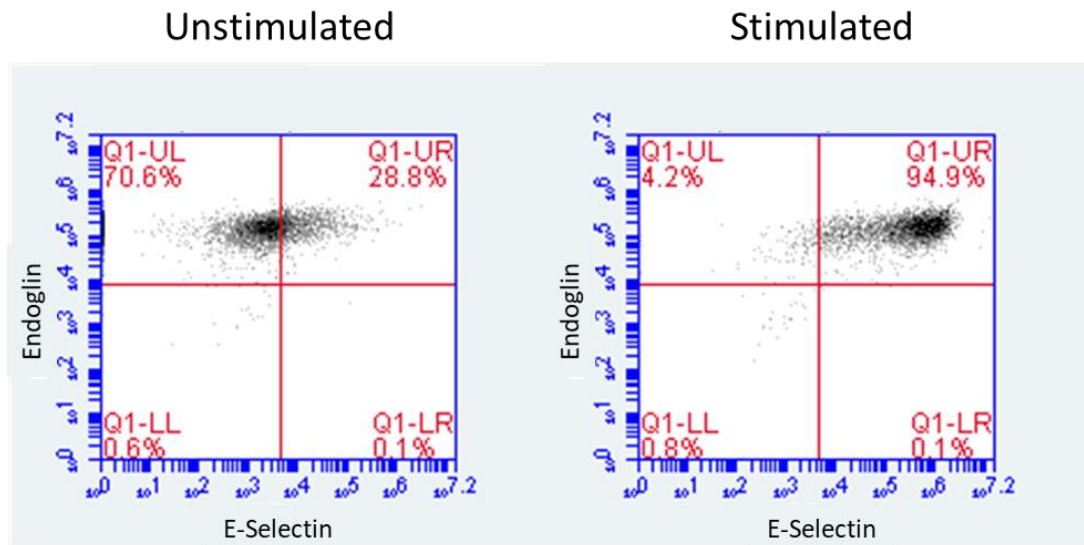


Figure 7. TNF- α Activates HUVECs to Overexpress E-Selectin.

HUVECs were incubated with TNF- α for 6 hours to activate E-selectin. Endoglin was used to ensure cells were still proliferating.

2.2.4. Static Chamber Setup

Aptamer modified PSM were conjugated to Alexa Flour 555 NHS ester (AF555; ThermoFisher Scientific, Waltham, MA, USA) via the remaining amine domains. Once HUVECs reached 80-90% confluency in 33 mm petri dishes they were stained with 15 μ l of long-chain dialkylcarbocyanine (DiO; ThermoFisher Scientific, Waltham, MA, USA), a live cell membrane stain, in 1 mL of M200 for 40 min at 37°C. The cells were then washed 3 times with M200 to remove excess DiO. The HUVECs were treated with 20 ng/mL of TNF- α for 6 hours at 37°C to activate the overexpression of E-selectin. HUVECs were washed once with PBS and then 8 million PSM mixed with fresh M200 were added to the cells. The particles were allowed to settle for 4 minutes before imaging commenced to lessen the effects of particle motion caused by pipetting the media onto the plate. An initial fluorescent image was taken of

the DiO stained cells to establish cell membrane locations (Figure 8). 1-minute-long fluorescent videos were taken using a Nikon Ti-E inverted microscope (Nikon, Tokyo, Japan) every other minute for 10-20 minutes. The exposure time for each frame was 300 ms using a 20X objective. The same experiment was also performed on unmodified dishes with PBS and with AF modified dishes with PBS or M200.

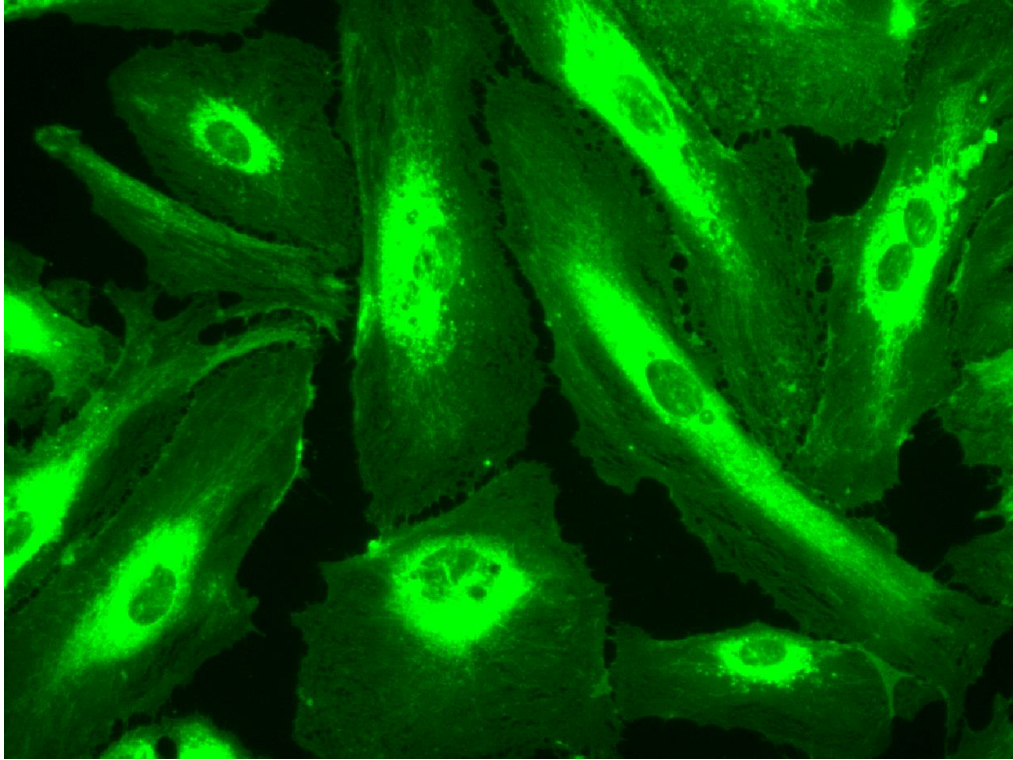


Figure 8. Fluorescent layer of HUVECs for the static chamber studies. HUVECs were stained with DiO for 40 minutes to stain the cell membranes and then incubated with TNF- α for 6 hours to activate E-selectin.

2.2.5. Flow Chamber Setup

A parallel plate flow chamber (Glycotech, Rockville, MD, USA) with a 0.005-inch-thick silicone gasket was utilized to imitate a physiologically relevant microvasculature environment (Figure 9A). The average shear rates in healthy mouse microvascular has been reported¹⁵⁶ to be $275 \pm 20 \text{ s}^{-1}$ whereas the average shear rates in mouse tumour microvascular is around $188 \pm 50 \text{ s}^{-1}$.¹⁵⁵ The velocity needed in the flow chamber to replicate the shear rate felt by a PSM in the mouse tumour microvasculature was determined using the shear rate equation for parallel plates,¹⁶¹

$$\gamma = \frac{6*Q}{w*h^2}, \quad \text{Equation 3}$$

where γ is the shear rate, Q is the flow rate, w is the width of the chamber ($w = 2.5$ mm), and h is the height between the two parallel plates ($h = 127$ μm).

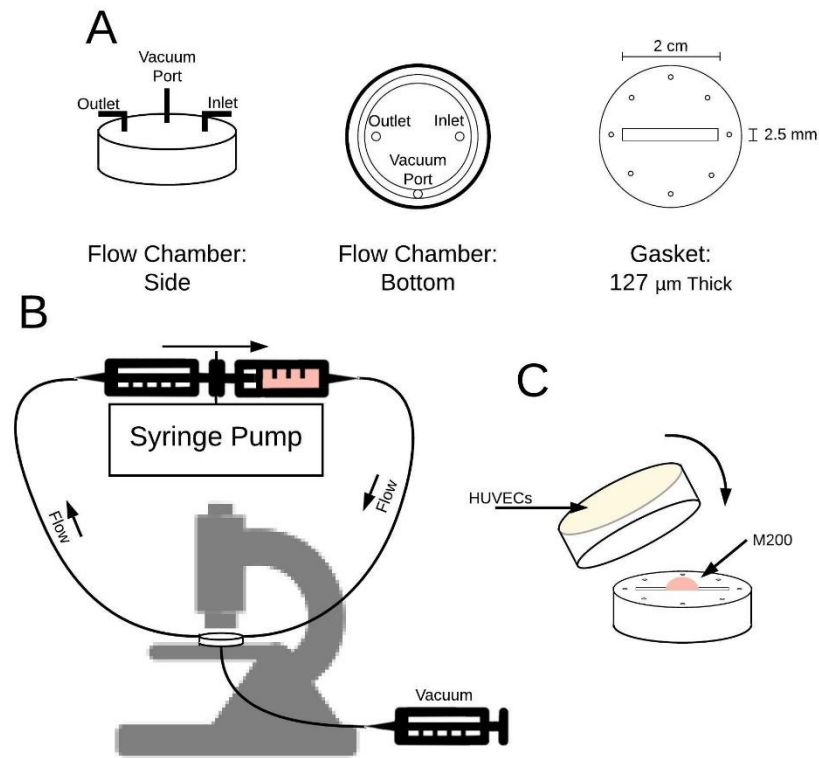


Figure 9. Flow Chamber Experimental Setup.

A) The flow chamber design allowed for inlet and outlet ports, as well as a vacuum port used to firmly hold the flow chamber to the plate. B) The imaging system consisted of the flow chamber, a syringe pump, three syringes, and a microscope. C) Plates were attached to the flow chamber upside down to ensure the area outside of the flow area remained dry to ensure a secure and stable seal. After the vacuum was in place, the system was inverted back to the cells being oriented below the flow chamber.

The flow rate used here was 70 $\mu\text{l}/\text{min}$ which gives a shear rate of approximately 174 s^{-1} , within one standard deviation of the average shear rate in the mouse tumour microvasculature. This flow rate was supplied by a PHD ULTRA™ Syringe Pump (Harvard Apparatus, Holliston, MA, USA) using two syringes, one infusing and one withdrawing on opposite sides of the flow chamber, to ensure a continuous and steady fluid flow (Figure 9B). The mean velocity was calculated from the flowrate using the equation,

$$Q = v * A , \quad \text{Equation 4}$$

where v is the average velocity and A is the cross-sectional area of the flow chamber ($A = 0.32 \text{ mm}^2$). The Reynolds number of the flow chamber was calculated using the equation, ¹⁶²

$$Re = \frac{Q * \rho}{\mu * w}, \quad \text{Equation 5}$$

where Re is the Reynolds number, ρ is the fluid density ($\rho \sim 1000 \text{ kg/m}^3$), and μ is the dynamic viscosity ($\mu \sim 1.002 \text{ mPa} \cdot \text{s}$). Reynolds number here was 0.47 which is well below the laminar-turbulent transition and the flow is therefore laminar after an entry length of $2.4 \text{ }\mu\text{m}$ as determined by, ¹⁶²

$$L = 0.04 * h * Re . \quad \text{Equation 6}$$

The forces against a PSM attached to the wall in the same position and orientation as shown in Figure 10 are a combination of form drag and skin friction. These were calculated using the drag equations (Figure 10),

$$F_{FD} = C_D * \rho * A * \frac{v_{avg}^2}{2}, \quad \text{Equation 7}$$

$$F_{SF} = \int_{surface} C_F * \frac{\rho * v(y)^2}{2} dA, \quad \text{Equation 8}$$

where,

$$v(y) = \frac{3}{2} * v * \left(1 - \left(\frac{y}{h}\right)^2\right), \quad \text{Equation 9}$$

$$C_F = \frac{0.664}{\sqrt{Re}}, \quad \text{Equation 10}$$

where F_{FD} is the form drag force, F_{SF} is the skin friction drag force where the surface of interest is a single base of the PSM, C_D is the drag coefficient for a thin disk ($C_D = 1.1$), $v(y)$ is the velocity of the fluid as a function of height (Figure 10), and C_F is the skin friction coefficient as defined by the Blasius solution. ¹⁶³ Due to its streamline shape, the force due to skin friction (0.31 fN or $3.1 \times 10^{-16} \text{ N}$) is two orders of magnitude larger than that of form drag (1.77 aN or $1.77 \times 10^{-18} \text{ N}$). The total amount of force on a PSM attached to the wall in the flow chamber is $\sim 0.312 \text{ fN}$ ($3.12 \times 10^{-16} \text{ N}$). This is the force the adhered PSM must resist if it is to stay attached to the vessel wall.

Although it was not investigated here, the adhesion force calculation could be used in future work to determine the attachment strength of each PSM. This would require an experiment in which PSM could adhere to a surface prior to experiencing the fluid shear forces. The flow rate would then be discretely increased and the number of particles that detach at each flow rate would be observed. From the flow rate and the above equations, the attachment force of each PSM could be ascertained.

Table 3. Flow Chamber Specifications.

Values either set or calculated for the setup described above. All force values are calculated for a PSM with a diameter of 2.6 μm and a height of 700 nm.

Specification	Value
Flow Rate (Q)	70 $\mu\text{l}/\text{min}$
Shear Rate (γ)	174 s^{-1}
Gasket Thickness / Chamber Height (h)	127 μm
Chamber Width (w)	2.5 mm
Chamber Length (l)	2 cm
Average Fluid Velocity (v)	3.65 mm/s
Fluid Density (ρ)	1000 kg/m^3
Dynamic Viscosity (μ)	1.002 mPa. s
Reynolds Number (Re)	0.47
Entry Length (L)	2.4 μm
Drag Coefficient (C_D)	1.1
Drag Force (F_{FD})	1.77 aN (1.77 X 10 ⁻¹⁸ N)
Skin Friction Coefficient (C_F)	0.97
Skin Friction Force (F_{SF})	0.31 fN (0.31 X 10 ⁻¹⁵ N)

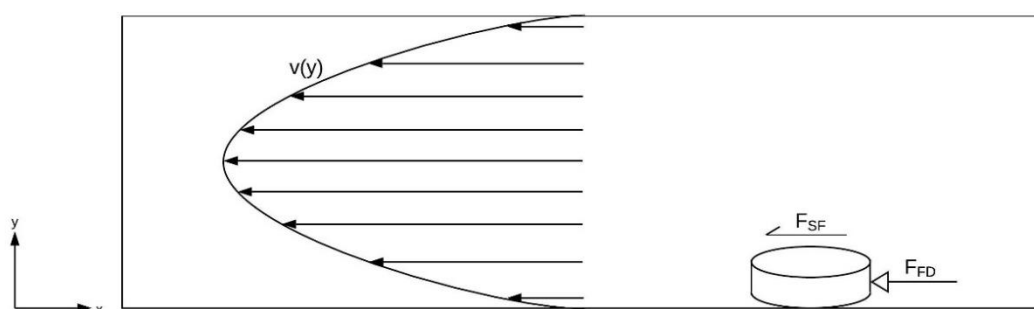


Figure 10. Forces on an Attached PSM.

The well-developed, parabolic velocity profile, $v(y)$, and the resulting drag forces on a stationary PSM (left). F_{SF} is the force of skin friction which acts on the top surface of the PSM. F_{FD} is the force of form drag which acts on the vertical surface of the PSM facing into the flow (right).

2.2.6. Flow Chamber Sample Preparation

Nine vials of each type of PSM modification (APTES, SCR, and ESTA) were prepared at a concentration of 55 million PSM in 3.5 mL M200 and were sonicated before each experiment. The PSM were tested on 3 different surfaces each: Attachment factor only, unstimulated HUVECs, and stimulated HUVECs. 18 plates of HUVECs were grown to 100% confluency. The 9 stimulation group plates of cells were incubated with 20 ng/mL of TNF- α for 6 hours at 37°C with 5% CO₂ to activate the cells to overexpress E-selectin. The incubation start time for each plate was staggered so that each plate finished incubating immediately prior to it being imaged. The tubing between the syringe and the flow chamber, or dead space, was filled with 1mL of PSM-free M200 to give the cells time to acclimate to the flow and clear away any debris before PSM were introduced and imaging began. Plates were attached to the flow chamber dry and upside down (Figure 9C). A single drop of M200 was placed on the flow chamber to keep the cells within the flow chamber wet and to prevent bubbles from forming. A vacuum seal was then applied to the edges of the chamber using a syringe. The chamber and cells were then inverted back upright, with the cells below the flow chamber.

2.2.7. Flow Chamber Imaging

Imaging was performed at the centre of the chamber, sufficiently far enough from the inlets and the chamber edges to avoid entry length or drag effects, providing a fully developed Poiseuille flow profile. All images and videos were acquired from a Nikon Ti-E inverted microscope (Nikon, Tokyo, Japan) surrounded by an incubation chamber (OKOLAB USA Inc., San Bruno, CA, USA) with a 10X objective. The stage was kept at 37°C with 5% CO₂ for the entirety of the imaging process. The microscope was focused on the top surface of the cells and a brightfield image was taken of the cells prior to the initiation of flow to help the software identify gaps in confluency. Also, by focusing on the top surface of the cells, only PSM rolling on the cell surfaces would be in focus. The PSM mixture was loaded into the syringe and flown over the HUVECs at 70 μ l/min for a total of 30 minutes for each plate, imaging in brightfield mode the entire time with 50 ms of exposure time for each frame. The imaging window was approximately 690 x 520 μ m, or 0.36 mm².

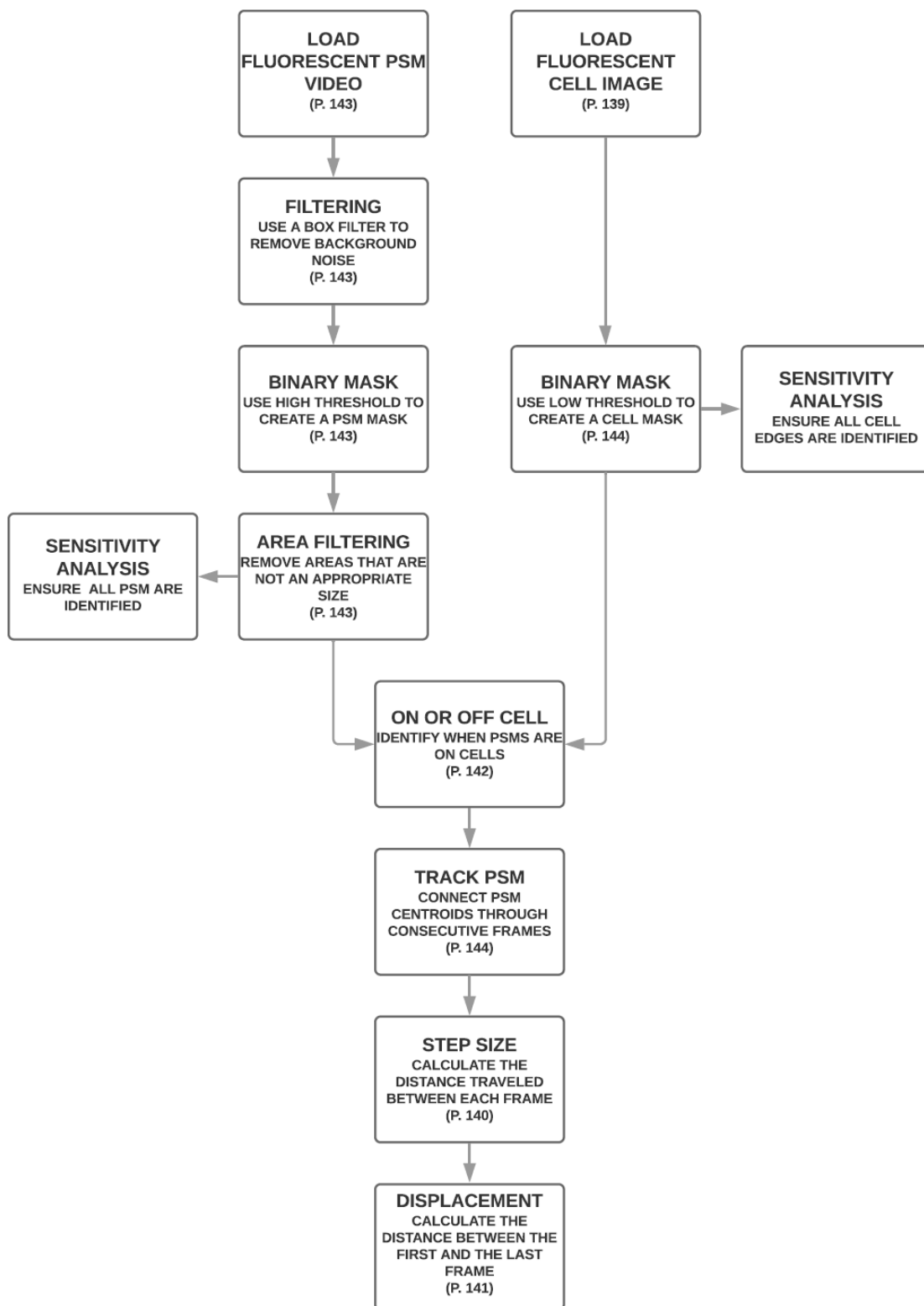


Figure 11. Static chamber analysis process diagram.

An overview of the data processing algorithm used to filter, identify, and analyse the Brownian motion of the PSM (page numbers refer to the respective piece of code in APPENDIX A).

2.2.8. Static Chamber Analysis

A MATLAB (2018a) algorithm was used to identify, track, and analyse PSM movement in the static chamber (Figure 11). These algorithms are included in the Appendices as MATLAB scripts (Appendix A). Fluorescent videos were loaded into MATLAB using the Bio-Formats software tool (The Open Microscopy Environment, The University of Dundee, Dundee, Scotland) and the timestamp from each frame was extracted from the metadata. A box filter was run over each frame to eliminate background white noise and the image was binarized to identify possible PSM. These PSM were then put through an area filter to ensure that no large aggregates or small noise was considered a PSM. The centroid of each PSM was recorded for each frame. To identify when a PSM was on or off a cell, the fluorescent image of the cell layer was loaded into MATLAB to create a binary mask. A very low threshold is used to ensure the entire cell is identified and then an area filter is used to remove any small background white noise (Figure 12).

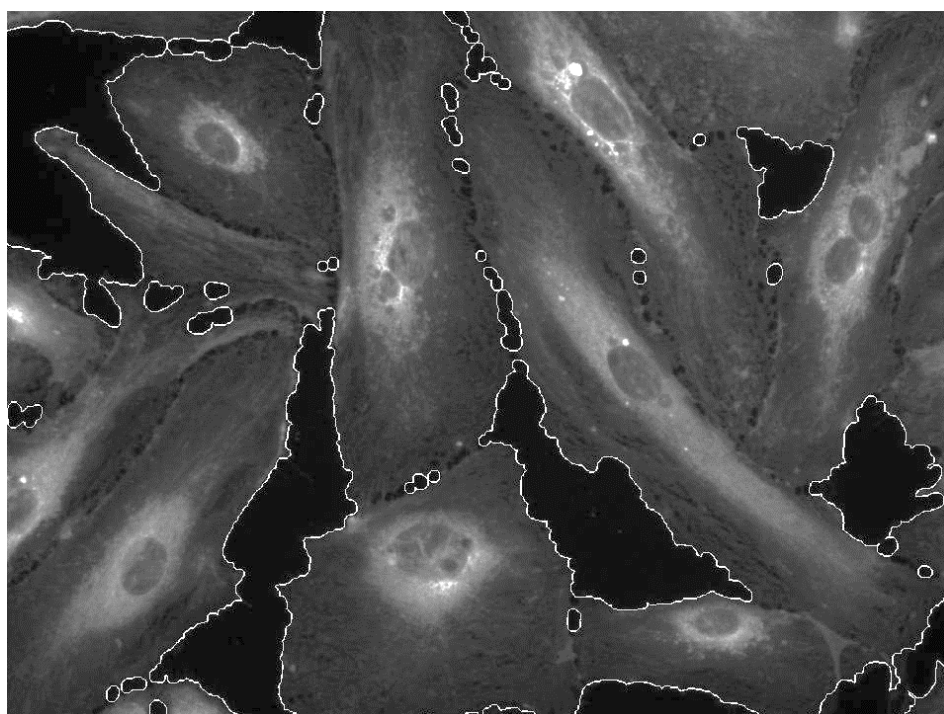


Figure 12. Static chamber HUVEC layer boundary detection.

Cell boundaries are detected using a binary threshold in MATLAB (2018a). The cell is stained with DiO and the boundary of each cell is marked with a white outline.

PSM were tracked through consecutive frames using recursion. Starting with the last frame, the centre of the top-left most PSM was identified. A circle was

visualized on the previous frame, centring on the previously identified centre point of the PSM, with a radius of 4 μm . The closest PSM identified in that circle was characterized as the same PSM and the distance between the two centre points was recorded (Figure 13). This process was then repeated for each previous frame, using the centre location of the PSM in its closest neighbouring frame until there were no more PSM identified within the circle. The program then jumped back to the last frame of the video and looked at the next PSM identified in that frame and repeated the process. Each time a PSM was identified, its location was deleted from the computer memory to ensure it could not be used again. Once all the PSM identified in the last frame of the video had been analysed, the program systematically stepped back a frame and tracked any PSM that had not been previously associated with another PSM. After all the PSM had been tracked, the distance between the first and the last location of each PSM was calculated to differentiate between adhered PSM that are tethered to a point but can still move small distances and random diffusive motion. Only particles that were tracked for at least half of the video length were considered and the total distance from origin was divided by the total number of frames tracked to normalize for particles that were not tracked throughout the entire video.

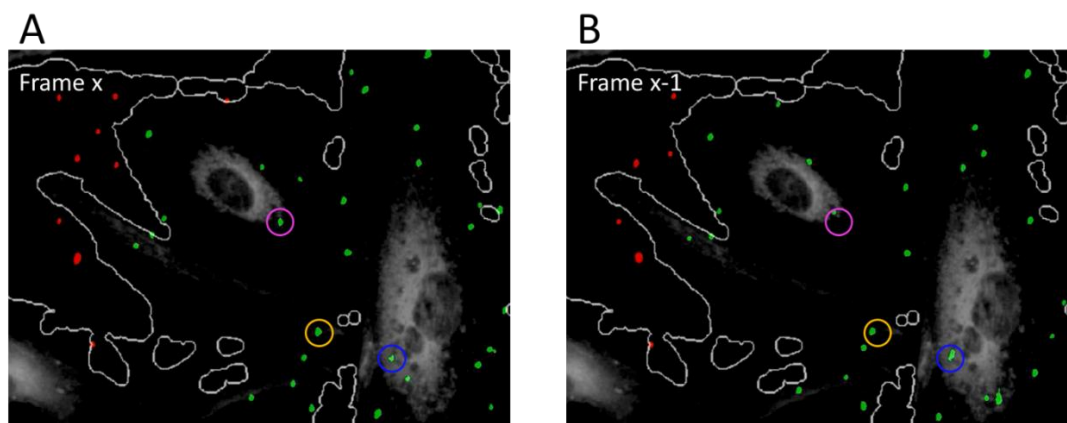


Figure 13. Static chamber PSM identification and tracking.

PSM identified by the software are characterized as on (green) or off (red) a cell. A radius of 4 μm ($\sim 2X$ the PSM diameter) around the centroid of the PSM from frame x (A) is considered in frame $x-1$ (B). If a centroid in frame $x-1$ lies within the circle, the PSM is matched (three examples shown in magenta, gold, and blue).

2.2.9. Flow Chamber Analysis

Videos were separated into 1-minute long segments to decrease memory usage and increase processing speed using the Nikon NIS Elements software (Nikon, Tokyo,

Japan). An algorithm was written in MATLAB (2018a) to identify cells and PSM, track PSM through consecutive frames, evaluate the velocity of the PSM, and identify attached PSM (Figure 14). These algorithms are included in the Appendices as MATLAB scripts (Appendix B). Brightfield images of the base layer of cells were manually checked prior to starting the flow to ensure the cells were 100% confluent (Figure 15). Brightfield videos were loaded into MATLAB via the Bio-Formats software and the timestamp from each frame was extracted from the metadata.

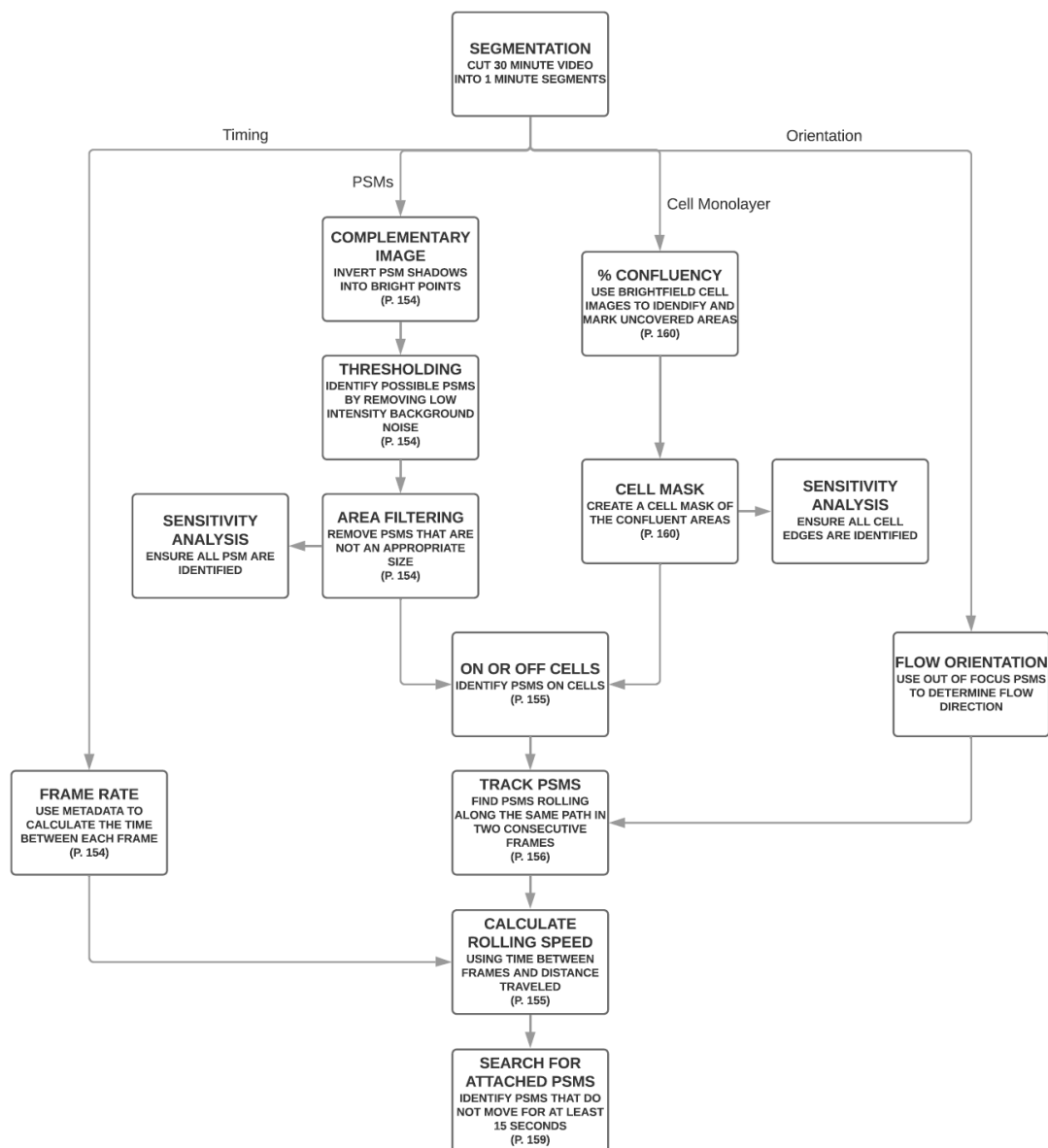


Figure 14. Flow chamber analysis process diagram.

An overview of the data processing algorithm used to filter, identify, and analyse the flow mechanics of the PSM (page numbers refer to the respective piece of code in APPENDIX B).

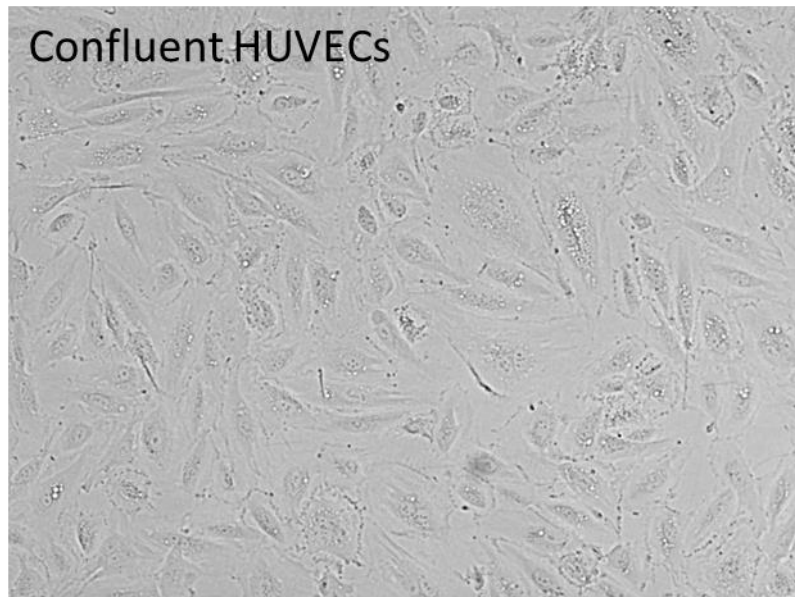
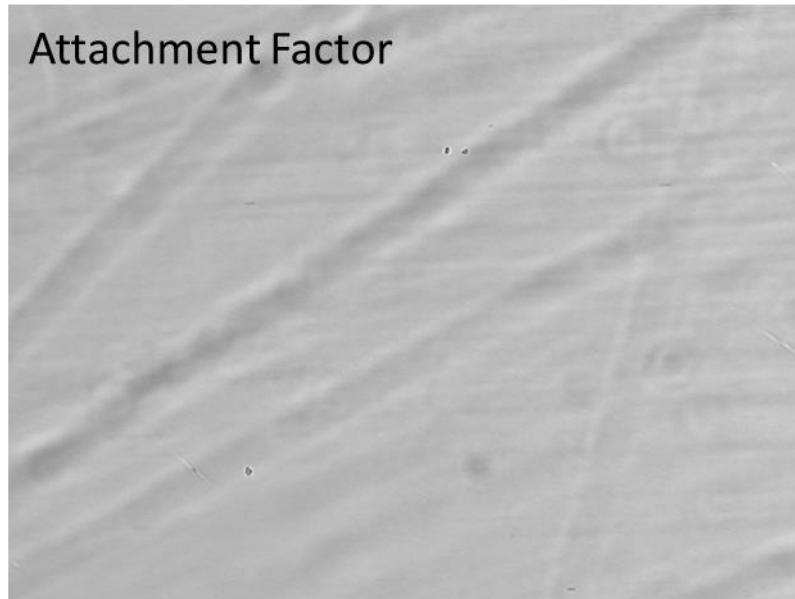


Figure 15. Identifying confluency of the HUVECs in the flow chamber.
Representative image of a flow chamber with a confluent layer of HUVECs.

Each frame of the video was processed individually to identify possible PSM. Due to the PSM blocking the brightfield light from reaching the camera, they create a dark spot on the image (Figure 16A). Therefore, the complementary image was put through a binary threshold with the threshold set 3.5 standard deviations above the mean intensity value of the complementary image. Since the cells are relatively transparent, this is an effective way to identify the opaque PSM. The binary images were filtered according to area, eccentricity, and solidity criteria because the approximate size and shape of a PSM is known (2.6 μm diameter, 0.7 μm tall). The

filtered spots are considered PSM and the centroid position of each is recorded (Figure 16B and C). While the exposure time is the minimum amount of time between frames, the microscope requires variable amounts of time to process one frame before starting to acquire the next. Therefore, a vector of time steps between each frame is calculated using the timestamps from the metadata.

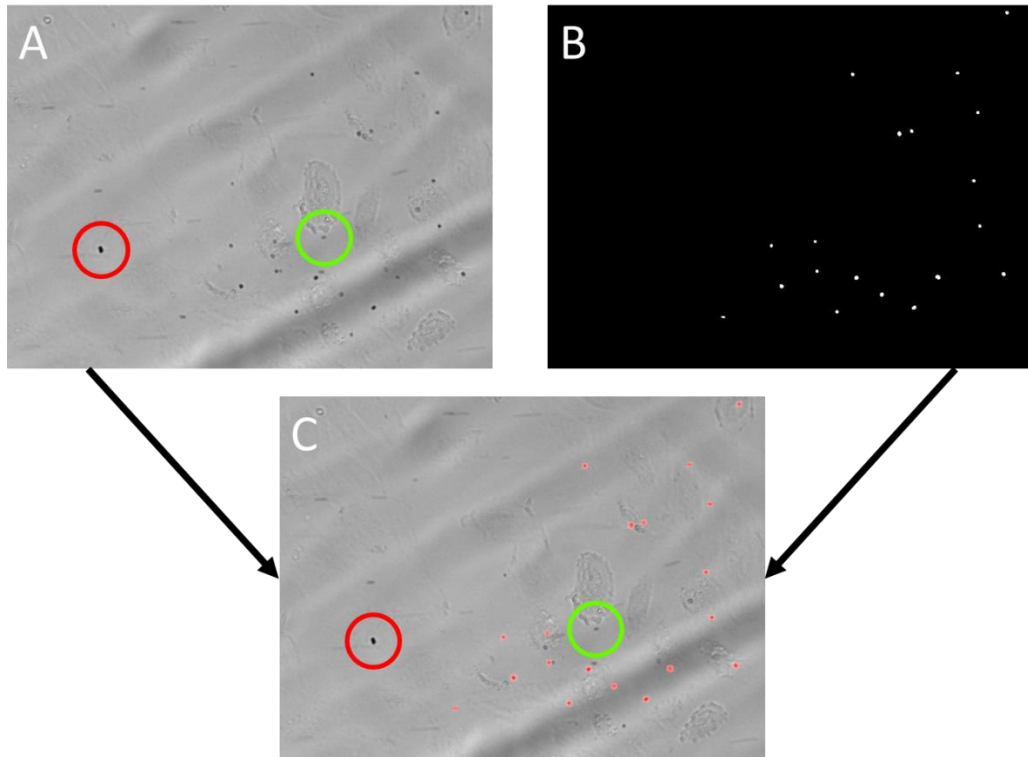


Figure 16. Flow chamber identification of PSM on HUVECs.

A) A single frame of flowing PSM. B) Binary representation identifying PSM. C) Overlay of the identified PSM on the original image. The red circles show an example of an object that was too large to be positively identified as a PSM. This is used to eliminate large conglomerates of PSM or cell debris. The green circles show an example of an object that was not dark enough to be considered a PSM. This is used to exclude cell organelles from being misidentified as PSM.

To identify PSM that had attached to the surface, the filtered binary representation of each frame in the minute-long video were summed together pixel-by-pixel and normalized by the total number of frames. PSM were considered attached if their normalized value was above 0.25, which is the same as 15 seconds of remaining attached. Therefore, when the binary threshold was applied, only PSM that were attached would be recognized and counted. The locations of the attached PSM in each minute-long video are checked against the locations of the attached PSM in the previous video to investigate the attachment and detachment rates. Centroids must be within a 4 μm radius to be counted as the same PSM.

It is nearly impossible to align the circular flow chamber exactly on the horizontal axis of the microscope and so the user is asked to define the orientation of the flow. Due to the 50 ms exposure time, PSM that are not rolling along the surface, and therefore moving much faster, are out of focus and have a long elliptical shape. A single image of the flowing PSM is displayed and the user picks two points along one of the elongated PSM. The line between these two points defines the orientation of the flow within the video. The inlet of the flow was always on the right side of the video and the outlet was on the left.

PSM were tracked through consecutive frames using recursion. Starting with the last frame, the centre of the top left most PSM was identified. A line was visualized on the previous frame, starting at the previously identified centre point of the PSM and tracking back to the right at the angle of the orientation of flow. The closest PSM identified on that line, or within 4 μm of that line, was characterized as the same PSM and the distance between the two centre points was recorded (Figure 17). This process was then repeated for each previous frame, using the centre location of the PSM in its closest neighbouring frame until there were no more PSM identified on that path. The program then jumped back to the last frame of the video and looked at the next PSM identified in that frame and repeated the process. Each time a PSM was identified, it was deleted from the computer memory to ensure it was not counted twice. Once all the PSM identified in the originating frame had been analysed, the program systematically stepped back a frame and tracked any PSM that had not been previously associated with another PSM.

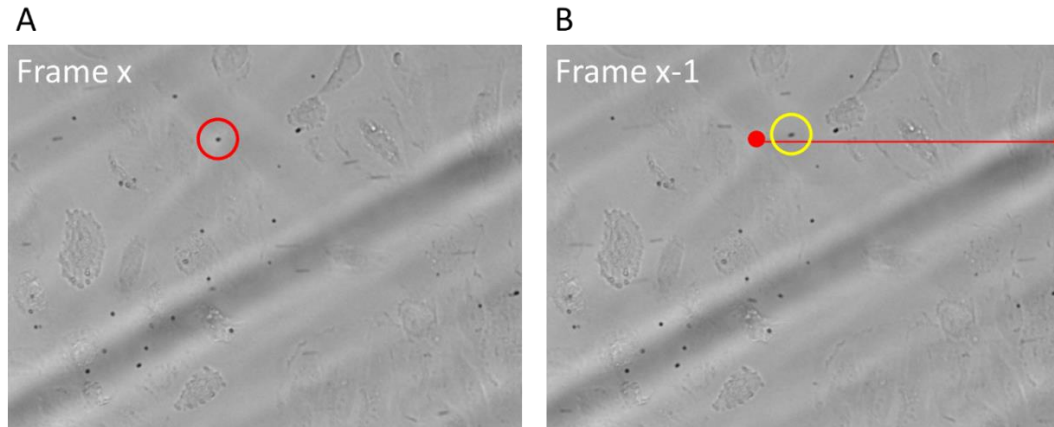


Figure 17. Flow chamber tracking PSM through consecutive frames.

A) The software identifies a PSM (circled in red). B) The software then looks at the previous frame and looks for the closest PSM (circled in yellow) along the flow orientation (red line) radiating from the centroid of the PSM identified in A (red dot).

These distances were then turned into velocities using the time steps calculated from the metadata. Using Equation 9 above, the maximum velocity of the fluid flow within 10 μm of the bottom surface of the flow chamber was calculated to be 417 $\mu\text{m/s}$. The PSM velocities were filtered to exclude anything above 417 $\mu\text{m/s}$ and anything below 3.6 $\mu\text{m/s}$, as these were considered attached PSM. The average velocity of each particle and the average velocity of all PSM in the minute-long video were calculated, as well as their associated standard deviations.

2.2.10. Statistical Analyses

In the static chamber experiments, only one sample was run for each petri dish modification and for each PSM modification. Therefore, no statistical analyses were performed on these experiments. All values displayed are the mean value of all the PSM in that video.

In the flow chamber experiments, of the 30-minute run time, the video with the shortest, continuous, uninterrupted footage was 22 minutes in length. Therefore, all videos were only considered through the 22nd minute. The data from all the PSM in a video were averaged per video and were considered one sample. The sample averages were then averaged together and compared for significance with a 2-tailed, unequal variance Student's t-test out of at most 3 samples. Student's t-tests were performed between the different PSM modifications for each plate surface and between each plate

surface for each PSM modification. Flow videos with less than 250 PSM flowing by in the entire 22-minute window (1/10 of the average count) were not considered in the statistical analysis. This removed one run from each of the following test cases: S-PSM on unstimulated HUVECs, E-PSM on unstimulated HUVECs, A-PSM on attachment factor, E-PSM on attachment factor. Those test cases were left with two independent runs each. All graphs show the sample mean values with error bars representing their standard deviation.

2.3. Results

2.3.1. PSM modification has little effect on Brownian motion step sizes

PSM were mixed with PBS or media and allowed to settle onto an unmodified, AF-modified, or stimulated HUVEC coated petri dish. 1-minute long videos were captured every other minute to observe the effects of Brownian motion on the PSM. Their movement paths were recorded and analysed to detect particle adhesion (Figure 21). On Average, all three types of PSM modifications interacted almost identically with each substrate (Figure 18). Contrary to the hypothesis that the ESTA would become attached to E-selectin expressed on the surface of HUVECs and in turn become stationary as seen in the unmodified plate group (Figure 18A), there was only a minute separation between the step sizes of the E- and S-PSM (Figure 18D). This could be due to a long tether length between connected PSM and the cells. A long tether would allow the E-PSM to continue to engage in Brownian motion without moving too far in any single direction.

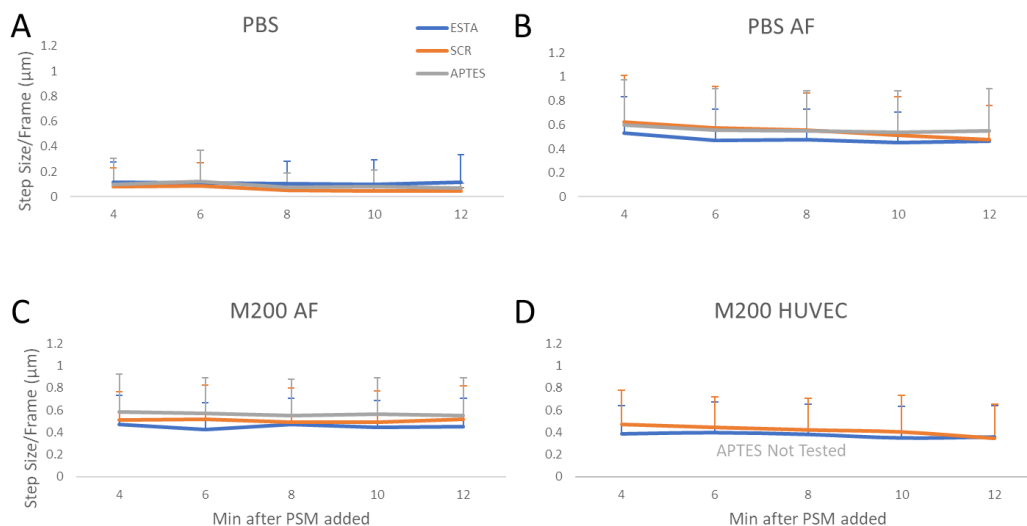


Figure 18. Brownian motion step size comparing PSM modifications on different substrates. Step size of E-, S-, and A-PSM at different time points on an unmodified substrate in PBS (A), on an AF modified substrate in PBS (B), on an AF modified substrate in M200 (C), and on HUVECs in M200 (D). Videos were captured for 1 minute, every other minute for 10 minutes and the average step size was reported for each video.

When the mean square displacement (MSD) is considered for each PSM, the averages continue to appear similar between the different substrate conditions and PSM configurations. While a majority of the PSM are consistently increasing their MSD proportional to time lag, as shown by the consistent, linear average MSD under each condition (Figure 19), there are a few instances of individual PSM reaching an asymptotic limit of how far they can travel over a certain amount of time, such as with APTES PSM on both attachment factor modified substrates and with the SCR and ESTA PSM on the stimulated HUVEC cells (Figure 19). These could indicate that some PSM are becoming entangled or tethered to the surface, therefore limiting their MSD.

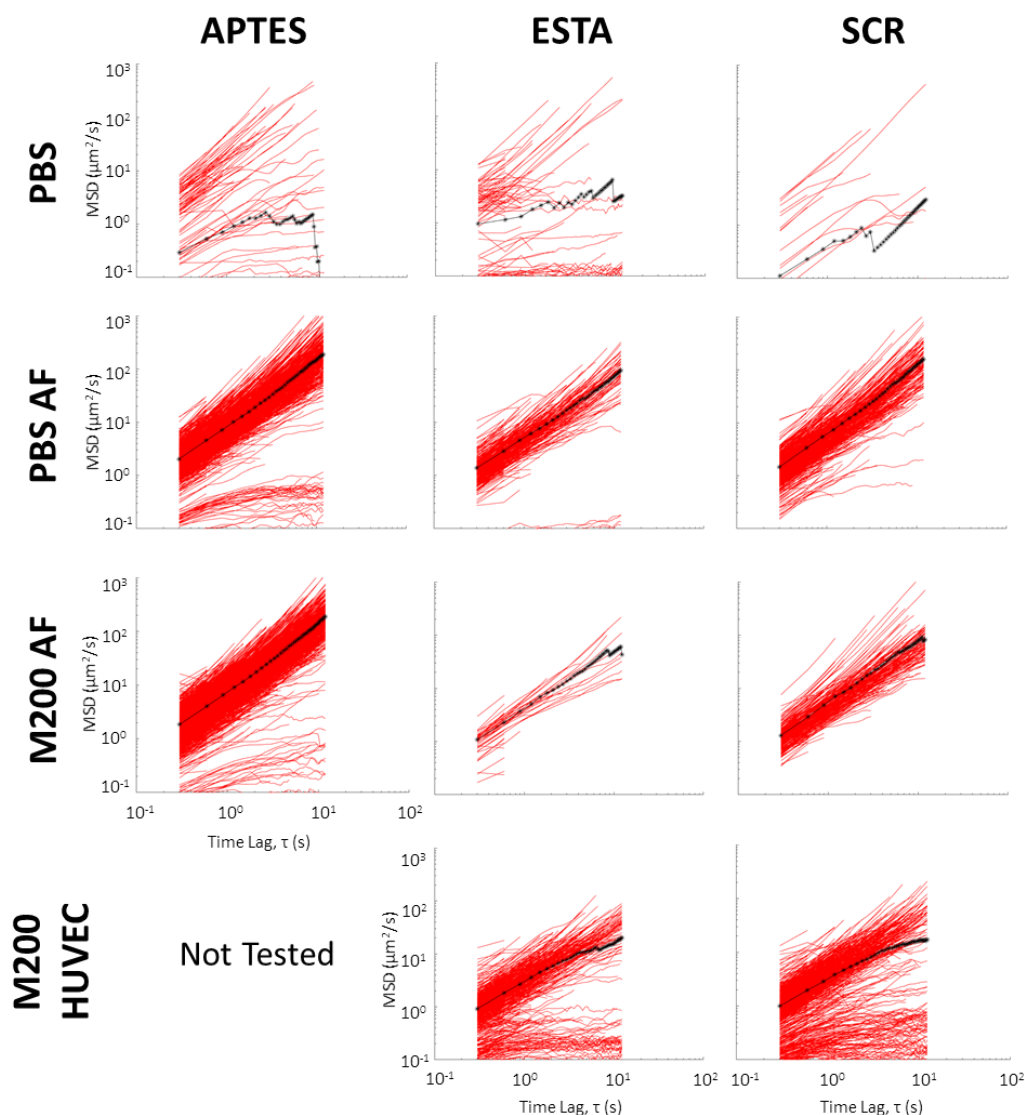


Figure 19. Mean square displacement (MSD) of PSM in static conditions.

Mean square displacements (MSD) of individual A-, S-, and E-PSM in static fluid across the microscope viewing field for each of the different substrates on a petri dish. Each red line represents a different individual PSM. The black line represents the average for each condition. The MSD plots include all recorded PSM from all 1-minute videos under the same conditions in the 10-minute observation window.

2.3.2. The addition of a protein interface increases PSM Brownian motility

When the surface modifications were compared for each PSM modification, there was a stark difference between the unmodified surface and the other surfaces. Interestingly, when PSM were deposited on the unmodified surface, there was little to no PSM movement caused by Brownian motion (Figure 18A, Figure 20, and Figure 21). This is most likely due to the lack of steric stability provided by the unmodified surface. Without a steric hinderance in place, PSM are likely to irreversibly adsorb

onto the surface due to short range electrostatic interactions.¹⁴⁵ However, once the plates were modified in any way, there was an immediate increase in the PSM motion. The jump in the step size increases by a very similar amount for each AF-modified substrate which may indicate that protein adsorption onto the PSM has little effect on their motility. When both the E- and S-PSM were added onto HUVECs, there was slight decrease in motility. This hints that there may be some interaction occurring between the aptamers and the cell surface.

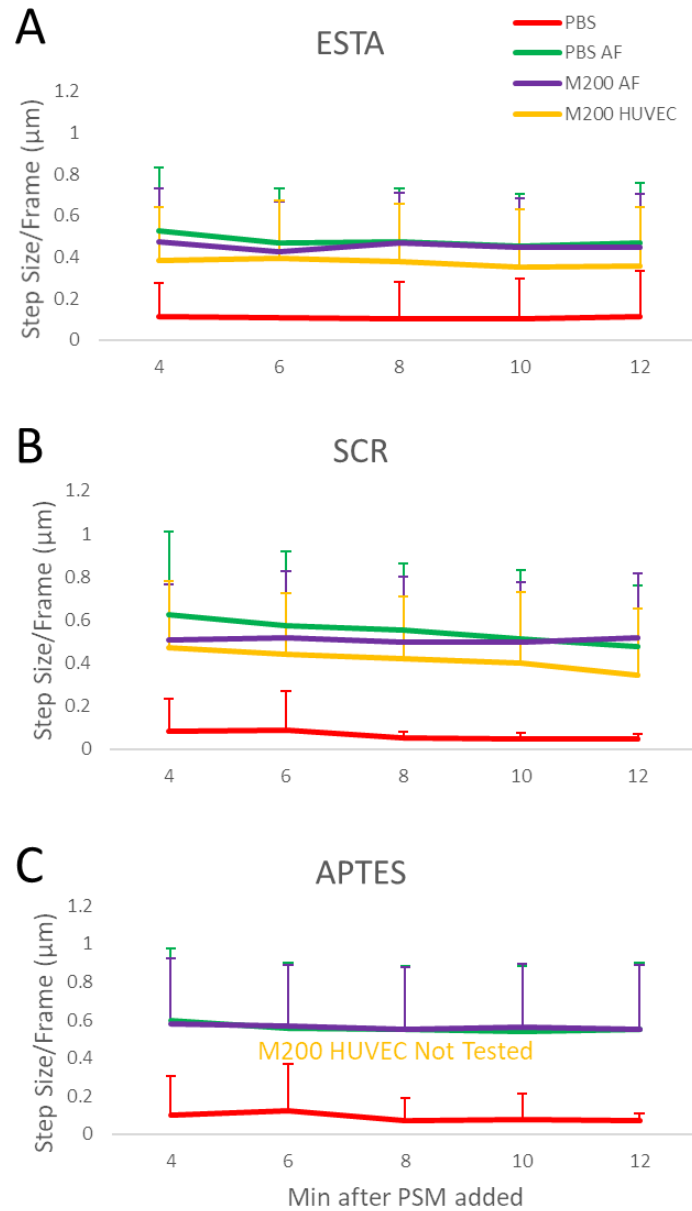


Figure 20. Brownian motion step size comparing different substrates on each PSM modification. Step size of E- (A), S- (B), and A-PSM (C) at different time points on an unmodified substrate in PBS, on an AF modified substrate in PBS, on an AF modified substrate in M200, and on HUVECs in M200. Videos were captured for 1 minute, every other minute for 10 minutes and the average step size was reported for each video.

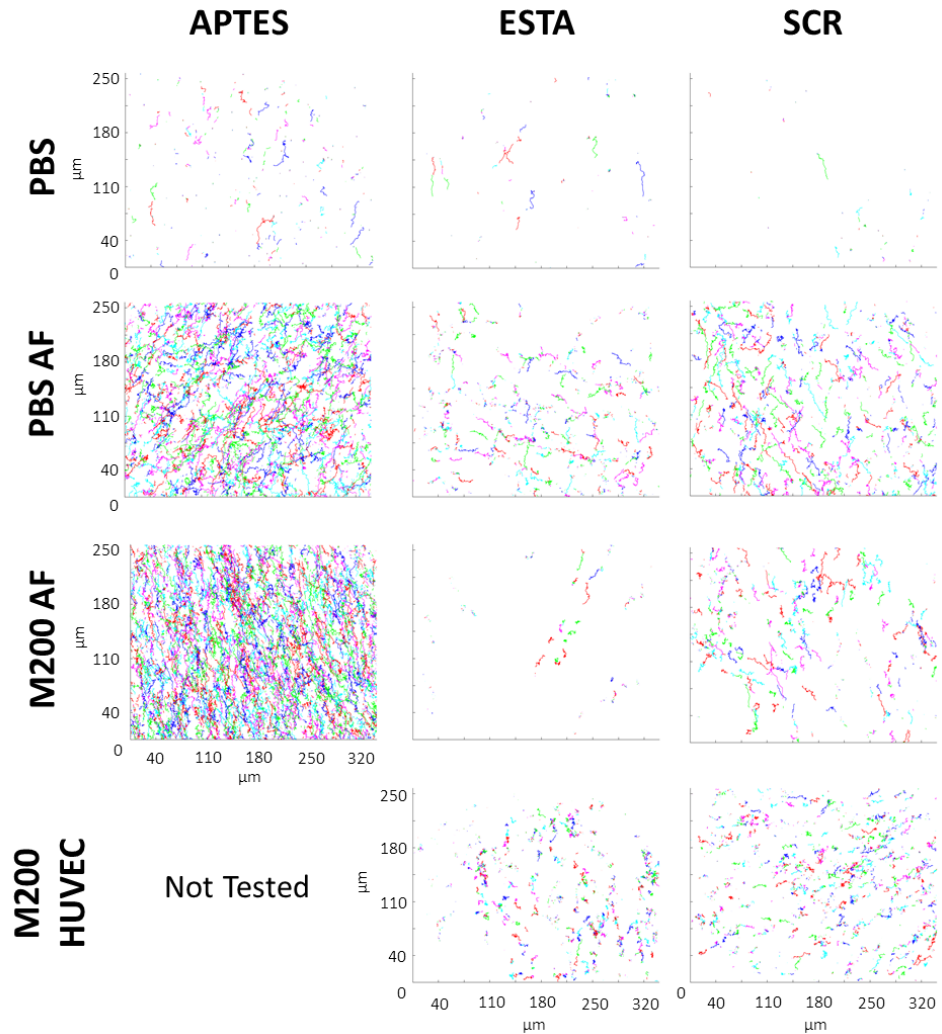


Figure 21. Individual PSM movement paths recorded across the substrates.

Movement paths of individual A-, S-, and E-PSM in static fluid across the microscope viewing field for each of the different substrates on a petri dish. Each colour path represents a different individual PSM. The movement paths include all recorded PSM from all 1-minute videos under the same conditions in the 10-minute observation window.

2.3.3. E-PSM have the overall least displacement

Another way of identifying PSM attachment is through displacement and so the distance between the origin and the final centroid of the PSM in each one-minute video was calculated. These values are normalized by the total number of frames the PSM was tracked through and only PSM that were tracked through at least half of the video were considered. When the different substrates are compared to one another, it can be inferred that every modification adheres to the unmodified substrate (Figure 22A). Once the petri dish is modified with a protein coating or a cell surface, the particles begin to move farther away from their origin. This agrees with the motility

observations from Figure 20 and is possibly due to steric interactions between the proteins on the petri dish and the compounds conjugated to the PSM. There is also a trend in the data indicating that adding serum and other growth supplements to the medium in which the PSM are dispersed will decrease the traveling distance although it did not affect the step size. It is possible that free proteins in the M200 are adsorbing onto the particle surface via ionic or hydrophobic interactions and creating some weak interactions with the different substrates. This adsorption could also create larger particles that would be less effected by the collisions with the fast-moving molecules in the fluid. Yet another trend indicates that the addition of living cells decreases the PSM displacement even more. As there was a similar drop in both the E- and S-PSM, this may be due to the peaks and valleys associated with a 2D cellular surface and the effect of gravity on the PSM. As the PSM settle over the first 4 minutes prior to imaging, they would accumulate in the valleys due to their low potential energy. These valleys, in conjunction with gravity, would restrict the directions of motion available to the PSM, thus restricting its motion.

When the modifications are compared to one another (Figure 22B), it is initially apparent that the E-PSM have the smallest displacement on every substrate. The trend in displacement seems to mimic itself between the E- and S- PSM, decreasing as the system becomes more complex while the A-PSM remain relatively constant. This may indicate that a similar interaction is occurring between the aptamer-modified PSM. An electrostatic attraction between the negatively charged aptamer-modified PSM and the positively charged AF gelatine surface could explain these proportional changes. Although, the decrease in the displacement of the E-PSM compared to the S-PSM on the HUVECs could imply E-PSM-cell interactions. Further studies on the binding kinetics of the ESTA and E-PSM on proteins other than E-selectin would be helpful in clarifying this quandary.

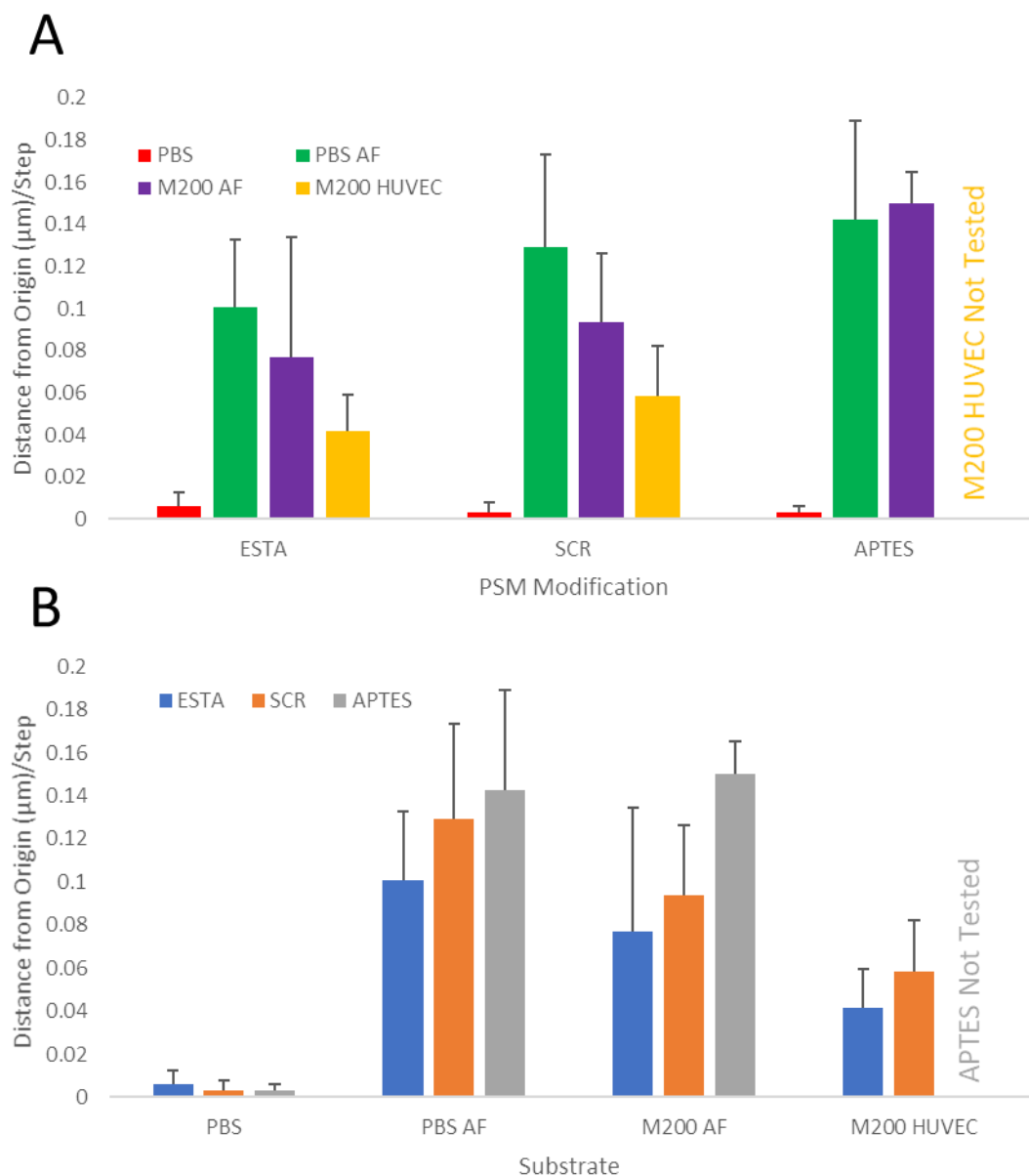


Figure 22. PSM displacement from origin after 1 minute.

A) Comparing the different substrates to one another for each PSM modification. B) Comparing the different PSM modifications to one another on each substrate. The average displacement PSM travelled from their origin at the beginning of the one-minute video is averaged across all 5 videos taken over the 10-minute imaging period. Standard deviation and significance are not reported because the values are only calculated from a single 10-minute imaging session on a single dish.

2.3.4. ESTA and SCR PSM modification reduce rolling velocity on HUVECs

While PSM rolled along the substrate surface, the Nikon microscope captured images at ~20 frames/second. After the software identified and tracked individual PSM through the frames, the average speed of each PSM was ascertained. As was expected

of the E-PSM, the rolling velocity slowed significantly when they were introduced to both the stimulated and unstimulated HUVEC substrate (Figure 23D). While this could indicate that the E-PSM are interacting with the E-selectin expressed on the HUVEC surface, the rolling speed of the S-PSM was slowed by similar amounts (Figure 23C). The SCR aptamer on the S-PSM should not selectively bind to E-selectin like the ESTA does. It is possible that this decrease in speed could simply be due to the uneven surface of the cell monolayer creating obstacles as the PSM roll along, which would also fit with the non-significant but similar decrease in rolling speed in the A-PSM on HUVECs (Figure 23B).

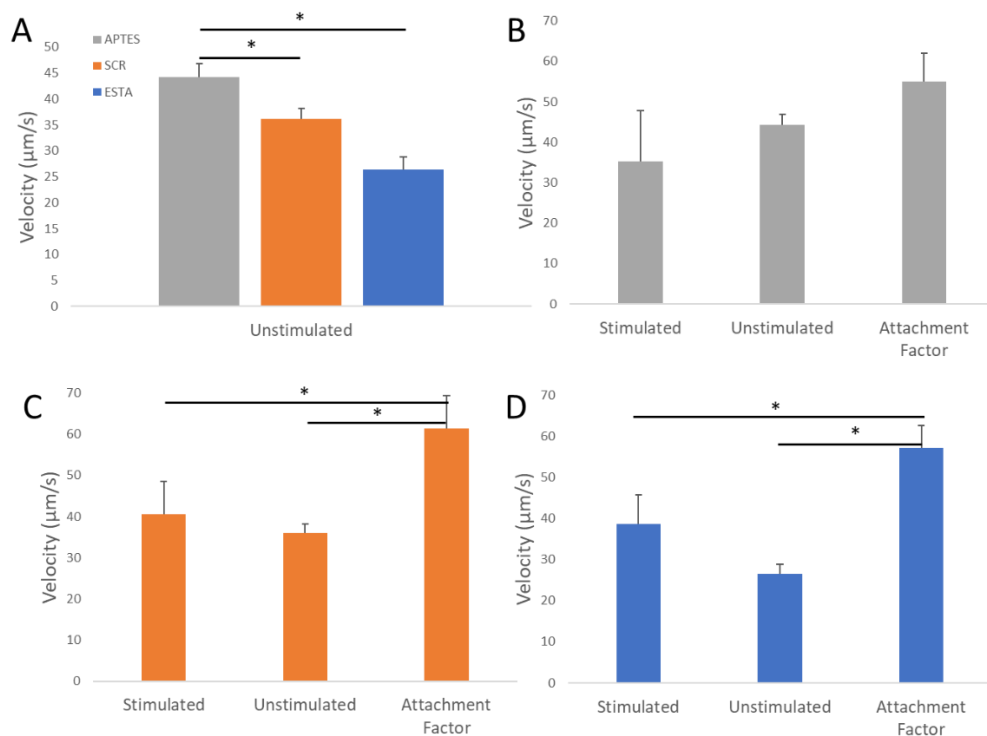


Figure 23. Tracking the rolling velocity of PSM.

A) A-PSM rolling on unstimulated HUVECs roll significantly faster than their aptamer-modified counterparts. B-D) Particle rolling velocities compared on the different substrates. Grey is the A-PSM, orange is the S-PSM, and blue is the E-PSM. * signifies a p value < 0.05 .

2.3.5. The addition of aptamers interferes with cellular interactions

Although the E-PSM slow their rolling velocity on cells as expected, it does not lead to a higher rate of attachment. After 22 minutes, the number of attached A-PSM significantly surpassed their aptamer bound counterparts when flown over the unstimulated HUVECs (Figure 24, Figure 25). This could be due to the aptamers

increasing the particle size or, more likely, the zeta potential of the PSM (Table 2). When the PSM were flown over attachment factor, the number of attached E-PSM was significantly greater than the A-PSM. This supports the idea that zeta potential is influencing the attachment of these particles. The aptamer coated PSM and the cell surface were both negatively charged¹⁷⁰, whereas the attachment factor created a positively charged surface¹⁷¹ and would therefore cause some attraction between the aptamer coated PSM and the surface. The reverse would also be true for the positively charged A-PSM. There was no significant difference between the PSM modification groups on the stimulated HUVECs. It may be possible that by treating the cells with TNF- α , the overexpression of E-selectin and other membrane proteins such as ICAM-1, IL-1, and tissue factor procoagulant¹⁶⁴ are generating a large quantity of steric interference, making it difficult for any particles to adhere or bind.

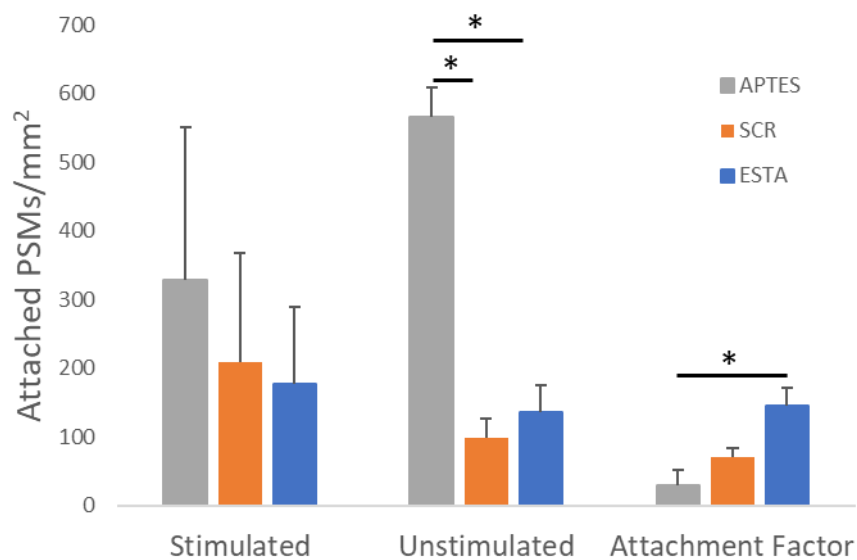


Figure 24. Tracking the number of attached PSM after 22 minutes.

After 22 minutes of flow, the total number of attached PSM/mm² was counted. A-PSM had a significant advantage on unstimulated HUVECs over both the E- and S-PSM. Grey is the A-PSM, orange is the S-PSM, and blue is the E-PSM. * signifies a p value < 0.05.

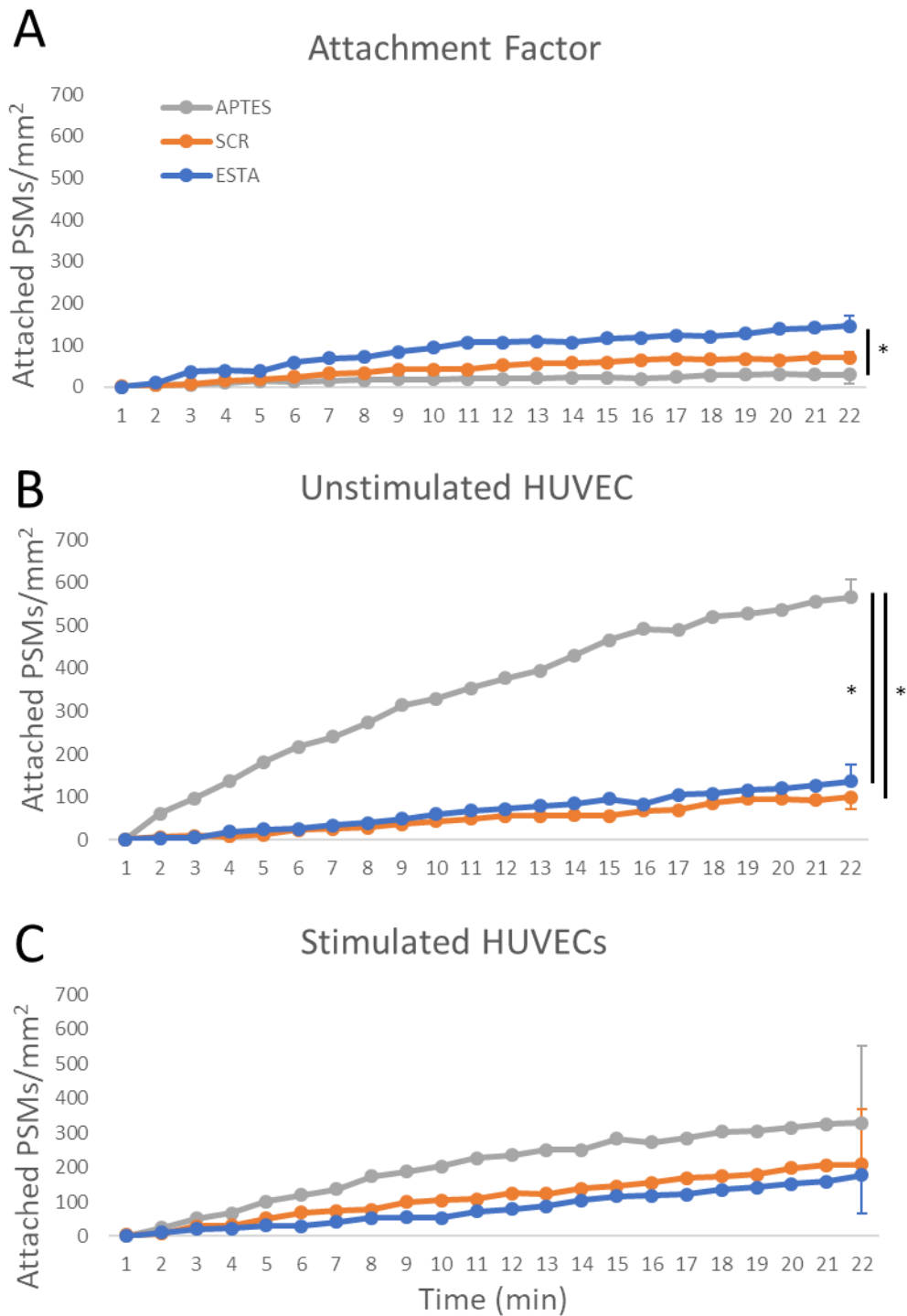


Figure 25. Adhesion kinetics of modified PSM on multiple substrates.

A) The attachment kinetics of the PSM onto attachment factor over 22 minutes. B) The attachment kinetics of the PSM onto unstimulated HUVECs over 22 minutes. C) The attachment kinetics of the PSM onto stimulated HUVECs over 22 minutes. Grey is the A-PSM, orange is the S-PSM, and blue is the E-PSM. * signifies a p value < 0.05.

2.3.6. A-PSM have a high affinity for endothelial cells

A-PSM are flown over the three different substrates at a flow rate of $70 \mu\text{L}/\text{min}$, which creates a shear rate of 174 s^{-1} at the luminal surface. The number of discoidal PSM attached to the surface are counted at the end of each minute-long video. As the PSM continued to pass over the substrate, the number of newly attached particles began to level out. This agrees with the saturation curves reported previously.¹⁵⁰ The number of attached A-PSM increased significantly when HUVECs were the substrate (Figure 26). When averaged over the 22-minute imaging period, the rate of accumulation increased from nearly $1.3 \text{ PSM}/\text{min}/\text{mm}^2$ on the attachment factor up to $25.8 \text{ PSM}/\text{min}/\text{mm}^2$ on the unstimulated HUVECs. This rate of increase is comprised of PSM that slow and arrest their movement as well as those that detach from the substrate and continue in the flow (Table 4). All substrates used the same flow rate and PSM rolling velocities remained unchanged (Figure 23B).

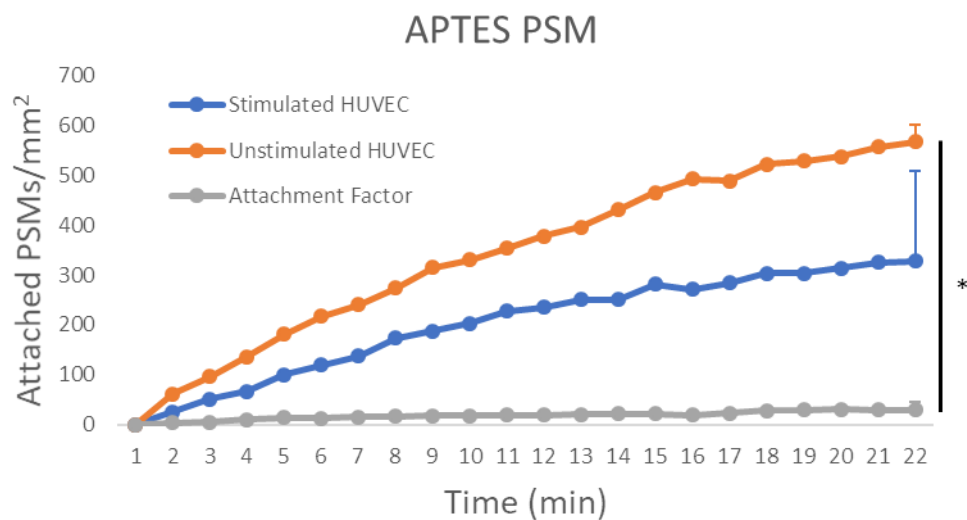


Figure 26. A-PSM attached to different substrates.

A) A-PSM adhere to HUVEC cells at a much higher rate than they attach to the gelatine ECM (AF). Grey is the AF ECM, orange is the unstimulated HUVEC, and blue is the HUVEC stimulated with $20 \text{ ng}/\text{mL}$ $\text{TNF-}\alpha$ to overexpress E-selectin. * signifies a p value < 0.05 .

Table 4. Rate of PSM Attachment and Detachment per square millimetre.

The attachment and detachment rates for each PSM functionalization group were calculated from the videos.

New Attached PSM/min/mm ²				Detached PSM/min/mm ²			
	APTES	SCR	ESTA		APTES	SCR	ESTA
Attachment Factor	3.3 ± 3.3	11.9 ± 6.9	17.5 ± 8.1	Attachment Factor	1.9 ± 2.2	8.1 ± 5.8	8.9 ± 5.9
Unstimulated HUVEC	60.3 ± 15.3	15.0 ± 11.7	16.4 ± 7.8	Unstimulated HUVEC	33.3 ± 18.1	10.6 ± 10.3	9.7 ± 6.1
Stimulated HUVEC	58.1 ± 28.6	32.8 ± 17.8	28.3 ± 21.9	Stimulated HUVEC	42.5 ± 23.9	21.9 ± 16.1	19.2 ± 18.9

2.4. Discussion

Discoidal PSM have been utilized *in vivo* for therapeutic cancer treatment with great success, especially when compared to systemic injections of the same therapeutics.^{140,144,165-166} The switch from spherical particles¹⁶⁷ to discoidal particles follows a line of flow chamber studies, modelling, and improvements to microparticle fabrication techniques. The development of the silicon electrochemical etching technique has allowed for the design and fabrication of bioinert microparticles in unique shapes and dimensions.⁴⁰ The shapes of these microparticles can be crafted to create favourable mechanical conditions for vascular adhesion. Modelling has predicted that the hydrodynamic forces on microparticles cause them to migrate towards and roll along the vessel walls, whereas nanoparticles (<100 nm) are too small for the forces to have the same effect.¹⁶⁸ This difference allows for microparticles to have a greater chance of breaking into the RBC free layer and targeting specific ligand indicators expressed on the vessel walls. Modelling has also projected that by injecting particles with oblong shapes (ellipsoids or, in this case, disks), a larger surface area is presented to the vessel wall for adhesion.¹⁶⁹ This oblong shape presents a larger percentage of the surface area to the cell surface, creating more binding opportunities. Flow chamber studies have demonstrated the advantage of discoidal microparticles over their spherical counterparts in their propensity to move toward the vessel walls in Poiseuille flow, adhere to ECM protein substrates,¹⁵⁰ and established the ability of unmodified spherical microparticles to adhere to the surface of endothelial cells.¹³⁹ While these studies are important precursors to understanding the mechanics around vascular adhesion of discoidal microparticles, they have yet to investigate how the discoidal microparticle interacts with live endothelial cells.

As hypothesized, E-PSM did show a decrease in their rolling speed when compared to A-PSM. Although the initial thought was that this was only due to the ESTA adhering to E-selectin expressed on the HUVEC surface, a similar, yet lesser, decrease in rolling speed was observed in the S-PSM. Therefore, the decrease in rolling speed might also come from another characteristic shared between the two modifications, such as zeta potential or steric hinderance. Decuzzi et al. demonstrated that spherical microparticles attached more preferentially to the ECM as compared to endothelial cells in the same flow chamber at low shear rates.¹⁴⁹ Decuzzi's endothelial cells were grown on a layer of gelatine ECM. It has been reported that cell membranes have a negative zeta potential due to their phospholipid bilayer,¹⁷⁰ gelatine has a positive zeta potential,¹⁷¹ and collagen (used in the current study) has a neutral to slightly positive zeta potential.¹⁷² The grafting density of aptamers onto the PSM in this study was 1.5X greater than that achieved by Mai et al.¹⁴⁴ and consequently, could have increased the magnitude of the negative zeta potential on the PSM. The spherical particles used in the Decuzzi et al. study have a similar negative zeta potential to the aptamer-modified PSM used here. As an increase in E-PSM adhesion to the ECM is also seen in this study, it is possible that there were electrostatic interactions affecting both outcomes. Decuzzi also modelled spherical particle margination to the vessels walls and found that electrostatic forces come into play within tens of nanometres of the cell surface.¹⁷³ As particles must be in contact with the cell surface to adhere, this could be where electrostatic repulsion is interfering with the aptamer-modified PSM's ability to adhere to HUVECs while allowing their adhesion to the AF substrate.

In order to test the effects of zeta-potential, future work should include studies with varying amounts of aptamer attached to the surface of the PSM. By varying the concentration of aptamer on the surface of the PSM, the zeta-potential can be tuned to a value that optimizes the strength of the bond with the ability to reach the bonds. Static studies, similar to the ones shown in this work, could be used to determine the effects of zeta-potential on adhesion to substrates with known electric surface charges. The optimized surface modifications could then be taken into the flow chamber to recreate this experiment to improve the understanding of PSM adhesion kinetics.

The concentration of E-selectin expression is another area where one could continue moving this work forward. Although functional, TNF- α does not provide a

consistent expression profile of E-selectin across the entire cell population, nor between experiments, and also stimulates the cells to also over-express other inflammatory proteins such as P-selectin and ICAM. By utilizing a cell line that selectively and consistently over-expresses E-selectin as the positive control, and a E-selectin knockout cell line as a negative control, the results of the static and flow chamber studies would have far less variability and identifying the causal relationship between the aptamer and its binding ligand would be more concrete.

Although the ESTA has been reported to preferentially adhere to HUVECs in static conditions ¹⁴³ and in vivo experiments have shown the accumulation of PSM in bone marrow metastasis increases with the addition of an ESTA modification, the mechanism is not apparent in these flow chamber experiments. Unfortunately, parallel plate flow chambers lack the ability to mimic the continuous heterogeneity of the tumour microvasculature. ³⁷ Due to an unbalanced overexpression of VEGF, tumour vessels grow at an abnormally fast rate. ³⁵ This leads to malformations in the tumour vasculature such as high tortuosity and permeability. ³² Once PSM reached the RBCFL, conservation of momentum would dictate that PSM flowing around a bend would be more likely to encounter a vessel wall as their momentum tried to maintain its course. With the increase in tumour vessel tortuosity comes more opportunities for a PSM to contact and attach to the endothelium. New 3D printing methods are allowing the development of tortuous flow chambers with known dimensions that can be used in microscopy experiments. ¹⁷⁴⁻¹⁷⁵ It has also been demonstrated by D'Apolito et al that when red blood cells (RBC) are present in the haematocrit, microparticles are bumped into the RBCFL next to the vessel walls and are then trapped there. ¹⁷⁶ This greatly increases the likelihood of PSM-endothelial cell interactions and, because they are on the edge of the Poiseuille flow, slows particles down, increasing the duration of PSM-cell interactions.

One possible future study should include testing the strength of these adhesions. Even though the A-PSM had the largest accumulation on the endothelial cells overall, they also showed the largest number of detachments out of the three groups (Table 4). The rate of detachment for each group was equal at approximately 50% of attached particles that attached to a substrate were subsequently knocked off or released. By running increasing flow rates over attached PSM and watching for

particles becoming detached, an adhesion strength can be calculated. If the ESTA does indeed have a higher adhesion strength to the endothelial cells as hypothesized, this could be a reason for the increased accumulation seen in the in vivo model.

Although A-PSM have a propensity to roll and adhere onto endothelial cell surfaces in physiological conditions, PSM modified with endothelial cell protein ligands have more difficulty adhering. The methodology used here, combined with the custom analysis software, has allowed for particle identification, tracking, and characterization under physiologically relevant flow conditions. This methodology could be easily adapted to track and analyse other types of flow chamber interaction in brightfield or fluorescent microscopy.

3. DENDRITIC CELL VACCINES IN COMBINATION WITH CHECKPOINT BLOCKADE THERAPY LEAD TO AN INCREASE IN INTRATUMOURAL TERTIARY LYMPHOID STRUCTURES

3.1. Introduction

Cancer immunotherapies aim to activate and stimulate the immune system against cancer. They have been applied to patients with great success, improving the survival rates and complete regression rates in patients with metastatic melanoma and renal cancer.¹⁷⁷ There are many factors that impede the ability of immune cells to effectively target and remove solid tumours, including blood vessel structure,³² a nutrient deficient microenvironment,³³ and the recruitment of immunosuppressive cells.³⁴ One treatment that targets immune suppressive mechanisms is the class of immune checkpoint blockade inhibitors, such as anti-PD1 and anti-CTLA4, which interfere with anti-immune signalling presented by cancerous cells.²⁹⁻³¹ Another treatment method that circumvents immune suppression is the adoptive transfer of tumour-antigen loaded dendritic cells (DC) into a patient.¹⁷⁸ The infused DCs can present the loaded tumour-antigens to other DCs and T cells, allowing for the formation of cytotoxic T cells as well as memory immunity, which in turn attack tumour cells. Xia et. al. have developed and previously reported on the use of a microparticle vaccine platform used in adoptive dendritic cell therapies (Figure 27).¹⁷⁹ Here, the combination of checkpoint blockade therapy is investigated with the particle loaded DC vaccine and its ability to stimulate anti-tumour immunity.

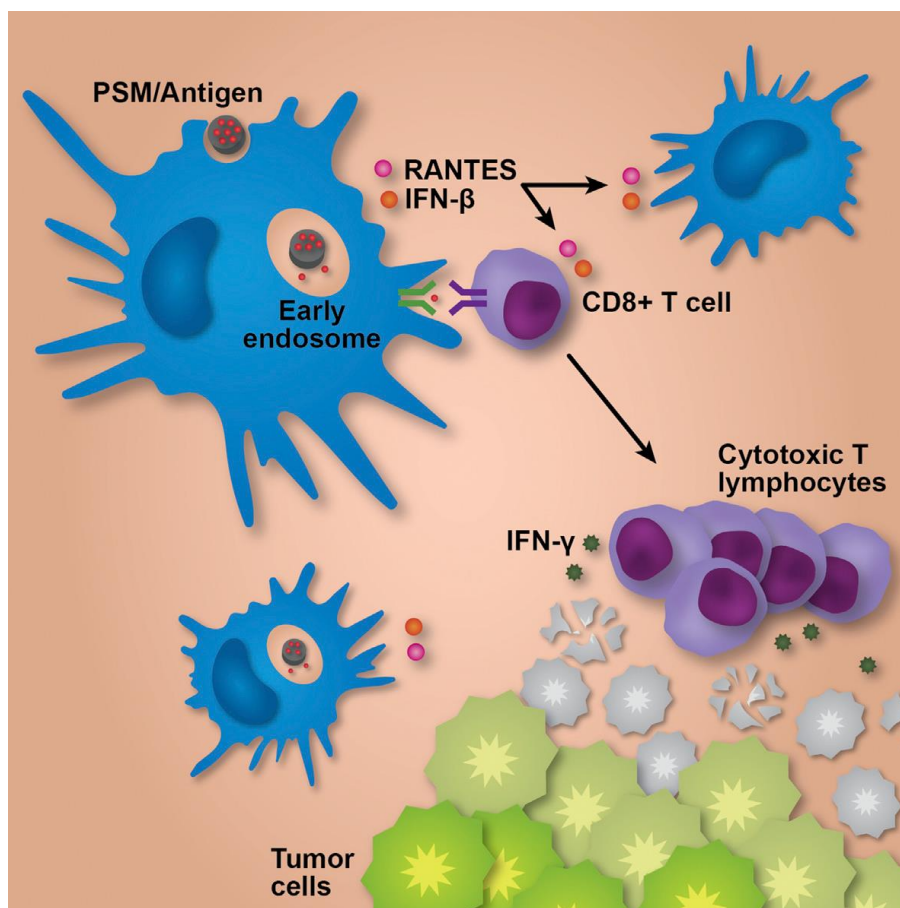


Figure 27. Representative diagram of the DC vaccination.

Antigen-loaded PSM are introduced to DCs in vitro. Activated DCs are intravenously injected into tumour bearing mice where they present the antigens to circulating T cells and DCs from the host. Activated T cells locate and attack tumour cells. Figure used with permission from Elsevier under the terms of the Creative Commons Attribution License (CC BY), <http://creativecommons.org/licenses/by/4.0/> (<https://doi.org/10.1016/j.celrep.2015.04.009>).¹⁷⁹

Discoidal porous silicon microparticles (PSM) are a degradable and biocompatible delivery method for small molecule drugs and other cancer therapeutics.¹⁸⁰⁻¹⁸³ By modifying the surface chemical makeup, the drug release rate of the PSM can be tailored to each patient or disease.^{141,183-184} The discoidal shape (1 μm diameter by 400 nm height) can be efficiently phagocytosed by DCs and sequestered in early endosomes where it can release its cargo for cross-presentation.¹⁷⁹ It has previously been shown that using PSM to deliver cancer antigens to antigen presenting cells (APC) improves the ability of the APC to stimulate anti-tumour immune responses.¹⁷⁹ HER2-loaded PSM enhance the activation and presentation of DCs in vitro which, when injected, enhance the activation of the immune response inside the tumour microenvironment, thus increasing the influx of CD11c+ DCs and tumour-specific cytotoxic T cells into the tumour and improving the patient outcome.

Tertiary lymphoid structures (TLS) are lymphoid structures that form outside of secondary lymphoid organs, such as the lymph nodes, in situations of chronic inflammation and are commonly considered a form of adaptive immunity. These lymphoid structures have been observed at primary tumour sites and represent one of the body's efforts to mount anti-tumour immunity.¹⁸⁵⁻¹⁸⁶ TLS in and around a tumour mass provide an environment in which naïve T cells can be presented tumour antigens by antigen presenting cells in close proximity to tumour cells.¹⁸⁷⁻¹⁸⁸ When TLS are associated with tumours, the prognosis is well correlated to prolonged patient survival¹⁸⁹⁻¹⁹⁰ with few instances of a negative relationship, usually relating to a late stage of the disease.¹⁹¹⁻¹⁹² In one study of breast cancer, TLS were associated with a lower risk of relapse and longer overall disease-free survival rates.¹⁹³⁻¹⁹⁴ Acting as a local immune hub, TLS reduce the distance that dendritic cells (DC) and other APCs must travel to coordinate the adaptive immune response.¹⁸⁵ In cancer, these structures are commonly found in the peritumoural space, where they resemble lymph nodes, but have also been seen and studied intratumorally.¹⁹⁵ Currently, there is not a gold standard for how to identify a TLS, but some identification methods include looking for clusters of B cells,^{193,196} follicular helper T cells,¹⁹⁷⁻¹⁹⁸ high endothelial venules (HEV),¹⁹⁴⁻¹⁹³ mature DCs,¹⁹⁸⁻²⁰¹ or any combination of these cell clusters (Figure 28). It was decided to look at immune cell clusters consisting of B cells, T cells, and HEVs.

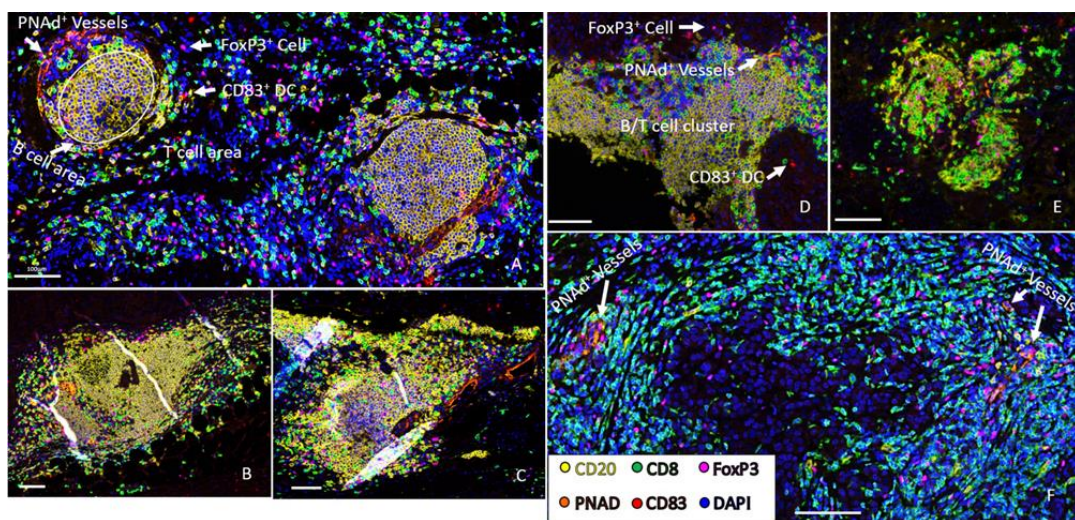


Figure 28. Tumour associated TLS.

Multispectral imaging of TLS in human melanoma metastases localized outside the tumour area (A), peritumourally at the peripheral edge of tumour (B–D), or intratumorally (E and F). Artefactual tissue creases due to sectioning can be seen in (B) and (C). Scale bars, 100 µm. Figure used with permission from *The Journal of Immunology*.¹⁹⁵

Cellular makeup, protein expression, and gene activation are all important indicators of cell process mechanisms, but a 3D analysis of the cellular environment is often discounted due to depth-of-imaging difficulties. One of the physical reasons that organic tissues are opaque and difficult to image is due to the variability between the refractive indices of the many structures that make up the tissue (e.g. extracellular fluid, lipids, cellular organelles, etc.). Clearing methods can be used to reduce the differences between refractive indices, causing the tissue to become optically transparent and allow for deep tissue imaging.²⁰² An optical tissue clearing method called iDISCO, along with improved staining and microscopy techniques, has been used to study the heterogeneity of intratumoural cellular diversity and vessel complexity.³⁷ The iDISCO and other tissue clearing protocols²⁰³ rely on reducing the change in refractive indices throughout the tissue. Each method has its own standardized protocol, but the basis of each requires completely dehydrating the tissue, thereby removing the extracellular fluid and free-floating lipids from the tissue, and then rehydrating the sample with a fluid that has a similar index of refraction as the ECM and cellular structures. Staining and imaging in 3D allows for complex analysis of entire organs of interest, up to 2 mm deep.²⁰⁴ As in the case of lymphoid tissues, the structure of the cellular environment is important for cellular communication and information transfer. In this work, iDISCO optical clearing was used to study the intratumoural immune response to therapeutic treatments and subsequent TLS formations.

3.2. Materials and Methods

Dr. Qing Shi, a Postdoctoral Fellow in Haifa Shen's lab at Houston Methodist Research Institute, Department of Nanomedicine, performed all the live animal work. His methods are reported below to give a complete report of this experiment. All sections that were performed by Dr. Qing Shi are marked as such.

3.2.1. Mice

The live mice were maintained by Dr. Qing Shi. Female 6–10-week-old Balb/c mice from Charles River were used in the tumour studies. All mice were maintained in a specific pathogen-free environment as part of the comparative medicine program at Houston Methodist Research Institute. All animal experiments were approved by

the Houston Methodist Research Institute IACUC and conducted in accordance with institutional guidelines.

3.2.2. Cell lines

The TUBO breast cancer cells were maintained by Dr. Qing Shi. TUBO cells were provided by Drs. Wei-Zen Wei (Wayne State University, Detroit, MI) and Guido Forni (University of Torin, Italy). Cells were maintained in 37°C 5% CO₂ incubators in DMEM media supplemented with foetal bovine serum (FBS) and Penicillin-Streptomycin (PS) antibiotics. TUBO is considered to be a nonmetastatic, cloned cell line established from a mammary carcinoma of the Her2-neu transgenic mice also on the Balb/c background.²⁰⁵

3.2.3. Preparation of Antigen-Encapsulated Liposomes and Loading into PSM

The treatment was prepared by Dr. Qing Shi. Liposomes were created as previously described.^{141,183} Briefly, three solutions were prepared: HER2 peptide antigen in H₂O, 1,2-dioleoyl-sn-glycero-3-phosphocholine (DOPC) in t-butanol at 20 mg/ml, and 0.1% Tween-20 in H₂O. All three solutions were then vortexed together in t-butanol for 1 minute. Samples were then freeze dried using a lyophilizer (LD85, MillRock). To load the liposomes into the PSM (1 µm in diameter and 400 nm high), the liposome powder was suspended in 50 µl of H₂O, followed by a brief sonication, and then the resulting suspension was added to dry PSM particles, followed by a brief sonication.

3.2.4. Bone Marrow Derived Dendritic Cells (BMDC)

The BMDCs were collected by Dr. Qing Shi. 6–10-week-old female Balb/c mice were used to isolate BMDCs.²⁰⁶ Briefly, bone marrow cells were collected from mouse femurs and tibias by flushing the bones with 1% FBS-containing PBS. Red blood cells were lysed with ACK lysis buffer (Lonza Inc.) and then resuspended in RPMI 1640 medium with 10% FBS, antibiotics, and 55 µM β-mercaptoethanol. The marrow cells were then cultured over a 7-day period with 20 ng/mL recombinant murine granulocyte-macrophage colony-stimulating factor (GM-CSF, Peprotech) and 20 ng/mL IL-4 (Peprotech). Cell culture medium was refreshed every other day. Non-

adherent cells were harvested as immature BMDCs. For in vivo studies, immature BMDCs were stimulated with 100 ng/ml LPS overnight and then primed with the PSM/HER2 antigen for 3 hours before intravenous (i.v.) injection into mice.

3.2.5. Live animal treatment protocol

Dr. Qing Shi inoculated and treated the live mice. TUBO cells (1×10^6 cells/mouse) suspended in 4°C Matrigel/PBS (1:3) were implanted into the mammary gland fat pad of 12 female Balb/c mice (6–10 weeks old). On day 7 after tumour inoculation, the mice were divided into groups (3 mice per group) and mice in the vaccine groups were given their first i.v. injections of the BMDC vaccine treatment (1×10^6 cells/mouse). On day 11 after tumour inoculation, an additional 3 naïve female Balb/c mice were inoculated with TUBO cells (1×10^6 cells/mouse) in the mammary gland fat pad to act as a small volume, early stage tumour control; these mice were not treated in any other way. On day 14 after tumour inoculation, mice in the vaccine groups were treated with another dose of the BMDC vaccine. 200 µg Rat IgG2a Isotype control (2A3, Bio X Cell) or anti-PD1 (RMP1-14, Bio X Cell) were administered through an intraperitoneal (i.p.) injection on day 11, day 14, and day 19 after tumour inoculation to all treatment groups. Tumour volume was recorded as $\frac{width \times length^2}{2}$. All mice were sacrificed on day 22 after tumour inoculation. Excised tumour samples were then cut into two halves, one half was used for flow cytometry while the other was used for tissue clearing and imaging.

3.2.6. Flow Cytometry

The flow cytometry of the tumour immune cell microenvironment was performed by Dr. Qing Shi. The first half of the tumour was used to obtain a single cell suspension digestion with 275 U/ml of type IV collagenase (Worthington Biochemical Corporation) and 375 U/ml DNase I (Applichem) at 37°C for 1 hr with 200 RPM shaking. Cells were stained with 2 µg/mL of CD45 and CD3 (BioLegend, San Diego, CA, USA) at 4°C for 30 mins. Flow cytometry was conducted on a LSRII (BD Falcon) and analysed by FlowJo software (TreeStar).

3.2.7. Sample Preparation

The second half of each tumour sample was cut into two approximately 2x2x2 mm cubes, one from the core and the other from the exterior 2 mm of the tumour sample (Figure 29). This size was chosen based on the largest working distance of the 10X objective on the confocal microscope (1.2mm). These cubes were then pre-treated for immunolabelling and clearing protocols according to the iDISCO protocol developed by Renier et al.²⁰⁴ Briefly, samples were fixed overnight in 4% paraformaldehyde, followed by two washes in 1X PBS. The tissue samples were then dehydrated with an increasing series of methanol/H₂O solutions (20%/40%/60%/80%/100%/100% methanol) and left overnight in 33% methanol/66% dichloromethane (DCM). After two washes with 100% methanol, the samples were bleached overnight in 5% H₂O₂ to remove pigmentation from the tissue and reduce autofluorescence. Post-bleaching, samples were rehydrated using a decreasing series of methanol/H₂O solutions (80%/60%/40%/20% methanol) and then fully rehydrated in PBS. Samples were washed twice in PBS/0.2% Triton-X prior to immunolabelling.

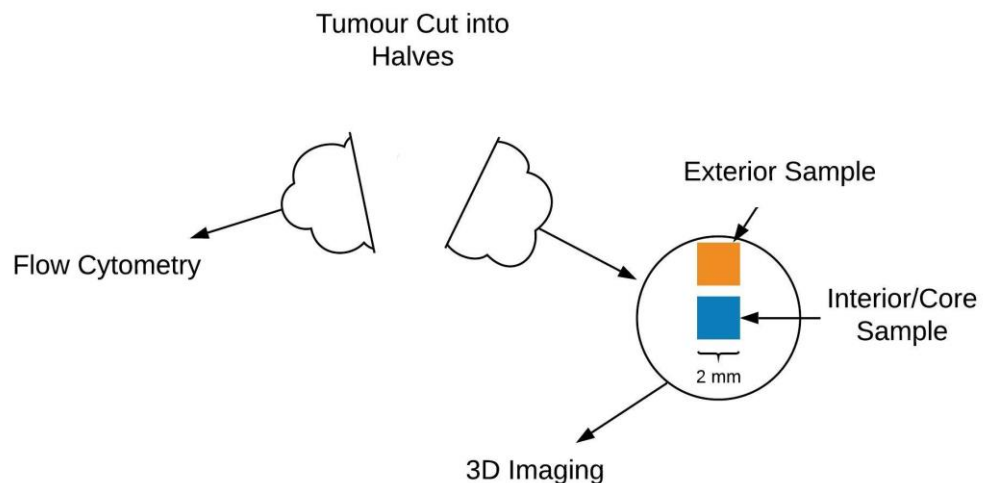


Figure 29. Tumour preparation.

After excision, tumour samples were cut in half. One half was used for flow cytometry, the other half was cut into two 2x2x2 mm cubes of tissue. One cube was taken from the interior, or core, of the sample and one from the exterior. These samples were processed and imaged in 3D on a confocal microscope.

3.2.8. Whole Tissue Immunolabelling

Pre-treated tissues were incubated in a permeabilization solution of PBS, 0.2% Triton X-100, 20% DMSO, and 0.3M glycine at 37 °C for 2 days and then with a blocking solution of PBS, 0.2% Triton X-100, 10% DMSO, and 6% FBS at 37 °C for 2 days. The samples were incubated with primary antibodies in PBS and 0.2% Tween-20 containing 10 mg ml⁻¹ heparin (PTwH), 5% DMSO, and 3% FBS at 37 °C for 4 days. The primary antibodies used in this study were anti-CD3 (1/200, rabbit polyclonal, ab5690, Abcam, Cambridge, UK), anti-CD45R (B220, 1/100, rat monoclonal, AB64100, Abcam, Cambridge, UK), and anti-MECA-79 (PNAd, 1/20, rat monoclonal, SC19602, Santa Cruz Biotechnology, Inc., Dallas, TX, USA). The samples were then washed in PTwH for 1 day and then incubated with secondary antibodies in PTwH and 3% FBS for 4 days at 37 °C. The secondary antibodies used in this study were Alexa Fluor (AF) 647 conjugated goat anti-rabbit IgG (H+L) cross-adsorbed (1/200, A-21244, ThermoFisher Scientific, Waltham, MA, USA), AF 555 conjugated donkey anti-rat IgG (h+L) preadsorbed (1/200, ab150154, Abcam, Cambridge, UK), and AF 488 conjugated goat anti-rat IgM (heavy chain) cross-adsorbed (1/100, A-21212, ThermoFisher Scientific, Waltham, MA, USA). All antibodies and their targets are shown in Table 5. Due to two of the primary antibodies having the same host species, this process was repeated for those 2 antibodies separately with 1 day of washing in PTwH in between. After incubation with the secondary antibodies, the samples were washed in PTwH for 1 day before beginning the tissue clearing process.

Table 5. Antibodies used for staining tumour samples.

A list of the antibodies used to stain the tumour samples for T-cells, B-cells, and HEV.

Target Protein	Primary Antibody (Host)	Secondary Antibody (Host)	Fluorescent Label on Secondary Antibody
T-Cells	Anti-CD3 (Rabbit)	Anti-rabbit IgG (Goat)	AF 647
B-Cells	Anti-CD45R (Rat)	Anti-Rat IgG (Donkey)	AF 555
High Endothelial Venules	Anti-MECA-79 (Rat)	Anti-Rat IgM (Goat)	AF 488

3.2.9. Tissue Clearing

The tissue clearing method was adapted from the iDISCO clearing method developed by Renier et. al.²⁰⁴ Briefly, the samples were dehydrated with an increasing series of methanol/H₂O solutions (20%/40%/60%/80%/100%/100% methanol) and then incubated with 66% DCM / 33% methanol for 3 hours. This was followed by 2, 15-minute washes with 100% DCM to remove any leftover methanol and completely dehydrate the sample. The samples were then incubated and stored in Dibenzyl Ether (DBE, $n = 1.562$), a fluid with a refractive index equivalent to the refractive index of the cell walls, until imaged.

3.2.10. Confocal Microscopy

For the cleared tissues to remain clear during imaging, they were kept in DBE throughout the imaging process. An imaging chamber was constructed out of two glass coverslips and a silicon gasket (FlexiPERM, Sarstedt, Nümbrecht, Germany) held together with a silicon glue (Kwik-Sil, World Precision Instruments, Sarasota, FL, USA). The wells were then filled with DBE and the tumour samples were placed inside before sealing the second coverslip on top (Figure 30). A Nikon A1R Confocal Laser Microscope System with a 10X objective (Nikon S Fluor, NA of 0.5, MRF00100, working distance of 1.2 mm) and a Nikon A1 Plus PMT were used to take Z stack images at 10 μm steps in the Z-direction. This creates an undersampled Z-dimension which was selected due to the large size of the tissues and time constraints with the institutionally shared confocal microscope. Although it is undersampled, the axial resolution ($\sim 5.5 \mu\text{m}$, calculated using Equation 2 and the size of the cells of interest ($\sim 13 \mu\text{m}$ in diameter) should both be large enough to be encompassed cells within at least two Z-planes. XY plane images were tiled and stitched to create larger fields of view with a resolution of 1.05 $\mu\text{m}/\text{pixel}$ (Figure 35A,C). Unfortunately, as the microscope images deeper into the sample, the amount of fluorescent signal returning to the sample is substantially lessened because of spherical aberrations. Spherical aberrations also decrease the clarity of the images due to the changes in the index of refraction. When light passes through an index of refraction change, the light refracts at a new angle that is dependent on its incident angle. This spreads out the focal point which accounts for a loss of fluorescent intensity and resolution. To account for this,

the objective used in this study was corrected to account for spherical aberration with the 0.17mm coverslip used in this study. Also, the tumour tissue is immersed in DCM, which has the same index of refraction as the cell walls and glass coverslip, therefore decreasing the changes in the index of refraction that the light must pass through on its way to the objective.

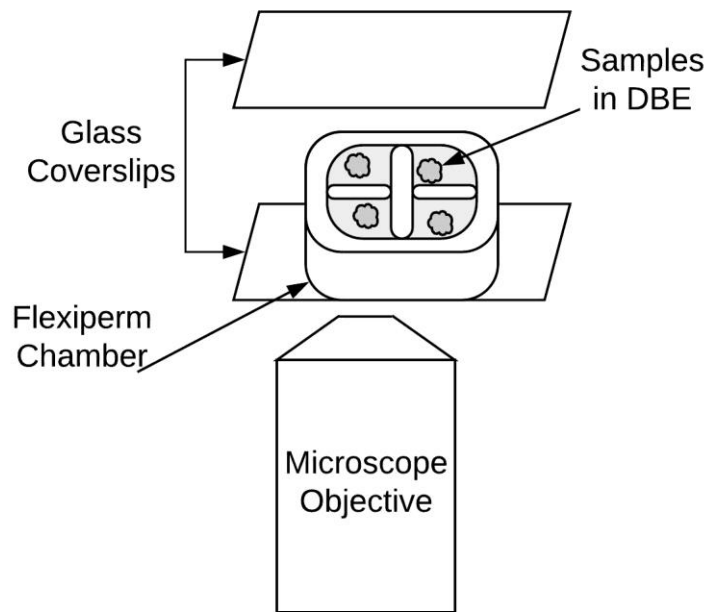


Figure 30. Setup for 3D imaging on the confocal microscope.

A FlexiPERM silicone chamber was sealed between two glass coverslips. The wells were filled with DBE and a single tissue sample was placed in each well. Z stacks were acquired from the bottom of the tumour sample to top.

3.2.11. Imaging Analysis

After the confocal images were acquired, an analysis software was built to determine the amount of TLS present inside of the tumour space and investigate the mechanisms of the vaccine's success. The software was first used to filter the images and identify the stained immune cells. The locations of those cells were then compared to the volume surrounding them to identify areas that were densely populated with immune cells. If a $100 \times 100 \times 100 \mu\text{m}^3$ volume contained at least 10 B cells, 20 T cells, and 50 μm worth of HEV, then it was considered a part of a TLS. The software then searched for locations where these volumes overlapped and grouped them into a single

TLS. TLS inside of the tumour space indicate a strong immune response to the cancer that could be a mechanism driving the effectiveness of the cancer vaccine.

In more detail, images were loaded into MATLAB R2018a via the Bio-Formats software tool (The Open Microscopy Environment, The University of Dundee, Dundee, Scotland) and one algorithm was developed to identify cells and vessels (Figure 35B and D), while a second algorithm was developed to find areas with diverse cell clustering (Figure 32). These algorithms are included in the Appendices as MATLAB scripts (Appendix C). The AF 488 and AF 555 channels (Table 5) were both seen to have extraneous autofluorescence from the extracellular matrix surrounding them. This autofluorescence showed the same structures in both channels so a compensation constant multiplier (C) was identified for each plane that minimized the absolute value of the difference between the two images. In order to identify the compensation constant multiplier, a range of constants was checked from 0 to 5 with a step size of 0.1. These constants were multiplied by the AF 488 image and subtracted from the AF 555 image and the absolute value was taken. These absolute values were compared to see which constant delivered the smallest result. The reciprocal of that value was used to multiply AF 555 image and subtract it from the AF 488 image. This was repeated for each new Z plane. The AF 488 and AF 555 channels (Table 5) were multiplied by their respective constant and subtracted from one another, pixel by pixel, to minimize the effects of autofluorescence (Figure 31). Any values that were less than zero after the subtraction were set to zero. This eliminated a large proportion of the extraneous and invalid information. The AF 647 channel (Table 5) did not have any obvious autofluorescence and so was not filtered in this way. A band pass filter was run over each channel to remove small, high frequency noise and broad, low frequency noise while maintaining cell-sized object prominence. A binary threshold was then applied to AF 488 image planes to create a mask of possible positive areas. If there were any positive areas in the binary mask that were smaller than a single cell or areas that exceeded the area of ten connected cells, these areas were set to zero to eliminate noise and large artefacts. Once all the planes had a representative binary mask, they were stacked into a 3D matrix and skeletonized to create a representation of the vasculature as a line with single-pixel width. This was accomplished by removing

positive pixels that were on the edge of a binary region, one layer at a time, until only a single line of pixels remained.

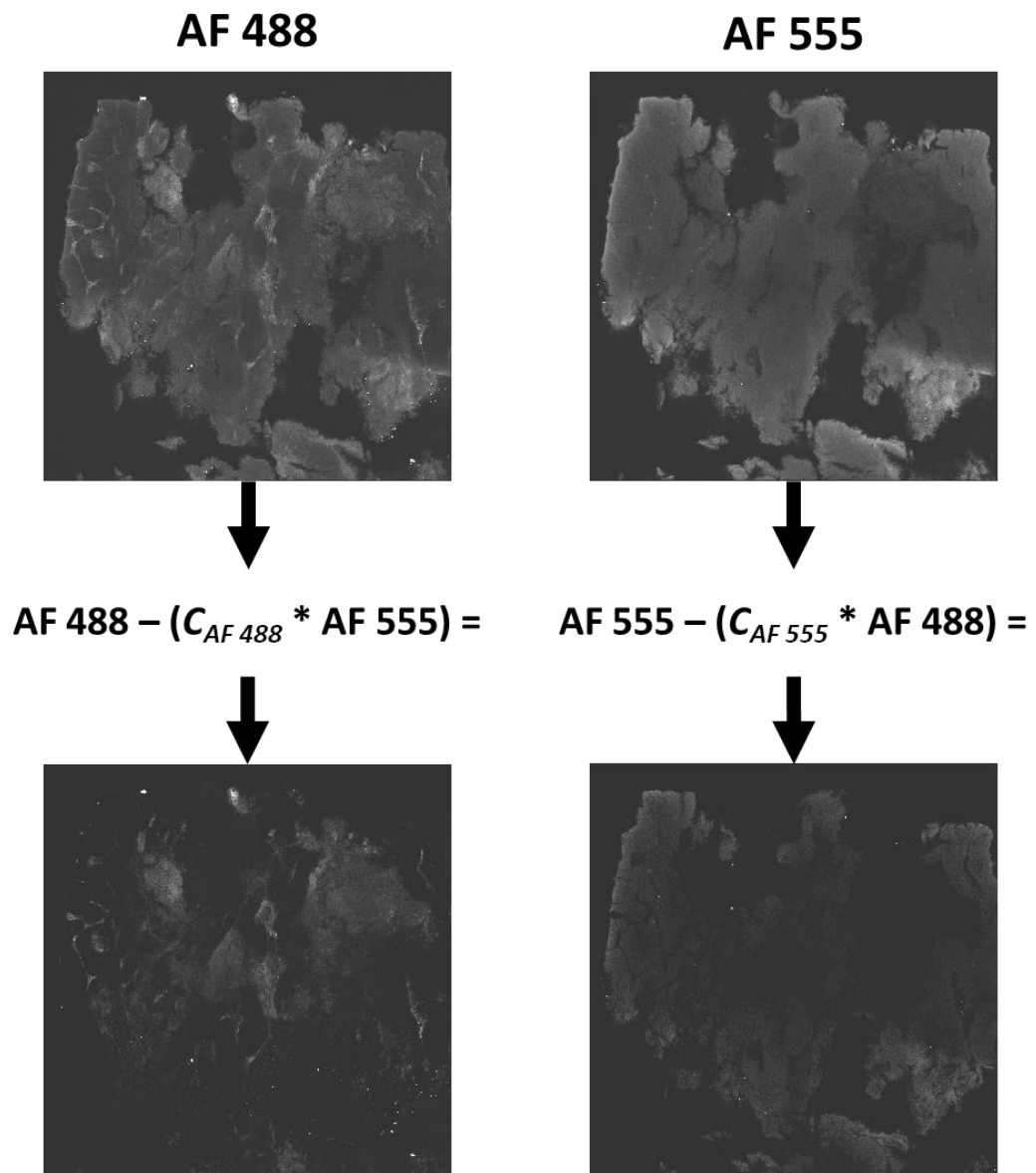


Figure 31. Subtract out autofluorescence.

Autofluorescence is subtracted out of the AF 488 (left) and AF 555 (right) channels (Table 5) by minimizing the absolute value of their differences. C is calculated independently for each channel by minimizing the absolute value of the equation, channel 1 – (C * channel 2).

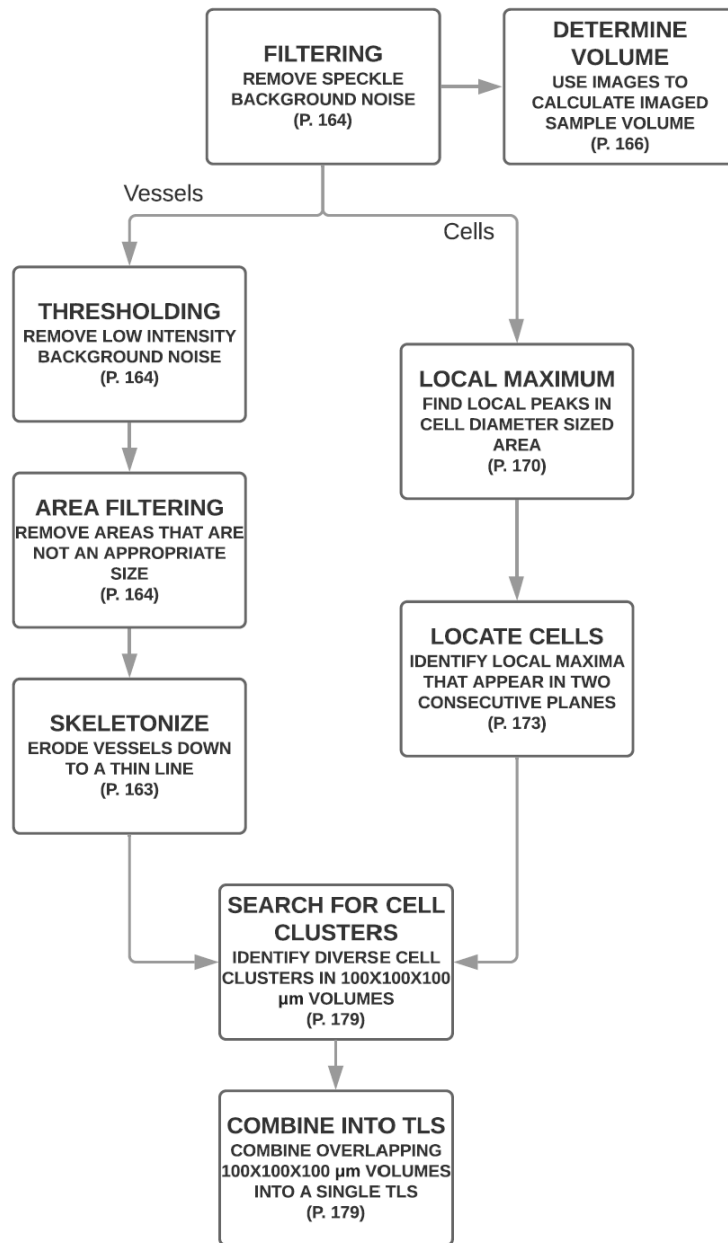


Figure 32. Overview of the data processing algorithm.

An overview of the data processing algorithm used to filter, identify, and analyse the 3D immune environment inside of tissue samples (page numbers refer to the respective piece of code in APPENDIX C).

When looking for B cells and T cells, local maximum pixel intensities were located within a single cell sized area. These locations were put through two tests to weed out noise from cells. The first test identified pixels around the local maximum with intensity values above a certain threshold and compared the resulting area with a minimum cell size. The second test compared the contrast of the interior pixel intensities of the cell with the pixel intensities of the area directly outside of the defined

cell area. If the mean value of the pixels outside of the cell area were similar to the mean pixel intensity values of those inside the cell, the local maximum lacked the contrast necessary to consider it a cell. The centroids of the remaining local maximums were recorded in each plane. To locate individual cell locations, the algorithm used recursion to identify where a local maximum appeared in the same x-y location in two consecutive z- planes. If the local maximum did not appear in two consecutive planes, it was considered noise because the diameter of a cell is larger than a single step in the z-direction. The centre points had to be within $1/4^{\text{th}}$ of the cell diameter to be considered the same cell (Figure 33).

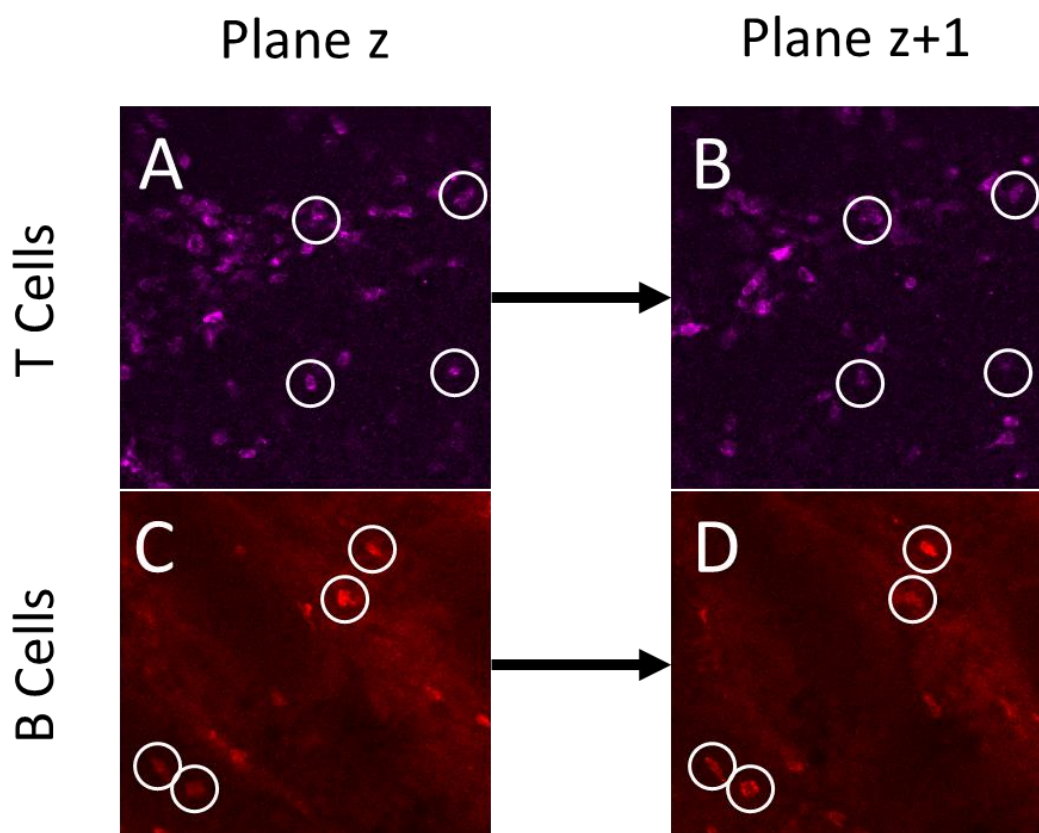


Figure 33. Cell identification in the tissue samples.

Cells were required to appear in two consecutive planes ($10\ \mu\text{m}$ step size) in the same location to be considered as a cell by the software. Representative consecutive planes of T cells (A and B) and B cells (C and D) are shown here with identified cells circled in white.

To determine the sample volume, all channels were summed pixel-by-pixel and a low binary threshold was applied to capture autofluorescence and scattered background noise. This allowed the program to calculate the area of the sample in that plane. This was done for each plane and these area values were summed and then

multiplied by the step size in the z-direction (10 μm) to calculate the imaged volume (Figure 34).

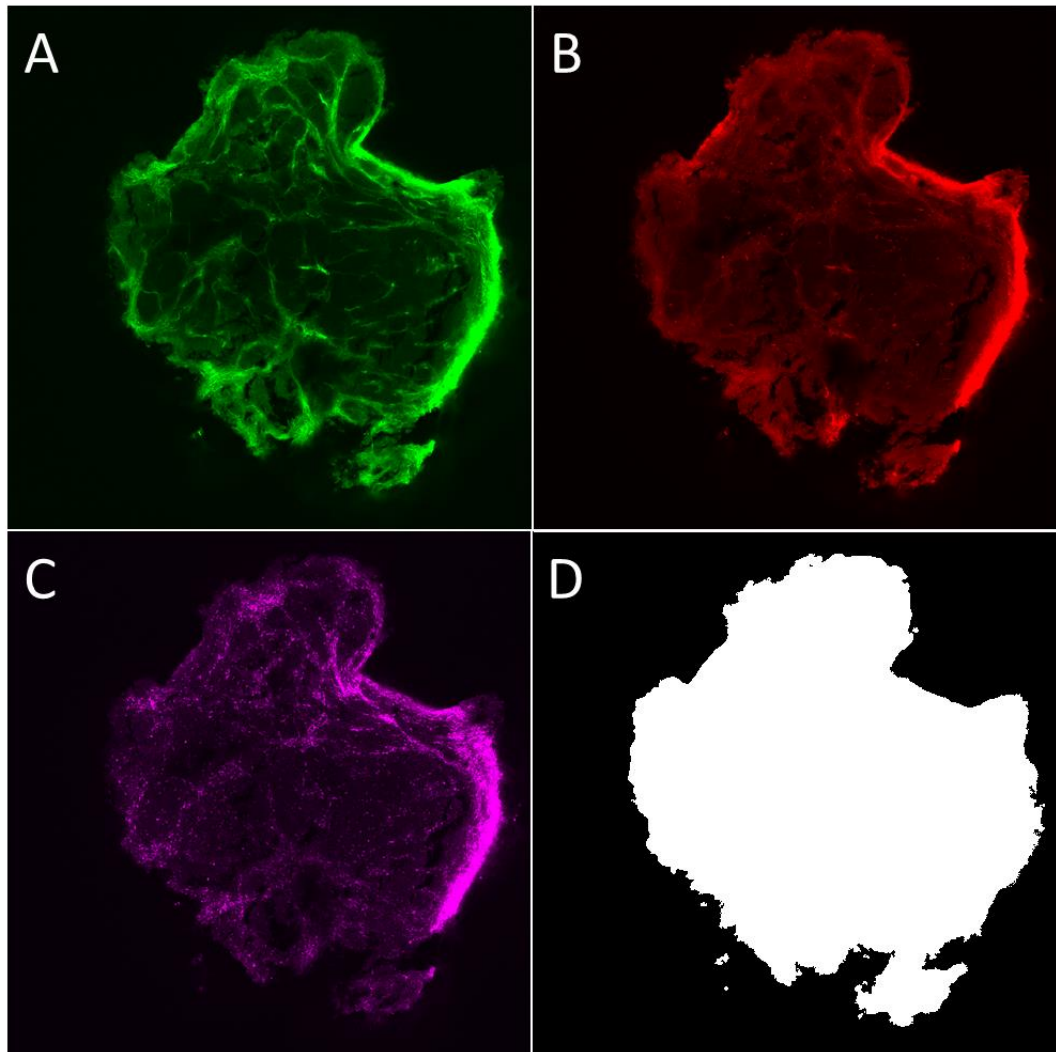


Figure 34. Using background autofluorescence to capture sample volume.

The pixel values from the three fluorescent channels (A-C) were summed together and a low threshold was used to create a binary mask of the current plane (D). The areas calculated from these planes were summed together and multiplied by the z-step size (10 μm) to calculate the imaged sample volume.

The second algorithm was focused on identifying small volumes with a highly concentrated and diverse cell profile using the cellular locations and skeletonized vessels. A $100 \times 100 \times 100 \mu\text{m}^3$ box was stepwise scanned across the entirety of the 3D image at 10 μm steps in every direction, counting the number of each type of cell inside the box at each scan point. Boxes were considered “hot spots” if the total number of each cell type (B cells, T cells, and HEVs) in the box were above its specific threshold (10, 20, and 5 cells, respectively; Figure 33). Since the software had skeletonized

HEVs down to only a single pixel thickness, 50 μm (or 50 pixels) worth of HEVs was considered 5 cells. These were then grouped together with any overlapping “hot spots” and as a collective, were considered a single TLS. If the volume of the newly defined TLS was less than double the size of the original scanning box ($<0.002 \text{ mm}^3$), the TLS was considered ill-defined and was excluded from the resulting statistics. The number of TLS in each sample was normalized by the sample volume to generate a TLS concentration.

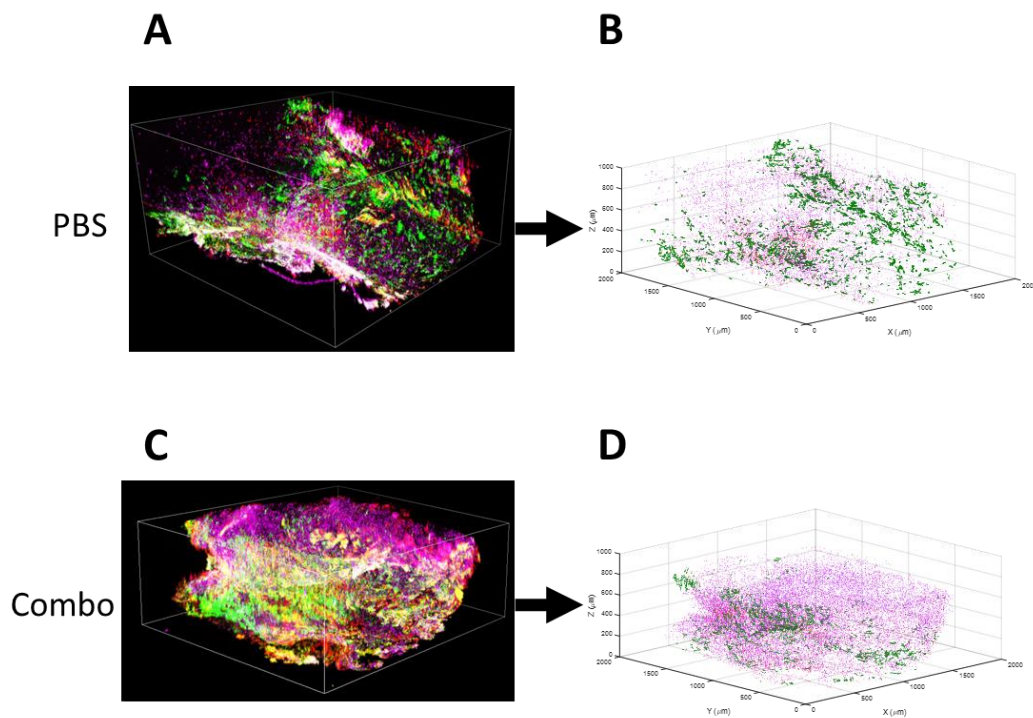


Figure 35. Cell and vessel segmentation.

(A,C) Filtered Nikon image files. (B,D) Digital representation of the image files with detected cell and vessel locations marked. Both the original and the digitized versions show CD3+ (magenta), B220+ (red), and HEV (green).

3.2.12. Error Analyses

A random number generator was used to generate a 3D image with dimensions equal to the imaged tissue samples ($\sim 2 \times 2 \times 1 \text{ mm}$) with 100 “cells” plotted in every two planes. Four different intensity levels of gaussian white noise were added on top of the cell signals with signal to noise ratios of 4, 2, 1.33, and 1. These test samples were run through the data analysis software to establish if the software can distinguish between background noise and whole cells within the collected data.

3.2.12.1. Changing analysis parameters affect the outcome

There are a few, very important parameters that, when changed, have a major effect on the results of the data analysis. These include the threshold for the minimum local maximum to consider for a cell and the size of the scanning box when searching for TLS. Some of the samples have weak signals, especially at large depths of imaging, due to either insufficient tissue clearing, poor antibody penetration, or a large autofluorescent background. To the human eye, it is still possible to differentiate these cells from the background noise, but this is difficult to translate into software. It was chosen to implement a low initial threshold and trust that the secondary checkpoints would protect the result. This leads to an initially high approximation of the number of cells in a single plane which is slowly eroded with each subsequent filter. Without a low threshold, the interior cells that have less fluorescent intensity would be indistinguishable.

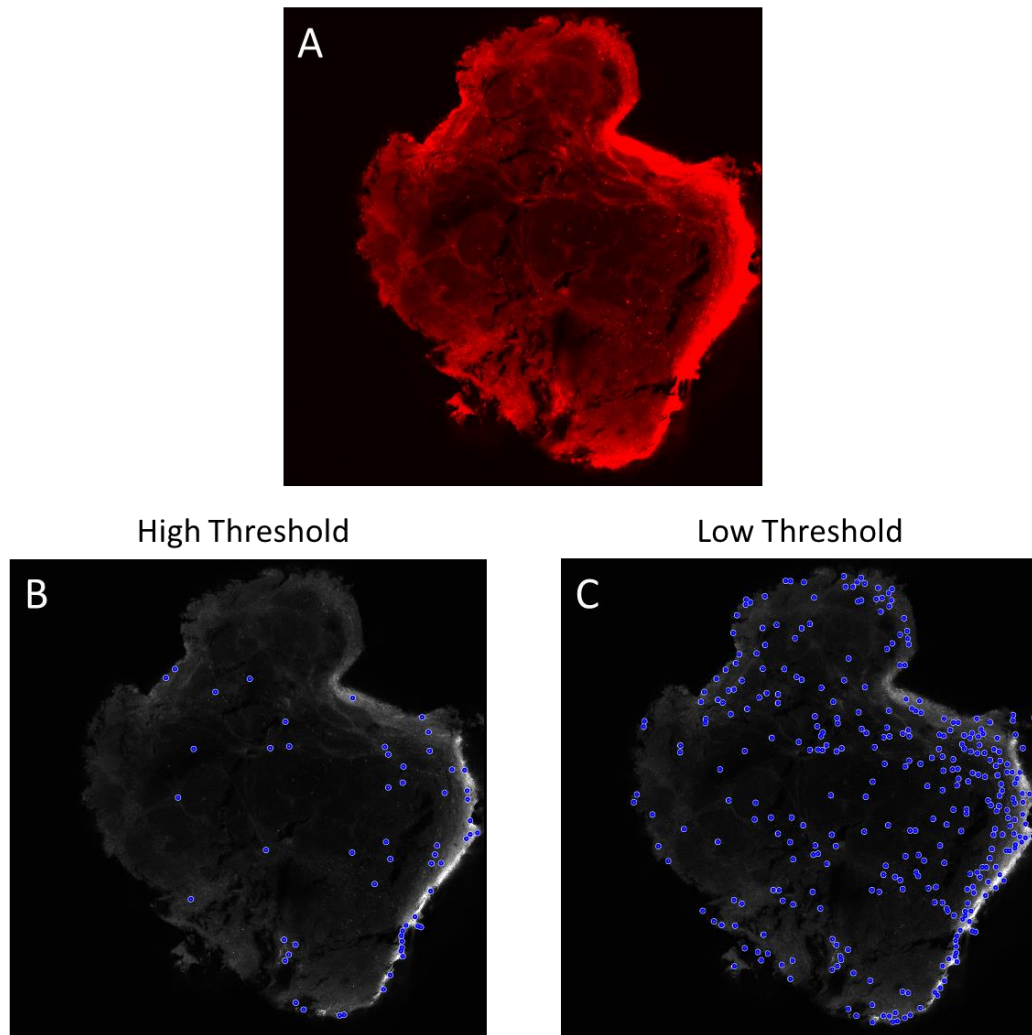


Figure 36. Low intensity threshold allows more cells to be identified.

The tissue sample (A) was stained for B cells. Using a high threshold (B) very few possible cell locations were identified from the background. Using a low threshold (C) increases the number of cell locations to be considered by the software.

The size of the scanning box has a significant effect on the size and number of final TLS. increasing the size of the scanning box should, in theory, increase the number of cells inside that box. Therefore, the cellular number thresholds must increase accordingly to account for the increase in scanned volume. When the scanning box is too large (e.g. 400x400x400 μm), even if it encloses an entire TLS, the concentration of cells is diluted and so there may not be enough cells to pass the TLS defining threshold. Even if the scanning box has a similar volume as that of a TLS (e.g. 200x200x200 μm), most TLS will not be a perfect sphere that fits inside the scanning box and could again fall short of the TLS defining threshold. A scanning box that is too small (e.g. 50x50x50 μm) could have TLS defining threshold values that are

too small to categorize as TLS. In these samples, no TLS were identified with scanning box sizes of 200x200x200 μm or above. A smaller scanning box could also define a single TLS as multiple TLS as the cell concentrations vary throughout the TLS (Figure 37). It was concluded that using a scanning box that was approximately half the size of the smallest TLS would give sufficient resolution as well as TLS specificity (Figure 45D and E).

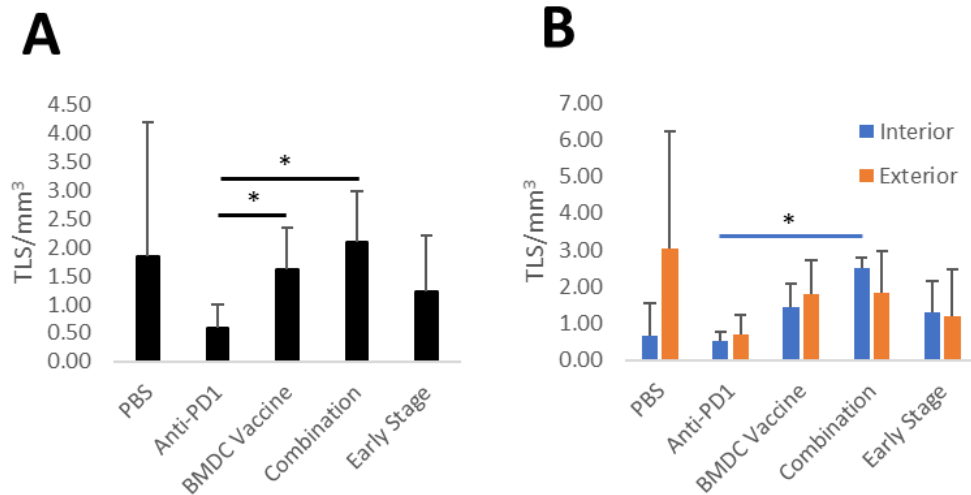


Figure 37. Examples of different sized scanning boxes.

A small scanning box (50x50x50 μm) requires small corresponding cell concentrations and therefore identifies a lot of regions overall (A) and between interior and exterior (B) when compared to the box used for the results presented here (100x100x100 μm ; Figure 45D and E). A box of equal volume to a TLS or larger requires higher corresponding cell concentrations and therefore identifies no TLS regions in these samples.

3.2.12.2. Analysis software limits the effect of white noise

When testing the effects of background white gaussian noise on the analysis, four different intensity levels were considered. These levels were dependent on the maximum intensity of an artificial cell. There are 5000 possible cells of each type to detect in this pseudo-sample. Although, due to the random placement of the cells, some are lost when the edges of the sample are truncated. Therefore, when the unaltered sample is put through the software, ~90% of the cells are detected (Figure 38A). When 0.25X and 0.5X noise was added (Figure 38B and C, respectively), the number of cells fluctuated by less than 5%. At 0.75X, the noise started to affect the initial stages of cell identification, including filtering and local contrast analysis (Figure 38D). This was corrected by the second half of the cell identification process, looking for the same

cell in two consecutive planes, and the cell count fluctuations only increased to within 10% of the unaltered cell count. Only once the maximum intensity of the noise matched the maximum intensity of the cell's fluorescence was there a significant change in cell detection, doubling the count of T cells (Figure 38E). Noise at this level was still unable to imitate a TLS to the system. Therefore, an assumption should be made that there will be at least one fluorescent cell of each type in each plane that is bright enough to prevent background noise from deceiving the software and increasing the cell concentrations.

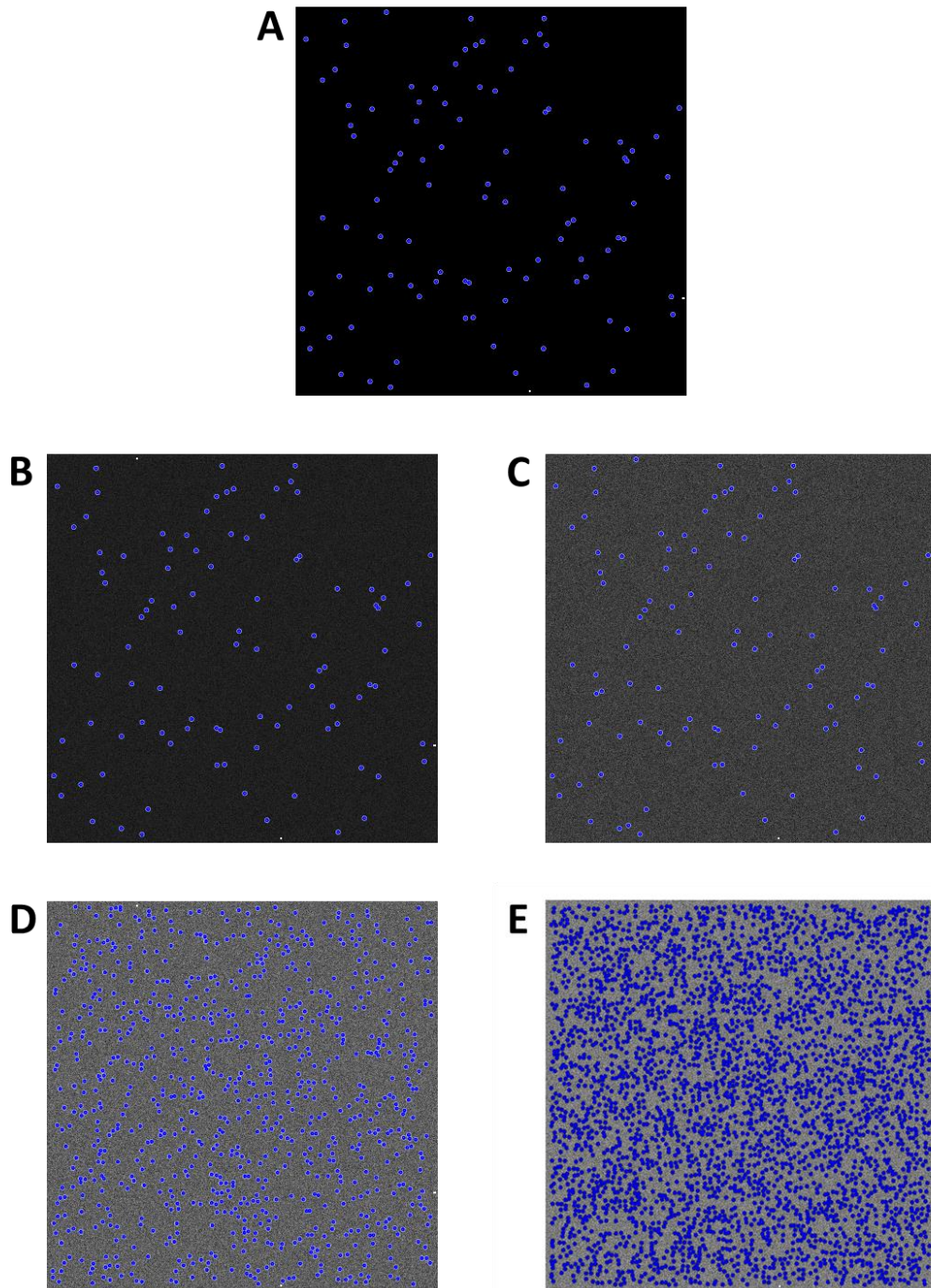


Figure 38. Background noise filtering and cell segmentation.

Different levels of noise were added to an unmodified test image (A) replicating an image of 100 cells. The different intensity levels include 0.25X (B), 0.5X (C), 0.75X (D), and 1X (E). Blue circles represent possible locations of cells.

3.2.13. Statistical Analyses

For the animal studies, each experiment group contained 3 animals. No randomization and/or blinding were used for the animal studies. No statistical method was used to predetermine sample group sizes. For the tumour sample clearing tests, all groups except the PBS control and combination treatment groups had three interior and three exterior samples, one of each coming from the same tumour in a single animal. Due to sample loss during processing, the PBS control group had two interior and two exterior samples, and the combination treatment group had two interior and three exterior samples. The values presented are the mean with standard deviation. A Student's t test (two tailed distribution, unequal variance) was used to determine all statistical significances, needing a p value < 0.05 to be considered significant. All statistical analyses were performed using Microsoft Excel (Microsoft Corporation, Redmond, Washington, USA) or GraphPad Prism 8 (GraphPad Software, San Diego, CA, USA).

3.3. Preliminary Results

These results were obtained by Dr. Qing Shi and inspired the following study into TLS inside the tumour space.

3.3.1. Combination treatment has a positive effect on tumour growth

Our lab has previously shown that the PSM/HER2-Primed DC vaccine ¹⁷⁹ significantly slows tumour growth. Here, Dr. Qing Shi reports similar tumour growth retardation using the vaccine/anti-PD1 combination treatment (Figure 39). The addition of anti-PD1 alongside the PSM/HER2 DC vaccine had a relatively small but positive effect on tumour growth when compared to that of the vaccine alone.

To ensure that the immune cell concentrations and configurations were not dependent on tumour volume, an early stage tumour control was included. This control group was inoculated 11 days after the inoculation of the other groups. The final volume of each treatment model is shown in Figure 39B. The early stage tumour control had a similar final volume to both the vaccinated and the combination treatment tumours.

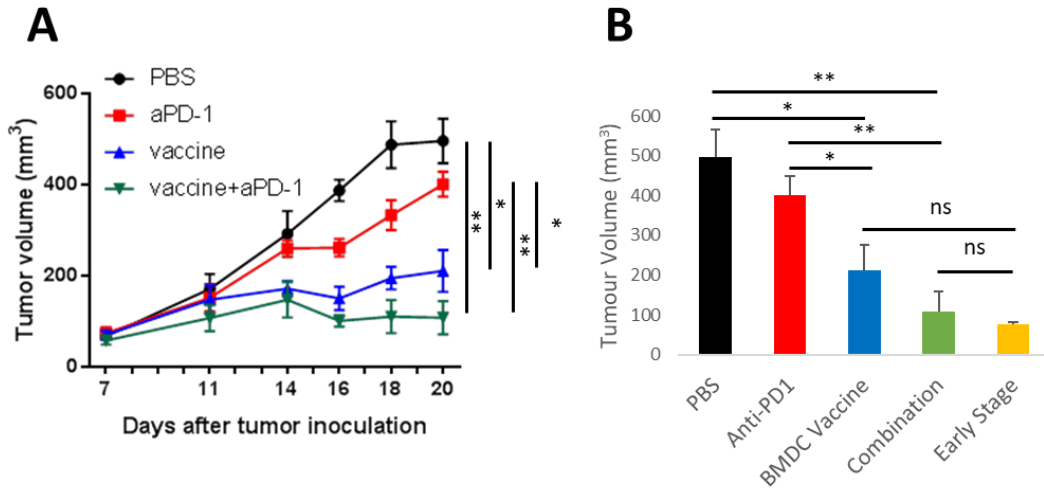


Figure 39. Treatment efficacy against TUBO solid tumours.

A) The tumour growth curve over 22 days. B) Final tumour volumes on day 20 including the early stage growth tumour. * indicates a p value <0.05, ** indicates a p value < 0.005, and *** indicates a p value < 0.0005.

3.3.2. All treatments show an ability to influence T cell recruitment

Dr. Qing Shi studied the cellular composition of solid tumours using flow cytometry. Treating mice with any of the therapeutic agents trended toward increasing the percentage of infiltrating CD3+ T cells in the tumour compared to other immune cells (Figure 40). In combination with the strong efficacy of the treatments, an interest was developed into the mechanism behind the response which inspired this study into TLS.

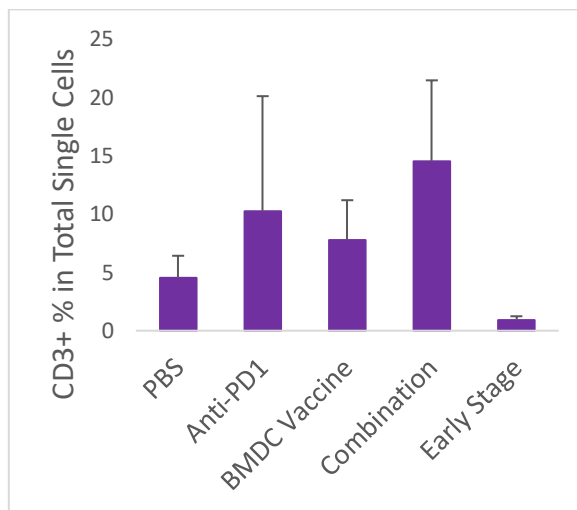


Figure 40. Flow cytometry shows a rise in T cells with treatment.

Treatment of the mice tends to increase the percentage of T cells in the total cell count after tumour sample digestions.

3.4. Results

3.4.1. Deep tissue staining and imaging

To study the 3D immune structure, tumour samples were optically cleared according to the iDISCO clearing protocol to increase the maximum imaging depth. Staining and imaging deep tissue structures (>100 nm) are constrained by the ability of antibodies to penetrate tissue samples and the optical laser scattering occurring at each cell wall interface due to changes in the index of refraction. This is commonly overcome by physically slicing the tissues into thin sections (~5-20 μm) thereby reducing the distance the antibodies must diffuse and limiting the number of scattering points that light will meet before being captured by the camera. In this study, the constraints are overcome chemically, using non-ionic detergents and organic solvents to permeabilize the tissue and clear away the steric interference caused by membrane lipids, and then decreasing the effects of light scattering by dehydrating the tissue and rehydrating it in DBE, an organic solvent with an index of refraction close to that of cell walls (Figure 41A). This allows the tissues to remain in larger sections and provide a 3D view of cellular structures without the risk of damaging or undersampling the tissue.

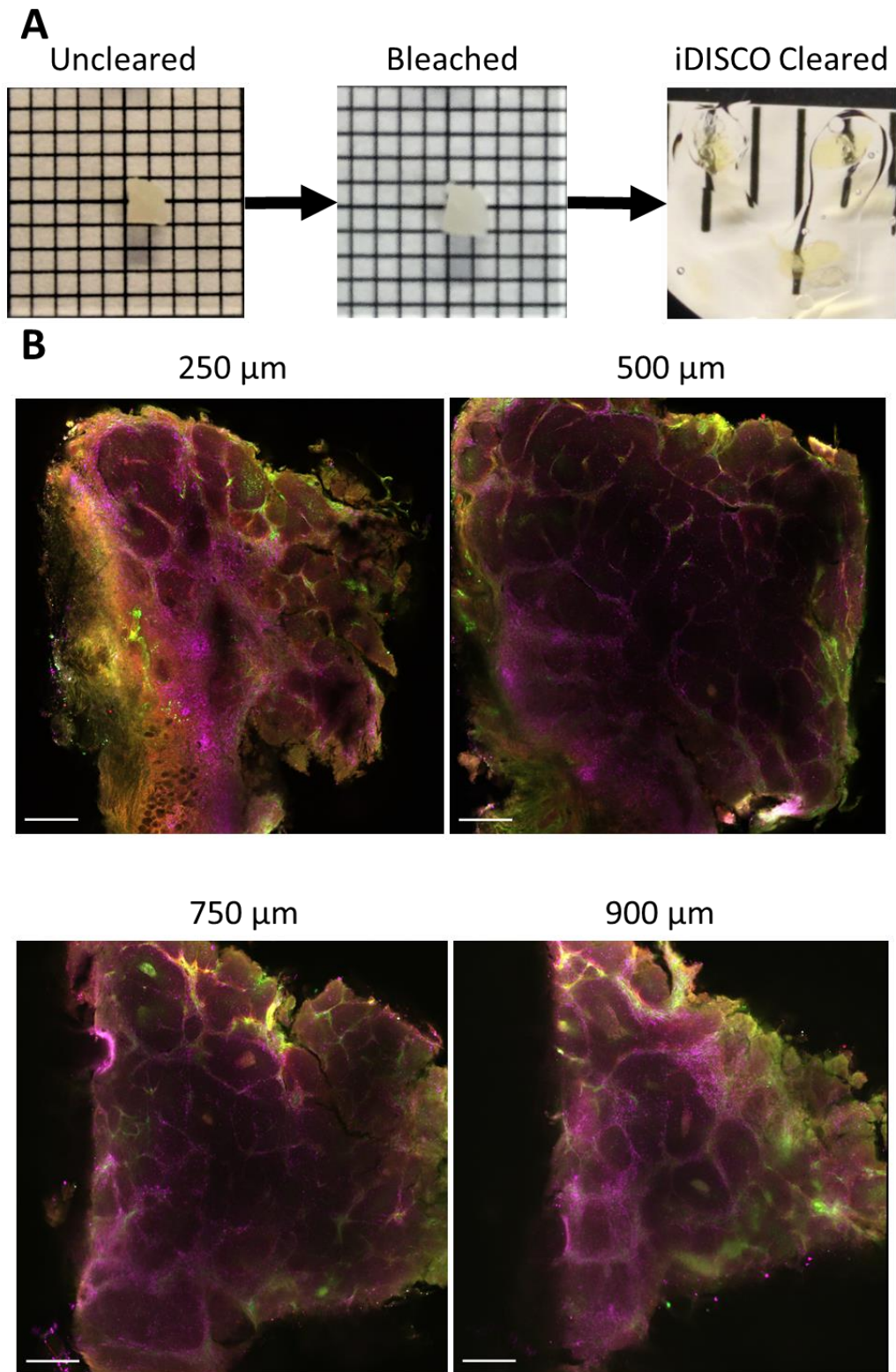
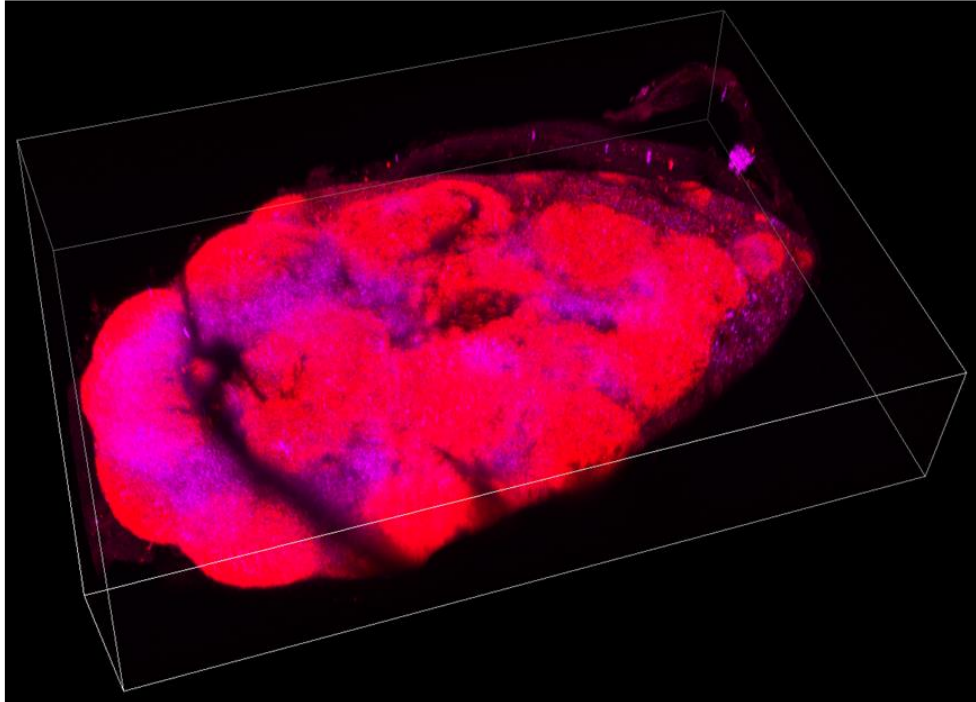


Figure 41. iDisco setup and depth visualization.

(A) 2 mm cubic tumour samples directly after dissection (left), after the sample was bleached overnight in H_2O_2 (middle), and after the sample was fully dehydrated and then rehydrated in DCM to create an optically clear tumour sample (3 tumour samples in the same image, right). The first two images were taken on 1 mm grid paper, the cleared image was photographed on a ruler with 1/16-inch apart ticks. (B) Tumour tissue at an imaging depth of 250 μm (top left), 500 μm (top right), 750 μm (bottom left), and 900 μm (bottom right) showing CD3+ (magenta), B220+ (red), and HEV (green). Scale bars represent 250 μm .

Many antibodies have already been proven to work with the iDISCO protocol,^{37,204} but this is the first time iDISCO has been used to investigate immune response specifically using the antibodies for CD3, B220, and MECA79 to look at T cells, B cells, and HEVs, respectively. The antibodies were tested on lymph nodes to ensure that these antibodies were functional and accurate when combined with the iDISCO protocol. The classical lymph node structure of a B cell germinal centre surrounded by T cells was easily identifiable (Figure 42). All three cell types were able to be seen at the maximum imaging depth inside the tumour tissues (900µm, Figure 41B).

A



B

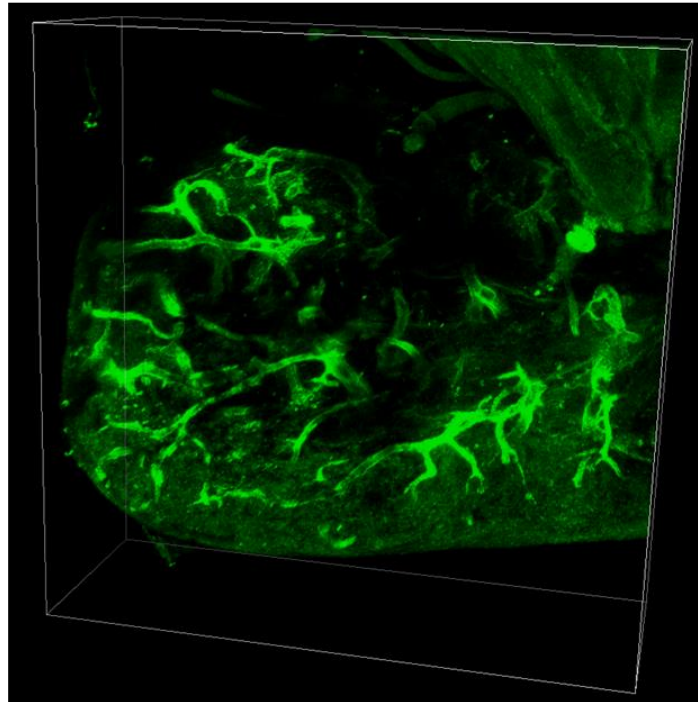


Figure 42. Lymph node stain confirmation.

(A) B cell germinal centres (red) and T cells (purple) stained in a lymph node. (B) HEV stained in a lymph node.

3.4.2. Vaccine and combination treatments initiate an increase in the concentration of tumour infiltrating CD3+ T Cells

The cellular composition of solid tumours was studied using flow cytometry. Treating mice with any of the therapeutic agents trended toward increasing the percentage of infiltrating CD3+ T cells (Figure 43A). Sections of the tumour samples were excised from the core and the exterior of the tumours and then optically cleared and stained for T cells, B cells, and HEVs to visualize the 3D immune structures inside the tumour. Image analysis software, written in MATLAB, identified a similar trend in the T cell infiltration as seen in the flow cytometry (Figure 43B). The software also revealed a significant difference between the combination treatment group and the untreated samples. The immune cell profiles of interior and exterior sections were compared against each other to better understand the depth of the T cell penetration. The main driving force of the increase in overall T cell infiltration was identified as the interior portion of the tumour sample (Figure 43C), as seen significantly in the combination treatment group and trending in the vaccination alone group. This is likely due to the increased activation of the T cells from the adoptive transfer of BMDCs. When concentration information was compiled for B cells, there was no significant difference between the treatment groups (Figure 44A). HEVs showed a difference between the PBS group and the early stage tumours, which could indicate that it takes time to develop these vessels, and between the PBS group and the anti-PD1 group (Figure 44). This significant decrease in HEVs in the anti-PD1 group, along with a similar decrease in the combination group, could be indicative of PD1 playing an important role in the development of HEV outside of lymphoid tissue. Zhu et al. reported that HEV are associated with tumour infiltrating lymphocytes with downregulated immune checkpoint proteins.²⁰⁸ It is possible that administering the anti-PD1 treatment provided the body with a surplus of these lymphocytes, alleviating the necessity for HEVs.

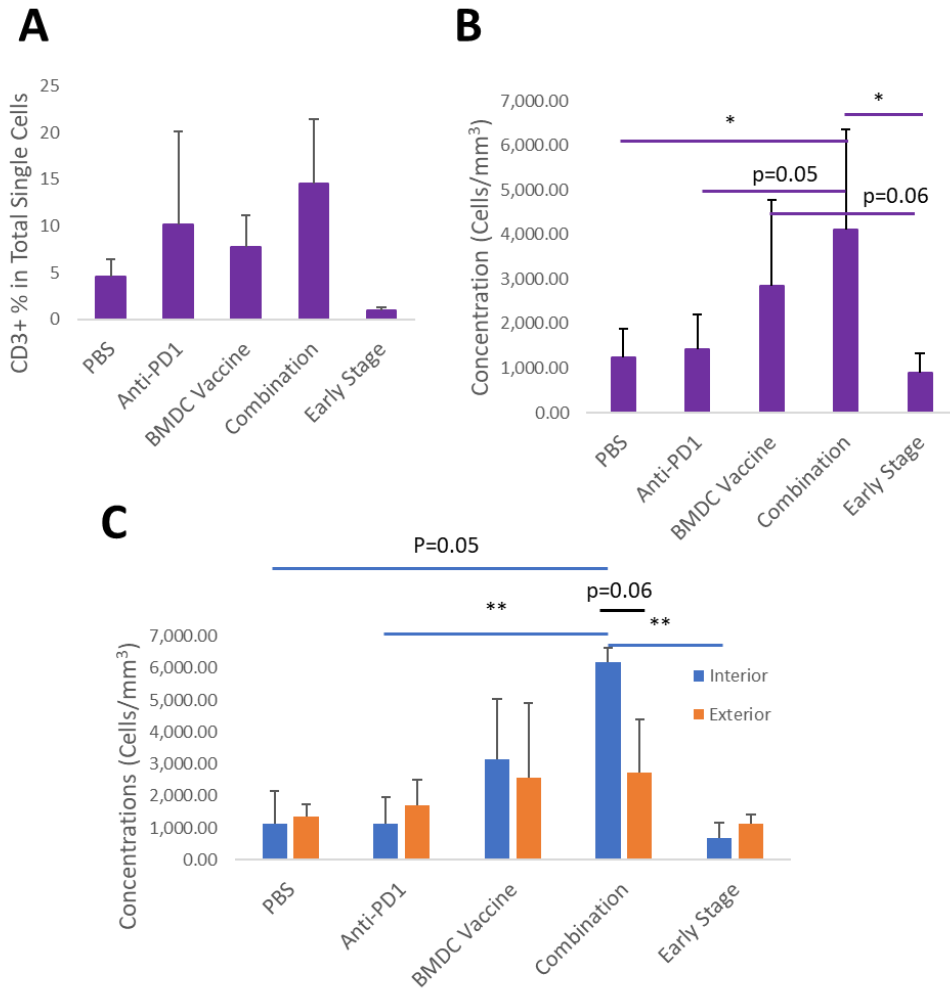


Figure 43. Comparison of the T cell penetration.

Cell concentrations from flow cytometry (A) and 3D analysis (B) show that the vaccine and combination treatments encourage an influx of T cells into the tumour. This is emphasised by the increase in T cells in the interior portion of the tumour compared to the exterior (C). * is for a p value < 0.05, ** is for a p value < 0.005, blue bars show significance between the interior samples from each treatment group, black bars show the significance between the interior and exterior samples of a single treatment group.

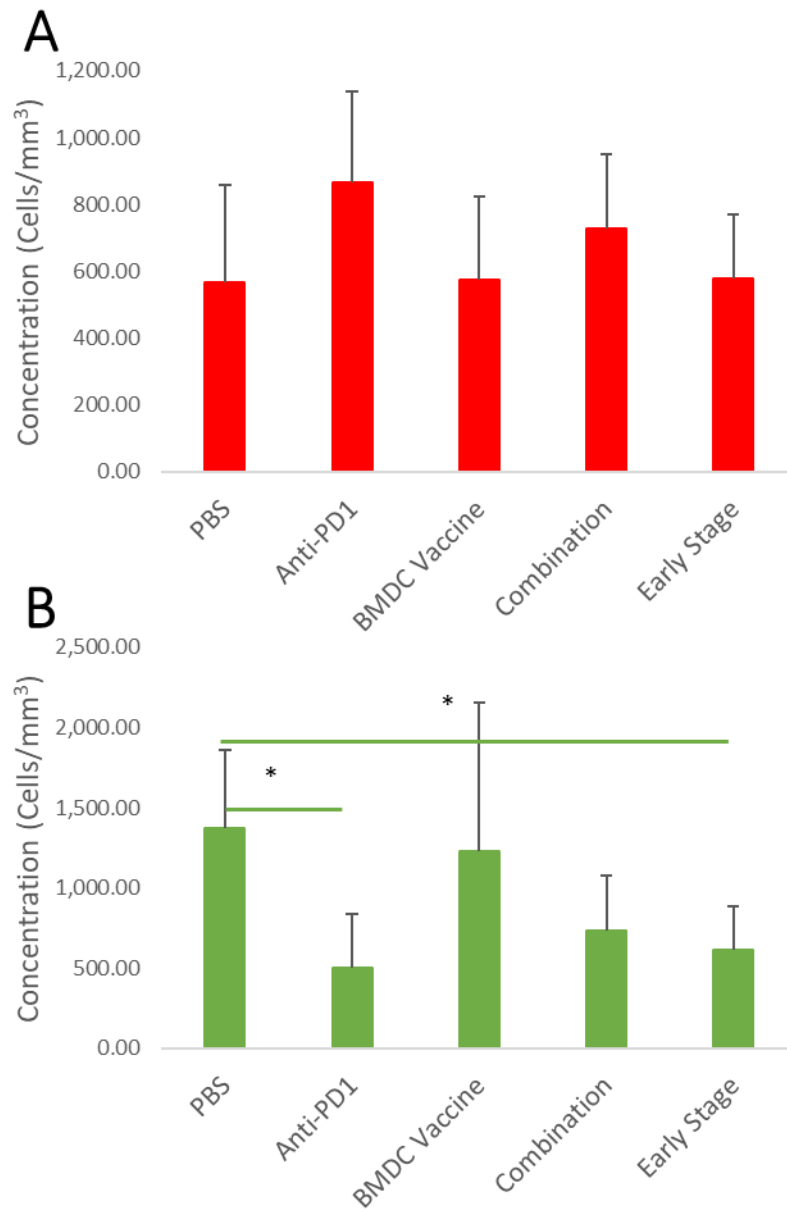


Figure 44. B cell and HEV concentrations inside the tissue samples.

A) Treatment has no significant effect on the concentration of B cells in the tumour space. B) Anti-PD1 shows a tendency to reduce the concentration of HEVs in the sample. * is for a p value < 0.05.

3.4.3. Combination treatment increases the concentration of TLS in the tumour

Optically cleared tumour sections were scanned for small volumes (100x100x100 μm cube) that contained all three cell types (Figure 45A, B, C). Previous studies have used T cells, B cells, and HEVs, together and individually, to identify TLS in tissue samples.^{193-194,196-198,201} Based on a meta-analysis of previously published data and images, TLS must consist of at least 10 B cells, 20 T cells, and 5

HEV within the defined scanning volume (Figure 46). Using these criteria, the MATLAB program identified TLS in each sample. Even with the concentrations of HEV and B cells remaining relatively constant (Figure 44), tumour sections treated with the vaccine therapy alone showed a significant increase in the concentration of TLS present within the overall tissue. Tumours treated with the combination therapy also saw an increase in TLS concentration but with a large variation (Figure 45D). The increase in T cell concentration correlates well with the increase in TLS concentration. When the interior and exterior sections were examined separately, the reason for the TLS concentration variation in the combination treatment group is made apparent. The concentration of TLS in the interior sections of the combination treated tumours showed a substantial increase over all other non-vaccinated groups whereas the exterior of the combination treatment tumours had the lowest TLS concentration of all groups (Figure 45E). The variance of the TLS concentrations in the exterior samples of all treatment groups were too large to illuminate any significant changes. This may be due to the differing tumour growth patterns (Figure 39A) and the heterogeneity that comes with each tumour and each patient. ³In theory, having TLS in the interior regions of the tumour mass decreases the lag time between immune cell communication. These local immune cell hubs provide quick and unfettered access to crosstalk for immune cells fighting cancer which could be major contributor to the slowed tumour growth.

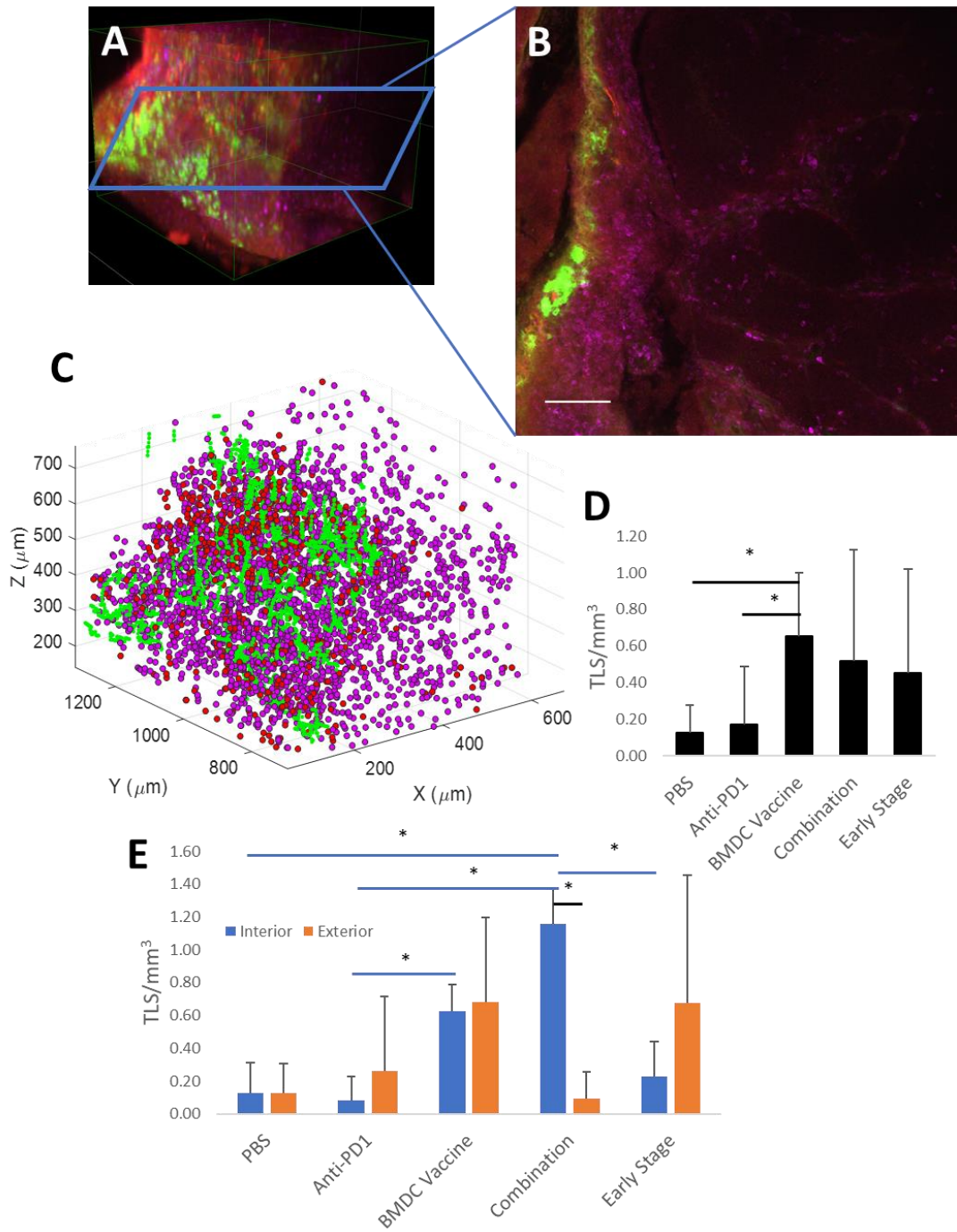


Figure 45. TLS identification and concentration inside tumour samples.

Area detected by the software to be TLS (A), shown in a 2D slice (B) and in a digital representation (C) showing CD3+ (magenta), B220+ (red), and HEV (green). The overall average number of TLS/mm³ (D) and the comparison of the interior and the exterior of the tumour (E). Scale bar represents 100 μ m. * is for a p value < 0.05, blue bars show significance between the interior samples from each treatment group, black bars show the significance between the interior and exterior of a single treatment group.

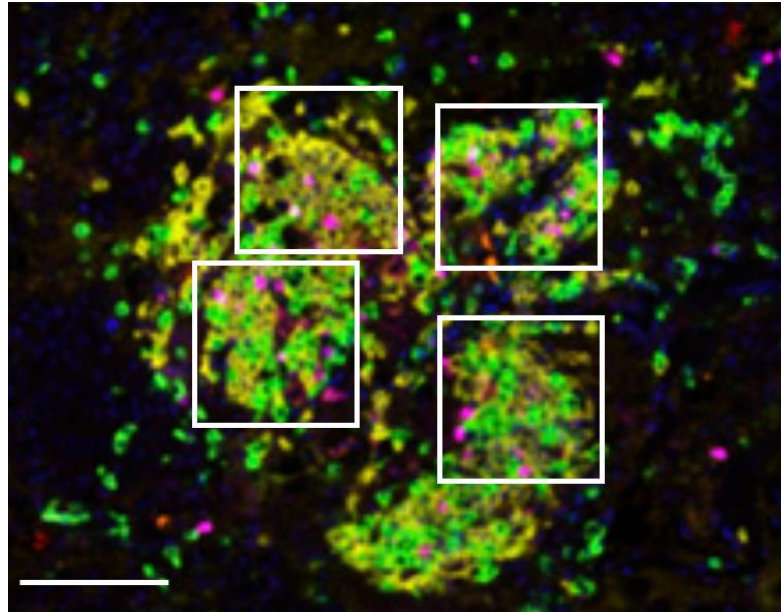


Figure 46. TLS meta-analysis.

A meta-analysis was conducted on a previously reported intratumoural TLS to determine minimum cell counts inside the given area. Cells were counted inside of white reference squares (100 μm in length). The top left box contains approximately 19 CD20 B cells, 12 CD8 T cells, and 5 HEVs. The top right box contains approximately 17 CD20 B cells, 22 CD8 T cells, and 3 HEVs. The bottom left box contains approximately 14 CD20 B cells, 24 CD8 T cells, and 6 HEVs. The bottom right box contains approximately 16 CD20 B cells, 20 CD8 T cells, and 4 HEVs. The threshold was set at 10 B cells, 20 T cells, and 5 HEVs. Scale bar represents 100 μm . Reproduced with permission from The Journal of Immunology.¹⁹⁵

3.5. Discussion

The presence of TLS in tumour masses have a positive association with improved patient outcomes.¹⁹⁹ In this study, it was seen that TLS are associated with slowed tumour growth progression. Moreover, it was characterized how three immunotherapies modify the immune environment inside tumour masses. When treated with both anti-PD1 and the PSM/peptide BMDC vaccine, there was an increase in the concentration of TLS which associated with decreased tumour growth. This work parallels work done by Zhu et al. where they reported that the formation of TLS in the tumour space is associated with a decrease in PD1+ CD8+ T cells and together, they suppress tumour growth.²⁰⁸ The increase in TLS concentration caused by the combination treatment may lead to a continuous, long-term activation of the immune system, and possibly, to an established cancer immunity, as was seen in the study by Xia et. al.¹⁷⁹

Xia et. al. have previously reported that the PSM/peptide BMDC vaccine can alter the tumour microenvironment, making it more receptive to immune recognition and creating a pro-inflammatory environment.¹⁷⁹ The results presented here support this claim, finding an increased influx of T cells into the tumour and an increased concentration of TLS following application of the vaccine. Subcutaneous (s.c.) injections of DC vaccines have been reported to incite the formation of TLS at the vaccination site in mice with B16 melanoma.²⁰⁹ While the s.c. injection would, in theory, keep a large portion of the DCs well confined in one area, increasing the potency of cellular signals to attract other immune cells to the area, the use of i.v. injection allows primed BMDCs to quickly spread throughout the entire body. As these primed BMDCs pass into lymph nodes, they present the loaded tumour antigens to other immune cells and allow tumour-specific T cells to activate and expand, eventually leading to TLS development in and around the tumour. Future biodistribution studies following the injected BMDCs are necessary to elucidate the development of these TLS in connection with where the BMDCs go and what types of cells they are communicating with. Another study should focus on confirming these results using immunohistochemistry, the gold standard of tissue staining. While iDISCO is a validated technique for immunostaining and imaging large tissue sections, the addition of histology slides would go a long way to support these claims. These studies are currently being run on a separate in vivo study to replicate these results and provide additional insight into the development of TLS.

TLS have generally been identified by observing a clustering of certain types of cells and/or gene expression in a small area.¹⁹⁹ At the time of this publishing, there are no defined cell concentrations in place for an immune structure to be considered a TLS. After analysing immunofluorescence images from published works on TLS,¹⁹⁵ it was concluded that a TLS could be identified when, inside a 100x100 μm area, there are at least 10 B cells, 20 T cells, and 5 HEV. This pattern also had to continue for at least a continuous 200x200 μm area to be of a size with the smaller TLS seen in literature.¹⁹⁵ In the same Engelhard review, they mention regions that may be precursors to a TLS (Figure 28F). Another metanalysis was done on these regions and it was observed that 2 B cells, 10 T cells, and 5 HEVs would fit inside the 100x100 μm search area (Figure 47A). The same software was used to analyse the samples used

in this study for these precursor TLS (Figure 47B). Yet again, the combination treatment group showed an advantage over the other treatment methods, except for the PBS group which had one significant outlier which brought the mean up and increased the variation. A well-defined trend develops in both the interior and exterior samples, which correlates well with the tumour growth curves.

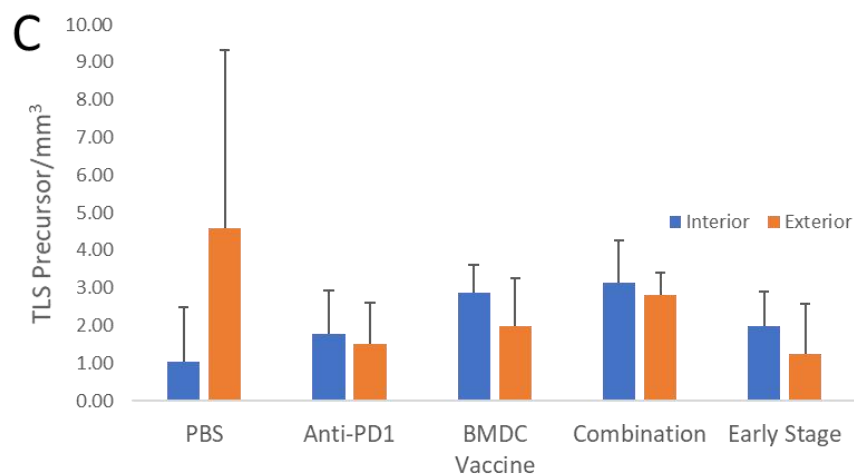
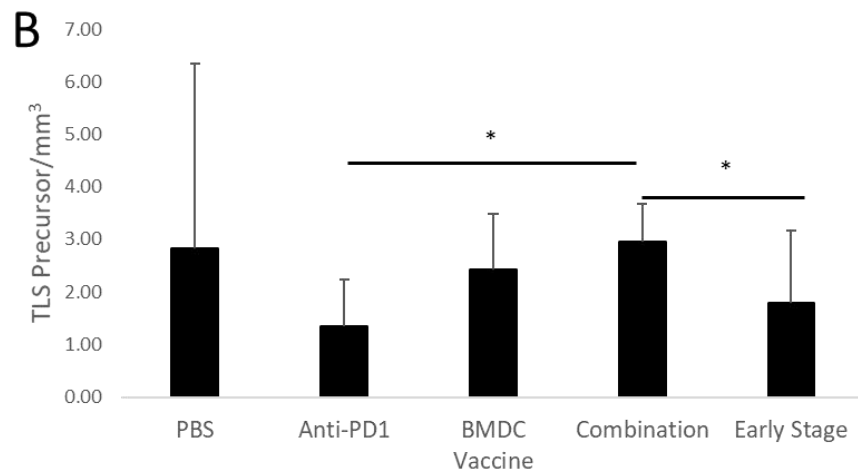
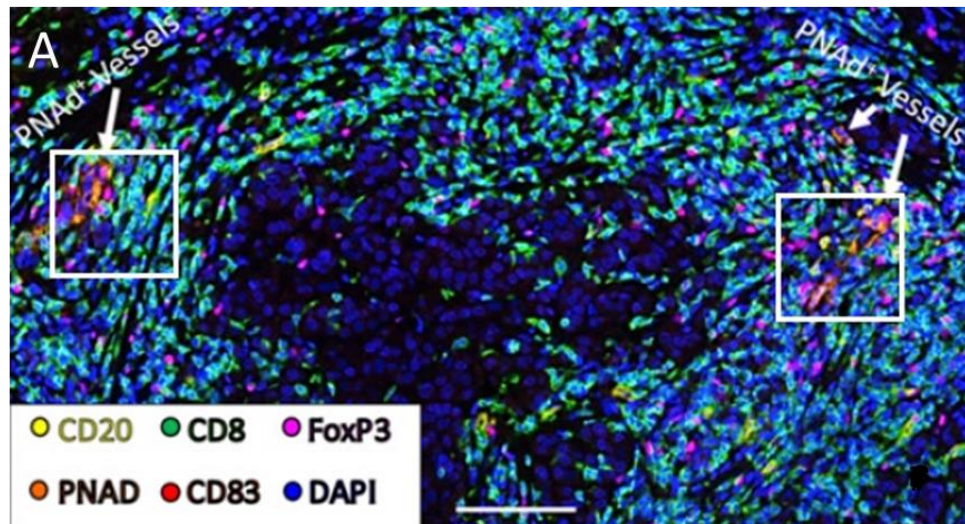


Figure 47. TLS precursors.

A) In the TLS precursor metanalysis, cells were counted inside of white reference squares (100 μm in length). The left box contains approximately 3 CD20 B cells, 13 CD8 T cells, and 6 HEVs. The right box contains approximately 4 CD20 B cells, 16 CD8 T cells, and 7 HEVs. The threshold was set at 2 B cells, 10 T cells, and 5 HEVs. Scale bar represents 100 μm . Reproduced with permission from The Journal of Immunology.¹⁹⁵ B) TLS precursor concentration averages in both interior and exterior samples combined. C) TLS precursor concentration averages in interior and exterior samples separately.

Another way of identifying TLS is by identifying the organizational structure of immune cells such as a B cell follicle surrounded by T cells, which resembles lymph node architecture. However, this structural definition might be too strict and miss detection of TLS in the interior of tumour masses. As seen by Engelhard et. al.,¹⁹⁵ intratumoural TLS are not nearly as well structured as when the TLS forms peritumourally or elsewhere in the body. This could be due to the increased density of cells within the tumour which would affect the ability of the immune cells to manoeuvre and position themselves. Thus, the search is not restricted to immune cell clusters that match the classical TLS organisational definition. While a 2D analysis by immunohistochemistry is common for finding the cellular composition of tissue samples, 3D imaging techniques collect uninterrupted continuous data without the risks of undersampling the tissues. 2D imaging has the chance of being out of alignment with the structure of interest, leaving only a small portion of the structure visible in any frame. If enough 2D slices are collected, a digital reconstruction could be formed to build a similar 3D volume, but each additional step increases the probability of a mistake. While the 3D sample preparation takes longer than a conventional tissue slide, tissue clearing is well-suited for searching for small structures inside a large volume.

When selecting the tumour sections for analysis, tumour samples were meticulously split into two groups, interior and exterior samples. Interior samples are used to investigate the immune response in the necrotic core of the tumour sample. Due to the rapid growth of the tumour, the core tends to become hypoxic and nutrient deficient and would therefore consist of more dead or dying tumour cells.²¹⁰ The exterior group of tumour sections are from the portion of the tumour that borders healthy tissue. These cells and the space around them should have an ample supply of nutrients and oxygen with which to continue multiplying and expanding. Due to the differences in the end tumour volumes between the treatment groups, it is difficult to compare the interior and the exterior sections without questioning if the results are volume driven. One mitigation for this quandary was to inoculate a second group of control mice eleven days after the other mice treatment groups were inoculated. This second control treatment group ended the study with similar tumour sizes to the

vaccinated and combination treatment groups and were therefore considered comparable for analysis with the other treatment methods.

An interesting trend appeared with the switch between interior and exterior T cell concentrations. The untreated and anti-PD1 samples showed a higher concentration of T cells in the exterior of the tumour mass than the interior. Yet both the groups treated with the BMDC vaccine had a higher concentration of T cells in the interior of the tumour mass. Although without further staining, it is unknown whether these are cancer promoting T cells (regulatory T cells) or cancer suppressing T cells (Cytotoxic, helper, natural killer, or memory). Building evidence supports the positive prognostic value of tumour-infiltrating leukocytes and has claimed that they are necessary for a sustained and effective immune response in breast cancer.²¹²⁻²¹³ The increase in intratumoural T cell concentration correlating with the shrinking tumour volume in the combination treatment group corroborates this claim. The increase in the T cell concentration in the core of the tumour could be related to the concentration of tumour cells in that area. Once the T cells have been activated by the HER2 antigen, they seek out areas of disease or inflammation. The tumour microenvironment is rife with inflammation which helps the tumour grow and spread.²¹⁴ For the purpose of this vaccine, the inflammation is an advantage that allows the newly activated T cells to penetrate the deepest areas of the tumour mass. By merging the BMDC/HER2 antigen vaccine with anti-PD1 checkpoint inhibition, it allows the T cells to attack cancer cells that are presenting PD-L1. Once these cells have undergone apoptosis, a positive feedback loop would be created to attract and activate more immune cells in the area, leading to a more efficient recruitment and development of TLS in the tumour space.

In conclusion, treatment with the BMDC vaccine combined with an anti-PD1 checkpoint blockade therapy can produce a strong, anti-tumour effect. This effect coincides with an increase in the presence of intratumoural TLS which are associated with positive patient prognoses. A new method for detecting TLS in large tissue samples was devised and these techniques were used to shed a new light on the connection between vaccines and TLS, as well as to accentuate the importance and value of 3D imaging.

4. ANALYSIS OF FLUORESCENCE IMAGING OF STAINED RED BLOOD CELLS WITH SIMULTANEOUS RESONANCE RAMAN PHOTOSTABILITY ANALYSIS

4.1. Introduction

Biomedical optical imaging has been revolutionized by recent developments in light sources and spectroscopic techniques.²¹⁵⁻²¹⁷ For example, fluorescence and Raman spectroscopies have been widely used in imaging a variety of biological systems including single biomolecules, cells, tissues, and whole organisms.²¹⁸⁻²²⁰ Different spectroscopic techniques have advantages and disadvantages which are often complementary and can be used selectively or combined to address specific problems. Fluorescence signals are typically orders of magnitude stronger than Raman signals. However, biological systems often lack chemically-specific fluorescence signals in the visible spectral range and require fluorescent dye staining to improve the imaging contrast. Staining complicates the experimental preparation process and may influence biological functions.²²¹ On the other hand, Raman spectroscopy is label-free, allowing study of the sample in its native state. Weak Raman signals may be enhanced using methods such as surface enhancement, coherence, or, as used here, electronic resonance effects. In resonance Raman scattering, the incident laser wavelength is aligned with the electronic transition of the molecule of interest. The transfer of energy between the photon and the molecule allows the molecule to bypass the virtual state up to the excited electronic state (Figure 48C). Resonance Raman scattering can increase the polarizability of the molecule and subsequently, the Raman intensity can also increase by up to a factor of 10^6 . Both fluorescence and resonance Raman imaging benefit from high intensity lasers, which can lead to laser-induced sample damage and disruption of biological functions. Photobleaching is caused by photon-induced chemical damage or covalent modifications.²²² To perform bioimaging using these techniques, the effects of the laser intensity on the photostability of biomolecules and live cells must be precisely determined. The challenges of rapid non-destructive spectroscopic imaging in biological media require developing new approaches for simultaneous bioimaging and estimation of the cell stability due to possible photodamage.

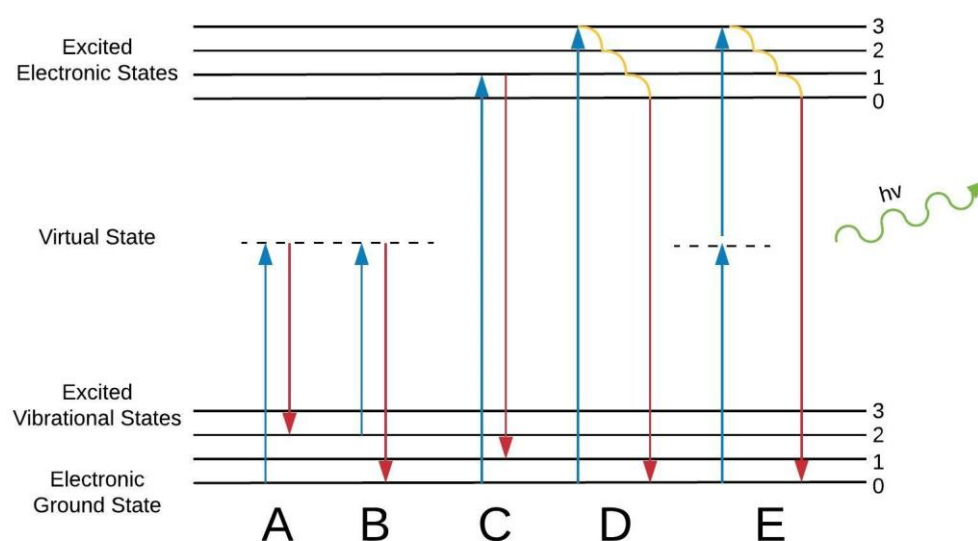


Figure 48. Jablonski diagram.

This figure is a reproduction of Figure 1 from the Introduction. Excitation (blue) and emission/relaxation (red) for Stokes Raman scattering (A), Anti-Stokes Raman scattering (B), resonance Raman scattering (C), fluorescence (D), and two-photon excitation fluorescence (E). The numbers represent the vibrational levels within the electronic states.

Here, optical imaging of simple stained cells, i.e. rhodamine 6G stained red blood cells (R6G-stained RBCs), is performed using a combination of fluorescence and resonance Raman scattering. R6G is a typical fluorescent dye used as the lasing gain media and for the staining of biological systems.²²³⁻²²⁴ It is also a common Raman reporter molecule used for improving the Raman imaging contrast and as a Raman marker in biosensing.²²⁵⁻²²⁸ R6G was specifically chosen because its fluorescence and resonance Raman signals are both relatively strong and may be obtained using 532 nm laser excitation. A simple biological cell, RBC, whose main constituent protein component, haemoglobin (Hb), was selected to provide a strong resonance Raman signal upon 532 nm laser excitation which coincides with the fluorescence and resonance Raman signals of R6G. By staining the RBC with R6G, a stained biological model system was designed which can be used for multi-component bioimaging within a single experimental setup using a single excitation source. This approach allows simultaneous exploration of several effects including fluorescence and resonance Raman bleaching of the stain, and photostability of the biological system.

RBCs are critical for the transport of oxygen to the cells in the body. They provide an ideal platform to study oxygenation capability using resonance Raman

spectroscopy.²²⁹⁻²³² Laser-induced photodamage in RBCs has also been probed using Raman spectroscopy.²³³ Bleaching of the fluorescence of R6G can be precisely controlled and monitored by tuning the laser intensity. Using this control parameter, Hb Raman signals are detected in the presence of the strong R6G fluorescence, and R6G Raman signals in the presence of a strong Hb Raman contribution. Since R6G forms a thin monolayer on the surface of RBCs, its Raman signal is weak compared to Hb which fills the interior of the cells. However, since R6G has a larger absorption coefficient and higher bleaching decay rates than Hb, the bleaching of both fluorescence and resonance Raman signals of R6G are achieved before any noticeable changes in the Raman spectra of Hb occur.²³⁴ The competition between these resonant signals allows the detection of R6G and Hb in the RBCs under various conditions while probing the photostability of stained RBCs. These results provide a step towards improving real-time multicomponent cellular imaging with high resolution and simultaneous photostability control.

4.2. Materials and methods

This work was a collaboration between students at Swansea University (SU) and Texas A&M University (TAMU). The experiment was performed by members of the Meissner lab and the Sokolov lab at TAMU and the data processing was completed by me at SU. Their experimental methods are reported below to give a complete report of this experiment and to provide a basis for the data analysis.

4.2.1. Cell Staining

As performed by students at TAMU. Whole bovine blood was provided by the Veterinary Medical Park, TAMU. 5 ml of blood was centrifuged at 9000 RPM for 5 minutes to separate the plasma, buffy coat, and RBCs. The buffy coat consists of leukocytes and platelets (Figure 49). The supernatant plasma and the buffy coat were pipetted off and discarded. RBCs were resuspended in 10 mL of phosphate buffered saline (PBS) and to ensure only RBCs remained, centrifuged at 9000 RPM for 3 minutes and the supernatant was again discarded. RBCs were stained with 1 mM R6G in PBS for 5 minutes at room temperature. R6G adsorbs onto the membrane of RBCs due to its lipophilic nature²²³ and forms a stained cell (Figure 50A). At this point, RBCs were split into two groups. Half of the RBCs were air dried on a glass slide and

then imaged immediately (referred to as “fresh” cells). The other half were stored at 4°C for 1-week and then air dried on a glass slide and imaged (referred to as “1-week old” cells). The RBCs underwent a week long storage at 4°C to ensure the cellular system stabilized after the effects of short term (a few hours) cell degradation in ambient conditions.

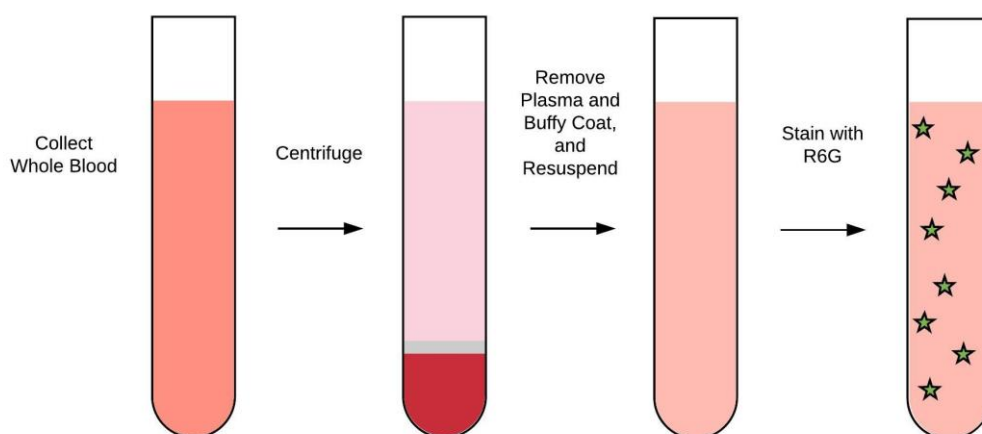


Figure 49. RBC collection and staining.

Bovine RBCs were centrifuged to separate the different cell types. The plasma, white blood cells, and platelets were removed with a pipette. Purified RBCs were resuspended in PBS and stained with R6G. Green stars represent the R6G.

4.2.2. Raman Spectroscopy

As performed by students at TAMU. A Raman confocal microscope (LabRAM HR Evolution, Horiba, Kyoto, Japan) was used to image the cells with a 532 nm excitation laser focused to a spot size of $\sim 1 \mu\text{m}$. Optical brightfield microscopy images were taken of unstained and R6G-stained RBCs to identify morphological changes from staining. An unaltered RBC typically presents a donut shape (Figure 50B) whereas a stained RBC will lose this shape and become more spherical due to echinocytosis (Figure 50D), which is common in stained bovine blood smears.²³⁴ A representative spectrum for both an unaltered RBC and a R6G-stained RBC were created from the same cells imaged in the brightfield (Figure 50C and E, respectively). These spectra were obtained by integrating over the whole area of the cell with 1 second of exposure time.

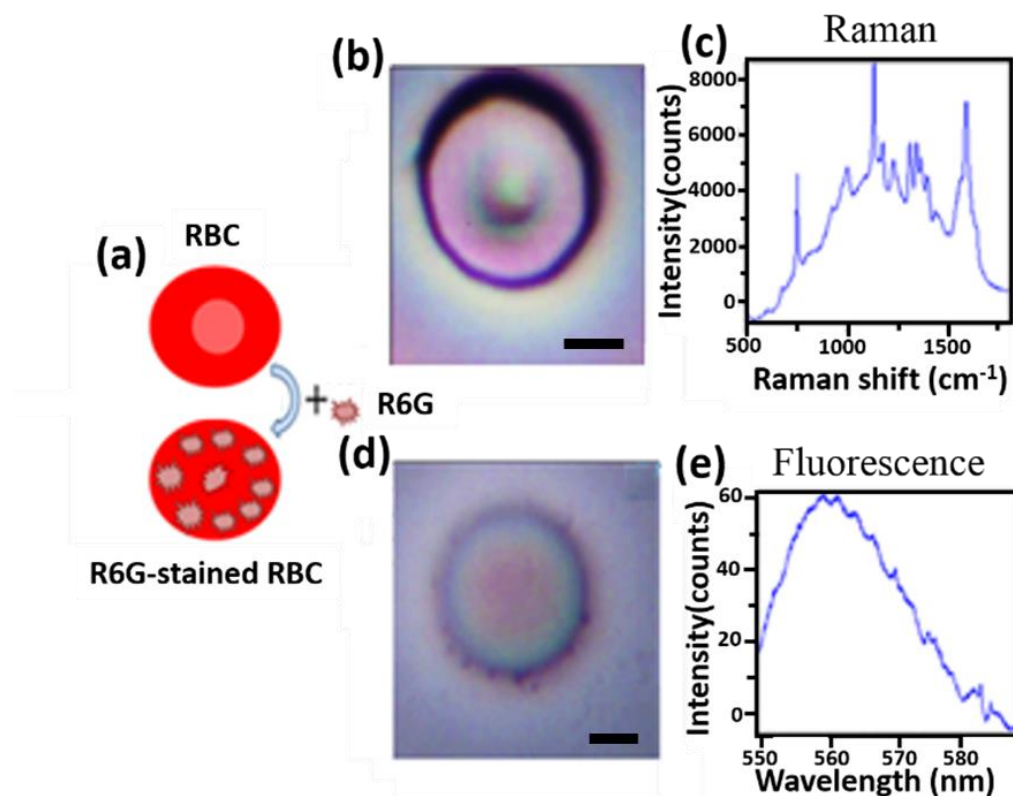


Figure 50. Overview of RBC staining and imaging.

(A) Schematic of the R6G-stained RBC preparation. Optical microscope images of RBC, $8 \times 8 \mu\text{m}$ (B) and R6G-stained RBC, $10 \times 10 \mu\text{m}$ (D) and the corresponding resonance Raman (C) and fluorescence (E) spectra. The spectrum of RBC is dominated by the resonance Raman signal of Hb, whereas the spectrum of R6G-stained RBC is dominated by the fluorescence of R6G (at low laser power). Scale bars represent $2 \mu\text{m}$. Reproduced with the permission of The Royal Society of Chemistry. *Analyst*, 2019, DOI: 10.1039/C9AN00757A – Reproduced by permission of The Royal Society of Chemistry. 235

4.2.3. Data processing algorithm

To separate the overlapping fluorescence and resonance Raman signals of R6G and the RBC, a data processing algorithm was developed (using MATLAB (2015b)), which is included as a MATLAB script in the Appendices (Appendix D). An overview of the data processing algorithm is shown in Figure 51. To reveal the Raman peaks, the raw spectrum data at each pixel was truncated to a region (approximately 20 cm^{-1} on either side of a peak) that encompasses the entirety of a single peak of interest (POI) for fitting. In addition to the 20 cm^{-1} on each side of the peak, an extra 10 cm^{-1} was included on each side to interpolate an accurate representation of any background around the POI such as that caused by neighbouring peaks or broad background signals. Background removal was done after filtering. In fluorescence fitting, it was assumed that there is only a single fluorescence peak in the spectrum. Therefore, the

raw spectrum data did not need to be truncated. If there were multiple fluorescence signals, the truncation function in the software could be used to isolate a single fluorescence peak.

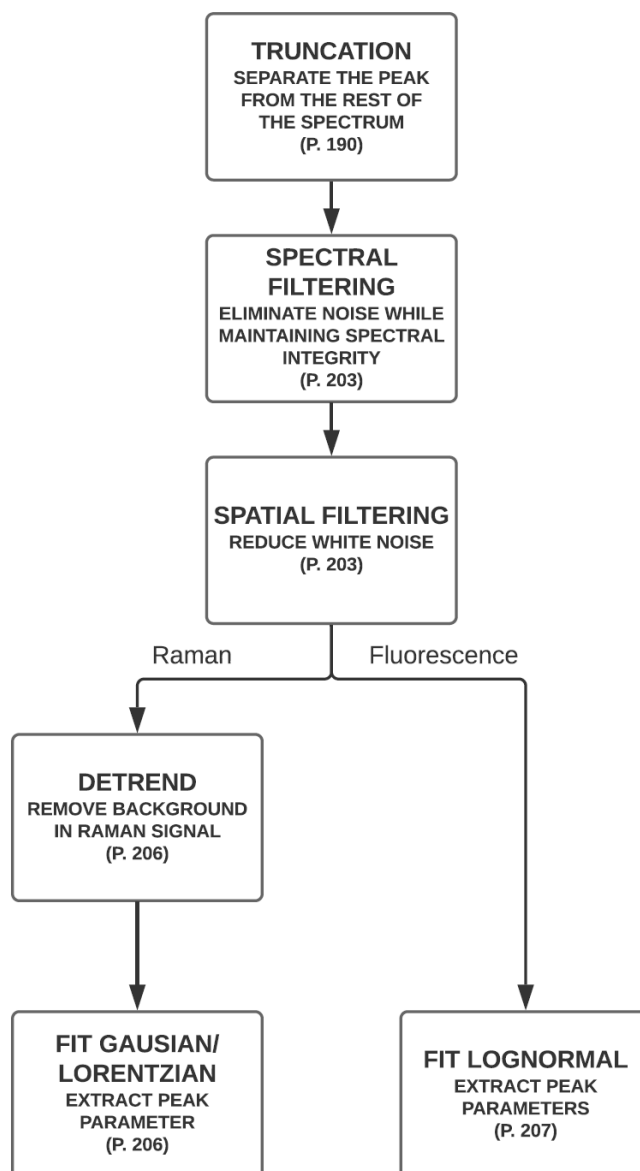


Figure 51. Data analysis process diagram.

Overview of the data processing algorithm for separating and analysing the fluorescence and resonance Raman spectra (page numbers refer to the respective piece of code in APPENDIX D). Reproduced with the permission of The Royal Society of Chemistry. Analyst, 2019, DOI: 10.1039/C9AN00757A – Reproduced by permission of The Royal Society of Chemistry.²³⁵

The truncated spectral data was then filtered with a Savitzky-Golay filter to remove noise while maintaining spectral integrity. Savitzky-Golay filters use a linear least squares polynomial fit of a chosen sized window and shift that window across the data set with the midpoint of each fit used as a new, filtered data point. Although a

boxcar filter's averaging technique would help to remove the high frequency noise, it has a greater chance of levelling narrow peaks. Due to the polynomial nature of the Savitzky-Golay filter, the chance of levelling a peak is reduced ²³⁷ and, hence, Savitzky-Golay filters are used by most spectroscopic software packages. ²³⁸ Many combinations of filter orders and window sizes were considered before choosing a 4th order filter with a 21-point window as the best compromise between the noise reduction, spectral integrity, and spectral feature width.

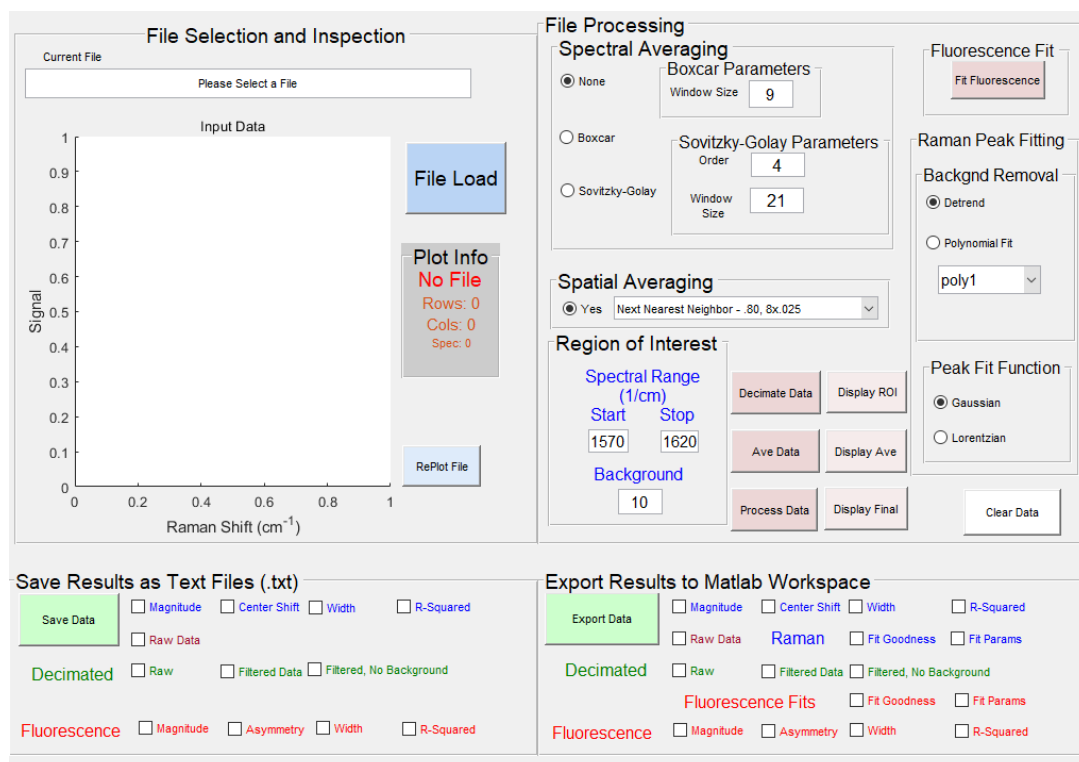
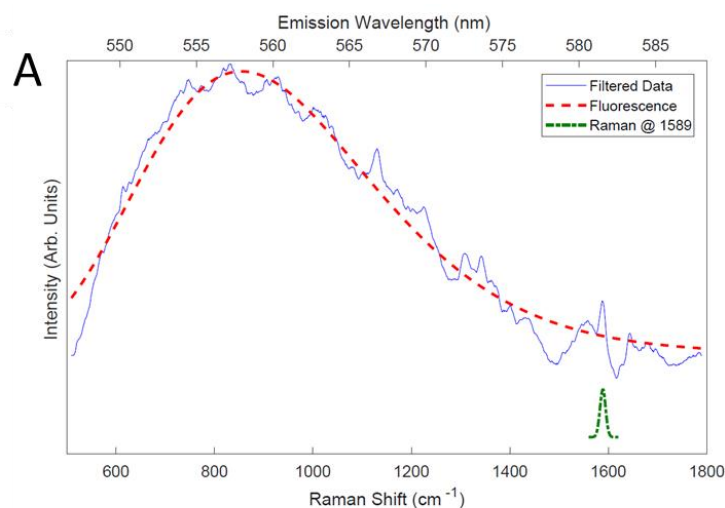


Figure 52. Graphical user interface (GUI) for the analysis software.
This GUI was used to analyse and save all RBC images.

A 3x3 pixel, next-nearest-neighbour spatial filter was then run over the entire image to reduce white noise. This filter weighs each point on the cell with its 8 closest neighbouring points to further reduce noise. Again, multiple variations of the nearest-neighbour (weighted only with the 4 closest neighbours) and next-nearest-neighbour spatial filters were considered. A next-nearest-neighbour filter was chosen as the best method of reducing noise while still maintaining high frequency peaks in the spectrum with the centre weighted at 80% and the surrounding 8 pixels equally weighted at 2.5%.

To remove background from Raman peaks caused by neighbouring Raman peaks and broader fluorescent signals, a linear function was used to fit the background between the combined 20 cm^{-1} of extra points (10 cm^{-1} from both the beginning and end of the data set). This fit was then subtracted from the truncated data set to eliminate any background created by fluorescence or neighbouring Raman peaks. A fitting function was used to extract information about the POI from the spectrum. The data was fit with either a Gaussian or a Lorentzian function and 4 parameters were measured and plotted in the spatial domain: normalized magnitude, centre, width, and R^2 (Figure 53). The “normalized magnitude” plot in Figure 53B shows where the vibrational signals corresponding to a chemical bond associated with the POI are concentrated. The “centre” plot in Figure 53C shows the location of the peak on the spectrum at each pixel. The “width” plot in Figure 53D was used to check for the uniformity of the Raman peaks throughout the cell. The “ R^2 ” plot in Figure 53E shows how well the Gaussian or Lorentzian functions fit the data. According to the R^2 values in this experiment, the Gaussian function was a cleaner fit to the Raman POI. The Raman peak fit at 1589 cm^{-1} is shown in Figure 53.



Raman Data: Spectral Ave: Sovitzky-Golay Ord: 4 Win: 21
 Spatial Ave: NN Uniform, Background: Poly, Fit: Gaussian, Range: 1570 - 1610 cm^{-1}

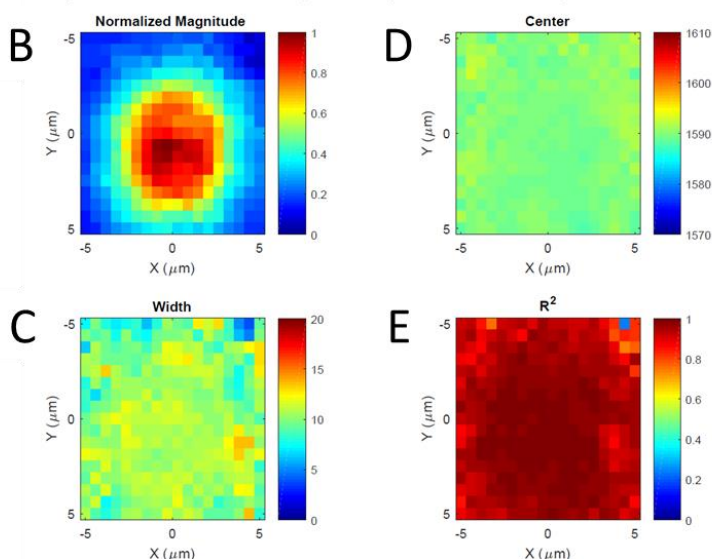


Figure 53. Raman signal extraction and fitting functions.

A) Simultaneous Raman scattering and fluorescence signal (blue) from an R6G-stained RBC (exposed to 532 nm laser at 0.2 mW for 0.1 seconds) with overlaid fitting curves of fluorescence at 555 nm (dashed red) and Raman scattering signal at 1589 cm^{-1} (dash-dotted green). B – E) the corresponding examples of the data processing software outputs identifying a Gaussian fit to a Raman peak around 1589 cm^{-1} . Reproduced with the permission of The Royal Society of Chemistry. Analyst, 2019, DOI: 10.1039/C9AN00757A – Reproduced by permission of The Royal Society of Chemistry.²³⁵

To eliminate fits that did not correspond to Raman peaks, the width and R^2 values were checked at each pixel to ensure a narrow, well-fit Raman peak. If either the value of R^2 was less than 0.2 or the width value was outside of the truncated boundaries, all four parameters at that pixel were set to zero (Figure 54A – C). When looking for a fluorescence peak, the fluorescence signal is much wider than any individual Raman peak and therefore, the impact of the Raman signal on the fluorescence fit is negligible. Also, since the entirety of the fluorescence signal is not

always collected in the range of this spectrum, subtracting the background has a chance of skewing the results. Therefore, no background was subtracted before fitting the data to a lognormal function,

$$\epsilon(\tilde{\nu}) = \frac{\epsilon_0 b}{\tilde{\nu} - a} \exp(-c^2) \exp \left\{ -\frac{1}{2c^2} \left[\ln \left(\frac{\tilde{\nu} - a}{b} \right) \right]^2 \right\}; \quad \text{Equation 11}$$

$$\tilde{\nu} > a,$$

where a, b, and c are the parameters related to the position of the mode, $\tilde{\nu}_0$ (cm^{-1}), the half-width, H (cm^{-1}), and the empirical measure of the skewness, ρ , by the equations,

$$c = \frac{\ln \rho}{\sqrt{2 \ln 2}}, \quad \text{Equation 12}$$

$$b = H \frac{\rho}{\rho^2 - 1} \exp(c^2), \quad \text{Equation 13}$$

$$a = \tilde{\nu}_0 - H \frac{\rho}{\rho^2 - 1}. \quad \text{Equation 14}$$

Siano et al., among others, observed that a lognormal function is a better fit for the fluorescence spectrum than a Gaussian function due to its ability to mimic the Gaussian while also allowing for asymmetric shapes.²³⁹ They were also able to use the function in Equation 11 to separate multiple fluorescent signals from one another when found in the same spectrum. Similar to the Raman image plots, the fluorescence image plots are composed of four subplots: normalized fluorescence intensity, asymmetry, width, and R^2 . If there is no fluorescence signal in the spectrum, an offset in magnitude of the filtered data would be likely where there is a significant contribution of the Raman signal. This is likely to occur because the background is not removed from the fluorescence data causing the “normalized fluorescence” plot to show a clear image of the cell even though the R^2 values are near zero. Therefore, if the R^2 value was less than 0.2, all the parameters associated with that pixel were set to zero (Figure 54D – F). This reduces the chances of a false positive identification of fluorescence in a spectrum. The “asymmetry” and “width” plots were both used to check uniformity across the cell. The R^2 plot was used to identify how well the lognormal curve was fit to the data. A typical fluorescence peak fit is shown in Figure

53A. The variables a , b , c , and ϵ_0 were all free coefficients that MATLAB could adjust to fit the fluorescence peak.

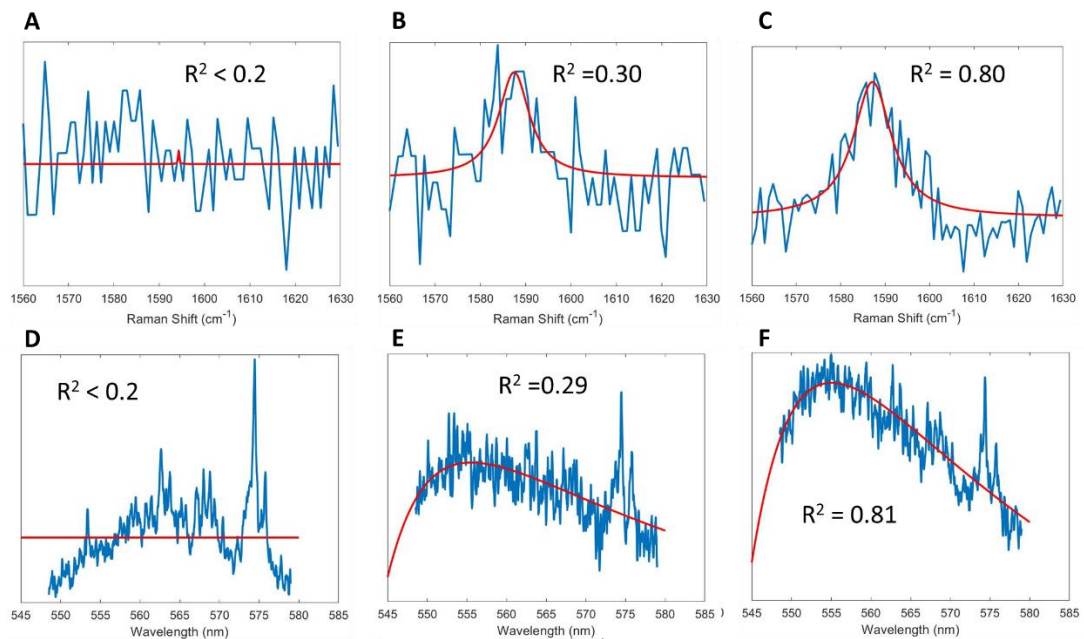


Figure 54. Comparison of R^2 values for Raman and fluorescence signals.

Any Raman fits with an R^2 value below 0.2 (A) will have all parameters at that pixel set to zero. Peaks with R^2 values of 0.30 (B) or 0.80 (C) can still be distinguished by eye and are therefore included in the study. Any fluorescence fits with an R^2 value below 0.2 (D) will have all parameters at that pixel set to zero. Peaks with R^2 values of 0.29 (E) and 0.81 (F) can still be distinguished by eye and are therefore included in the study. Rd lines indicate the best fit line for the sections shown. A-C use a Lorentzian fit function, D-F use a lognormal fit function.

The filtered data from the fluorescence processing was also used to identify the pixels that encompassed the surface of the RBC, from which a spatially averaged spectrum for each cell was generated (Figure 58M – R). The Raman and fluorescence images were used to identify where the cell boundaries lie in terms of the pixels. Then, a rectangular area of the pixels that reside entirely within the cell were summed and averaged to create a spectrum plot that is representative of the cell.

4.3. Results

4.3.1. Filter optimization

Choosing the correct filter can have a large influence on the sensitivity of the system. For Raman peaks, it is important to improve the signal-to-noise ratio (SNR) without losing the resolution on the peaks. By utilizing a Savitzky-Golay filter instead of a low-pass box car filter, the sharpness of the peaks was maintained (Figure 55).

Each coloured curve represents the collected spectrum at a single pixel associated with a location on the surface of the RBC. Only 9 pixels from the cells surface are represented in Figure 55 for clarity. The Raman shift of different compounds have the potential to be very close together and so if a filter widens a peak too much, information could be lost as the surrounding peaks coalesce. To compare the two, a 4th order Savitzky-Golay filter with a window size of 21 cm⁻¹ was compared to a boxcar filter with a window size of 9 cm⁻¹. Both filters reduced the background noise to a similar degree but the Savitzky-Golay filter is seen to maintain the sharpness of its peaks. This can be observed in the FWHM of the main 1589 cm⁻¹ peak and the dampening of the smaller peaks surrounding it after filtering. The boxcar filter (Figure 55B) produces an average FWHM of 5.9 cm⁻¹ as compared to 5.0 cm⁻¹ for the Savitzky-Golay filter (Figure 55C).

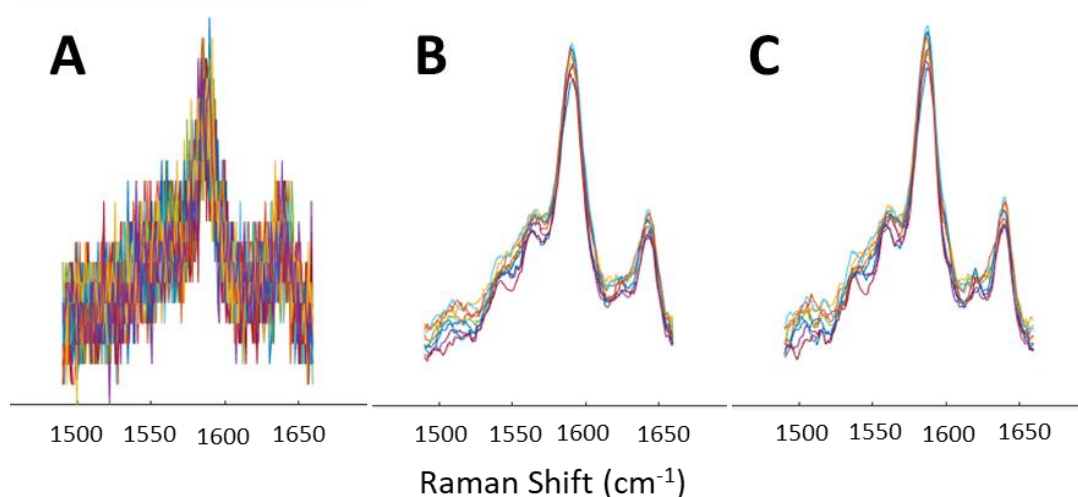


Figure 55. Comparison of spectral filters.

Raman signals are not apparent in the unfiltered signal (A). A Savitzky-Golay filter (C) was chosen over a box car filter (B) because of its ability to maintain high frequency peaks while improving the SNR. Coloured lines represent the spectrums for each separate pixel. Only 9 pixels are represented here for clarity.

A similar phenomenon occurs when the Savitzky-Golay filter parameters shift (Figure 56). When the window size is small, a large proportion of high frequency noise continues to pass through the filter. As the window size increases, the filter lets through less and less high frequency noise until eventually it begins to act like a box filter, removing all the high frequency components from the signal and widens the peaks. A low order Savitzky-Golay filter will have more low pass filter properties than a high order filter. In Figure 56, when a 2nd order filter with a window size of 41 cm⁻¹ is run

over the spectrum, the FWHM of the fit function around the 1589 cm^{-1} peak increases from 8.0 to 11.1 cm^{-1} , demonstrating that a low order filter with a large window size will eliminate all high frequency noise at the expense of spectral resolution. When an 8th order filter with a window size of 11 cm^{-1} is run over the spectrum, the FWHM of the fit function around the 1589 cm^{-1} peak remains constant but the R^2 value decreases from 0.86 to 0.68 , demonstrating that a high order filter with a small window size will lessen the credibility of the software. In some scenarios, the overall effect of the filter can be very similar, such as when a 4th order, 11 cm^{-1} filter is compared with an 8th order, 21 cm^{-1} filter. The rationale behind this is that by doubling the size of the filtering window while also doubling the size of the order of the filter allows the same ratio of local maximum and local minimums per cm^{-1} . These filters will act similarly in situations with lower frequency signals but change as the frequency of the signal increases. By compromising between the removal of high frequency noise and the dampening of sharp peaks, a 4th order filter with a window size of 21 cm^{-1} was selected for this study.

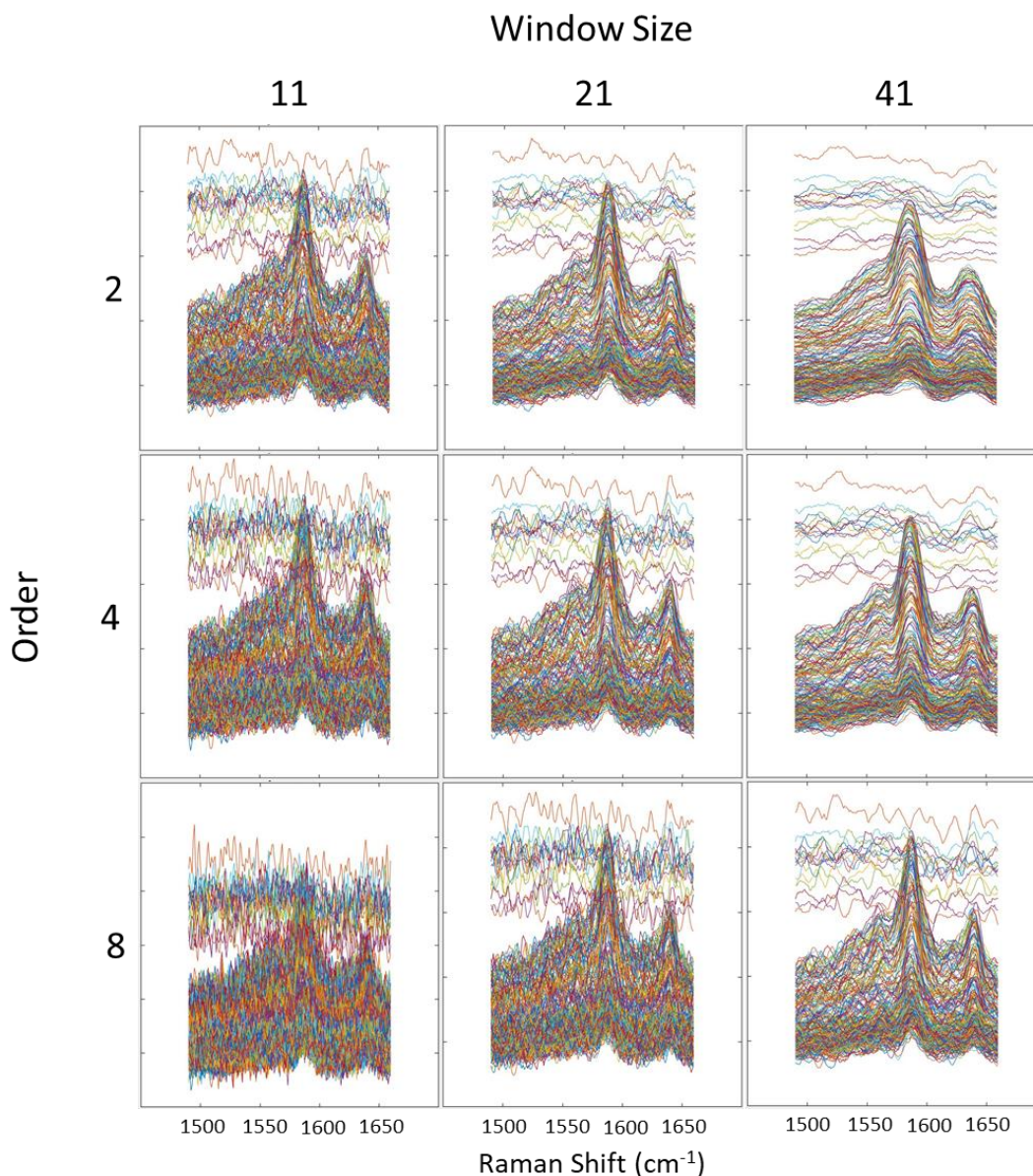


Figure 56. Comparison of Savitzky-Golay filter parameters.

As the order of the Savitzky-Golay filter increases, it allows more high frequency signals to pass. As the window size increases, the peaks begin to flatten. Coloured lines represent the spectrums for each separate pixel. Flat lines are spectra from pixels that were not on the RBC.

Spatial filters are also employed to help eliminate extraneous background noise and increase the SNR (Figure 57). Nearest neighbour and next nearest neighbour spatial filters did not show a distinguishable difference in this study, only observing a change from 0.86 to 0.89 in the R^2 values from fitting the 1589 cm^{-1} peak, respectively. The variation between the two systems was mainly seen between different weighting strategies. When the spatial filter gave uniform weight to all neighbours, a large portion of the random noise was removed while still maintaining the prominent peak

shapes, improving the R^2 value of the fit around the 1589 cm^{-1} peak from 0.77 to 0.86 with the nearest neighbour filter and from 0.78 to 0.89 with the inclusion of the next nearest neighbours. This filtering approach is adept at removing high frequency noise in systems where there is a constant concentration of the compound of interest across the entire cell. If relied upon too heavily, variations across the cell could be dampened and go undiscovered. For this reason, a next nearest neighbour filter, weighted 80% to the pixel of interest and 2.5% to the neighbouring 8 pixels, was adopted. This marginally boosted the SNR while also allowing variation to be discovered around the cell.

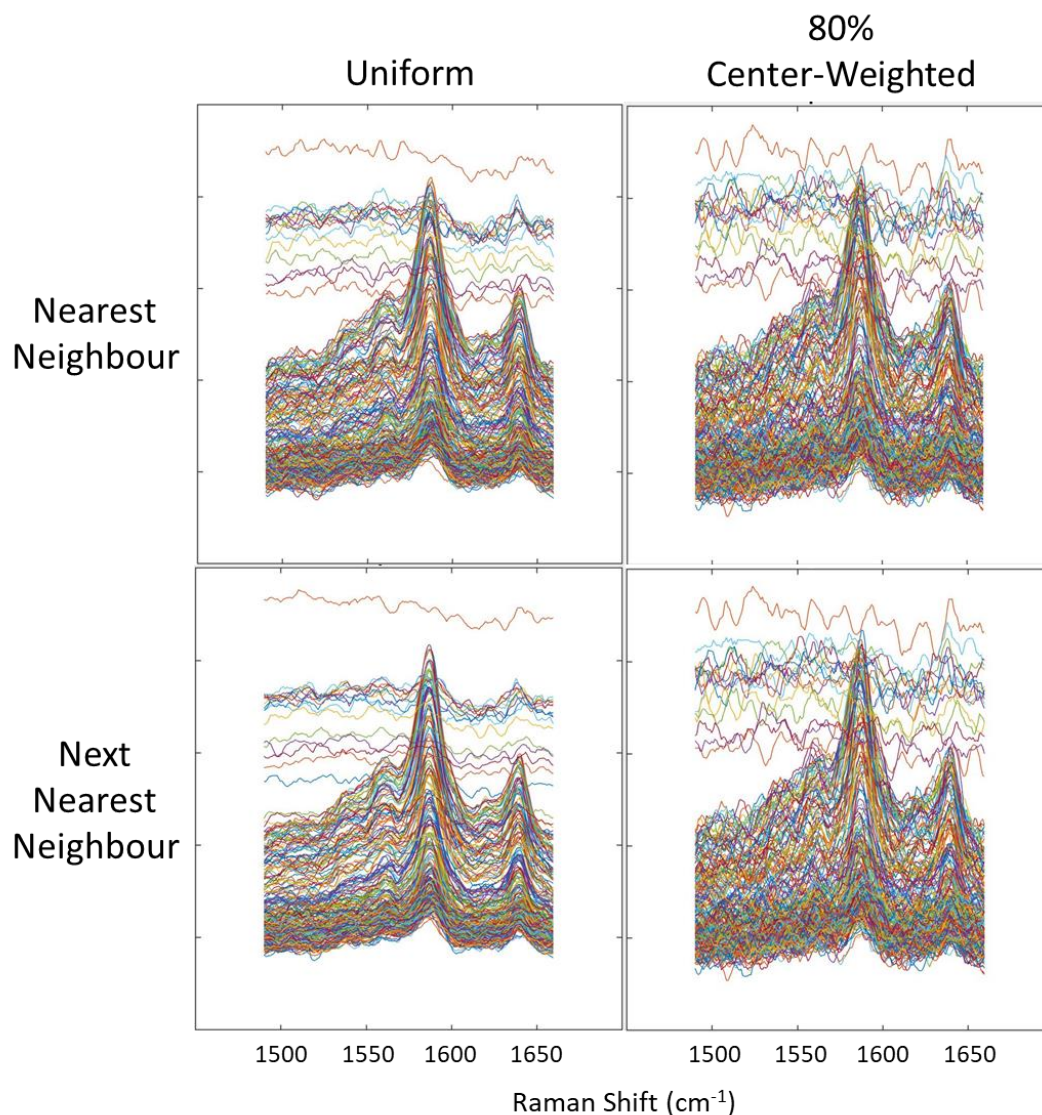


Figure 57. Comparison of spatial filters.

Uniform spatial filters can reduce unique peaks that may be present in one part of the cell but not another, therefore reducing the resolution of the system. By expanding from nearest neighbour to next nearest neighbour, the chances of an individual having a large effect on its neighbours is reduced while some extraneous noise is removed. Coloured lines represent the spectrums for each separate pixel. Flat lines are spectra from pixels that were not on the RBC.

4.3.2. Photobleaching unveils Raman signal

To investigate the effects of the laser power on imaging and photodamage, 1-week old cells were selected to suppress the effects of early aging during the imaging process. Due to the relative weakness of the Raman signals, the complete acquisition of all the spectroscopic images for one cell using all the different laser powers takes several hours. Therefore, it is desirable to select a cell which has already been aged for several days to eliminate the effects of early cell aging and degradation. To evaluate

the photostability of the stained cells, spectroscopic measurements were performed on 1-week old, R6G-stained RBCs using 532 nm laser excitation. The combined fluorescence and resonance Raman spectra of each cell were obtained with 1 second accumulation time at 0.2 μ W, 2 μ W, 20 μ W, 75 μ W, 0.2 mW, and 0.3 mW laser powers consecutively on the same cell (Figure 58). Due to the consecutive nature of these experiments on the same cell, the total, cumulative energy that the cells were subjected to was larger than the dose from the current laser power (

Table 6). The spectra were integrated over the whole cell and show a large contribution of the R6G fluorescence at low laser powers. As the laser power increased, the fluorescence was gradually photobleached and the Hb and R6G Raman signals started to become more prevalent (Figure 58M – R). The corresponding fluorescence (555 nm) and Raman (1589 cm^{-1} band) images (Figure 58 A – F and Figure 58 G – L, respectively) show the normalized magnitude of their respective peak at each pixel across the cell. These images reveal the anti-correlated dependence of the two signals on laser power. Both the fluorescence and the Raman cell images are clearly recognizable in the 75 μ W laser power with 1 second accumulation time (Figure 58D and J), providing the optimal compromise of SNR for the two signals. Therefore, a fresh, unstained RBC at 75 μ W laser power (Figure 59) was chosen as the control for the data analysis throughout this paper. All reported ratios are in comparison to this control.

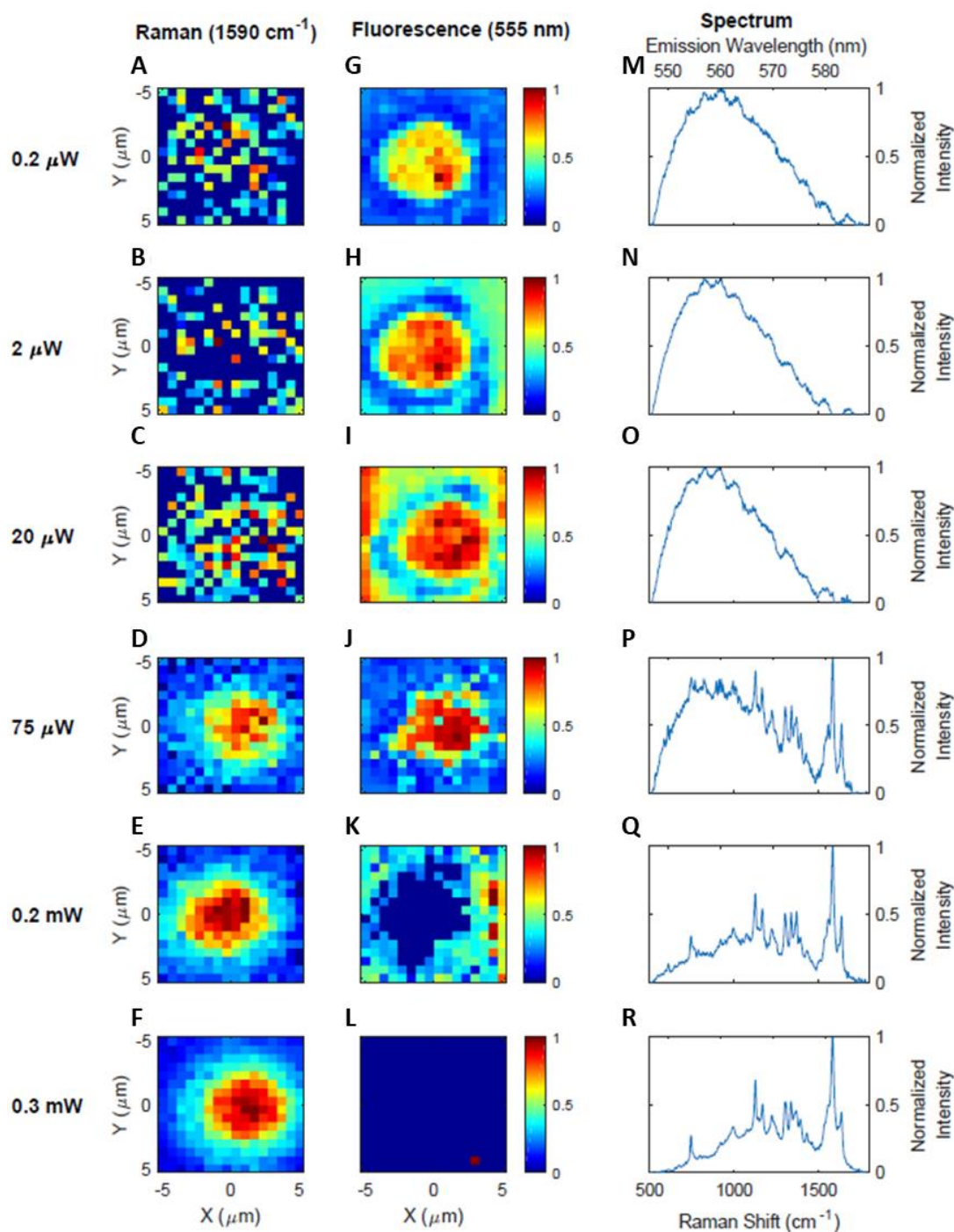


Figure 58. Photostability of the R6G fluorescence.

Resonance Raman (A – F) and fluorescence (G – L) images and their corresponding spectra (M – R) of the 1-week old, R6G-stained RBC. The images were obtained by fitting a gaussian curve to the 1589 cm^{-1} Raman band of Hb in (A – F) and fitting a lognormal curve to the 555 nm fluorescence band of R6G in (G – L) at each pixel and displaying the normalized peak magnitude of each fit as a colour map. The spectra and images show the gradual transition between the fluorescence and Raman signals as a function of the laser power that was varied from $0.2\ \mu\text{W}$ to $0.3\ \text{mW}$ on the sample.

Reproduced with the permission of The Royal Society of Chemistry. *Analyst*, 2019, DOI: 10.1039/C9AN00757A – Reproduced by permission of The Royal Society of Chemistry.²³⁵

Table 6. Cumulative dose at each power level.

Each cell was consecutively exposed to lasers of increasing power creating a larger energy dose per cell.

Laser Power (μW)	Cumulative Dose (J/cm^2)
0.2	20
2	220
20	2,220
75	9,720
200	29,720
300	59,720

The spectrum and images at the 75 μW laser power show clear signatures of the broad R6G fluorescence band at 555 nm and narrow Raman peaks of Hb (Figure 58D, J, and P). For higher laser powers, the fluorescence signal of R6G is photobleached (Figure 58Q and R) and the Raman signals dominate. The spectral analysis and band assignment were performed based on the previous work performed on RBCs²⁴⁰⁻²⁴³ and R6G.²⁴⁴ Raman bands corresponding to 17 unique vibrational transitions were identified.

Table 7 shows the measured Raman shifts and band assignments for the Raman spectra of the unstained RBC control at 75 μW laser power, the fresh, R6G-stained RBC at 75 μW and 0.3 mW, and the 1-week old, R6G-stained RBC at 75 μW , 0.3 mW, and 0.9 mW. These Raman shifts were obtained by the fitting procedure described above in the data analysis section.

Table 7 also shows that the Raman shifts fluctuate within several cm^{-1} of their true centre value. These small aberrations are likely due to overlap between neighbouring signals and the fluorescence background. However, the relative separation between the vibrational transitions is on the order of tens of cm^{-1} which allows them to be clearly distinguishable.

Table 7. Raman shifts and band assignments.

Raman shifts (in cm^{-1}) and band assignments of the unstained RBC control at 75 μW laser power (control), the fresh, R6G-stained RBC at 75 μW and 0.3 mW, and the 1-week old, R6G-stained RBC at 75 μW , 0.3 mW, and 0.9 mW. (Abbreviations: v, ip: in plane mode, Y, op: out of plane mode, str: stretching). Reproduced with the permission of The Royal Society of Chemistry. *Analyst*, 2019, DOI: 10.1039/C9AN00757A – Reproduced by permission of The Royal Society of Chemistry. ²³⁵

Peak no.	Unstained RBC at 75 μW (control)	Fresh, R6G-stained RBC at 75 μW	Fresh, R6G-stained RBC at 0.3 mW	1-week old, R6G-stained RBC at 75 μW	1-week old, R6G-stained RBC at 0.3 mW	1-week old, R6G-stained RBC at 0.9 mW	RBC band assignment	R6G band assignment
1	748	749	749	747	747	747	ν_{15} , Υ_1	--
2	760	764	762	760	760	759	ν_{15} , Υ_1	--
3	768	768	774	771	773	771	--	C-H op bend
4	1129	1131	1132	1129	1128	1129	ν_5	C-H ip bend
5	1144	1148	1146	1145	1146	1146	ν_{14}	--
6	1156	1165	1156	1154	1157	1165	ν_{44}	--
7	1173	1173	1172	1170	1171	1172	ν_{30}	--
8	1307	1312	1311	1308	1307	1308	ν_{21}	Aromatic C-C str
9	1339	1346	1344	1343	1340	1341	ν_{41}	--
10	1360	1368	1364	1360	1360	1360	ν_4^d	Aromatic C-C str
11	1374	1380	1376	1373	1374	1372	ν_4	--
12	1395	1401	1399	1397	1395	1396	ν_{20}	--
13	1544	1547	1549	1546	1544	1545	ν_{11}	--
14	1559	1565	1564	1563	1561	1560	ν_2	--
15	1587	1590	1589	1587	1586	1586	ν_{37}	--
16	1625	1625	1625	1625	1625	1625	ν_{10} , ν (C=C)	--
17	1641	1641	1643	1640	1640	1640	--	Aromatic C-C str

4.3.3. R6G is successfully attached to RBCs

Figure 59 shows a comparison of the Raman spectra of the unstained RBC control with the fresh R6G-stained RBC at 75 μW laser power. All spectral band intensities were normalized to the maximum of the 1589 cm^{-1} band, which was the highest intensity band observed in all the spectra in this work. The signals of the R6G-stained RBC show strong bands at 770 cm^{-1} (peak 3) and 1645 cm^{-1} (peak 17) relative

to their neighbouring peaks. These peaks are attributed to the C-H out-of-plane mode bending and the aromatic C-C stretching of the R6G, respectively. The ratios of the other bands attributed to an RBC in the 1500 – 1650 cm^{-1} spectral range do not noticeably change, instead remaining at a ratio very close to 1:1, indicating the R6G staining does not adversely affect the Raman spectra of the RBC.²⁴⁰

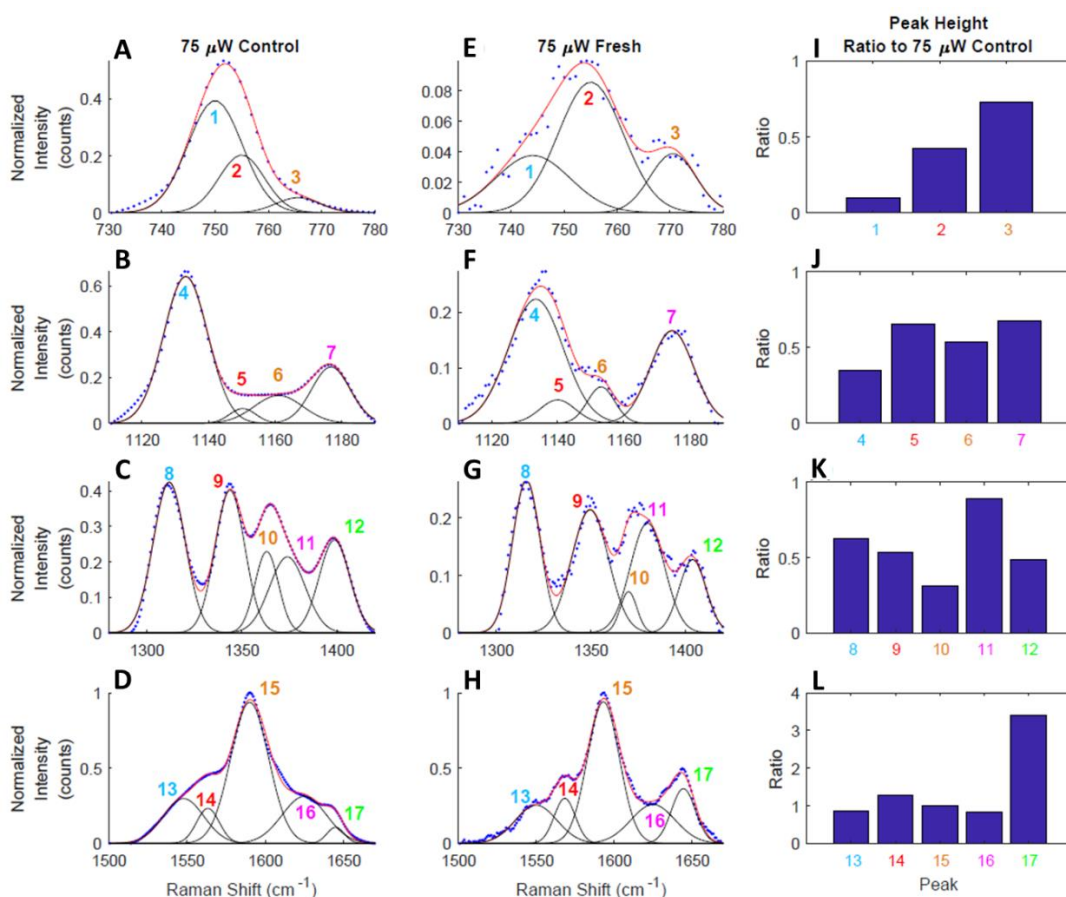


Figure 59. RBCs with and without R6G.

Raman spectra of the unstained RBC control (A – D) and fresh R6G-stained RBC (E – H) normalized to the intensity of the 1589 cm^{-1} band at 75 μW laser power show the absence of the significant staining effects on RBC. The observed strong intensities of bands 3 and 17 are attributed to the resonance Raman signals of R6G. The blue dots represent the filtered intensity data, the red line represents the best-fit of the data, and the black gaussians represent the components that sum together to create the red, best-fit line. Peak height ratios of the normalized peak intensities of the fresh R6G-stained RBC to the normalized peak intensities of the unstained RBC control at 75 μW were computed (I – L). Reproduced with the permission of The Royal Society of Chemistry. Analyst, 2019, DOI: 10.1039/C9AN00757A – Reproduced by permission of The Royal Society of Chemistry.²³⁵

4.3.4. Aging may reduce the intensity of the Raman signal

Figure 60 presents the Raman spectral analysis of aging effects on the R6G-stained RBC samples at 0.3 mW excitation. The 0.3 mW laser power was used to lessen the influence of the fluorescence peak via photobleaching and only characterize

the ageing effects on the Raman signals. All spectral band intensities were normalized to the maximum intensity of the 1589 cm^{-1} band. No significant changes were observed in the Raman spectra of the 1-week old, R6G-stained RBC compared to the fresh R6G-stained RBC in the 1500 – 1650 cm^{-1} spectral range, except for a small reduction of the intensity of the R6G band 17. However, changes in the relative ratios of the bands in the other three spectral ranges, namely, the 730 – 780 cm^{-1} (Figure 60A and E), 1120 – 1180 cm^{-1} (Figure 60B and F), and 1300 – 1400 cm^{-1} (Figure 60C and G) ranges were observed to mainly increase relative to the unstained RBC control at 75 μW (Figure 60I – K). Since there is no fluorescence to bleach in the unstained RBC, it works well as a control for this experiment. The only two peaks that showed a decrease in their ratio were peak 3 and 17 which were earlier identified as R6G specific peaks. It is possible that resuspending the RBCs and storing them for an extra week created an extra-long washing step that the fresh cells weren't subjected to, leading to a decrease in the amount of R6G on these RBCs. This could also lead to a decrease in the overall fluorescence of the system, allowing the Raman peaks of the RBC to become more pronounced in the regions that heavily overlap with the fluorescence signal. This would also explain why there is very little change in the 1500-1650 cm^{-1} region, as the effect of the fluorescence is already limited in that region.

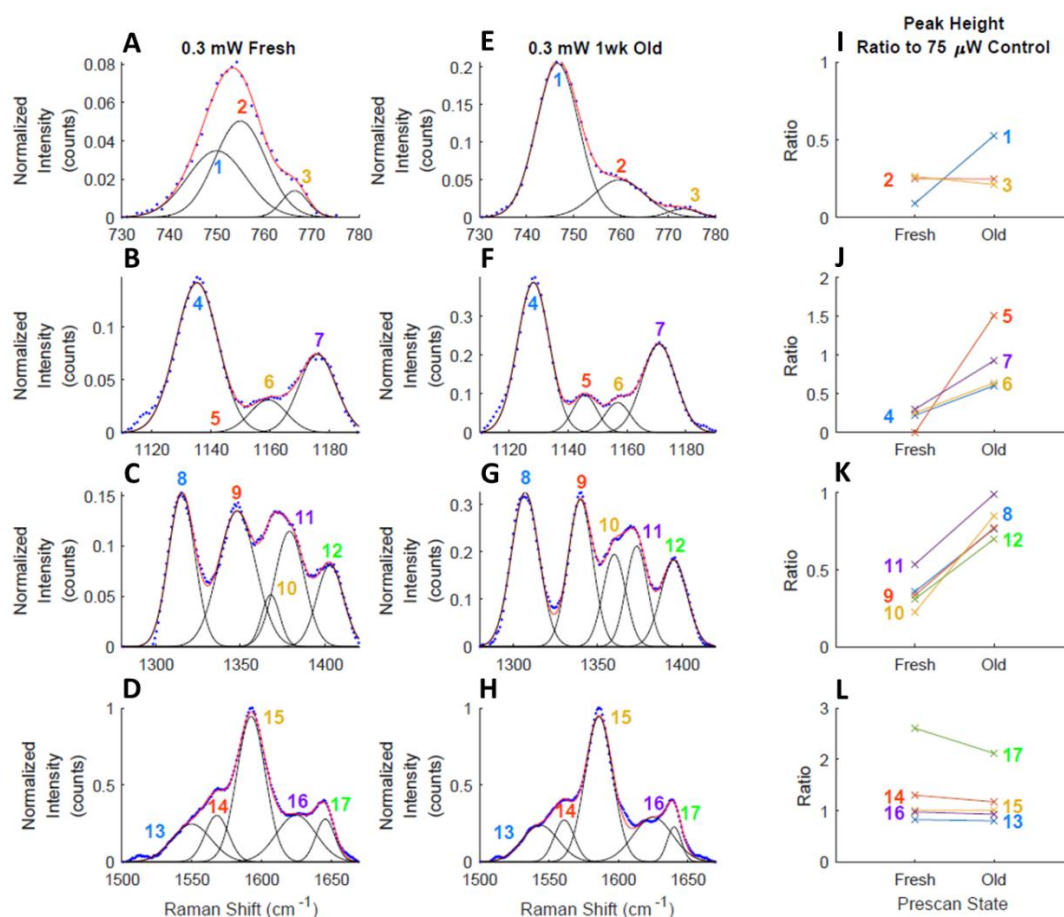


Figure 60. Fresh and 1-week old, R6G stained RBCs compared to unstained control.

Raman spectra of the fresh (A – D) and 1-week old (E – H), R6G-stained RBCs at 0.3 mW laser power normalized to the intensity of the 1589 cm^{-1} band show the absence of relative differences to their neighbouring peaks due to aging of the stained RBCs. 0.3 mW laser power was used to lessen the influence of the fluorescence peak and only characterize the ageing effects on the Raman signals. The blue dots represent the filtered intensity data, the red line represents the fit of the data, and the black gaussians represent the components that sum together to create the red, best-fit line. Peak height ratios of the normalized peak intensities of the fresh and 1-week old, R6G-stained RBCs at 0.3 mW laser power to the corresponding normalized peak intensities of the unstained RBC control at 75 μW (Figure 59A – D) show a trend for RBC Raman peaks to become more defined at lower Raman shift levels after aging (I – L). Reproduced with the permission of The Royal Society of Chemistry. Analyst, 2019, DOI: 10.1039/C9AN00757A – Reproduced by permission of The Royal Society of Chemistry.²³⁵

4.3.5. R6G Raman signal shows signs of photodamage at high laser intensities

Figure 61 shows the Raman spectra of the 1-week old, R6G-stained RBC obtained at 75 μW (Figure 61A – D) and 0.9 mW (Figure 61E – H) laser powers, and the ratios of all tested laser powers to the unstained RBC control at 75 μW (Figure 61I – L). All spectral band intensities were normalized to the maximum intensity of the 1589 cm^{-1} band. The spectra show a steady decrease in the relative Raman band intensities of the R6G bands 3 (at 770 cm^{-1}) and 17 (1645 cm^{-1}) relative to the unstained

RBC control (Figure 61I and L, respectively) as the laser power increases. As hypothesized previously, this may be attributed to the photobleaching of the R6G fluorescence and resonance Raman signals. The relative ratios of the other RBC bands do not change drastically with laser power, which reveals the photostability of the RBC Raman signals even after R6G-staining and consequent bleaching of the R6G signals.

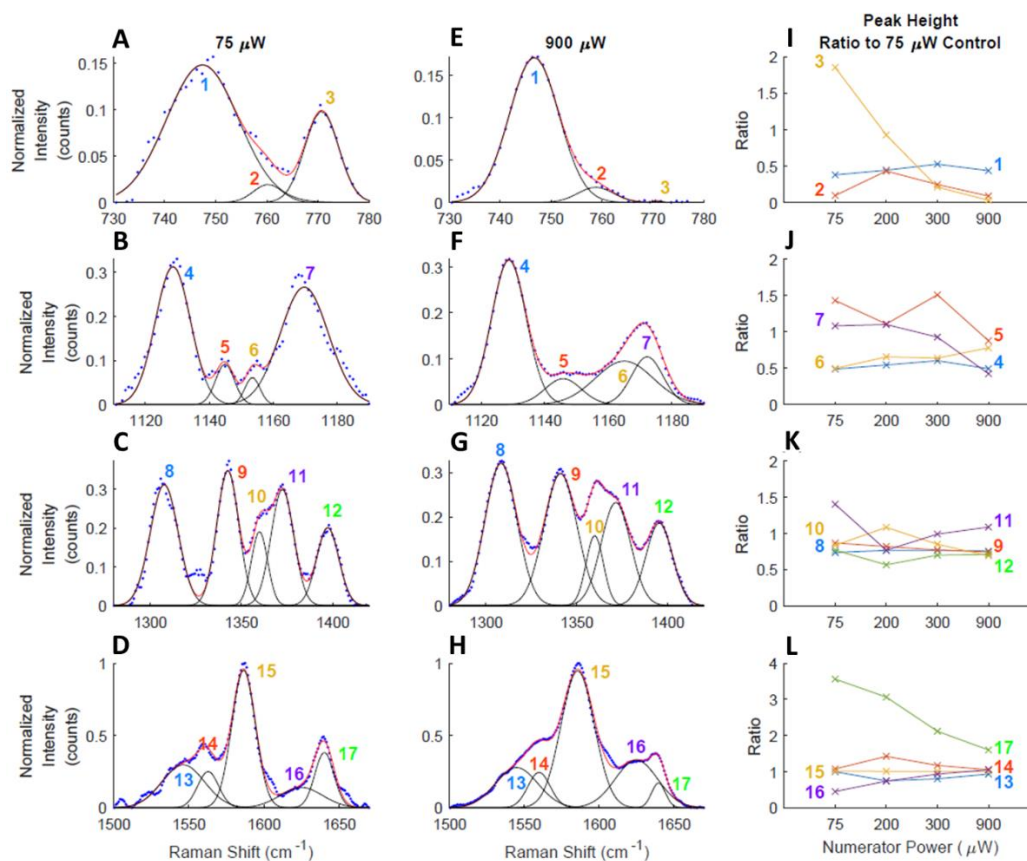


Figure 61. 1-week old, R6G-stained RBCs at varying power levels.

Raman spectra of the 1-week old, R6G-stained RBC at 75 μW (A – D) and at 0.9 mW (E – H) laser powers normalized to the intensity of the 1589 cm⁻¹ band show the absence of the significant laser power effects on the Hb signals of RBC and the significant spectral differences of the bands 3 and 17 due to the photobleaching of R6G fluorescence and resonance Raman. The blue dots represent the filtered intensity data, the red line represents the fit of the data, and the black gaussians represent the components that sum together to create the red, best-fit line. Peak height ratios of the normalized peak intensities of the 1-week old, R6G-stained RBC at different laser powers to the normalized peak intensities of the unstained RBC control at 75 μW (Figure 59) show the gradual photobleaching of peaks 3 and 17 (I – L). Reproduced with the permission of The Royal Society of Chemistry. *Analyst*, 2019, DOI: 10.1039/C9AN00757A – Reproduced by permission of The Royal Society of Chemistry. 235

4.4. Discussion

When comparing the Raman signals between the stained and unstained cells (Figure 59), fresh and aged cells (Figure 60), and among the different laser powers (Figure 61), care must be taken for the quantitative analysis of the peak height ratios.

For example, the overall shapes of the Raman spectra of the stained and the unstained RBCs in Figure 59 are similar apart from peaks 3 and 17, which are attributed almost exclusively to R6G. The four strong Raman peaks (13 – 16 in the 1500 cm^{-1} – 1650 cm^{-1} spectral range) do not show any significant changes with a peak ratio of ~ 1 . However, the peak height ratios of the other peaks in the 1120 cm^{-1} – 1180 cm^{-1} and 1300 cm^{-1} – 1400 cm^{-1} ranges are approximately two times smaller, and the peak height ratios are even smaller in the 730 cm^{-1} – 780 cm^{-1} range. These peak height ratio variations are attributed to the decrease of the SNR of the Raman signals extracted from the large background fluorescence. Indeed, the overlap between the Raman and fluorescence signals is strongest for the 730 cm^{-1} – 780 cm^{-1} range and weakest for the 1500 cm^{-1} – 1650 cm^{-1} spectral range, as can be seen in Figure 53A. Therefore, it is concluded that the peak height ratios provide the most useful information when they are compared with neighbouring peaks in the same spectral range. The quantitative comparison of the peaks located in different spectral ranges should be limited and performed with care when a large contribution of background fluorescence is present.

Similar effects are observed by comparing Raman spectra of the fresh and 1-week old, R6G-stained RBCs in Figure 60. The overall spectral shape in all the spectral ranges remain consistent even though the peak heights vary. All peaks in the 1500 cm^{-1} – 1650 cm^{-1} range maintain their peak ratio, except peak 17 which decreases slightly, probably due to a slow desorption of R6G during aging. Contrary to the results in Figure 59, most of the peaks in the 1120 cm^{-1} – 1180 cm^{-1} and 1300 cm^{-1} – 1400 cm^{-1} ranges in Figure 60 become approximately two times stronger but do not change their relative peak ratios. This agrees with the interpretation of the partial masking of the Raman signals by the fluorescence background from R6G. During the cell aging, R6G partially desorbs, reducing the amount of fluorescence and, therefore, leads to the increase in the measured Raman signals of RBC.

In summary, simultaneous spectroscopic bioimaging and photostability analysis of R6G stained RBCs was performed using fluorescence and resonance Raman scattering of 532 nm laser excitation. A data processing algorithm was developed for separating these signals and as a result, observed no significant photodamage of RBC under the complete photobleaching of R6G. Excitation laser power dependence was explored and spectral ranges were identified which provide

reliable spectroscopic signatures of RBC photodamage and R6G bleaching. This general approach may be used to investigate the photostability of different staining molecules by matching the corresponding excitation laser frequency or even applied to more complex systems such as leukocytes, cancer, and other types of cells to investigate the effects of cellular staining. Additionally, this approach can be extended to coherent Raman spectroscopy, such as coherent anti-Stokes Raman scattering (CARS), which will allow for the fast imaging of cells in a few seconds time which would decrease the effects of photobleaching. Finally, this approach may also be used to investigate photostability effects in cells and biomolecules using surface-enhanced Raman scattering (SERS) and surface-enhanced fluorescence (SEF) spectroscopies as well as their interplay, as these techniques are based on strong electric fields generated by the plasmonic nanoparticles.

5. SUMMARY OF CONCLUSIONS

5.1. PSM Adhesion Kinetics

In this work, the rolling speed and the adhesion kinetics of discoidal porous silicon microparticles (PSM) was investigated on endothelial cells by replicating the average shear rates seen in tumour microvasculature. As an initial study, static studies observed the effects of Brownian motion on the PSM. Without steric interactions created by either the gelatine basement membrane or the cells, particle motion was wholly subdued. Once steric interactions were introduced, PSM modified with the E-selectin binding specific ESTA aptamer (E-PSM) consistently moved a smaller distance from their starting position than the SCR or the APTES modified PSM, indicating that attractive forces were at play causing a reduction in their overall displacement. When PSM were introduced to cells at a physiologically representative shear rate in a flow chamber, the E-PSM showed a propensity to roll slower on the surface of the cells than the other two PSM modification groups. Although when the total adhesion of PSM was considered, the APTES modified PSM were the most likely to stop and adhere to the cell surface. It is possible that this contradiction is the result of an electrical repulsion between the aptamer modified particles and the cell surface preventing the PSM from fully attaching to the E-selectin on the cell surface. Since the aptamer is conjugated to the APTES, another possibility could be that the linker arm between the PSM and the cell is long enough to allow flow to get between them and pry the PSM off.

The development of this particle tracking software adds another tool to the biologist toolkit when studying drug delivery and attachment kinetics in flow. As previously discussed in the introduction, research is becoming more and more specialized as new methodologies are imagined and introduced into literature and common practice. By making this well commented software free and publicly available via GitHub, future scientists with a small amount of computer science understanding will be able to utilize it as is or adjust pieces as needed to fit their desired research analysis outcomes.

Continuing efforts for this study have many avenues to still be pursued, especially because the experiments collected here do not seem to coincide with the

previous *in vivo* successes attributed to E-PSM. While the shear rates used in this experiment mimicked the average shear rates seen in tumours, the average value could still be too high for PSM to overcome. Studying the shear rate levels around the average would examine if PSM only adhere in the tumour vessels with the lowest shear rates. Another helpful area to expand upon would be the actual adhesion strength of the PSM to the cells. This could be studied by allowing the PSM to settle onto endothelial cells before gradually ramping up the flow speed. By tracking PSM that become detached from the surface, the adhesion strength can be calculated. Also, by varying the shape of the flow chamber, it would be possible to study the effect of vessel tortuosity on the PSM adhesion characteristics. The possibility of turbulence and sharp turns in the tumour vasculature could give PSM more chances to interact with the vessel walls. To properly replicate the environment inside the vessels, introducing a physiologically relevant concentration of RBCs or a particle of similar size and density would recreate the RBC free layer that microparticles are known to get trapped in. This layer keeps microparticles sequestered to a small gap between the RBC dense central region of the vessels and the vessel walls, increasing the likelihood of PSM-endothelial cell interactions. Finally, to investigate the possibility of electrical interactions interfering with E-PSM adhesion, the concentration of ESTA adhered to the surface of the PSM should be varied. ESTA and SCR are negatively charged aptamers, so by controlling its concentration on the surface of the PSM, the zeta potential can be tuned to different values.

5.2. Tumour Vaccine TLS Development

In this study, the use of optical tissue clearing and whole tissue imaging was explored to search for and define tertiary lymphoid structures (TLS) inside the tumour space. After performing a meta-analysis of previously studied tertiary lymphoid structures, a rough concentration of B cells, T cells, and HEVs were used to define the minimum requirements for a TLS. Before the writing of this Thesis, there have been no explicit cellular requirements defined to identify TLS. Using this definition, TLS were able to be identified and quantified inside the tumour space using the iDISCO whole tissue imaging technique. This process was applied to a study investigating the effectiveness of a DC cancer vaccine therapy developed by Dr. Haifa Shen's lab at Houston Methodist Research Institute. When the DC cancer vaccine was combined

with an anti-PD1 checkpoint therapy, the number of TLS present inside the tumour space increased significantly. This also correlated with an improvement in the tumour growth curve and a decreased tumour size. The thought behind the increased TLS presence with the combination treatment is an avalanche theory. The cancer antigen primed DCs activate the immune system but the PD-L1 on the cancerous cells still prevents some of them from being attacked. Once the anti-PD1 is introduced, the immune system can freely target and destroy cancerous cells. Once the tumour mass has been identified, immune cells can signal a larger response from the host, eventually leading to the development of a TLS in the area of attack.

Being able to identify TLS and other 3D cellularly defined structures with software, instead of with human eyes, could dramatically increase the amount of information available to doctors in a timely manner. Currently, all histopathology is individually analysed by a qualified pathologist on 2D slides. These slides rely on structures of interest being visible in all dimensions so that a pathologist could identify it from a 2D projection. By using the software designed here or others like it, the pathologist role could become one of supervision and affirmation. ImageJ has already been utilized by many scientists to speed up 2D histopathological analysis.²⁴⁵⁻²⁴⁹ On top of all the time saved from digitizing 2D analyses, an analysis of a 3D sample allows for the software to identify structures, protein expression, and cellular density that could be missed by 2D analyses by eye. 3D tissue imaging does require longer processing times, but the amount of information gained from these methods would be substantial.

Immunofluorescence histology would go a long way to further examine this phenomenon and move this work forward. Histology samples are the current standard for identifying TLS and so would help to validate these results and the analysis method itself. It would also be productive to identify where primed DCs go after they are injected into the bloodstream. It is assumed that some will make it to the lymph nodes where they can continue to activate the immune system against the cancer, but it could also be possible that some end up in the tumour and are the beginnings of these TLS. Another future project would be to optimize the staining protocol to increase the signal to noise ratio. Although information can be extracted from low SNR images, having

clear images pre-processing would allow for higher confidence in the analysis methodology.

5.3. Simultaneous Collection of Overlapping Raman and Fluorescent Signals

In this publication, a method was developed for identifying Raman and fluorescence peaks in simultaneous overlapping spectra. Using MATLAB, the fluorescence signal was able to be identified and subtracted from the spectra. Once that was accomplished, the resonance Raman peaks were visible and identifiable. The MATLAB software was utilized to explore the effects of photodamage on Raman, resonance Raman, and fluorescence signals inside an RBC. The resonance Raman and the fluorescence signals both showed photo damage from high laser power excitation sources.

Until now, fluorescence signals have overwhelmed Raman signals and discouraged scientists from utilizing both imaging modalities in unison. By developing a methodology for uncovering the Raman signals from beneath the fluorescent signal, a new tool is now available to the scientific community. Fluorescence can be employed to identify cells of interest that are expressing a certain protein while the Raman uncovers compositional differences between the chemical structures inside the cell. One could even take a cell like that has been genetically modified to express a fluorescent protein and track the molecular changes in or around the cell as the fluorescent protein expression increases and decreases.

Raman spectroscopy has been used to diagnose breast cancer in patients with 94% sensitivity and 96% specificity.²⁵⁰ By implementing the software developed here with that diagnostic method, fluorescent markers could be included that would allow for a more detailed diagnosis into what type of breast cancer might be present. This could also be combined with the optical clearing method from Chapter 3 to elucidate the host's immune response inside the tumour mass. These techniques would give doctors pertinent information that will help them personalize the cancer therapy to each individual patient.

6. REFERENCES

1. Schlotter, C.M., Vogt, U., Allgayer, H. and Brandt, B., 2008. Molecular targeted therapies for breast cancer treatment. *Breast cancer research*, 10(4), p.211.
2. Palumbo, M.O., Kavan, P., Miller, W., Panasci, L., Assouline, S., Johnson, N., Cohen, V., Patenaude, F., Pollak, M., Jagoe, R.T. and Batist, G., 2013. Systemic cancer therapy: achievements and challenges that lie ahead. *Frontiers in pharmacology*, 4, p.57.
3. Ziemys, A., Yokoi, K., Kai, M., Liu, Y.T., Kojic, M., Simic, V., Milosevic, M., Holder, A. and Ferrari, M., 2018. Progression-dependent transport heterogeneity of breast cancer liver metastases as a factor in therapeutic resistance. *Journal of controlled release*, 291, pp.99-105.
4. Wibroe, P.P., Ahmadvand, D., Oghabian, M.A., Yaghmur, A. and Moghimi, S.M., 2016. An integrated assessment of morphology, size, and complement activation of the PEGylated liposomal doxorubicin products Doxil®, Caelyx®, DOXOrubicin, and SinaDoxosome. *Journal of Controlled Release*, 221, pp.1-8.
5. Swain, S.M., Baselga, J., Kim, S.B., Ro, J., Semiglazov, V., Campone, M., Ciruelos, E., Ferrero, J.M., Schneeweiss, A., Heeson, S. and Clark, E., 2015. Pertuzumab, trastuzumab, and docetaxel in HER2-positive metastatic breast cancer. *New England Journal of Medicine*, 372(8), pp.724-734.
6. Venuta, A., Wolfram, J., Shen, H. and Ferrari, M., 2017. Post-nano strategies for drug delivery: multistage porous silicon microvectors. *Journal of Materials Chemistry B*, 5(2), pp.207-219.
7. DeVita, V.T. and Chu, E., 2008. A history of cancer chemotherapy. *Cancer research*, 68(21), pp.8643-8653.
8. Wang, A.Z., Langer, R. and Farokhzad, O.C., 2012. Nanoparticle delivery of cancer drugs. *Annual review of medicine*, 63, pp.185-198.
9. Iyer, A.K., Khaled, G., Fang, J. and Maeda, H., 2006. Exploiting the enhanced permeability and retention effect for tumor targeting. *Drug discovery today*, 11(17-18), pp.812-818.
10. Nakamura, Y., Mochida, A., Choyke, P.L. and Kobayashi, H., 2016. Nanodrug delivery: is the enhanced permeability and retention effect sufficient for curing cancer?. *Bioconjugate chemistry*, 27(10), pp.2225-2238.
11. Truong, N.P., Whittaker, M.R., Mak, C.W. and Davis, T.P., 2015. The importance of nanoparticle shape in cancer drug delivery. *Expert opinion on drug delivery*, 12(1), pp.129-142.
12. Huang, X., Li, L., Liu, T., Hao, N., Liu, H., Chen, D. and Tang, F., 2011. The shape effect of mesoporous silica nanoparticles on biodistribution, clearance, and biocompatibility in vivo. *ACS nano*, 5(7), pp.5390-5399.
13. Yu, T., Hubbard, D., Ray, A. and Ghandehari, H., 2012. In vivo biodistribution and pharmacokinetics of silica nanoparticles as a function of geometry, porosity and surface characteristics. *Journal of controlled release*, 163(1), pp.46-54.
14. Geyer, C.E., Forster, J., Lindquist, D., Chan, S., Romieu, C.G., Pienkowski, T., Jagiello-Gruszfeld, A., Crown, J., Chan, A., Kaufman, B. and Skarlos, D., 2006. Lapatinib plus capecitabine for HER2-positive advanced breast cancer. *New England journal of medicine*, 355(26), pp.2733-2743.
15. Burstein, H.J., Sun, Y., Dirix, L.Y., Jiang, Z., Paridaens, R., Tan, A.R., Awada, A., Ranade, A., Jiao, S., Schwartz, G. and Abbas, R., 2010. Neratinib, an

- irreversible ErbB receptor tyrosine kinase inhibitor, in patients with advanced ErbB2-positive breast cancer. *J Clin Oncol*, 28(8), pp.1301-1307.
16. Lambert, J.M. and Chari, R.V., 2014. Ado-trastuzumab Emtansine (T-DM1): an antibody–drug conjugate (ADC) for HER2-positive breast cancer.
 17. Nakada, T., Sugihara, K., Jikoh, T., Abe, Y. and Agatsuma, T., 2019. The latest research and development into the antibody–drug conjugate,[fam-] trastuzumab deruxtecan (ds-8201a), for HER2 cancer therapy. *Chemical and Pharmaceutical Bulletin*, 67(3), pp.173-185.
 18. Cancer Research UK. (2019). *Breast cancer incidence (invasive) statistics*. [online] Available at: <https://www.cancerresearchuk.org/>
 19. Torchilin, V.P., 2005. Recent advances with liposomes as pharmaceutical carriers. *Nature reviews Drug discovery*, 4(2), pp.145-160.
 20. Hofheinz, R.D., Gnad-Vogt, S.U., Beyer, U. and Hochhaus, A., 2005. Liposomal encapsulated anti-cancer drugs. *Anti-cancer drugs*, 16(7), pp.691-707.
 21. Lee, T.R., Choi, M., Kopacz, A.M., Yun, S.H., Liu, W.K. and Decuzzi, P., 2013. On the near-wall accumulation of injectable particles in the microcirculation: smaller is not better. *Scientific reports*, 3, p.2079.
 22. Chaffer, C.L. and Weinberg, R.A., 2011. A perspective on cancer cell metastasis. *science*, 331(6024), pp.1559-1564.
 23. Patanaphan, V.I.N.I.T.A., Salazar, O.M. and Risco, R.A.F.A.E.L., 1988. Breast cancer: metastatic patterns and their prognosis. *Southern medical journal*, 81(9), pp.1109-1112.
 24. Key, T.J., Verkasalo, P.K. and Banks, E., 2001. Epidemiology of breast cancer. *The lancet oncology*, 2(3), pp.133-140.
 25. Amoldoni, J.T., Baungartner, W. and Canache, N.T., *DERMATITIS POST RADIOTERAPIA EN CÁNCER DE MAMA. A propósito de un caso.(Dermatitis Actínica)*.
 26. O'Connor, C. and Adams, J. (2010). *Essentials of Cell Biology*. Cambridge, MA: NPG Education.
 27. Langley, R.R. and Fidler, I.J., 2011. The seed and soil hypothesis revisited—The role of tumor-stroma interactions in metastasis to different organs. *International journal of cancer*, 128(11), pp.2527-2535.
 28. Paget, S., 1889. The distribution of secondary growths in cancer of the breast. *The Lancet*, 133(3421), pp.571-573.
 29. Hodi, F.S., O'day, S.J., McDermott, D.F., Weber, R.W., Sosman, J.A., Haanen, J.B., Gonzalez, R., Robert, C., Schadendorf, D., Hassel, J.C. and Akerley, W., 2010. Improved survival with ipilimumab in patients with metastatic melanoma. *New England Journal of Medicine*, 363(8), pp.711-723.
 30. Postow, M.A., Callahan, M.K. and Wolchok, J.D., 2015. Immune checkpoint blockade in cancer therapy. *Journal of clinical oncology*, 33(17), p.1974.
 31. Sharma, P., Wagner, K., Wolchok, J.D. and Allison, J.P., 2011. Novel cancer immunotherapy agents with survival benefit: recent successes and next steps. *Nature Reviews Cancer*, 11(11), p.805.
 32. Nagy, J.A., Chang, S.H., Dvorak, A.M. and Dvorak, H.F., 2009. Why are tumour blood vessels abnormal and why is it important to know?. *British journal of cancer*, 100(6), p.865.
 33. Anderson, K.G., Stromnes, I.M. and Greenberg, P.D., 2017. Obstacles posed by the tumor microenvironment to T cell activity: a case for synergistic therapies. *Cancer cell*, 31(3), pp.311-325.

34. Haddad, R. and Saldanha-Araujo, F., 2014. Mechanisms of T-cell immunosuppression by mesenchymal stromal cells: what do we know so far?. *BioMed research international*, 2014.
35. Dvorak, H.F., 2003. How tumors make bad blood vessels and stroma. *The American journal of pathology*, 162(6), p.1747.
36. Roberts, W.G. and Palade, G.E., 1997. Neovasculature induced by vascular endothelial growth factor is fenestrated. *Cancer research*, 57(4), pp.765-772.
37. Tanaka, N., Kanatani, S., Tomer, R., Sahlgren, C., Kronqvist, P., Kaczynska, D., Louhivuori, L., Kis, L., Lindh, C., Mitura, P. and Stepulak, A., 2017. Whole-tissue biopsy phenotyping of three-dimensional tumours reveals patterns of cancer heterogeneity. *Nature Biomedical Engineering*, 1(10), p.796.
38. Coussens, L.M. and Werb, Z., 2002. Inflammation and cancer. *Nature*, 420(6917), p.860.
39. Picker, L.J., Kishimoto, T.K., Smith, C.W., Warnock, R.A. and Butcher, E.C., 1991. ELAM-1 is an adhesion molecule for skin-homing T cells. *Nature*, 349(6312), p.796.
40. Mann, A.P., Tanaka, T., Somasunderam, A., Liu, X., Gorenstein, D.G. and Ferrari, M., 2011. E-Selectin-Targeted Porous Silicon Particle for Nanoparticle Delivery to the Bone Marrow. *Advanced Materials*, 23(36), pp.H278-H282.
41. Tasciotti, E., Liu, X., Bhavane, R., Plant, K., Leonard, A.D., Price, B.K., Cheng, M.M.C., Decuzzi, P., Tour, J.M., Robertson, F. and Ferrari, M., 2008. Mesoporous silicon particles as a multistage delivery system for imaging and therapeutic applications. *Nature nanotechnology*, 3(3), p.151.
42. Stephens, D.J. and Allan, V.J., 2003. Light microscopy techniques for live cell imaging. *science*, 300(5616), pp.82-86.
43. Wolf, D.E., 2007. Fundamentals of fluorescence and fluorescence microscopy. *Methods in cell biology*, 81, pp.63-91.
44. Lichtman, J.W. and Conchello, J.A., 2005. Fluorescence microscopy. *Nature methods*, 2(12), pp.910-919.
45. Daly, C.J. and McGrath, J.C., 2003. Fluorescent ligands, antibodies, and proteins for the study of receptors. *Pharmacology & therapeutics*, 100(2), pp.101-118.
46. Helmchen, F. and Denk, W., 2005. Deep tissue two-photon microscopy. *Nature methods*, 2(12), p.932.
47. Jacques, S.L., 2013. Optical properties of biological tissues: a review. *Physics in Medicine & Biology*, 58(11), p.R37.
48. Overall, N.J., 2000. Confocal Raman microscopy: why the depth resolution and spatial accuracy can be much worse than you think. *Applied Spectroscopy*, 54(10), pp.1515-1520.
49. Serda, R.E., Gu, J., Bhavane, R.C., Liu, X., Chiappini, C., Decuzzi, P. and Ferrari, M., 2009. The association of silicon microparticles with endothelial cells in drug delivery to the vasculature. *Biomaterials*, 30(13), pp.2440-2448.
50. Rayleigh, L., 1896. XV. On the theory of optical images, with special reference to the microscope. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 42(255), pp.167-195.
51. Abbe, E., 1873. Beiträge zur Theorie des Mikroskops und der mikroskopischen Wahrnehmung. *Archiv für mikroskopische Anatomie*, 9(1), pp.413-418.
52. Engstrom, R.W., 1947. Multiplier photo-tube characteristics: Application to low light levels. *JOSA*, 37(6), pp.420-431.
53. Ferraro, J.R., 2003. *Introductory raman spectroscopy*. Elsevier.

54. Le Ru, E. and Etchegoin, P., 2008. Principles of Surface-Enhanced Raman Spectroscopy: and related plasmonic effects. Elsevier.
55. Kallaway, C., Almond, L.M., Barr, H., Wood, J., Hutchings, J., Kendall, C. and Stone, N., 2013. Advances in the clinical application of Raman spectroscopy for cancer diagnostics. *Photodiagnosis and photodynamic therapy*, 10(3), pp.207-219.
56. Byrne, H.J., Baranska, M., Puppels, G.J., Stone, N., Wood, B., Gough, K.M., Lasch, P., Heraud, P., Sulé-Suso, J. and Sockalingum, G.D., 2015. Spectropathology for the next generation: Quo vadis?. *Analyst*, 140(7), pp.2066-2073.
57. Meyer, T., Bergner, N., Krafft, C., Akimov, D., Dietzek, B., Popp, J., Bielecki, C., Romeike, B.F., Reichart, R. and Kalff, R., 2011. Nonlinear microscopy, infrared, and Raman microspectroscopy for brain tumor analysis. *Journal of biomedical optics*, 16(2), p.021113.
58. Fullwood, L.M., Clemens, G., Griffiths, D., Ashton, K., Dawson, T.P., Lea, R.W., Davis, C., Bonnier, F., Byrne, H.J. and Baker, M.J., 2014. Investigating the use of Raman and immersion Raman spectroscopy for spectral histopathology of metastatic brain cancer and primary sites of origin. *Analytical Methods*, 6(12), pp.3948-3961.
59. Gajjar, K., Heppenstall, L.D., Pang, W., Ashton, K.M., Trevisan, J., Patel, I.I., Llabjani, V., Stringfellow, H.F., Martin-Hirsch, P.L., Dawson, T. and Martin, F.L., 2013. Diagnostic segregation of human brain tumours using Fourier-transform infrared and/or Raman spectroscopy coupled with discriminant analysis. *Analytical Methods*, 5(1), pp.89-102.
60. Krafft, C., Neudert, L., Simat, T. and Salzer, R., 2005. Near infrared Raman spectra of human brain lipids. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 61(7), pp.1529-1535.
61. Stone, N., Baker, R., Rogers, K., Parker, A.W. and Matousek, P., 2007. Subsurface probing of calcifications with spatially offset Raman spectroscopy (SORS): future possibilities for the diagnosis of breast cancer. *Analyst*, 132(9), pp.899-905.
62. Crow, P., Uff, J.S., Farmer, J.A., Wright, M.P. and Stone, N., 2004. The use of Raman spectroscopy to identify and characterize transitional cell carcinoma in vitro. *BJU international*, 93(9), pp.1232-1236.
63. Lin, D., Feng, S., Pan, J., Chen, Y., Lin, J., Chen, G., Xie, S., Zeng, H. and Chen, R., 2011. Colorectal cancer detection by gold nanoparticle based surface-enhanced Raman spectroscopy of blood serum and statistical analysis. *Optics express*, 19(14), pp.13565-13577.
64. Dekker, E. and Fockens, P., 2005. Advances in colonic imaging: new endoscopic imaging methods. *European journal of gastroenterology & hepatology*, 17(8), pp.803-808.
65. Stone, N., Stavroulaki, P., Kendall, C., Birchall, M. and Barr, H., 2000. Raman spectroscopy for early detection of laryngeal malignancy: preliminary results. *The Laryngoscope*, 110(10), pp.1756-1763.
66. Huang, Z., McWilliams, A., Lui, H., McLean, D.I., Lam, S. and Zeng, H., 2003. Near-infrared Raman spectroscopy for optical diagnosis of lung cancer. *International journal of cancer*, 107(6), pp.1047-1052.
67. Horsnell, J.D., Smith, J.A., Sattlecker, M., Sammon, A., Christie-Brown, J., Kendall, C. and Stone, N., 2012. Raman spectroscopy—a potential new method

- for the intra-operative assessment of axillary lymph nodes. *the surgeon*, 10(3), pp.123-127.
68. Lloyd, G.R., Orr, L.E., Christie-Brown, J., McCarthy, K., Rose, S., Thomas, M. and Stone, N., 2013. Discrimination between benign, primary and secondary malignancies in lymph nodes from the head and neck utilising Raman spectroscopy and multivariate analysis. *Analyst*, 138(14), pp.3900-3908.
 69. Stone, N., Kendall, C., Shepherd, N., Crow, P. and Barr, H., 2002. Near-infrared Raman spectroscopy for the classification of epithelial pre-cancers and cancers. *Journal of Raman spectroscopy*, 33(7), pp.564-573.
 70. Kendall, C., Stone, N., Shepherd, N., Geboes, K., Warren, B., Bennett, R. and Barr, H., 2003. Raman spectroscopy, a potential tool for the objective identification and classification of neoplasia in Barrett's oesophagus. *The Journal of Pathology: A Journal of the Pathological Society of Great Britain and Ireland*, 200(5), pp.602-609.
 71. Bergholt, M.S., Zheng, W., Lin, K., Ho, K.Y., Teh, M., Yeoh, K.G., So, J.Y. and Huang, Z., 2011. In vivo diagnosis of esophageal cancer using image-guided Raman endoscopy and biomolecular modeling. *Technology in cancer research & treatment*, 10(2), pp.103-112.
 72. Crow, P., Molckovsky, A., Stone, N., Uff, J., Wilson, B. and WongKeeSong, L.M., 2005. Assessment of fiberoptic near-infrared Raman spectroscopy for diagnosis of bladder and prostate cancer. *Urology*, 65(6), pp.1126-1130.
 73. Patel, I.I. and Martin, F.L., 2010. Discrimination of zone-specific spectral signatures in normal human prostate using Raman spectroscopy. *Analyst*, 135(12), pp.3060-3069.
 74. Patel, I.I., Trevisan, J., Singh, P.B., Nicholson, C.M., Krishnan, R.G., Matanhelia, S.S. and Martin, F.L., 2011. Segregation of human prostate tissues classified high-risk (UK) versus low-risk (India) for adenocarcinoma using Fourier-transform infrared or Raman microspectroscopy coupled with discriminant analysis. *Analytical and bioanalytical chemistry*, 401(3), pp.969-982.
 75. Crow, P., Uff, J.S., Farmer, J.A., Wright, M.P. and Stone, N., 2004. The use of Raman spectroscopy to identify and characterize transitional cell carcinoma in vitro. *BJU international*, 93(9), pp.1232-1236.
 76. Rashid, N., Nawaz, H., Poon, K.W., Bonnier, F., Bakhiet, S., Martin, C., O'Leary, J.J., Byrne, H.J. and Lyng, F.M., 2014. Raman microspectroscopy for the early detection of pre-malignant changes in cervical tissue. *Experimental and molecular pathology*, 97(3), pp.554-564.
 77. Patel, I.I., Trevisan, J., Evans, G., Llabjani, V., Martin-Hirsch, P.L., Stringfellow, H.F. and Martin, F.L., 2011. High contrast images of uterine tissue derived using Raman microspectroscopy with the empty modelling approach of multivariate curve resolution-alternating least squares. *Analyst*, 136(23), pp.4950-4959.
 78. Krishna, C.M., Prathima, N.B., Malini, R., Vadhiraaja, B.M., Bhatt, R.A., Fernandes, D.J., Kushtagi, P., Vidyasagar, M.S. and Kartha, V.B., 2006. Raman spectroscopy studies for diagnosis of cancers in human uterine cervix. *Vibrational Spectroscopy*, 41(1), pp.136-141.
 79. Lyng, F.M., Faoláin, E.O., Conroy, J., Meade, A.D., Knief, P., Duffy, B., Hunter, M.B., Byrne, J.M., Kelehan, P. and Byrne, H.J., 2007. Vibrational spectroscopy for cervical cancer pathology, from biochemical analysis to diagnostic tool. *Experimental and molecular pathology*, 82(2), pp.121-129.

80. Bonnier, F., Knief, P., Lim, B., Meade, A.D., Dorney, J., Bhattacharya, K., Lyng, F.M. and Byrne, H.J., 2010. Imaging live cells grown on a three dimensional collagen matrix using Raman microspectroscopy. *Analyst*, 135(12), pp.3169-3177.
81. Notingher, I. and Hench, L.L., 2006. Raman microspectroscopy: a noninvasive tool for studies of individual living cells in vitro. *Expert review of medical devices*, 3(2), pp.215-234.
82. Zhang, X., Roeffaers, M.B., Basu, S., Daniele, J.R., Fu, D., Freudiger, C.W., Holtom, G.R. and Xie, X.S., 2012. Label-free live-cell imaging of nucleic acids using stimulated Raman scattering microscopy. *ChemPhysChem*, 13(4), pp.1054-1059.
83. Kneipp, J., Kneipp, H., Rajadurai, A., Redmond, R.W. and Kneipp, K., 2009. Optical probing and imaging of live cells using SERS labels. *Journal of Raman Spectroscopy: An International Journal for Original Work in all Aspects of Raman Spectroscopy, Including Higher Order Processes, and also Brillouin and Rayleigh Scattering*, 40(1), pp.1-5.
84. Mellon, M.G., 1950. Analytical absorption spectroscopy.
85. Instruments, M., 11. Size theory. Zetasizer nano series user manual, pp.11-1.
86. Instruments, M., 11. Zeta potential theory. Zetasizer nano series user manual, pp.13-1.
87. Lawrence, M.B., McIntire, L.V. and Eskin, S.G., 1987. Effect of flow on polymorphonuclear leukocyte/endothelial cell adhesion.
88. Davies, P.F., 1995. Flow-mediated endothelial mechanotransduction. *Physiological reviews*, 75(3), pp.519-560.
89. Giavazzi, R., Foppolo, M., Dossi, R. and Remuzzi, A., 1993. Rolling and adhesion of human tumor cells on vascular endothelium under physiological flow conditions. *The Journal of clinical investigation*, 92(6), pp.3038-3044.
90. Patil, V.R.S., Campbell, C.J., Yun, Y.H., Slack, S.M. and Goetz, D.J., 2001. Particle diameter influences adhesion under flow. *Biophysical Journal*, 80(4), pp.1733-1743.
91. Bakker, D.P., Van der Plaats, A., Verkerke, G.J., Busscher, H.J. and Van Der Mei, H.C., 2003. Comparison of velocity profiles for different flow chamber designs used in studies of microbial adhesion to surfaces. *Appl. Environ. Microbiol.*, 69(10), pp.6280-6287.
92. Lavis, L.D., 2011. Histochemistry: live and in color. *Journal of Histochemistry & Cytochemistry*, 59(2), pp.139-145.
93. Stockton, G.W. and Smith, I.C., 1976. A deuterium nuclear magnetic resonance study of the condensing effect of cholesterol on egg phosphatidylcholine bilayer membranes. I. Perdeuterated fatty acid probes. *Chemistry and physics of lipids*, 17(2-3), pp.251-263.
94. Mahadevan-Jansen, A. and Richards-Kortum, R.R., 1996. Raman spectroscopy for the detection of cancers and precancers. *Journal of biomedical optics*, 1(1), pp.31-71.
95. Haka, A.S., Shafer-Peltier, K.E., Fitzmaurice, M., Crowe, J., Dasari, R.R. and Feld, M.S., 2005. Diagnosing breast cancer by using Raman spectroscopy. *Proceedings of the National Academy of Sciences*, 102(35), pp.12371-12376.
96. Stone, N., Kendall, C., Smith, J., Crow, P. and Barr, H., 2004. Raman spectroscopy for identification of epithelial cancers. *Faraday discussions*, 126, pp.141-157.

97. Huang, Z., McWilliams, A., Lui, H., McLean, D.I., Lam, S. and Zeng, H., 2003. Near-infrared Raman spectroscopy for optical diagnosis of lung cancer. *International journal of cancer*, 107(6), pp.1047-1052.
98. Brown, J.Q., Vishwanath, K., Palmer, G.M. and Ramanujam, N., 2009. Advances in quantitative UV–visible spectroscopy for clinical and pre-clinical application in cancer. *Current opinion in biotechnology*, 20(1), pp.119-131.
99. Heidari, A. and Gobato, R., 2018. Ultraviolet Photoelectron Spectroscopy (UPS) and Ultraviolet–Visible (UV–Vis) Spectroscopy Comparative Study on Malignant and Benign Human Cancer Cells and Tissues with the Passage of Time under Synchrotron Radiation. *Parana Journal of Science and Education*, 4(6), pp.18-33.
100. Xu, R., 2015. Light scattering: A review of particle characterization applications. *Particuology*, 18, pp.11-21.
101. Brown, W. ed., 1993. *Dynamic light scattering: the method and some applications* (Vol. 313). Oxford: Clarendon Press.
102. Xu, R., 2001. *Particle characterization: light scattering methods* (Vol. 13). Springer Science & Business Media.
103. Yellen, B.B., Forbes, Z.G., Halverson, D.S., Fridman, G., Barbee, K.A., Chorny, M., Levy, R. and Friedman, G., 2005. Targeted drug delivery to magnetic implants for therapeutic applications. *Journal of magnetism and magnetic materials*, 293(1), pp.647-654.
104. Smid, M., Wang, Y., Zhang, Y., Sieuwerts, A.M., Yu, J., Klijn, J.G., Foekens, J.A. and Martens, J.W., 2008. Subtypes of breast cancer show preferential site of relapse. *Cancer research*, 68(9), pp.3108-3114.
105. Kim, M.S., Kim, T., Kong, S.Y., Kwon, S., Bae, C.Y., Choi, J., Kim, C.H., Lee, E.S. and Park, J.K., 2010. Breast cancer diagnosis using a microfluidic multiplexed immunohistochemistry platform. *PloS one*, 5(5).
106. Klaes, R., Benner, A., Friedrich, T., Ridder, R., Herrington, S., Jenkins, D., Kurman, R.J., Schmidt, D., Stoler, M. and von Knebel Doeberitz, M., 2002. p16INK4a immunohistochemistry improves interobserver agreement in the diagnosis of cervical intraepithelial neoplasia. *The American journal of surgical pathology*, 26(11), pp.1389-1399.
107. Shah, R.B., 2013. Clinical applications of novel ERG immunohistochemistry in prostate cancer diagnosis and management. *Advances in anatomic pathology*, 20(2), pp.117-124.
108. Jagirdar, J., 2008. Application of immunohistochemistry to the diagnosis of primary and metastatic carcinoma to the lung. *Archives of pathology & laboratory medicine*, 132(3), pp.384-396.
109. Ollion, J., Cochenec, J., Loll, F., Escudé, C. and Boudier, T., 2013. TANGO: a generic tool for high-throughput 3D image analysis for studying nuclear organization. *Bioinformatics*, 29(14), pp.1840-1841.
110. Doube, M., Klosowski, M.M., Arganda-Carreras, I., Cordelières, F.P., Dougherty, R.P., Jackson, J.S., Schmid, B., Hutchinson, J.R. and Shefelbine, S.J., 2010. BoneJ: free and extensible bone image analysis in ImageJ. *Bone*, 47(6), pp.1076-1079.
111. Gilles, J.F., Dos Santos, M., Boudier, T., Bolte, S. and Heck, N., 2017. DiAna, an ImageJ tool for object-based 3D co-localization and distance analysis. *Methods*, 115, pp.55-64.

112. Tinevez, J.Y., Perry, N., Schindelin, J., Hoopes, G.M., Reynolds, G.D., Laplantine, E., Bednarek, S.Y., Shorte, S.L. and Eliceiri, K.W., 2017. TrackMate: An open and extensible platform for single-particle tracking. *Methods*, 115, pp.80-90.
113. Murray, J.I., Bao, Z., Boyle, T.J. and Waterston, R.H., 2006. The lineaging of fluorescently-labeled *Caenorhabditis elegans* embryos with StarryNite and AceTree. *Nature protocols*, 1(3), p.1468.
114. Sage, D., Neumann, F.R., Hediger, F., Gasser, S.M. and Unser, M., 2005. Automatic tracking of individual fluorescence particles: application to the study of chromosome dynamics. *IEEE transactions on image processing*, 14(9), pp.1372-1383.
115. Jaqaman, K., Loerke, D., Mettlen, M., Kuwata, H., Grinstein, S., Schmid, S.L. and Danuser, G., 2008. Robust single-particle tracking in live-cell time-lapse sequences. *Nature methods*, 5(8), p.695.
116. De Chaumont, F., Dallongeville, S., Chenouard, N., Hervé, N., Pop, S., Provoost, T., Meas-Yedid, V., Pankajakshan, P., Lecomte, T., Le Montagner, Y. and Lagache, T., 2012. Icy: an open bioimage informatics platform for extended reproducible research. *Nature methods*, 9(7), p.690.
117. Chenouard, N., Bloch, I. and Olivo-Marin, J.C., 2013. Multiple hypothesis tracking for cluttered biological image sequences. *IEEE transactions on pattern analysis and machine intelligence*, 35(11), pp.2736-3750.
118. Crocker, J.C. and Grier, D.G., 1996. Methods of digital video microscopy for colloidal studies. *Journal of colloid and interface science*, 179(1), pp.298-310.
119. Sergé, A., Bertaux, N., Rigneault, H. and Marguet, D., 2008. Dynamic multiple-target tracing to probe spatiotemporal cartography of cell membranes. *Nature methods*, 5(8), pp.687-694.
120. Sbalzarini, I.F. and Koumoutsakos, P., 2005. Feature point tracking and trajectory analysis for video imaging in cell biology. *Journal of structural biology*, 151(2), pp.182-195.
121. Cordelières, F.P., Petit, V., Kumasaka, M., Debeir, O., Letort, V., Gallagher, S.J. and Larue, L., 2013. Automated cell tracking and analysis in phase-contrast videos (iTrack4U): development of Java software based on combined mean-shift processes. *PloS one*, 8(11).
122. Liedtke, C., Mazouni, C., Hess, K.R., André, F., Tordai, A., Mejia, J.A., Symmans, W.F., Gonzalez-Angulo, A.M., Hennessy, B., Green, M. and Cristofanilli, M., 2008. Response to neoadjuvant therapy and long-term survival in patients with triple-negative breast cancer. *Journal of clinical oncology*, 26(8), pp.1275-1281.
123. Diel, I.J., Solomayer, E.F. and Bastert, G., 2000. Treatment of metastatic bone disease in breast cancer: bisphosphonates. *Clinical breast cancer*, 1(1), pp.43-51.
124. Mundy, G.R., 2002. Metastasis: Metastasis to bone: causes, consequences and therapeutic opportunities. *Nature Reviews Cancer*, 2(8), p.584.
125. Walker, M.S., Miller, P.J., Namjoshi, M., Houts, A.C., Stepanski, E.J. and Schwartzberg, L.S., 2013. Relationship between incidence of fracture and health-related quality-of-life in metastatic breast cancer patients with bone metastases. *Journal of medical economics*, 16(1), pp.179-189.
126. Hines, S.L., Sloan, J.A., Atherton, P.J., Perez, E.A., Dakhil, S.R., Johnson, D.B., Reddy, P.S., Dalton, R.J., Mattar, B.I. and Loprinzi, C.L., 2010. Zoledronic

- acid for treatment of osteopenia and osteoporosis in women with primary breast cancer undergoing adjuvant aromatase inhibitor therapy. *The Breast*, 19(2), pp.92-96.
127. Jones, D.H., Nakashima, T., Sanchez, O.H., Kozieradzki, I., Komarova, S.V., Sarosi, I., Morony, S., Rubin, E., Sarao, R., Hojilla, C.V. and Komnenovic, V., 2006. Regulation of cancer cell migration and bone metastasis by RANKL. *Nature*, 440(7084), p.692.
 128. Martin, M., Bell, R., Bourgeois, H., Brufsky, A., Diel, I., Eniu, A., Fallowfield, L., Fujiwara, Y., Jassem, J., Paterson, A.H. and Ritchie, D., 2012. Bone-related complications and quality of life in advanced breast cancer: results from a randomized phase III trial of denosumab versus zoledronic acid. *Clinical Cancer Research*, 18(17), pp.4841-4849.
 129. Coleman, R., Gnant, M., Morgan, G. and Clezardin, P., 2012. Effects of bone-targeted agents on cancer progression and mortality. *Journal of the National Cancer Institute*, 104(14), pp.1059-1067.
 130. Gnant, M., Mlineritsch, B., Luschin-Ebengreuth, G., Kainberger, F., Kässmann, H., Piswanger-Sölkner, J.C., Seifert, M., Ploner, F., Menzel, C., Dubsky, P. and Fitzal, F., 2008. Adjuvant endocrine therapy plus zoledronic acid in premenopausal women with early-stage breast cancer: 5-year follow-up of the ABCSG-12 bone-mineral density substudy. *The lancet oncology*, 9(9), pp.840-849.
 131. Lórinicz, T., Tímár, J. and Szendrői, M., 2004. Alterations of microvascular density in bone metastases of adenocarcinomas. *Pathology Oncology Research*, 10(3), pp.149-153.
 132. Moore, M.A., 2012. Waking up HSCs: a new role for E-selectin. *Nature medicine*, 18(11), p.1613.
 133. Zarbock, A., Ley, K., McEver, R.P. and Hidalgo, A., 2011. Leukocyte ligands for endothelial selectins: specialized glycoconjugates that mediate rolling and signaling under flow. *Blood*, 118(26), pp.6743-6751.
 134. Winkler, I.G., Barbier, V., Nowlan, B., Jacobsen, R.N., Forristal, C.E., Patton, J.T., Magnani, J.L. and Lévesque, J.P., 2012. Vascular niche E-selectin regulates hematopoietic stem cell dormancy, self renewal and chemoresistance. *Nature medicine*, 18(11), p.1651.
 135. Mitchell, M.J., Chen, C.S., Ponmudi, V., Hughes, A.D. and King, M.R., 2012. E-selectin liposomal and nanotube-targeted delivery of doxorubicin to circulating tumor cells. *Journal of controlled release*, 160(3), pp.609-617.
 136. Jubeli, E., Moine, L., Vergnaud-Gauduchon, J. and Barratt, G., 2012. E-selectin as a target for drug delivery and molecular imaging. *Journal of Controlled Release*, 158(2), pp.194-206.
 137. Tasciotti, E., Liu, X., Bhavane, R., Plant, K., Leonard, A.D., Price, B.K., Cheng, M.M.C., Decuzzi, P., Tour, J.M., Robertson, F. and Ferrari, M., 2008. Mesoporous silicon particles as a multistage delivery system for imaging and therapeutic applications. *Nature nanotechnology*, 3(3), p.151.
 138. Namdee, K., Thompson, A.J., Charoenphol, P. and Eniola-Adefeso, O., 2013. Margination propensity of vascular-targeted spheres from blood flow in a microfluidic model of human microvessels. *Langmuir*, 29(8), pp.2530-2535.
 139. Decuzzi, P., Godin, B., Tanaka, T., Lee, S.Y., Chiappini, C., Liu, X. and Ferrari, M., 2010. Size and shape effects in the biodistribution of intravascularly injected particles. *Journal of Controlled Release*, 141(3), pp.320-327.

140. Xu, R., Huang, Y., Mai, J., Zhang, G., Guo, X., Xia, X., Koay, E.J., Qin, G., Erm, D.R., Li, Q. and Liu, X., 2013. Multistage Vectored siRNA Targeting Ataxia-Telangiectasia Mutated for Breast Cancer Therapy. *Small*, 9(9-10), pp.1799-1808.
141. Tanaka, T., Mangala, L.S., Vivas-Mejia, P.E., Nieves-Alicea, R., Mann, A.P., Mora, E., Han, H.D., Shahzad, M.M., Liu, X., Bhavane, R. and Gu, J., 2010. Sustained small interfering RNA delivery by mesoporous silicon particles. *Cancer research*, 70(9), pp.3687-3696.
142. Shen, H., Rodriguez-Aguayo, C., Xu, R., Gonzalez-Villasana, V., Mai, J., Huang, Y., Zhang, G., Guo, X., Bai, L., Qin, G. and Deng, X., 2013. Enhancing chemotherapy response with sustained EphA2 silencing using multistage vector delivery. *Clinical Cancer Research*, 19(7), pp.1806-1815.
143. Mann, A.P., Somasunderam, A., Nieves-Alicea, R., Li, X., Hu, A., Sood, A.K., Ferrari, M., Gorenstein, D.G. and Tanaka, T., 2010. Identification of thioaptamer ligand against E-selectin: potential application for inflamed vasculature targeting. *PloS one*, 5(9), p.e13050.
144. Mai, J., Huang, Y., Mu, C., Zhang, G., Xu, R., Guo, X., Xia, X., Volk, D.E., Lokesh, G.L., Thiviyanathan, V. and Gorenstein, D.G., 2014. Bone marrow endothelium-targeted therapeutics for metastatic breast cancer. *Journal of controlled release*, 187, pp.22-29.
145. Duncan, G.A., Gerecht, S., Fairbrother, D.H. and Bevan, M.A., 2016. Diffusing Colloidal Probes of kT-Scale Biomaterial–Cell Interactions. *Langmuir*, 32(46), pp.12212-12220.
146. Nelson, P.C., Zurla, C., Brogioli, D., Beausang, J.F., Finzi, L. and Dunlap, D., 2006. Tethered particle motion as a diagnostic of DNA tether length. *The Journal of Physical Chemistry B*, 110(34), pp.17260-17267.
147. Lawrence, M.B., McIntire, L.V. and Eskin, S.G., 1987. Effect of flow on polymorphonuclear leukocyte/endothelial cell adhesion. *Blood*, 70(5), pp.1284-1290.
148. Li, Q., Wayman, A., Lin, J., Fang, Y., Zhu, C. and Wu, J., 2016. Flow-enhanced stability of rolling adhesion through E-selectin. *Biophysical journal*, 111(4), pp.686-699.
149. Decuzzi, P., Gentile, F., Granaldi, A., Curcio, A., Causa, F., Indolfi, C., Netti, P. and Ferrari, M., 2007. Flow chamber analysis of size effects in the adhesion of spherical particles. *International journal of nanomedicine*, 2(4), p.689.
150. Gentile, F., Chiappini, C., Fine, D., Bhavane, R.C., Peluccio, M.S., Cheng, M.M.C., Liu, X., Ferrari, M. and Decuzzi, P., 2008. The effect of shape on the margination dynamics of non-neutrally buoyant particles in two-dimensional shear flows. *Journal of biomechanics*, 41(10), pp.2312-2318.
151. Brown, D.C. and Larson, R.S., 2001. Improvements to parallel plate flow chambers to reduce reagent and cellular requirements. *BMC immunology*, 2(1), p.9.
152. Liang, S., Slattery, M.J. and Dong, C., 2005. Shear stress and shear rate differentially affect the multi-step process of leukocyte-facilitated melanoma adhesion. *Experimental cell research*, 310(2), pp.282-292.
153. Nagaoka, T. and Yoshida, A., 2006. Noninvasive evaluation of wall shear stress on retinal microcirculation in humans. *Investigative ophthalmology & visual science*, 47(3), pp.1113-1119.

154. Koutsiaris, A.G., Tachmitzi, S.V., Batis, N., Kotoula, M.G., Karabatsas, C.H., Tsironi, E. and Chatzoulis, D.Z., 2007. Volume flow and wall shear stress quantification in the human conjunctival capillaries and post-capillary venules in vivo. *Biorheology*, 44(5-6), pp.375-386.
155. Fisher, D.T., Muhitch, J.B., Kim, M., Doyen, K.C., Bogner, P.N., Evans, S.S. and Skitzki, J.J., 2016. Intraoperative intravital microscopy permits the study of human tumour vessels. *Nature communications*, 7, p.10684.
156. Kruskal, J.B., Thomas, P., Kane, R.A. and Goldberg, S.N., 2004. Hepatic perfusion changes in mice livers with developing colorectal cancer metastases. *Radiology*, 231(2), pp.482-490.
157. Ma, S., Tian, X.Y., Zhang, Y., Mu, C., Shen, H., Bismuth, J., Pownall, H.J., Huang, Y. and Wong, W.T., 2016. E-selectin-targeting delivery of microRNAs by microparticles ameliorates endothelial inflammation and atherosclerosis. *Scientific reports*, 6, p.22910.
158. Brown, D.C. and Larson, R.S., 2001. Improvements to parallel plate flow chambers to reduce reagent and cellular requirements. *BMC immunology*, 2(1), p.9.
159. Godin, B., Chiappini, C., Srinivasan, S., Alexander, J.F., Yokoi, K., Ferrari, M., Decuzzi, P. and Liu, X., 2012. Discoidal porous silicon particles: fabrication and biodistribution in breast cancer bearing mice. *Advanced functional materials*, 22(20), pp.4225-4235.
160. Linstrom, P.J. and Mallard, W.G., 2001. The NIST Chemistry WebBook: A chemical data resource on the internet. *Journal of Chemical & Engineering Data*, 46(5), pp.1059-1063.
161. Bacabac, R.G., Smit, T.H., Cowin, S.C., Van Loon, J.J., Nieuwstadt, F.T., Heethaar, R. and Klein-Nulend, J., 2005. Dynamic shear stress in parallel-plate flow chambers. *Journal of biomechanics*, 38(1), pp.159-167.
162. Schlichting, H. and Gersten, K., 2016. *Boundary-layer theory*. Springer.
163. White, F., 2011. *Fluid Mechanics*. 7th ed. Singapore: McGraw-Hill, pp.477-478.
164. Pober, J.S. and Cotran, R.S., 1990. Cytokines and endothelial cell biology. *Physiological reviews*, 70(2), pp.427-451.
165. Xu, R., Zhang, G., Mai, J., Deng, X., Segura-Ibarra, V., Wu, S., Shen, J., Liu, H., Hu, Z., Chen, L. and Huang, Y., 2016. An injectable nanoparticle generator enhances delivery of cancer therapeutics. *Nature biotechnology*, 34(4), p.414.
166. Mi, Y., Mu, C., Wolfram, J., Deng, Z., Hu, T.Y., Liu, X., Blanco, E., Shen, H. and Ferrari, M., 2016. A micro/nano composite for combination treatment of melanoma lung metastasis. *Advanced healthcare materials*, 5(8), pp.936-946.
167. Jiang, W., Gupta, R.K., Deshpande, M.C. and Schwendeman, S.P., 2005. Biodegradable poly (lactic-co-glycolic acid) microparticles for injectable delivery of vaccine antigens. *Advanced drug delivery reviews*, 57(3), pp.391-410.
168. Müller, K., Fedosov, D.A. and Gompper, G., 2014. Margination of micro- and nano-particles in blood flow and its effect on drug delivery. *Scientific reports*, 4, p.4871.
169. Decuzzi, P. and Ferrari, M., 2006. The adhesive strength of non-spherical particles mediated by specific interactions. *Biomaterials*, 27(30), pp.5307-5314.
170. Bondar, O.V., Saifullina, D.V., Shakhmaeva, I.I., Mavlyutova, I.I. and Abdullin, T.I., 2012. Monitoring of the zeta potential of human cells upon

- reduction in their viability and interaction with polymers. *Acta Naturae* (англоязычная версия), 4(1 (12)).
171. Liu, Y., Zhang, C., Li, X. and Wu, D., 2017. A self-monitored fluorescence DNA anti-counterfeiting system based on silica coated SYBR Green I/DNA gelatin nanoparticles. *Journal of Materials Chemistry C*, 5(24), pp.5939-5948.
 172. Andrade, Â.L., Ferreira, J.M.F. and Domingues, R.Z., 2004. Zeta potential measurement in bioactive collagen. *Materials Research*, 7(4), pp.631-634.
 173. Decuzzi, P., Lee, S., Bhushan, B. and Ferrari, M., 2005. A theoretical model for the margination of particles within blood vessels. *Annals of biomedical engineering*, 33(2), pp.179-190.
 174. Lee, V.K., Lanzi, A.M., Ngo, H., Yoo, S.S., Vincent, P.A. and Dai, G., 2014. Generation of multi-scale vascular network system within 3D hydrogel using 3D bio-printing technology. *Cellular and molecular bioengineering*, 7(3), pp.460-472.
 175. Saggiomo, V. and Velders, A.H., 2015. Simple 3D printed scaffold-removal method for the fabrication of intricate microfluidic devices. *Advanced Science*, 2(9), p.1500125.
 176. D'apolito, R., Tomaiuolo, G., Taraballi, F., Minardi, S., Kirui, D., Liu, X., Cevenini, A., Palomba, R., Ferrari, M., Salvatore, F. and Tasciotti, E., 2015. Red blood cells affect the margination of microparticles in synthetic microcapillaries and intravital microcirculation as a function of their size and shape. *Journal of Controlled Release*, 217, pp.263-272.
 177. Pardoll, D.M., 2012. The blockade of immune checkpoints in cancer immunotherapy. *Nature Reviews Cancer*, 12(4), p.252.
 178. Steinman, R.M. and Banchereau, J., 2007. Taking dendritic cells into medicine. *Nature*, 449(7161), p.419.
 179. Xia, X., Mai, J., Xu, R., Perez, J.E.T., Guevara, M.L., Shen, Q., Mu, C., Tung, H.Y., Corry, D.B., Evans, S.E. and Liu, X., 2015. Porous silicon microparticle potentiates anti-tumor immunity by enhancing cross-presentation and inducing type I interferon response. *Cell reports*, 11(6), pp.957-966.
 180. Chen, X., Iliopoulos, D., Zhang, Q., Tang, Q., Greenblatt, M.B., Hatziapostolou, M., Lim, E., Tam, W.L., Ni, M., Chen, Y. and Mai, J., 2014. XBP1 promotes triple-negative breast cancer by controlling the HIF1 α pathway. *Nature*, 508(7494), p.103.
 181. Dave, B., Granados-Principal, S., Zhu, R., Benz, S., Rabizadeh, S., Soon-Shiong, P., Yu, K.D., Shao, Z., Li, X., Gilcrease, M. and Lai, Z., 2014. Targeting RPL39 and MLF2 reduces tumor initiation and metastasis in breast cancer by inhibiting nitric oxide synthase signaling. *Proceedings of the National Academy of Sciences*, 111(24), pp.8838-8843.
 182. Shen, H., Rodriguez-Aguayo, C., Xu, R., Gonzalez-Villasana, V., Mai, J., Huang, Y., Zhang, G., Guo, X., Bai, L., Qin, G. and Deng, X., 2013. Enhancing chemotherapy response with sustained EphA2 silencing using multistage vector delivery. *Clinical Cancer Research*, 19(7), pp.1806-1815.
 183. Xu, R., Huang, Y., Mai, J., Zhang, G., Guo, X., Xia, X., Koay, E.J., Qin, G., Erm, D.R., Li, Q. and Liu, X., 2013. Multistage Vectored siRNA Targeting Ataxia-Telangiectasia Mutated for Breast Cancer Therapy. *Small*, 9(9-10), pp.1799-1808.
 184. Shen, J., Xu, R., Mai, J., Kim, H.C., Guo, X., Qin, G., Yang, Y., Wolfram, J., Mu, C., Xia, X. and Gu, J., 2013. High capacity nanoporous silicon carrier for

- systemic delivery of gene silencing therapeutics. *ACS nano*, 7(11), pp.9867-9880.
185. Weih, F., Gräbner, R., Hu, D., Beer, M. and Habenicht, A.J., 2012. Control of dichotomic innate and adaptive immune responses by artery tertiary lymphoid organs in atherosclerosis. *Frontiers in physiology*, 3, p.226.
 186. Neyt, K., Perros, F., GeurtsvanKessel, C.H., Hammad, H. and Lambrecht, B.N., 2012. Tertiary lymphoid organs in infection and autoimmunity. *Trends in immunology*, 33(6), pp.297-305.
 187. GeurtsvanKessel, C.H., Willart, M.A., Bergen, I.M., van Rijt, L.S., Muskens, F., Elewaut, D., Osterhaus, A.D., Hendriks, R., Rimmelzwaan, G.F. and Lambrecht, B.N., 2009. Dendritic cells are crucial for maintenance of tertiary lymphoid structures in the lung of influenza virus–infected mice. *Journal of Experimental Medicine*, 206(11), pp.2339-2349.
 188. Halle, S., Dujardin, H.C., Bakocevic, N., Fleige, H., Danzer, H., Willenzon, S., Suezter, Y., Hämmerling, G., Garbi, N., Sutter, G. and Worbs, T., 2009. Induced bronchus-associated lymphoid tissue serves as a general priming site for T cells and is maintained by dendritic cells. *Journal of Experimental Medicine*, 206(12), pp.2593-2601.
 189. Hiraoka, N., Ino, Y. and Yamazaki-Itoh, R., 2016. Tertiary lymphoid organs in cancer tissues. *Frontiers in immunology*, 7, p.244.
 190. Sautès-Fridman, C., Lawand, M., Giraldo, N.A., Kaplon, H., Germain, C., Fridman, W.H. and Dieu-Nosjean, M.C., 2016. Tertiary lymphoid structures in cancers: prognostic value, regulation, and manipulation for therapeutic intervention. *Frontiers in immunology*, 7, p.407.
 191. Bento, D.C., Jones, E., Junaid, S., Tull, J., Williams, G.T., Godkin, A., Ager, A. and Gallimore, A., 2015. High endothelial venules are rare in colorectal cancers but accumulate in extra-tumoral areas with disease progression. *Oncoimmunology*, 4(3), p.e974374.
 192. Goc, J., Fridman, W.H., Sautès-Fridman, C. and Dieu-Nosjean, M.C., 2013. Characteristics of tertiary lymphoid structures in primary cancers. *Oncoimmunology*, 2(12), p.e26836.
 193. Martinet, L., Garrido, I., Filleron, T., Le Guellec, S., Bellard, E., Fournie, J.J., Rochaix, P. and Girard, J.P., 2011. Human solid tumors contain high endothelial venules: association with T- and B-lymphocyte infiltration and favorable prognosis in breast cancer. *Cancer research*, 71(17), pp.5678-5687.
 194. Lee, H.J., Park, I.A., Song, I.H., Shin, S.J., Kim, J.Y., Yu, J.H. and Gong, G., 2016. Tertiary lymphoid structures: prognostic significance and relationship with tumour-infiltrating lymphocytes in triple-negative breast cancer. *Journal of clinical pathology*, 69(5), pp.422-430.
 195. Engelhard, V.H., Rodriguez, A.B., Mauldin, I.S., Woods, A.N., Peske, J.D. and Slingluff, C.L., 2018. Immune cell infiltration and tertiary lymphoid structures as determinants of antitumor immunity. *The Journal of Immunology*, 200(2), pp.432-442.
 196. Germain, C., Gnjjatic, S., Tamzalit, F., Knockaert, S., Remark, R., Goc, J., Lepelley, A., Becht, E., Katsahian, S., Bizouard, G. and Validire, P., 2014. Presence of B cells in tertiary lymphoid structures is associated with a protective immunity in patients with lung cancer. *American journal of respiratory and critical care medicine*, 189(7), pp.832-844.

197. Gu-Trantien, C., Loi, S., Garaud, S., Equeter, C., Libin, M., De Wind, A., Ravoet, M., Le Buanec, H., Sibille, C., Manfouo-Foutsop, G. and Veys, I., 2013. CD4⁺ follicular helper T cell infiltration predicts breast cancer survival. *The Journal of clinical investigation*, 123(7), pp.2873-2892.
198. de Chaisemartin, L., Goc, J., Damotte, D., Validire, P., Magdeleinat, P., Alifano, M., Cremer, I., Fridman, W.H., Sautès-Fridman, C. and Dieu-Nosjean, M.C., 2011. Characterization of chemokines and adhesion molecules associated with T cell presence in tertiary lymphoid structures in human lung cancer. *Cancer research*, 71(20), pp.6391-6399.
199. Dieu-Nosjean, M.C., Antoine, M., Danel, C., Heudes, D., Wislez, M., Poulot, V., Rabbe, N., Laurans, L., Tartour, E., de Chaisemartin, L. and Lebecque, S., 2008. Long-term survival for patients with non-small-cell lung cancer with intratumoral lymphoid structures. *Journal of Clinical Oncology*, 26(27), pp.4410-4417.
200. Goc, J., Germain, C., Vo-Bourgais, T.K.D., Lupo, A., Klein, C., Knockaert, S., de Chaisemartin, L., Ouakrim, H., Becht, E., Alifano, M. and Validire, P., 2014. Dendritic cells in tumor-associated tertiary lymphoid structures signal a Th1 cytotoxic immune contexture and license the positive prognostic value of infiltrating CD8⁺ T cells. *Cancer research*, 74(3), pp.705-715.
201. Martinet, L., Filleron, T., Le Guellec, S., Rochaix, P., Garrido, I. and Girard, J.P., 2013. High endothelial venule blood vessels for tumor-infiltrating lymphocytes are associated with Lymphotoxin β -producing dendritic cells in human breast cancer. *The Journal of Immunology*, 191(4), pp.2001-2008.
202. Spalteholz, W., 1914. Über das Durchsichtigmachen von menschlichen und tierischen Präparaten und seine theoretischen Bedingungen, nebst Anhang.
203. Azaripour, A., Lagerweij, T., Scharfbillig, C., Jadcak, A.E., Willershausen, B. and Van Noorden, C.J., 2016. A survey of clearing techniques for 3D imaging of tissues with special reference to connective tissue. *Progress in histochemistry and cytochemistry*, 51(2), pp.9-23.
204. Renier, N., Wu, Z., Simon, D.J., Yang, J., Ariel, P. and Tessier-Lavigne, M., 2014. iDISCO: a simple, rapid method to immunolabel large tissue samples for volume imaging. *Cell*, 159(4), pp.896-910.
205. Roy, L.D., Pathangey, L.B., Tinder, T.L., Schettini, J.L., Gruber, H.E. and Mukherjee, P., 2009. Breast cancer-associated metastasis is significantly increased in a model of autoimmune arthritis. *Breast Cancer Research*, 11(4), p.R56.
206. Lutz, M.B., Kukutsch, N., Ogilvie, A.L., Röbner, S., Koch, F., Romani, N. and Schuler, G., 1999. An advanced culture method for generating large quantities of highly pure dendritic cells from mouse bone marrow. *Journal of immunological methods*, 223(1), pp.77-92.
207. Dieu-Nosjean, M.C., Goc, J., Giraldo, N.A., Sautès-Fridman, C. and Fridman, W.H., 2014. Tertiary lymphoid structures in cancer and beyond. *Trends in immunology*, 35(11), pp.571-580.
208. Zhu, G., Nemoto, S., Mailloux, A.W., Perez-Villaruel, P., Nakagawa, R., Falahat, R., Berglund, A.E. and Mulé, J.J., 2018. induction of Tertiary lymphoid structures With antitumor Function by a lymph node-Derived stromal cell line. *Frontiers in immunology*, 9.
209. Mac Keon, S., Gazzaniga, S., Mallerman, J., Bravo, A.I., Mordoh, J. and Wainstok, R., 2010. Vaccination with dendritic cells charged with

- apoptotic/necrotic B16 melanoma induces the formation of subcutaneous lymphoid tissue. *Vaccine*, 28(51), pp.8162-8168.
210. Lee, S.Y., Ju, M.K., Jeon, H.M., Jeong, E.K., Lee, Y.J., Kim, C.H., Park, H.G., Han, S.I. and Kang, H.S., 2018. Regulation of tumor progression by programmed necrosis. *Oxidative medicine and cellular longevity*, 2018.
 211. Dieu-Nosjean, M.C., Giraldo, N.A., Kaplon, H., Germain, C., Fridman, W.H. and Sautès-Fridman, C., 2016. Tertiary lymphoid structures, drivers of the anti-tumor responses in human cancers. *Immunological reviews*, 271(1), pp.260-275.
 212. Ruffell, B., Au, A., Rugo, H.S., Esserman, L.J., Hwang, E.S. and Coussens, L.M., 2012. Leukocyte composition of human breast cancer. *Proceedings of the National Academy of Sciences*, 109(8), pp.2796-2801.
 213. Meng, S., Li, L., Zhou, M., Jiang, W., Niu, H. and Yang, K., 2018. Distribution and prognostic value of tumor-infiltrating T cells in breast cancer. *Molecular medicine reports*, 18(5), pp.4247-4258.
 214. Lorusso, G. and Rüegg, C., 2008. The tumor microenvironment and its contribution to tumor evolution toward metastasis. *Histochemistry and cell biology*, 130(6), pp.1091-1103.
 215. Yun, S.H. and Kwok, S.J., 2017. Light in diagnosis, therapy and surgery. *Nature biomedical engineering*, 1(1), p.0008.
 216. Ramser, K., 2012. Raman spectroscopy of single cells for biomedical applications. *Applications of Raman Spectroscopy to Biology* ed. Mahmoud, G. IOS Press, Amsterdam, The Netherlands, pp.106-147.
 217. Balas, C., 2009. Review of biomedical optical imaging—a powerful, non-invasive, non-ionizing technology for improving in vivo diagnosis. *Measurement science and technology*, 20(10), p.104020.
 218. Smith, R., Wright, K.L. and Ashton, L., 2016. Raman spectroscopy: an evolving technique for live cell studies. *Analyst*, 141(12), pp.3590-3600.
 219. Germond, A., Kumar, V., Ichimura, T., Moreau, J., Furusawa, C., Fujita, H. and Watanabe, T.M., 2017. Raman spectroscopy as a tool for ecology and evolution. *Journal of The Royal Society Interface*, 14(131), p.20170174.
 220. Wood, B.R., Stoddart, P.R. and McNaughton, D., 2011. Molecular imaging of red blood cells by raman spectroscopy. *Australian Journal of Chemistry*, 64(5), pp.593-599.
 221. Das, N., Dai, Y., Liu, P., Hu, C., Tong, L., Chen, X. and Smith, Z., 2017. Raman plus X: biomedical applications of multimodal raman spectroscopy. *Sensors*, 17(7), p.1592.
 222. Song, L., Hennink, E.J., Young, I.T. and Tanke, H.J., 1995. Photobleaching kinetics of fluorescein in quantitative fluorescence microscopy. *Biophysical journal*, 68(6), pp.2588-2600.
 223. Wang, J.H., Bartlett, J.D., Dunn, A.C., Small, S., Willis, S.L., Driver, M.J. and Lewis, A.L., 2005. The use of rhodamine 6G and fluorescence microscopy in the evaluation of phospholipid-based polymeric biomaterials. *Journal of microscopy*, 217(3), pp.216-224.
 224. Berns, M.W., Siemens, A.E. and Walter, R.J., 1984. Mitochondrial fluorescence patterns in rhodamine 6G-stained myocardial cells in vitro. *Cell biophysics*, 6(4), pp.263-277.
 225. He, X.N., Gao, Y., Mahjouri-Samani, M., Black, P.N., Allen, J., Mitchell, M., Xiong, W., Zhou, Y.S., Jiang, L. and Lu, Y.F., 2012. Surface-enhanced Raman

- spectroscopy using gold-coated horizontally aligned carbon nanotubes. *Nanotechnology*, 23(20), p.205702.
226. Li, P., Zhou, X., Liu, H., Yang, L. and Liu, J., 2013. Surface-enhanced Raman evidence for Rhodamine 6 G and its derivative with different adsorption geometry to colloidal silver nanoparticle. *Journal of Raman Spectroscopy*, 44(7), pp.999-1003.
 227. Henry, A.I., Sharma, B., Cardinal, M.F., Kurouski, D. and Van Duyne, R.P., 2016. Surface-enhanced Raman spectroscopy biosensing: in vivo diagnostics and multimodal imaging. *Analytical chemistry*, 88(13), pp.6638-6647.
 228. Willets, K.A., Stranahan, S.M. and Weber, M.L., 2012. Shedding light on surface-enhanced Raman scattering hot spots through single-molecule super-resolution imaging. *The journal of physical chemistry letters*, 3(10), pp.1286-1294.
 229. Egawa, T. and Yeh, S.R., 2005. Structural and functional properties of hemoglobins from unicellular organisms as revealed by resonance Raman spectroscopy. *Journal of inorganic biochemistry*, 99(1), pp.72-96.
 230. Rousseau, D.L. and Ondrias, M.R., 1983. Resonance Raman scattering studies of the quaternary structure transition in hemoglobin. *Annual review of biophysics and bioengineering*, 12(1), pp.357-380.
 231. Hoey, S., Brown, D.H., McConnell, A.A., Smith, W.E., Marabani, M. and Sturrock, R.D., 1988. Resonance Raman spectroscopy of hemoglobin in intact cells: a probe of oxygen uptake by erythrocytes in rheumatoid arthritis. *Journal of inorganic biochemistry*, 34(3), pp.189-199.
 232. Wood, B.R., Caspers, P., Puppels, G.J., Pandiancherri, S. and McNaughton, D., 2007. Resonance Raman spectroscopy of red blood cells using near-infrared laser excitation. *Analytical and bioanalytical chemistry*, 387(5), pp.1691-1703.
 233. Snook, R.D., Harvey, T.J., Correia Faria, E. and Gardner, P., 2008. Raman tweezers and their application to the study of singly trapped eukaryotic cells. *Integrative Biology*, 1(1), pp.43-52.
 234. Eggeling, C., Volkmer, A. and Seidel, C.A., 2005. Molecular photobleaching kinetics of rhodamine 6G by one-and two-photon induced confocal fluorescence microscopy. *ChemPhysChem*, 6(5), pp.791-804.
 235. Talib, A.J., Fisher, A., Voronine, D.V., Sinyukov, A.M., Lopez, S.C.B., Ambardar, S., Meissner, K.E., Scully, M.O. and Sokolov, A.V., 2019. Fluorescence imaging of stained red blood cells with simultaneous resonance Raman photostability analysis. *Analyst*, 144(14), pp.4362-4370.
 236. Jain, N.C., 1993. *Essentials of veterinary hematology* (No. SF769. 5 J25).
 237. Savitzky, A. and Golay, M.J., 1964. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, 36(8), pp.1627-1639.
 238. Člupek, M., Matějka, P. and Volka, K., 2007. Noise reduction in Raman spectra: Finite impulse response filtration versus Savitzky–Golay smoothing. *Journal of Raman Spectroscopy: An International Journal for Original Work in all Aspects of Raman Spectroscopy, Including Higher Order Processes, and also Brillouin and Rayleigh Scattering*, 38(9), pp.1174-1179.
 239. Siano, D.B. and Metzler, D.E., 1969. Band shapes of the electronic spectra of complex molecules. *The Journal of Chemical Physics*, 51(5), pp.1856-1861.
 240. Wood, B.R. and McNaughton, D., 2002. Raman excitation wavelength investigation of single red blood cells in vivo. *Journal of Raman Spectroscopy*, 33(7), pp.517-523.

241. Bankapur, A., Zachariah, E., Chidangil, S., Valiathan, M. and Mathur, D., 2010. Raman tweezers spectroscopy of live, single red and white blood cells. *PLoS one*, 5(4), p.e10427.
242. Rusciano, G., De Luca, A., Pesce, G. and Sasso, A., 2008. Raman tweezers as a diagnostic tool of hemoglobin-related blood disorders. *Sensors*, 8(12), pp.7818-7832.
243. Asghari-Khiavi, M., Mechler, A., Bambery, K.R., McNaughton, D. and Wood, B.R., 2009. A resonance Raman spectroscopic investigation into the effects of fixation and dehydration on heme environment of hemoglobin. *Journal of Raman Spectroscopy: An International Journal for Original Work in all Aspects of Raman Spectroscopy, Including Higher Order Processes, and also Brillouin and Rayleigh Scattering*, 40(11), pp.1668-1674.
244. Wahadoszamen, M., Rahaman, A., Hoque, N.M., I Talukder, A., Abedin, K.M. and Haider, A.F.M., 2015. Laser Raman spectroscopy with different excitation sources and extension to surface enhanced Raman spectroscopy. *Journal of Spectroscopy*, 2015.
245. Girish, V. and Vijayalakshmi, A., 2004. Affordable image analysis using NIH Image/ImageJ. *Indian journal of cancer*, 41(1), p.47.
246. Ramadhani, D., Rahardjo, T. and Nurhayati, S., 2012. Automated Measurement of Haemozoin (Malarial Pigment) Area in Liver Histology Using ImageJ 1.6. In *Proceeding of 6th Electrical Power, Electronics Communication, Control and Informatics Seminar (EECCIS)*, Malang.
247. Riddle, A., Dean, J., Buser, J.R., Gong, X., Maire, J., Chen, K., Ahmad, T., Cai, V., Nguyen, T., Kroenke, C.D. and Hohimer, A.R., 2011. Histopathological correlates of magnetic resonance imaging-defined chronic perinatal white matter injury. *Annals of neurology*, 70(3), pp.493-507.
248. Vrekoussis, T., Chaniotis, V., Navrozoglou, I., Dousias, V., Pavlakis, K., Stathopoulos, E.N. and Zoras, O., 2009. Image analysis of breast cancer immunohistochemistry-stained sections using ImageJ: an RGB-based model. *Anticancer research*, 29(12), pp.4995-4998.
249. Dundar, M.M., Badve, S., Bilgin, G., Raykar, V., Jain, R., Sertel, O. and Gurcan, M.N., 2011. Computerized classification of intraductal breast lesions using histopathological images. *IEEE Transactions on Biomedical Engineering*, 58(7), pp.1977-1984.
250. Haka, A.S., Shafer-Peltier, K.E., Fitzmaurice, M., Crowe, J., Dasari, R.R. and Feld, M.S., 2005. Diagnosing breast cancer by using Raman spectroscopy. *Proceedings of the National Academy of Sciences*, 102(35), pp.12371-12376.

APPENDIX A

Appendix A shows the data analysis software for the static chamber studies written in MATLAB (2018a). Full code downloadable from

<https://github.com/afishtex/APPLICATION-OF-OPTICAL-ANALYSES-TO-CANCER-THERAPEUTICS-AND-DIAGNOSTICS>.

```
% Author: Andrew P. Fisher
% Centre of NanoHealth, College of Science,
% Swansea University, Singleton Park, Swansea, SA2 8PP, United Kingdom
```

```
%% load images
date = 20171017;
psm = 'APTES';
media = 'PBS\';
row = 1;
col = 75;
passage = 7;
fileext = '.nd2';
obj = 20;
umpxl = 0.18; %um/pixel at 20x
psmSplit = strsplit(psm);
if length(psmSplit) > 1
    psmSplit{2} = [' ' psmSplit{2}];
else
    psmSplit{2} = '';
end
filepath = ['S:\Andrew_Fisher\Static Chamber\' num2str(date) '\Brownian Motion\'...
    media psm '\']; %starting folder
savepath = [filepath 'Analysis\'];
xlsavepath = 'S:\Andrew_Fisher\Static Chamber\Particle Motion Analysis.xlsx';
numOfVids = 1;
i = 1;
while i <= numOfVids
    %% load particle locations
    %filenamestd = ['Stimulated P' num2str(passage) ' HUVEC w ' psmSplit{1}...
        % ' PSM ' num2str(obj) 'x' num2str(i-1)];
    filenamestd = [psmSplit{1} ' PSM ' num2str(obj) 'x' num2str(i-1)];

    myFileFolderInfo = dir([filepath '*' fileext]);
    myFileFolderInfo([myFileFolderInfo.bytes] < 10000000) = [];
    numOfVids = length(myFileFolderInfo);
    filename = [filepath myFileFolderInfo(i).name];
    res = [1440 1920];
    [filtBW, BWprops, times, delta, framerate] = ...
        LoadVideoND2_v2(filename, filenamestd, savepath, res);
    numberOfFrames = length(times(:,1));
    %rename bf images to all match
    cellfilename = [filepath psmSplit{1} ' FITC' psmSplit{2} '.tif'];

    cellimg = imread(cellfilename);

    %% select and save roi
    savename1 = [filenamestd ' roi.mat'];
```

```

savename2 = [filenamestd ' masks.mat'];
if exist([savepath savename2], 'file') ~= 2
    masks = CellMasks_v2(imresize(elling, res));
    roiF = getframe;
    roiI = frame2im(roiF);
    roi = imresize(roiI,res);
    save([savepath savename1], 'roi');
    save([savepath savename2], 'masks');
else
    load([savepath savename1]);
    load([savepath savename2]);
end

%% track particles
savename4 = [filenamestd ' distances-20.mat'];
savename10 = [filenamestd ' numParticles-20.mat'];
if exist([savepath savename4], 'file') ~= 2
    startloc = length(BWprops);
    startparticles = 0;
    distances = TrackParticle(BWprops, startloc, startparticles, startparticles, masks);
    numParticles = max(distances(1,:));
    save([savepath savename4], 'distances');
    save([savepath savename10], 'numParticles');
else
    load([savepath savename4]);
    load([savepath savename10]);
end

%% measure segment distances
savename5 = [filenamestd ' onDist-20.mat'];
savename6 = [filenamestd ' offDist-20.mat'];
savename7 = [filenamestd ' onCount-20.mat'];
savename8 = [filenamestd ' offCount-20.mat'];
if exist([savepath savename5], 'file') ~= 2
    onDist = NaN(1,length(distances(1,:)));
    offDist = NaN(1,length(distances(1,:)));
    onCount = 0;
    offCount = 0;
    for k = 1:max(distances(1,:))
        tempdist = distances(:, distances(1,:) == k);
        tempdistoncell = tempdist(:, tempdist(3,:) == 1);
        tempdistoffcell = tempdist(:, tempdist(3,:) == 0);
        if ~isempty(tempdistoncell)
            for n = 1:length(tempdistoncell(1,:))-1
                if tempdistoncell(2,n) == tempdistoncell(2,n+1) - 1
                    onCount = onCount+1;
                    onDist(onCount) = tempdistoncell(4,n);
                end
            end
        end
        if ~isempty(tempdistoffcell)
            for m = 1:length(tempdistoffcell(1,:))-1
                if tempdistoffcell(2,m) == tempdistoffcell(2,m+1) - 1
                    offCount = offCount+1;
                    offDist(offCount) = tempdistoffcell(4,m);
                end
            end
        end
    end
end

```

```

end
onDist = onDist(~isnan(onDist));
offDist = offDist(~isnan(offDist));
onAvg = mean(onDist);
onStd = std(onDist);
offAvg = mean(offDist);
offStd = std(offDist);
save([savepath savename5], 'onAvg');
save([savepath savename6], 'offAvg');
save([savepath savename7], 'onCount');
save([savepath savename8], 'offCount');
else
load([savepath savename5]);
load([savepath savename6]);
load([savepath savename7]);
load([savepath savename8]);
end

%% Total Distance Travelled from origin
savename11 = [filenamestd ' totOnDistFromOrigin-20.mat'];
savename12 = [filenamestd ' totOffDistFromOrigin-20.mat'];
savename13 = [filenamestd ' meanOnDistFromOrigin-20.mat'];
savename14 = [filenamestd ' meanOffDistFromOrigin-20.mat'];
if exist([savepath savename11], 'file') ~= 2
totOnDistFromOrigin = nan(3,numParticles);
totOffDistFromOrigin = nan(3,numParticles);
for q = 1:numParticles
tempdist = distances(:,distances(1,:)==q);
if length(tempdist(3,:))>1
if nnz(tempdist(3,:)) == length(tempdist(3,:))
totOnDistFromOrigin(1,q) = q;
totOnDistFromOrigin(2,q) = pdist([tempdist(5,end) tempdist(6,end)];...
[tempdist(5,1) tempdist(6,1)]);
%check to see if the particle moves farther than its longest
%step to tell if the particle is stuck
if totOnDistFromOrigin(2,q) <= 2*max(tempdist(4,:))
totOnDistFromOrigin(3,q) = 1;
end
elseif nnz(~tempdist(3,:)) == length(tempdist(3,:))
totOffDistFromOrigin(1,q) = q;
totOffDistFromOrigin(2,q) = pdist([tempdist(5,end) tempdist(6,end)];...
[tempdist(5,1) tempdist(6,1)]);
%check to see if the particle moves farther than its longest
%step to tell if the particle is stuck
if totOffDistFromOrigin(2,q) <= 2*max(tempdist(4,:))
totOffDistFromOrigin(3,q) = 1;
end
end
end
end
end
totOnDistFromOrigin = totOnDistFromOrigin(:,~isnan(totOnDistFromOrigin(1,:)));
meanOnDistFromOrigin = mean(totOnDistFromOrigin(2,:));
totOffDistFromOrigin = totOffDistFromOrigin(:,~isnan(totOffDistFromOrigin(1,:)));
meanOffDistFromOrigin = mean(totOffDistFromOrigin(2,:));
save([savepath savename11], 'totOnDistFromOrigin');
save([savepath savename12], 'totOffDistFromOrigin');
save([savepath savename13], 'meanOnDistFromOrigin');
save([savepath savename14], 'meanOffDistFromOrigin');

```



```

else
    load([savepath savename11]);
    load([savepath savename12]);
    load([savepath savename13]);
    load([savepath savename14]);
end

%% particle color changer
savename9 = [filenamestd 'filtBW.mat'];
if exist([savepath savename9], 'file') ~= 2
    for p = length(distances(1,:))-1:1
        if distances(3,p) == 1
            j = 1;
            while j <= length(BWprops{distances(2,p)})
                if BWprops{distances(2,p)}(j).Centroid(1) == distances(5,p) &&...
                    BWprops{distances(2,p)}(j).Centroid(2) == distances(6,p)
                    filtBW{distances(2,p)}(BWprops{distances(2,p)}(j).PixelIdxList+...
                        (res(1)*res(2)))=1;
                    filtBW{distances(2,p)}(BWprops{distances(2,p)}(j).PixelIdxList)=0;
                    j = length(BWprops{distances(2,p)}) + 1;
                end
                j = j+1;
            end
            if j ~= length(BWprops{distances(2,p)}) + 2
                sprintf('if failed at %d, %d', p,j);
            end
        else
            k = 1;
            while k <= length(BWprops{distances(2,p)})
                if BWprops{distances(2,p)}(k).Centroid(1) == distances(5,p) &&...
                    BWprops{distances(2,p)}(k).Centroid(2) == distances(6,p)
                    filtBW{distances(2,p)}(BWprops{distances(2,p)}(k).PixelIdxList+...
                        (res(1)*res(2)))=0;
                    filtBW{distances(2,p)}(BWprops{distances(2,p)}(k).PixelIdxList)=1;
                    k = length(BWprops{distances(2,p)}) + 1;
                end
                k = k+1;
            end
            if k ~= length(BWprops{distances(2,p)}) + 2
                sprintf('else failed at %d, %d', p,k);
            end
        end
    end
    save([savepath savename9], 'filtBW');
else
    load([savepath savename9]);
end

%% save blended video
savename3 = ['color Blended ' filenamestd '.avi'];
if exist([savepath savename3], 'file') ~= 2
    vw = VideoWriter([savepath savename3]);
    vw.FrameRate = framerate;
    open(vw);
    for n = 1:numberOfFrames-1 %% change when changing folders
        video = imfuse(roi,filtBW{n},'blend');
        writeVideo(vw, video);
    end
end

```

```

        close(vw);
    end
    xlsxwrite(xlsavepath, {numParticles offAvg}, num2str(date), [char(col) num2str(row+i)...
        ':' char(col+1) num2str(row+i)]);
    xlsxwrite(xlsavepath, {offCount offStd}, num2str(date), [char(col+3) num2str(row+i) ':'...
        char(col+4) num2str(row+i)]);
    i = i+1;
    clearvars distances filtBW BWprops roi roiIroiF video onDist offDist masks
end

```

```

function [ filtBW, BWprops, times, delta, framerate ] = ...
    LoadVideoND2_v2( filename, filenamestd, savepath, res)
%LoadVideoND2_v2 loads videos and identifies particles
%
%% load particle locations
savename4 = [filenamestd ' particle locs.mat'];
savename5 = [filenamestd ' particle props.mat'];
savename6 = [filenamestd ' frame times.mat'];
savename7 = [filenamestd ' frame deltas.mat'];
if exist([savepath savename4], 'file') ~= 2
    BWprops = cell(1,1);
    delta = zeros(1,1);
    reader = bfGetReader(filename);
    %extract filter
    datasize = reader.getImageCount();
    filtBW = cell(datasize,1);
    BWprops = cell(datasize,1);
    times = zeros(datasize, 2);
    %extract timing
    timingmetastr = char(reader.getSeriesMetadata());
    [timingstartindex, timingstopindex] = regexp(timingmetastr, 'timestamp #[0-9]+=[0-9]+.[0-9]+');
    h = waitbar(0, 'Processing...');
    for n = 1:datasize
        %extract timing
        timeframestart = find(timingmetastr(timingstartindex(n):timingstopindex(n)) == '#')...
            +timingstartindex(n);
        timestart = find(timingmetastr(timingstartindex(n):timingstopindex(n)) == '=')...
            +timingstartindex(n);
        times(n,1) = str2double(timingmetastr(timeframestart:timestart-2));
        times(n,2) = str2double(timingmetastr(timestart:timingstopindex(n)));
        %extract frames and change to binary
        img = bfGetPlane(reader, n);
        initialGrey = mat2gray(img);
        gry = imadjust(initialGrey,[mean(initialGrey(:)); mean(initialGrey(:))+...
            3*std(initialGrey(:))], [0; 1]); %change with each image
        filtgry = imfilter(gry,ones(5,5) / 25);
        BW = imbinarize(filtgry,0.5);
        BW2 = bwmorph(BW, 'bridge');
        filtBW{n} = bwareafilt(BW2,[10,400]);
        filtBW{n}(:, :, 2) = zeros(res);
        filtBW{n}(:, :, 3) = zeros(res);
        BWprops{n} = regionprops(filtBW{n}, 'Centroid', 'PixelList', 'PixelIdxList');
        waitbar(n/datasize,h);
    end
    times = sort(times,1); %make sure times are in the correct frame order
    delta = zeros(1,length(times)-1);
    for i = 2:length(times)
        delta(i-1) = times(i,2)-times(i-1,2); %time between each frame
    end
end

```

```

end
framerate = 1/(sum(delta)/length(times)); % frames per second

save([savepath savename4], 'filtBW');
save([savepath savename5], 'BWprops');
save([savepath savename6], 'times');
save([savepath savename7], 'delta');
close(h);
else
load([savepath savename4], 'filtBW');
load([savepath savename5], 'BWprops');
load([savepath savename6], 'times');
load([savepath savename7], 'delta');
framerate = 1/(sum(delta)/length(times)); % frames per second
end
end

```

```

function masks = CellMasks_v2( cellimg )
%CellMasks_v2 creates a binary mask of the cell layer in the field of view

initialGrey = mat2gray(cellimg);
BW = imbinarize(initialGrey,0.05);
masks = bwareafilt(BW,[50,inf]);
BWoutline = bwperim(masks);
Segout = cellimg;
Segout(BWoutline) = max(Segout(:)); %add color
figure, imshow(Segout);
end

```

```

function [ distances, particle, prevparticle, BWprops ]...
= TrackParticle( BWprops, loc, particle, prevparticle, masks, prevCenter )
%TRACKPARTICLE Track brownian motion of particles within a certain radius
%
dbstop if error
try
distances = [];
while ~isempty(BWprops)
if loc<=0
particle = particle+1;
break;
end
if loc > length(BWprops)
particle = particle+1;
loc = length(BWprops);
end
if isempty(BWprops{loc})
BWprops(loc) = [];
loc = loc-1;
if loc ~= length(BWprops)
break;
end
end
else
if loc == length(BWprops)
particle = particle+1;
prevCenter = BWprops{loc}(1).Centroid;
if masks(round(prevCenter(2)), round(prevCenter(1))) == 1

```

```

        OnCell = true;
    else
        OnCell = false;
    end
    firstdistances = [particle; loc; OnCell; NaN;...
        BWprops{loc}(1).Centroid(1); BWprops{loc}(1).Centroid(2)];
    BWprops{loc}(1) = [];
    [prevdistances, particle, prevparticle, BWprops ]...
        = TrackParticle( BWprops, loc-1, particle, prevparticle,...
            masks, prevCenter );
    if isempty(prevdistances)
        firstdistances(1) = NaN;
    else
        firstdistances(1) = prevdistances(1,1);
    end
    distances = [prevdistances firstdistances distances];
else
    for i = 1:length(BWprops{loc})
        if abs(pdist([BWprops{loc}(i).Centroid; prevCenter])) < 20
            if diff([prevparticle particle]) > 1
                particle = prevparticle+1;
            end
            prevparticle = particle;
            if masks(round(BWprops{loc}(i).Centroid(2)),...
                round(BWprops{loc}(i).Centroid(1))) == 1
                OnCell = true;
            else
                OnCell = false;
            end
            distances = [particle; loc; OnCell; pdist([prevCenter;...
                BWprops{loc}(i).Centroid]);...
                BWprops{loc}(i).Centroid(1); BWprops{loc}(i).Centroid(2)];
            prevCenter = BWprops{loc}(i).Centroid;
            BWprops{loc}(i) = [];
            [prevdistances, particle, prevparticle, BWprops ]...
                = TrackParticle( BWprops, loc-1, particle, prevparticle,...
                    masks, prevCenter );
            distances = [prevdistances distances];
            break;
        end
    end
end
break;
end
end
catch ME
    disp(ME.stack(1));
    throw(ME);
end
end

```

APPENDIX B

Appendix B shows the data analysis software for the flow chamber studies written in MATLAB (2018a). The *BFMATLAB* library was utilized from Bio-Formats at The Open Microscopy Environment. Full code downloadable from <https://github.com/afishtex/APPLICATION-OF-OPTICAL-ANALYSES-TO-CANCER-THERAPEUTICS-AND-DIAGNOSTICS>.

```
% Author: Andrew P. Fisher
% Centre of NanoHealth, College of Science,
% Swansea University, Singleton Park, Swansea, SA2 8PP, United Kingdom

%can take a folder of nd2 files and pull out the fluorescent ones and then
%calculate average speed and std of the particles in each with relation to
%its flowrate

dbstop if error
filestart = 'F:\Shared\Data exchange\Fisher\Flow Chamber\';
date = '';
surface = '';
particle = '';
run = 1;
filepath = [filestart date '\' surface particle '\' num2str(run) '\']; %starting folder
fileext = '.nd2';
myFileFolderInfo = dir([filepath '*' fileext]);
numfiles = length(myFileFolderInfo);
savepath = [filepath, 'Analysis1\'];
if ~exist(savepath,'file')
    mkdir(savepath);
end
playVideo = 0; %show video and trace particle/flow orientation
saveVideo = 0;
stoppedParticles = 0;
saveWorkspace = 1;
saveData = 1;
distPerPixel = 0.36; %0.18 um/pixel at 20X, 0.36 um/pixel at 10X
numberOfParticles = zeros(numfiles,1);
avgSpeedPerVid = NaN(numfiles,1);
stdSpeedPerVid = NaN(numfiles,1);
speedums = cell(numfiles,1); %how to declare length of speedums when
% some files have different frame rates
numberOfOnParticles = zeros(numfiles,1);
avgOnSpeedPerVid = NaN(numfiles,1);
stdOnSpeedPerVid = NaN(numfiles,1);
speedonums = cell(numfiles,1);
numberOfOffParticles = zeros(numfiles,1);
avgOffSpeedPerVid = NaN(numfiles,1);
stdOffSpeedPerVid = NaN(numfiles,1);
speedoffums = cell(numfiles,1);
BWprops = cell(numfiles,1);
deltatimes = cell(numfiles,1);
stoppedParticleLocs = cell(numfiles+1,1);
averages = cell(numfiles,1);
onaverages = cell(numfiles,1);
```

```

offaverages = cell(numfiles,1);
totalStoppedParticles = zeros(1,numfiles);
totalStoppedOnParticles = zeros(1,numfiles);
totalStoppedOffParticles = zeros(1,numfiles);
newStoppedParticles = zeros(1,numfiles);
newStoppedOnParticles = zeros(1,numfiles);
newStoppedOffParticles = zeros(1,numfiles);
celling = false(1440, 1920);
excelSheets = {'Number Of Moving Particles/Video' 'Number of Stopped Particles/Video'...
'Number of New Stopped Particles/Video' 'Average Speed/Video' 'Standard Deviation/Video'...
'Average Particle Speed/Video' 'Particle Standard Deviation/Video'...
'Number Of Moving Particles On Cells/Video' 'Number of Stopped Particles On Cells/Video'...
'Number of New Stopped Particles On Cells/Video' 'Average Speed On Cells/Video'...
'Standard Deviation On Cells/Video' 'Average Particle Speed On Cells/Video'...
'Particle Standard Deviation On Cells/Video'...
'Number Of Moving Particles Off Cells/Video' 'Number of Stopped Particles Off Cells/Video'...
'Number of New Stopped Particles Off Cells/Video' 'Average Speed Off Cells/Video'...
'Standard Deviation Off Cells/Video' 'Average Particle Speed Off Cells/Video'...
'Particle Standard Deviation Off Cells/Video'...
'Average Speed/Flow Rate' 'Standard Deviation/Flow Rate' 'Average Particle Speed/Flow Rate'...
'Particle Standard Deviation/Flow Rate'...
'Average Speed On Cells/Flow Rate' 'Standard Deviation On Cells/Flow Rate'...
'Average Particle Speed On Cells/Flow Rate' 'Particle Standard Deviation On Cells/Flow Rate'...
'Average Speed Off Cells/Flow Rate' 'Standard Deviation Off Cells/Flow Rate'...
'Average Particle Speed Off Cells/Flow Rate' 'Particle Standard Deviation Off Cells/Flow Rate' "};
xlsavename = [date ' ' particle ' Particle Speeds Angled Line w gaps.xlsx'];
xlsavepath = [savepath xlsavename];
% crossArea = .254*2.5;
filecount = 1;
startcount = 1;
orientation = 180;
totalstoppedparticleimage = zeros(1440, 1920);
try
while filecount <= numfiles
if myFileFolderInfo(filecount).bytes > 10000000
if filecount == 1
[~, stopindex] = regexp(myFileFolderInfo(filecount).name, '[0-9]+ul-min');
wksavename = ['Workspace ' myFileFolderInfo(filecount).name(1:stopindex)];
wksavepath = [savepath wksavename '.mat'];
end
if ~exist(wksavepath,'file')
tic

[BWprops{filecount}, deltatimes{filecount}, stoppedParticleLocs{filecount+1},...
orientation, totalstoppedparticleimage] = ...
LoadVideoBF(myFileFolderInfo(filecount).folder,...
myFileFolderInfo(filecount).name, playVideo, saveVideo, orientation,...
totalstoppedparticleimage);
toc
elseif filecount == 1
load(wksavepath, 'BWprops', 'deltatimes', 'stoppedParticleLocs', 'orientation',...
'totalstoppedparticleimage');
end
if ~isempty(BWprops{filecount})
[flowstartindex, flowstopindex] = regexp(myFileFolderInfo(filecount).name,...
'[0-9]+ul-min');
flowrate = str2double(myFileFolderInfo(filecount).name(flowstartindex:flowstopindex-6));
if filecount == 1
prevflowrate = flowrate;

```

```

end
if flowrate ~= prevflowrate
    tempspd = [];
    temppartspd = [];
    for i = startcount:filecount-1
        if isempty(speedums{i})
            tempspd = [tempspd NaN];
        else
            tempspd = [tempspd speedums{i}(3,:)];
        end
        if isempty(averages{i})
            temppartspd = [temppartspd NaN];
        else
            temppartspd = [temppartspd averages{i}(2,:)];
        end
    end
    avgSpeedFlowRate = [prevflowrate, mean(tempspd,'omitnan')];
    stdSpeedFlowRate = [prevflowrate, std(tempspd,'omitnan')];
    avgParticleSpeedFlowRate = [prevflowrate, mean(temppartspd,'omitnan')];
    stdParticleSpeedFlowRate = [prevflowrate, std(temppartspd,'omitnan')];

    temponspd = [];
    temponpartspd = [];
    for k = startcount:filecount-1
        if isempty(speedonums{k})
            temponspd = [temponspd NaN];
        else
            temponspd = [temponspd speedonums{k}(3,:)];
        end
        if isempty(onaverages{k})
            temponpartspd = [temponpartspd NaN];
        else
            temponpartspd = [temponpartspd onaverages{k}(2,:)];
        end
    end
    avgOnSpeedFlowRate = [prevflowrate, mean(temponspd,'omitnan')];
    stdOnSpeedFlowRate = [prevflowrate, std(temponspd,'omitnan')];
    avgOnParticleSpeedFlowRate = [prevflowrate, mean(temponpartspd,'omitnan')];
    stdOnParticleSpeedFlowRate = [prevflowrate, std(temponpartspd,'omitnan')];

    tempoffspd = [];
    tempoffpartspd = [];
    for k = startcount:filecount-1
        if isempty(speedoffums{k})
            tempoffspd = [tempoffspd NaN];
        else
            tempoffspd = [tempoffspd speedoffums{k}(3,:)];
        end
        if isempty(offaverages{k})
            tempoffpartspd = [tempoffpartspd NaN];
        else
            tempoffpartspd = [tempoffpartspd offaverages{k}(2,:)];
        end
    end
    avgOffSpeedFlowRate = [prevflowrate, mean(tempoffspd,'omitnan')];
    stdOffSpeedFlowRate = [prevflowrate, std(tempoffspd,'omitnan')];
    avgOffParticleSpeedFlowRate = [prevflowrate, mean(tempoffpartspd,'omitnan')];
    stdOffParticleSpeedFlowRate = [prevflowrate, std(tempoffpartspd,'omitnan')];

```

```

if saveWorkspace
    save(wksavepath);
end
if saveData
    excelSheets(2,22:33) = {avgSpeedFlowRate(2) stdSpeedFlowRate(2)...
        avgParticleSpeedFlowRate(2) stdParticleSpeedFlowRate(2)...
        avgOnSpeedFlowRate(2) stdOnSpeedFlowRate(2) avgOnParticleSpeedFlowRate(2)...
        stdOnParticleSpeedFlowRate(2) avgOffSpeedFlowRate(2) stdOffSpeedFlowRate(2)...
        avgOffParticleSpeedFlowRate(2) stdOffParticleSpeedFlowRate(2)};
    xlswrite(xlsavepath, excelSheets(:,:), num2str(prevflowrate));
end
prevflowrate = flowrate;
startcount = filecount;
excelSheets = {'Number Of Moving Particles/Video'...
    'Number of Stopped Particles/Video' 'Number of New Stopped Particles/Video'...
    'Average Speed/Video' 'Standard Deviation/Video'...
    'Average Particle Speed/Video' 'Particle Standard Deviation/Video'...
    'Number Of Moving Particles On Cells/Video'...
    'Number of Stopped Particles On Cells/Video'...
    'Number of New Stopped Particles On Cells/Video'...
    'Average Speed On Cells/Video' 'Standard Deviation On Cells/Video'...
    'Average Particle Speed On Cells/Video'...
    'Particle Standard Deviation On Cells/Video'...
    'Number Of Moving Particles Off Cells/Video'...
    'Number of Stopped Particles Off Cells/Video'...
    'Number of New Stopped Particles Off Cells/Video'...
    'Average Speed Off Cells/Video' 'Standard Deviation Off Cells/Video'...
    'Average Particle Speed Off Cells/Video'...
    'Particle Standard Deviation Off Cells/Video'...
    'Average Speed/Flow Rate' 'Standard Deviation/Flow Rate'...
    'Average Particle Speed/Flow Rate' 'Particle Standard Deviation/Flow Rate'...
    'Average Speed On Cells/Flow Rate' 'Standard Deviation On Cells/Flow Rate'...
    'Average Particle Speed On Cells/Flow Rate'...
    'Particle Standard Deviation On Cells/Flow Rate'...
    'Average Speed Off Cells/Flow Rate' 'Standard Deviation Off Cells/Flow Rate'...
    'Average Particle Speed Off Cells/Flow Rate'...
    'Particle Standard Deviation Off Cells/Flow Rate' "};
stoppedParticles = 0;
end
[speedums{filecount}, speedonums{filecount}, speedoffums{filecount},...
    numberOfParticles(filecount)] = CalculateSpeed(BWprops{filecount},...
    deltatimes{filecount}, distPerPixel, orientation, cellimg);
if numberOfParticles(filecount)
    [numberOfParticles(filecount), speedums{filecount}, averages{filecount},...
        avgSpeedPerVid(filecount), stdSpeedPerVid(filecount),...
        avgSpeedPerPartPerVid(filecount), stdSpeedPerPartPerVid(filecount)] =...
        AverageSpeeds(flowrate, speedums{filecount});
    [numberOfOnParticles(filecount), speedonums{filecount}, onaverages{filecount},...
        avgOnSpeedPerVid(filecount), stdOnSpeedPerVid(filecount),...
        avgOnSpeedPerPartPerVid(filecount), stdOnSpeedPerPartPerVid(filecount)] =...
        AverageSpeeds(flowrate, speedonums{filecount});
    [numberOfOffParticles(filecount), speedoffums{filecount}, offaverages{filecount},...
        avgOffSpeedPerVid(filecount), stdOffSpeedPerVid(filecount),...
        avgOffSpeedPerPartPerVid(filecount), stdOffSpeedPerPartPerVid(filecount)] =...
        AverageSpeeds(flowrate, speedoffums{filecount});
    %pass cellimg into this to use it to decide on v off cell attachments
    [totalStoppedParticles(filecount), newStoppedParticles(filecount)] =...
        CountNewParticles(stoppedParticleLocs(filecount), stoppedParticleLocs(filecount+1)...
        ,ones(size(cellimg)));
end

```



```

%pass celling into this to use it to decide on v off cell attachments
[totalStoppedOnParticles(filecount), newStoppedOnParticles(filecount)] =...
    CountNewParticles(stoppedParticleLocs(filecount),...
        stoppedParticleLocs(filecount+1),cellimg);
%pass celling into this to use it to decide on v off cell attachments
[totalStoppedOffParticles(filecount), newStoppedOffParticles(filecount)] =...
    CountNewParticles(stoppedParticleLocs(filecount),...
        stoppedParticleLocs(filecount+1),imcomplement(cellimg));
if playVideo
    figure(3);
    curaxes = axes;
    img = zeros(size(initialimage{ 1 }));
    for m = 1:length(initialimage)-1
        tempimg = imbinarize(img,0.5);
        if nonzeros(speedums{ filecount }(2,:) == m)
            row = floor(speedums{ filecount }(5,speedums{ filecount }(2,:) == m));
            col = floor(speedums{ filecount }(4,speedums{ filecount }(2,:) == m));
            for u = 1:length(col)
                x1=row(u)-10;
                x2=row(u)+10;
                y1=col(u)-10;
                y2=col(u)+10;
                if x1 < 1
                    x1=1;
                end
                if y1<1
                    y1=1;
                end
                if x2>length(tempimg(:,1))
                    x2 = length(tempimg(:,1));
                end
                if y2>length(tempimg(1,:))
                    y2 = length(tempimg(1,:));
                end
                tempimg(x1:x2, y1:y2)=1;
                tempimg(x1+3:x2-3, y1+3:y2-3)=0;
            end
        end
        imshow(tempimg, 'Parent', curaxes); %play video
        pause(1/10);
    end
end
excelSheets(filecount-startcount+2,1:21) = { numberOfParticles(filecount)...
    totalStoppedParticles(filecount) newStoppedParticles(filecount)...
    avgSpeedPerVid(filecount) stdSpeedPerVid(filecount)
avgSpeedPerPartPerVid(filecount)...
    stdSpeedPerPartPerVid(filecount)...
    numberOfOnParticles(filecount) totalStoppedOnParticles(filecount)...
    newStoppedOnParticles(filecount) avgOnSpeedPerVid(filecount)
stdOnSpeedPerVid(filecount)...
    avgOnSpeedPerPartPerVid(filecount) stdOnSpeedPerPartPerVid(filecount)...
    numberOfOffParticles(filecount) totalStoppedOffParticles(filecount)...
    newStoppedOffParticles(filecount) avgOffSpeedPerVid(filecount)...
    stdOffSpeedPerVid(filecount) avgOffSpeedPerPartPerVid(filecount)...
    stdOffSpeedPerPartPerVid(filecount) };
if isempty(speedums{ filecount })
    temp = NaN;
else
    temp = speedums{ filecount }(3,:);

```

```

end
if isempty(averages{filecount})
    temppart = NaN;
else
    temppart = averages{filecount}(2,:);
end
excelSheets(1, 35+2*(filecount-startcount)) = {'Raw Data'};
excelSheets(2:length(temp(~isnan(temp)))+1, 35+2*(filecount-startcount)) =...
    num2cell(temp(~isnan(temp)));
excelSheets(1, 36+2*(filecount-startcount)) = {'Averages'};
excelSheets(2:length(temppart(~isnan(temppart)))+1, 36+2*(filecount-startcount)) =...
    num2cell(temppart(~isnan(temppart)));
clearvars temp temppart;
end
filecount=filecount+1;
else
myFileFolderInfo(filecount) = [];
numberOfParticles(filecount) = [];
avgSpeedPerVid(filecount) = [];
stdSpeedPerVid(filecount) = [];
speedums(filecount,:) = [];
numberOfOnParticles(filecount) = [];
avgOnSpeedPerVid(filecount) = [];
stdOnSpeedPerVid(filecount) = [];
speedonums(filecount,:) = [];
numberOfOffParticles(filecount) = [];
avgOffSpeedPerVid(filecount) = [];
stdOffSpeedPerVid(filecount) = [];
speedoffums(filecount,:) = [];
BWprops(filecount,:) = [];
deltatimes(filecount,:) = [];
stoppedParticleLocs(filecount) = [];
averages(:,filecount) = [];
onaverages(:,filecount) = [];
offaverages(:,filecount) = [];
totalStoppedParticles(filecount) = [];
totalStoppedOnParticles(filecount) = [];
totalStoppedOffParticles(filecount) = [];
newStoppedParticles(filecount) = [];
newStoppedOnParticles(filecount) = [];
newStoppedOffParticles(filecount) = [];
numfiles = length(myFileFolderInfo);
end
else
cellfile = regexp(myFileFolderInfo(filecount).name, 'FITC');
if cellfile
    cellimg = LoadCellsConfluent(myFileFolderInfo(filecount).folder,...
        myFileFolderInfo(filecount).name);
end
myFileFolderInfo(filecount) = [];
numberOfParticles(filecount) = [];
avgSpeedPerVid(filecount) = [];
stdSpeedPerVid(filecount) = [];
speedums(filecount,:) = [];
numberOfOnParticles(filecount) = [];
avgOnSpeedPerVid(filecount) = [];
stdOnSpeedPerVid(filecount) = [];
speedonums(filecount,:) = [];
numberOfOffParticles(filecount) = [];

```

```

    avgOffSpeedPerVid(filecount) = [];
    stdOffSpeedPerVid(filecount) = [];
    speedoffums(filecount,:) = [];
    BWprops(filecount,:) = [];
    deltatimes(filecount,:) = [];
    numfiles = length(myFileFolderInfo);
end
end

%%used to capture the data from the final video in a folder
if filecount > 1
    tempspd = [];
    temppartspd = [];
    for k = startcount:filecount-1
        if isempty(speedums{k})
            tempspd = [tempspd NaN];
        else
            tempspd = [tempspd speedums{k}(3,:)];
        end
        if isempty(averages{k})
            temppartspd = [temppartspd NaN];
        else
            temppartspd = [temppartspd averages{k}(2,:)];
        end
    end
    avgSpeedFlowRate = [prevflowrate, mean(tempspd,'omitnan')];
    stdSpeedFlowRate = [prevflowrate, std(tempspd,'omitnan')];
    avgParticleSpeedFlowRate = [prevflowrate, mean(temppartspd,'omitnan')];
    stdParticleSpeedFlowRate = [prevflowrate, std(temppartspd,'omitnan')];

    temponspd = [];
    temponpartspd = [];
    for k = startcount:filecount-1
        if isempty(speedonums{k})
            temponspd = [temponspd NaN];
        else
            temponspd = [temponspd speedonums{k}(3,:)];
        end
        if isempty(onaverages{k})
            temponpartspd = [temponpartspd NaN];
        else
            temponpartspd = [temponpartspd onaverages{k}(2,:)];
        end
    end
    avgOnSpeedFlowRate = [prevflowrate, mean(temponspd,'omitnan')];
    stdOnSpeedFlowRate = [prevflowrate, std(temponspd,'omitnan')];
    avgOnParticleSpeedFlowRate = [prevflowrate, mean(temponpartspd,'omitnan')];
    stdOnParticleSpeedFlowRate = [prevflowrate, std(temponpartspd,'omitnan')];

    tempoffspd = [];
    tempoffpartspd = [];
    for k = startcount:filecount-1
        if isempty(speedoffums{k})
            tempoffspd = [tempoffspd NaN];
        else
            tempoffspd = [tempoffspd speedoffums{k}(3,:)];
        end
        if isempty(offaverages{k})
            tempoffpartspd = [tempoffpartspd NaN];
        end
    end
end

```

```

else
    tempoffpartspd = [tempoffpartspd offaverages{k}(2,:);
end
end
avgOffSpeedFlowRate = [prevflowrate, mean(tempoffspd,'omitnan')];
stdOffSpeedFlowRate = [prevflowrate, std(tempoffspd,'omitnan')];
avgOffParticleSpeedFlowRate = [prevflowrate, mean(tempoffpartspd,'omitnan')];
stdOffParticleSpeedFlowRate = [prevflowrate, std(tempoffpartspd,'omitnan')];

if saveWorkspace
    save(wksavepath);
end
if saveData
    excelSheets(2,22:33) = {avgSpeedFlowRate(2) stdSpeedFlowRate(2)...
        avgParticleSpeedFlowRate(2) stdParticleSpeedFlowRate(2)...
        avgOnSpeedFlowRate(2) stdOnSpeedFlowRate(2) avgOnParticleSpeedFlowRate(2)...
        stdOnParticleSpeedFlowRate(2)...
        avgOffSpeedFlowRate(2) stdOffSpeedFlowRate(2) avgOffParticleSpeedFlowRate(2)...
        stdOffParticleSpeedFlowRate(2)};
    xlswrite(xlsavepath, excelSheets(:,:), num2str(prevflowrate));
end
end
allPeak = PlotHistogram(averages);
if allPeak
    title({filepath; 'All Averages'});
    savefig([savepath 'AllAverageParticleSpeedHistogram']);
end
onPeak = PlotHistogram(onaverages);
if onPeak
    title({filepath; 'On Averages'});
    savefig([savepath 'OnAverageParticleSpeedHistogram']);
end
offPeak = PlotHistogram(offaverages);
if offPeak
    title({filepath; 'Off Averages'});
    savefig([savepath 'OffAverageParticleSpeedHistogram']);
end
normalizedtotalstoppedparticleimage = mat2gray(totalstoppedparticleimage/numfiles);
bwnormalizedtotalstoppedparticleimage = imbinarize(normalizedtotalstoppedparticleimage, 0.25);
stoppedparticleregions = regionprops(bwnormalizedtotalstoppedparticleimage,...
    normalizedtotalstoppedparticleimage, 'WeightedCentroid', 'MeanIntensity');
histogram([stoppedparticleregions.MeanIntensity].*100,...
    ceil(max([stoppedparticleregions.MeanIntensity].*100) -...
    min([stoppedparticleregions.MeanIntensity].*100)));
xlabel('% of Video Stationary');
ylabel('Number of Particles');
if ~isempty(stoppedparticleregions)
    title({filepath; 'Length of Stopped Particles'});
    savefig([savepath 'LengthOfStoppedParticles']);
end

catch ME
    disp(ME.stack(1));
    throw(ME);
end

function [ BWprops, delta, stoppedparticleprops, orientation, totalstoppedparticleimage ] =...
    LoadVideoBF( path, name, vid, saveV, orientation, totalstoppedparticleimage )

```

```

%LoadVideoBF Loads the video into matlab
% also detects particle locations in each frame
dbstop if error
filename = [path '\' name];
try
    BWprops = cell(1,1);
    delta = zeros(1,1);
    reader = bfGetReader(filename);
    filtermetastr = char(reader.getGlobalMetadata());
    %extract filter
    fluor = regexp(filtermetastr, 'BF1');
    clearvars filtermetastr filename;
    if fluor %check difference between TRITC and BF
        datasize = reader.getImageCount();
        img = cell(datasize,1);
        filtBW3 = cell(datasize,1);
        BWprops = cell(datasize,1);
        times = zeros(datasize, 2);
        %extract timing
        timingmetastr = char(reader.getSeriesMetadata());
        [timingstartindex, timingstopindex] = regexp(timingmetastr,...
            'timestamp #[0-9]+=[0-9]+.[0-9]+);

        if vid
            figure(1);
            currAxes = axes;
            figure(2);
            otherAxes = axes;
        end
        h = waitbar(0,[name ' Please wait...']);
        for n = 1:datasize
            %extract timing
            timeframestart = find(timingmetastr(timingstartindex(n):timingstopindex(n)) == '#')...
                +timingstartindex(n);
            timestart = find(timingmetastr(timingstartindex(n):timingstopindex(n)) == '=')...
                +timingstartindex(n);
            times(n,1) = str2double(timingmetastr(timeframestart:timestart-2));
            times(n,2) = str2double(timingmetastr(timestart:timingstopindex(n)));
            %extract frames and change to binary
            img = imcomplement(bfGetPlane(reader, n));

            initialGrey = mat2gray(img);
            BW = imbinarize(initialGrey,mean(initialGrey(:))+3.5*std(initialGrey(:)));
            %a little more than a 1um particle up to a 3um particle area
            filtBW = bwareafilt(BW,[40,100]);
            filtBW2 = bwpropfilt(filtBW,'Eccentricity',[0 0.75]);
            filtBW3{n} = bwpropfilt(filtBW2,'Solidity',[0.9 1]);
            BWprops{n} = regionprops(filtBW3{n}, 'Eccentricity', 'Orientation',...
                'Area', 'Centroid', 'Solidity');
            if vid
                imshow(filtBW3{n}, 'Parent', currAxes); %play video
                imshow(initialGrey, 'Parent', otherAxes);
            end
            waitbar(n/datasize,h);
        end
        close(h);
        if isnan(orientation)
            figure;
            imshow(initialGrey);

```

```

        direction = imline(gca);
        position = wait(direction);
        %y positions swapped due to the y axis being zero in the top
        %left corner and increasing as it goes down
        orientation = atan2d(position(2,2)-position(1,2),position(2,1)-position(1,1));
    end
    times = sort(times,1); %make sure times are in the correct frame order
    delta = zeros(1,length(times)-1);
    for i = 2:length(times)
        delta(i-1) = times(i,2)-times(i-1,2); %time between each frame
    end
    framerate = 1/(sum(delta)/length(times)); %frames per second
    savename = ['Binary ' name(1:end-4)];
    savepath = [path '\Analysis1\' savename];
    if saveV
        v = VideoWriter(savepath);
        v.FrameRate = framerate;
        open(v);
        g = waitbar(0,'Saving Video, Please wait...');
    end
    individualstoppedparticleimage = zeros(1440,1920);
    for j = 1:length(filtBW3)
        if saveV
            writeVideo(v,uint8(filtBW3{j})*255);
            waitbar(j/length(filtBW3),g);
        end
        individualstoppedparticleimage = individualstoppedparticleimage+filtBW3{j};
    end
    normalizedstoppedparticleimage = individualstoppedparticleimage./length(filtBW3);
    totalstoppedparticleimage = totalstoppedparticleimage+normalizedstoppedparticleimage;
    bwstoppedparticle = imbinarize(mat2gray(normalizedstoppedparticleimage));
    bw2stoppedparticle = bwpropfilt(bwstoppedparticle,'Eccentricity',[0 0.5]);
    stoppedparticleprops = regionprops(bw2stoppedparticle, 'Centroid');
    % figure;
    % imshow(bw2stoppedparticle);
    % greystoppedparticleimage = mat2gray(normalizedstoppedparticleimage);
    % individualstoppedparticleregions = regionprops(bw2stoppedparticle,...
    % greystoppedparticleimage, 'WeightedCentroid', 'MeanIntensity');
    % figure;
    % histogram([individualstoppedparticleregions.MeanIntensity],20);
    if saveV
        close(g);
        close(v);
    end
    end
    reader.close();
catch ME
    disp(ME.stack(1));
    throw(ME);
end

end

function [ speed, speedon, speedoff, numparticles ] =...
    CalculateSpeed( BWprops, deltatimes, umPerPix, orient, masks )
%CALCULATESPEED calculates the speed of the tracked particles
%
    dbstop if error

```

```

speed = [];
speedon = [];
speedoff = [];
numparticles = 0;
startparticles = 0;
startloc = length(BWprops);
try
    distance = TrackParticle( BWprops, startloc, startparticles, startparticles, orient);
    if ~isempty(distance)
        speed(1:2,:) = distance(1:2,:);
        speed(4:6,:) = distance(4:6,:);
        for i = 1:length(distance(3,:))
            speed(3,i) = distance(3,i)*umPerPix./deltatimes(distance(2,i));
            if masks(round(speed(5,i)), round(speed(4,i))) == 1
                speedon = [speedon speed(:,i)];
            else
                speedoff = [speedoff speed(:,i)];
            end
        end
        numparticles = speed(1,1);
    else
        speed = NaN(3,1);
    end
catch ME
    disp(ME.stack(1));
    throw(ME);
end
end

```

```

function [ distances, particle, prevparticle, BWprops ] =...
    TrackParticle( BWprops, loc, particle, prevparticle, orient, prevcent, missedparticle )
% TRACKPARTICLE tracks individual particles through multiple frames
% using recursion
dbstop if error
try
    distances = [];
    while ~isempty(BWprops)
        if loc <= 0
            particle = particle+1;
            break;
        end
        if loc > length(BWprops)
            particle = particle+1;
            loc = length(BWprops);
        end
        if isempty(BWprops{loc})
            BWprops(loc) = [];
            loc = loc-1;
            if loc ~= length(BWprops)
                break;
            end
        end
        else
            if loc == length(BWprops)
                particle = particle+1;
                prevcent = BWprops{loc}(1).Centroid;
                BWprops{loc}(1) = [];
                missedparticle = 0;
                [ prevdistances, particle, prevparticle, BWprops ] =...

```

```

        TrackParticle( BWprops, loc-1, particle, prevparticle,...
        orient, prevcent, missedparticle );
    distances = [prevdistances distances];
else
%       figure(4); hold on; plot(prevcent(1), prevcent(2),'x');...
%       xlim([0 1920]); ylim([0 1440]); pause(1/1000);
    a = tand(orient);
    b = -1;
    c = prevcent(2)-tand(orient)*prevcent(1);
    i=1;
    while i <= length(BWprops{loc})
        %because particles are being tracked from the end of
        %the video to the beginning, particles with centers to
        %the left of the last particle should not be
        %considered. particles who are less than 10 pixels off
        %of the slope line should be considered.
        if BWprops{loc}(i).Centroid(1) >= prevcent(1)-10 &&...
            abs(a*BWprops{loc}(i).Centroid(1)+b*BWprops{loc}(i).Centroid(2)+c)...
            /sqrt(a^2+b^2)< 20
            missedparticle = 0;
            if diff([prevparticle particle]) > 1
                particle = prevparticle+1;
            end
            prevparticle = particle;
            distances = [particle; loc; pdist([prevcent; BWprops{loc}(i).Centroid]);...
                min([max([BWprops{loc}(i).Centroid(1) 1] 1440)];...
                min([max([BWprops{loc}(i).Centroid(2) 1] 1920)]; BWprops{loc}(i).Eccentricity];

            prevcent = BWprops{loc}(i).Centroid;
            BWprops{loc}(i) = [];
            [ prevdistances, particle, prevparticle, BWprops ] = TrackParticle( BWprops, loc-1,...
                particle, prevparticle, orient, prevcent, missedparticle );
            distances = [prevdistances distances];
            %prevent code from hitting missed particles after successfully finding a particle
            missedparticle = 1;
            break;
        end
        i=i+1;
    end
    if loc <= length(BWprops) && i == length(BWprops{loc})+1 && ~missedparticle
        missedparticle = 1;
        if diff([prevparticle particle]) > 1
            particle = prevparticle+1;
        end
        distances = [particle; loc; NaN; prevcent(1); prevcent(2); NaN];
        [ prevdistances, particle, prevparticle, BWprops ] =...
            TrackParticle( BWprops, loc-1, particle, prevparticle,...
                orient, prevcent, missedparticle );
        if isempty(prevdistances)
            distances = [];
            break;
        end
        prevdistances(3,end) = prevdistances(3,end)/2;
        distances(3) = prevdistances(3,end);
        distances(4:6) = [min([max([prevdistances(4,end)+cosd(orient)*distances(3) 1] 1440)];...
            min([max([prevdistances(5,end)+sind(orient)*distances(3) 1] 1920)];
prevdistances(6,end)];
        distances = [prevdistances distances];
    end
end

```



```

        break;
    end
end
end
catch ME
    disp(ME.stack(1));
    throw(ME);
end
end
end

```

```

function [ numberOfParticles, speed, averages, avgSpeedPerVid, stdSpeedPerVid,
avgSpeedPerPartPerVid,...
    stdSpeedPerPartPerVid ] = AverageSpeeds( flowrate, speed )
%AverageSpeeds averages the speeds of individual particles over the entire
% time they're in the field of view
dbstop if error
try
    if isempty(speed)
        %totalStoppedParticles = 0;
        numberOfParticles = 0;
        averages = [];
        avgSpeedPerVid = NaN;
        stdSpeedPerVid = NaN;
        avgSpeedPerPartPerVid = NaN;
        stdSpeedPerPartPerVid = NaN;
    else
        %  $V = (3/4)Q(H^2-y^2)/(WH^3)$  um/s within 10um of the surface (cell height between 1-4
um in vitro;
        %http://www.sciencedirect.com/science/article/pii/S0006349500765714) DOF ~ 700nm
        maximum = ((3/4)*(flowrate*10^9/60)*(127^2-117^2))/(2500*127^3);
        minimum = 3.6;% if it moves farther than 1.8 um, it cannot move back because of the
        % 10 pixel backward limit, doubled for insurance. cannot use equation because
        % the whole point is to account for drag on the cells

        speed(:,speed(3,:)<minimum)=NaN;
        %removes velocities moving faster than the flow profile would allow
        speed(:,speed(3,:)>maximum)=NaN;
        %removes outliers in hopes of removing false information from the datasets
        speed(:,isoutlier(speed(3,:))) = NaN;
        p = 1;
        tempspeeds = [];
        averages = [];
        crowspeedums = speed(:,~isnan(speed(1,:)));
        while p < length(crowspeedums(1,:))
            k = crowspeedums(1,p);
            while p < length(crowspeedums(1,:)) && crowspeedums(1,p) == k
                tempspeeds = [tempspeeds crowspeedums(3,p)];
                p = p+1;
            end
            if length(tempspeeds)>1 %ensure there is enough points to create a reasonable average
                averages = [averages [k;mean(tempspeeds)]];
            end
            tempspeeds = [];
        end
        numberOfParticles = length(averages);
        if ~isempty(speed)
            avgSpeedPerVid = mean(mean(speed(3,:),'omitnan'),'omitnan');
            stdSpeedPerVid = std(speed(3,:),'omitnan');
        end
    end
end

```

```

else
    avgSpeedPerVid = NaN;
    stdSpeedPerVid = NaN;
end
if ~isempty(averages)
    avgSpeedPerPartPerVid = mean(averages(2,:));
    stdSpeedPerPartPerVid = std(averages(2,:));
else
    avgSpeedPerPartPerVid = NaN;
    stdSpeedPerPartPerVid = NaN;
end
end
catch ME
    disp(ME.stack(1));
    throw(ME);
end
end
end

```

```

function [totalparticles, newparticles] = CountNewParticles( old, present, cellimg )
%CountNewParticles counts how many new particles have become stuck in a one
%minute window
n = 1;
while n <= length(old{1})
    temp = old{1};
    if ~cellimg(round(temp(n).Centroid(2)), round(temp(n).Centroid(1)))
        old{1}(n) = [];
        n = n-1;
    end
    n = n+1;
end
m = 1;
while m <= length(present{1})
    temp = present{1};
    if ~cellimg(round(temp(m).Centroid(2)), round(temp(m).Centroid(1)))
        present{1}(m) = [];
        m = m-1;
    end
    m = m+1;
end
newparticles = length(present{1});
if ~isempty(old{1})
    for i = 1:length(present{1})
        for j = 1:length(old{1})
            tempres = present{1};
            tempold = old{1};
            if pdist([tempres(i).Centroid; tempold(j).Centroid]) < 10
                newparticles = newparticles-1;
            end
        end
    end
else
    newparticles = length(present{1});
end
totalparticles = length(present{1});
end
end

```

```

function filtBW2 = LoadCellsConfluent( path, name )
%LoadCellsConfluent loads and locates the cell locations in the frame
% based on fluorescent images
dbstop if error
filename = [path '\' name];
try
    reader = bfGetReader(filename);
    %extract frames and change to binary
    img = bfGetPlane(reader, 1);

    initialGrey = mat2gray(img);
    adj = imadjust(initialGrey);
    BW = imbinarize(adj,0.3);
    dilatedImage = imdilate(BW,strel('disk',2));
    BW2 = bwmorph(dilatedImage, 'bridge');
    BW3 = imfill(BW2,'holes');
    filtBW = bwareafilt(BW3,[500,inf]);%a little more than a 1um particle up to a 3um particle area
%    figure;
%    imshow(filtBW); %play video
%    figure;
%    imshow(initialGrey);
    filtBW2 = imresize(filtBW,[1440 1920]);
    reader.close();
catch ME
    disp(ME.stack(1));
    throw(ME);
end

end

```

```

function complete = PlotHistogram( average )
%PlotHistogram plots a histogram of the average speeds of the particles

complete = 0;
lognorm2 = fitype(@ (a,b,c,I0,offset,x) offset+(I0*b./(x-a)).*...
    exp(-c^2).*exp(-(1/(2*c^2))*(log((x-a)/b)).^2));
x = 0:10:300;
n = zeros(length(average(:,1)),length(x));
y = zeros(1,length(average(:,1)));
for u = 1:length(average(:,1))
    if average{u}
        n(u,:) = hist(average{u}(2,:),x);
    end
end
if ~isempty(n)
    for v = 1:length(n(1,:))
        y(v) = mean(n(:,v));
    end
    [F, ~] = fit(x',y',lognorm2,'StartPoint',[mean([min(x)-eps -max(x)]),...
        500,.5,max(y)-mean(y(end-20:end)),mean(y(end-20:end))],...
        'Upper',[min(x)-eps,2000,10,max(y),max(y)],'Lower',[-max(x),eps,eps,0,0]);
    xres = 0:1:300;
    funct = F.offset+(F.I0*F.b./(xres-F.a)).*exp(-F.c^2).*exp(-(1/(2*F.c^2))*...
        (log((xres-F.a)/F.b)).^2);
    [pks, locs] = findpeaks(funct,xres);
    figure;
    bar(x, n');
    legend('show');

```

```
hold on;
plot(F);
hold off;
xlabel('Average Particle Speed (\mum/sec)');
ylabel('Frequency');
text(locs+10, pks, [num2str(locs) ' \mum/s']);
complete = 1;
end
end
```

APPENDIX C

Appendix C shows the data analysis software for the TLS studies written in MATLAB (2018a). The function *bpass* was written by John C. Crocker and David G. Grier at the University of Chicago in 1997 and the function *pkfnd* was created by Eric R. Dufresne at Yale University in 2005. Both are freely distributed for use by the public. Full code downloadable from <https://github.com/afishtex/APPLICATION-OF-OPTICAL-ANALYSES-TO-CANCER-THERAPEUTICS-AND-DIAGNOSTICS>.

```
% Author: Andrew P. Fisher
% Centre of NanoHealth, College of Science,
% Swansea University, Singleton Park, Swansea, SA2 8PP, United Kingdom

samples = ['1.1A'; '1.1B'; '1.2A'; '1.3B'; '2.1A'; '2.1B'; '2.2A'; '2.2B';...
          '2.3A'; '2.3B'; '3.1A'; '3.1B'; '3.2A'; '3.2B'; '3.3A'; '3.3B'; '4.1A';...
          '4.1B'; '4.2A'; '4.2B'; '4.3B'; '5.1B'; '5.2A'; '5.2B'; '5.3A'; '5.3B'; '5.4A'];

toPlot = 0;
for i = 1:length(samples(:,1))
    tertiaryLymphoidStructure(samples(i,:), toPlot);
end

function tertiaryLymphoidStructure(sample,toPlot)
    version = 'v2';
    filepath = 'C:\Users\Andrew\Documents\Clearing TLS\';
    loadpath = [filepath 'tifs\Raw\'];
    savepath = [filepath version '\Analysis\' sample '\'];
    if ~exist(savepath, 'dir')
        mkdir(savepath);
    end
    filename = [sample];
    file = [loadpath filename];
    savefile = [savepath sample(1) '-' sample(3:end) ' filtered'];
    cellDiameterPx1 = 12;
    zStep = 10; %microns between images
    %number of consecutive planes that are required to consider a cell to be a cell
    numPlanes = 2;

    %% Load planes
    [planecentersDAPI, binaryFITC, planecentersTRITC, planecentersCy5,...
     voxel_size, imageSpecs, mmVolume, tumorMask] =...
        loadTifImages(file, cellDiameterPx1, savefile, zStep);
    boxDim = [ceil(100/voxel_size(1)),ceil(100/voxel_size(1)),10];
            %[100 um X 100 um X 100 um]

    %% Find areas that are likely cells
    if ~exist([savefile ' centers3DDAPI.mat'],'file') || ~exist([savefile...
        ' centers3DTRITC.mat'],'file') || ~exist([savefile...
        ' centers3DCy5.mat'],'file') || ~exist([savefile...
        ' skelFITC.mat'],'file')

        h = waitbar(0, [sample ': Finding DAPI Cells...']);
```

```

centers3DDAPI = findCenters3D(imageSpecs, planecentersDAPI,...
    cellDiameterPxl, numPlanes);

waitbar(0.25, h, [sample ': Finding TRITC Cells...']);
centers3DTRITC = findCenters3D(imageSpecs, planecentersTRITC,...
    cellDiameterPxl, numPlanes);

waitbar(0.5, h, [sample ': Finding Cy5 Cells...']);
centers3DCy5 = findCenters3D(imageSpecs, planecentersCy5,...
    cellDiameterPxl, numPlanes);

centers3DDAPI(~tumorMask)=false;
centers3DTRITC(~tumorMask)=false;
centers3DCy5(~tumorMask)=false;
binaryFITC(~tumorMask)=false;

waitbar(0.75, h, [sample ': Skeletonizing FITC Vessels...']);
dirtySkelFITC = bwskel(binaryFITC);
cleanedSkelFITC = bwmorph3(dirtySkelFITC, 'clean');
skelFITC = cleanedSkelFITC;

skelprops = regionprops3(skelFITC, 'Volume', 'VoxelIdxList');
for i = 1:height(skelprops)
    %not any smaller than a single cell or any larger than a
    %string of 50 cells
    if skelprops.Volume(i) > 500 || skelprops.Volume(i) < 10

        skelFITC(skelprops.VoxelIdxList{i})=false;
    end
end

save([savefile ' centers3DDAPI.mat'], 'centers3DDAPI');
save([savefile ' centers3DTRITC.mat'], 'centers3DTRITC');
save([savefile ' centers3DCy5.mat'], 'centers3DCy5');
save([savefile ' skelFITC.mat'], 'skelFITC');
close(h);
else
    load([savefile ' centers3DDAPI.mat']);
    load([savefile ' centers3DTRITC.mat']);
    load([savefile ' centers3DCy5.mat']);
    load([savefile ' skelFITC.mat']);
end

clear planecentersDAPI binaryFITC planecentersTRITC planecentersCy5 tumorMask;

%% Plot cells in 3D
marker_size = 0;
h = waitbar(0, [sample ': Plotting DAPI Cells...']);
[marker_size, totalDAPICells] = plot3DCenters(centers3DDAPI, imageSpecs,...
    cellDiameterPxl, marker_size, toPlot, 'b');
waitbar(0.25, h, [sample ': Plotting TRITC Cells...']);
[marker_size, totalTRITCCells] = plot3DCenters(centers3DTRITC, imageSpecs,...
    cellDiameterPxl, marker_size, toPlot, 'r');
waitbar(0.5, h, [sample ': Plotting Cy5 Cells...']);
[marker_size, totalCy5Cells] = plot3DCenters(centers3DCy5, imageSpecs,...
    cellDiameterPxl, marker_size, toPlot, 'm');
waitbar(0.75, h, [sample ': Plotting FITC Vessels...']);
plotVessels(skelFITC, imageSpecs, toPlot);
close(h);

```

```

% hold off;
concDAPI = totalDAPICells/mmVolume;
concTRITC = totalTRITCCells/mmVolume;
concCy5 = totalCy5Cells/mmVolume;
%because its a skeleton and the average diameter of a cell is around 10um
totalFITCCells = sum(sum(sum(skelFITC)))/10;
concFITC = totalFITCCells/mmVolume;

h = waitbar(0, [sample ': Scanning DAPI Cells...']);
% if ~exist([savefile ' boxNumDAPI.mat'],'file')
%   boxNumDAPI = boxConcentrations(centers3DDAPI,boxDim);
%   save([savefile ' boxNumDAPI.mat'],'boxNumDAPI','-v7.3');
% else
%   load([savefile ' boxNumDAPI.mat']);
% end
waitbar(0.25, h, [sample ': Scanning TRITC Cells...']);
if ~exist([savefile ' boxNumTRITC.mat'],'file')
    boxNumTRITC = boxConcentrations(centers3DTRITC,boxDim);
    save([savefile ' boxNumTRITC.mat'],'boxNumTRITC','-v7.3');
else
    load([savefile ' boxNumTRITC.mat']);
end
waitbar(0.5, h, [sample ': Scanning Cy5 Cells...']);
if ~exist([savefile ' boxNumCy5.mat'],'file')
    boxNumCy5 = boxConcentrations(centers3DCy5,boxDim);
    save([savefile ' boxNumCy5.mat'],'boxNumCy5','-v7.3');
else
    load([savefile ' boxNumCy5.mat']);
end
waitbar(0.75, h, [sample ': Scanning FITC Cells...']);
if ~exist([savefile ' boxNumFITC.mat'],'file')
    boxNumFITC = boxConcentrations(skelFITC,boxDim);
    save([savefile ' boxNumFITC.mat'],'boxNumFITC','-v7.3');
else
    load([savefile ' boxNumFITC.mat']);
end
close(h);
%clear skelFITC centers3DTRITC centers3DCy5 centers3DDAPI;
printExcel(sample, [savepath(1:end-6) '\Combined Hot Spots' sample '.xlsx'],...
    boxDim, totalTRITCCells,totalCy5Cells,totalFITCCells,mmVolume,...
    boxNumTRITC,boxNumCy5,boxNumFITC,centers3DTRITC,centers3DCy5,...
    skelFITC,toPlot);
end

```

```

function [ centersDAPI, binaryFITC, centersTRITC, centersCy5, voxel_size,...
    imageSpecs, mmVolume, tumorMask ] = loadTifImages( file, cellSizePx1,...
    savefile, zStep )
%loadTifImages Searches for cell sized fluorescence
% outputs center values for all colors
oneroundavgmin = 0;
info = imfinfo([file '.tif']);
%% extract dimensions
numImages = length(info);
umPx1 = 1.05;%microns/pixel
voxel_size = [umPx1, zStep]; %microns
frameWidth = info(1).Width; % pixels
frameLength = info(1).Height; % pixels
umVolume = 0;

```

```

numberOfColors = 4;
planes = numImages/numberOfColors;
imageSpecs = [frameWidth, frameLength, planes];
tumorMask = false(imageSpecs);
centersDAPI = cell(1,numImages/numberOfColors);
fillFactor = 0.66;

if ~exist([savefile ' centersTRITC.mat'],'file')

%% extract frames and change to binary
centersDAPI = cell(1,numImages/numberOfColors);
binaryFITC = false(frameWidth, frameLength, numImages/numberOfColors);
centersTRITC = cell(1,numImages/numberOfColors);
centersCy5 = cell(1,numImages/numberOfColors);
h = waitbar(0, 'Loading...');
tic;
for n = 1:numImages/numberOfColors
    %% import and identify centers of DAPI images
    %currently only used to find tumour volume shape, adjust to
    %look like other color profiles if necessary
    img1 = imread([file '.tif'],n*4-3);
    initialGreyDAPI = mat2gray(img1);

    Y = imboxfilt(imadjust(initialGreyDAPI),21);
    bw1 = imbinarize(Y);
    clear img1 initialGreyDAPI Y;

    %% Import and find FITC vessels
    img2 = imread([file '.tif'], n*4-2);
    volFITC = mat2gray(img2);
    img3 = imread([file '.tif'], n*4-1);
    volTRITC = mat2gray(img3);
    img4 = imread([file '.tif'], n*4);
    volCy5 = mat2gray(img4);
    %remove background from fitc; channels shouldn't overlap
    [initialGreyFITC,~] = calcComp(volFITC,volTRITC);
    %remove background from tritc; channels shouldn't overlap
    [initialGreyTRITC,~] = calcComp(volTRITC,volFITC);
    %remove background from cy5; channels shouldn't overlap
    initialGreyCy5 = volCy5;
    initialGreyFITC(initialGreyFITC<0) = 0;
    initialGreyTRITC(initialGreyTRITC<0) = 0;
    initialGreyCy5(initialGreyCy5<0) = 0;
    clear img2 img3 img4;

    %filter out small noise, similar to denoising but works between
    %large items
    gry2 = bpass(initialGreyFITC,3,cellSizePx1);
    % threshold of 50 based on choice from 5.2B
    bw2 = logical(imbinarize(gry2,max(gry2(:))/10));
    %from a single cell area to much larger to account for long branches
    %of vasculature all in one plane
    bw2filt = bwareafilt(bw2,[ceil(pi*(cellSizePx1/3)^2) 1000]);

    binaryFITC(:, :, n) = bw2filt;

%     figure; imshow(gry2);
%     figure; imshow(binaryFITC(:, :, n));

```



```

clear A2 B2 J2 gry2 bw2 initialGreyFITC;

%% import and identify centers of TRITC images
gry3 = bpass(initialGreyTRITC,3,cellSizePx1);
centersTRITC{n} = pkfnd(gry3, max(max(gry3))/25,cellSizePx1, fillFactor);
%   radiiTRITC = ones(length(centersTRITC{n}(:,1)),1)*(cellSizePx1/2);
%
%   figure; imshow(gry3);
%   viscircles([centersTRITC{n}(:,2) centersTRITC{n}(:,1)],...
%   radiiTRITC,'Color','b');

clear gry3 bw3 gry3filt initialGreyTRITC;

%% Import and find centers of cy5 images
gry4 = bpass(initialGreyCy5,3,cellSizePx1); % use 3 on next pass
centersCy5{n} = pkfnd(gry4, max(max(gry4))/50,cellSizePx1, fillFactor);
%   radiiCy5 = ones(length(centersCy5{n}(:,1)),1)*(cellSizePx1/2);
%
%   figure; imshow(gry4);
%   viscircles([centersCy5{n}(:,2) centersCy5{n}(:,1)], radiiCy5,'Color','b');

clear img4 bw4 gry4filt initialGreyCy5;

%% show image and digital cell location overlay
%   rgb = zeros(imageSpecs(1), imageSpecs(2),3);
%   rgb(:,1) = initialGreyTRITC;
%   rgb(:,2) = initialGreyFITC;
%   rgb(:,3) = initialGreyCy5;
%   for m = 1:length(centersTRITC{n}(:,1))
%       rgb(centersTRITC{n}(m,2),centersTRITC{n}(m,1),:) = 1;
%   end
%   for p = 1:length(centersCy5{n}(:,1))
%       rgb(centersCy5{n}(p,2),centersCy5{n}(p,1),:) = 1;
%   end
%   skeleton = bwskel(binaryFITC(:,n));
%   for r = 1:imageSpecs(1)
%       for q = 1:imageSpecs(2)
%           if skeleton(r,q)
%               rgb(r,q,:) = 1;
%           end
%       end
%   end
%   figure; imshow(rgb);

%% Volume Estimation
sliceMask = false(frameWidth,frameLength);
area = 0;
grys = bw1+imbinarize(volFITC,max(max(volFITC))/50)+...
    imbinarize(volTRITC,max(max(volTRITC))/50)+...
    imbinarize(volCy5,max(max(volCy5))/50);
for j = 1:10:frameLength-10 %step through lengths
    start = inf;
    finish = -inf;
    %search 10 different lengths for the largest one and use that for all 10
    for k = 0:9
        %alleviates the problem of a limited number of cells used
        %to try and calculate a volume
        tempstart = find(grys(:,j+k)>0,1,'first'); %index in pixels

```

```

tempfinish = find(grys(:,j+k)>0,1,'last'); %index in pixels
if tempstart < start
    start = tempstart; %index in pixels
end
if tempfinish > finish
    finish = tempfinish; %index in pixels
end
end
end
if start ~= inf && finish ~= -inf && start ~= finish
    width = (finish-start)*umPx1; %um
    area = area+width*umPx1*10; %um^2
    sliceMask(start:finish,j:j+9) = true;
end
end
umVolume = umVolume+(area*zStep); %um^3
sliceMask(~imerode(sliceMask,strel('disk',50))) = false;
tumorMask(:,:,n) = sliceMask;

%% Display time remaining
oneround = toc;
oneroundavgmin = (oneroundavgmin*(n-1)+oneround/60)/n;
waittimemin = round(oneroundavgmin*(planes-n),4,'significant');
clear bw1 gry2 gry3 gry4 grys sliceMask;
tic;
waitbar(n/(numImages/numberOfColors),h,['Loading...' newline...
    num2str(waittimemin) ' min remaining']);
end
mmVolume = umVolume/10^9;
binaryFITC = logical(binaryFITC);
% save([savefile ' centersDAPI.mat'], 'centersDAPI');
save([savefile ' binaryFITC.mat'], 'binaryFITC');
save([savefile ' centersTRITC.mat'], 'centersTRITC');
save([savefile ' centersCy5.mat'], 'centersCy5');
save([savefile ' mmVolume.mat'], 'mmVolume');
save([savefile ' tumorMask.mat'], 'tumorMask');
save([savefile ' imageSpecs.mat'], 'imageSpecs');
save([savefile ' voxel_size.mat'], 'voxel_size');
close(h);
else
% load([savefile ' centersDAPI.mat']);
load([savefile ' binaryFITC.mat']);
load([savefile ' centersTRITC.mat']);
load([savefile ' centersCy5.mat']);
load([savefile ' mmVolume.mat']);
load([savefile ' tumorMask.mat']);
load([savefile ' imageSpecs.mat']);
load([savefile ' voxel_size.mat']);
end
end
end

```

```

function [filtImg, compFactor] = calcComp( giver, taker )
%calcComp Caclulates the compensation factor between two images to subtract
% out background autofluorescence
compFactor = 0;
minimum = inf;
for i = 0:0.1:5
    summation = sum(sum(abs(giver-taker*i)));
    if summation < minimum

```

```

        minimum = summation;
        compFactor = i;
        filtImg = giver-taker*i;
    end
end
% figure; imshow(giver);
% figure; imshow(filtImg);
end

```

```

function res = bpass(image_array,lnoise,lobject,threshold)
%
% NAME:
%     bpass
% PURPOSE:
%     Implements a real-space bandpass filter that suppresses
%     pixel noise and long-wavelength image variations while
%     retaining information of a characteristic size.
%
% CATEGORY:
%     Image Processing
% CALLING SEQUENCE:
%     res = bpass( image_array, lnoise, lobject )
% INPUTS:
%     image: The two-dimensional array to be filtered.
%     lnoise: Characteristic lengthscale of noise in pixels.
%             Additive noise averaged over this length should
%             vanish. May assume any positive floating value.
%             May be set to 0 or false, in which case only the
%             highpass "background subtraction" operation is
%             performed.
%     lobject: (optional) Integer length in pixels somewhat
%             larger than a typical object. Can also be set to
%             0 or false, in which case only the lowpass
%             "blurring" operation defined by lnoise is done,
%             without the background subtraction defined by
%             lobject. Defaults to false.
%     threshold: (optional) By default, after the convolution,
%             any negative pixels are reset to 0. Threshold
%             changes the threshold for setting pixels to
%             0. Positive values may be useful for removing
%             stray noise or small particles. Alternatively, can
%             be set to -Inf so that no thresholding is
%             performed at all.
%
% OUTPUTS:
%     res: filtered image.
% PROCEDURE:
%     simple convolution yields spatial bandpass filtering.
% NOTES:
%     Performs a bandpass by convolving with an appropriate kernel. You can
%     think of this as a two part process. First, a lowpassed image is
%     produced by convolving the original with a gaussian. Next, a second
%     lowpassed image is produced by convolving the original with a boxcar
%     function. By subtracting the boxcar version from the gaussian version, we
%     are using the boxcar version to perform a highpass.
%
%     original - lowpassed version of original => highpassed version of the
%     original

```

```

%
% Performing a lowpass and a highpass results in a bandpassed image.
%
% Converts input to double. Be advised that commands like 'image' display
% double precision arrays differently from UINT8 arrays.

% MODIFICATION HISTORY:
%     Written by David G. Grier, The University of Chicago, 2/93.
%
%     Greatly revised version DGG 5/95.
%
%     Added /field keyword JCC 12/95.
%
%     Memory optimizations and fixed normalization, DGG 8/99.
%     Converted to Matlab by D.Blair 4/2004-ish
%
%     Fixed some bugs with conv2 to make sure the edges are
%     removed D.B. 6/05
%
%     Removed inadvertent image shift ERD 6/05
%
%     Added threshold to output. Now sets all pixels with
%     negative values equal to zero. Gets rid of ringing which
%     was destroying sub-pixel accuracy, unless window size in
%     cntrd was picked perfectly. Now cntrd gets sub-pixel
%     accuracy much more robustly ERD 8/24/05
%
%     Refactored for clarity and converted all convolutions to
%     use column vector kernels for speed. Running on my
%     macbook, the old version took ~1.3 seconds to do
%     bpass(image_array,1,19) on a 1024 x 1024 image; this
%     version takes roughly half that. JWM 6/07
%
%     This code 'bpass.pro' is copyright 1997, John C. Crocker and
%     David G. Grier. It should be considered 'freeware'- and may be
%     distributed freely in its original form when properly attributed.

if nargin < 3, lobject = false; end
if nargin < 4, threshold = 0; end

normalize = @(x) x/sum(x);

image_array = double(image_array);

if lnoise == 0
    gaussian_kernel = 1;
else
    gaussian_kernel = normalize(...
        exp(-((-ceil(5*lnoise):ceil(5*lnoise))/(2*lnoise)).^2));
end

if lobject
    boxcar_kernel = normalize(...
        ones(1,length(-round(lobject):round(lobject))));
end

% JWM: Do a 2D convolution with the kernels in two steps each. It is
% possible to do the convolution in only one step per kernel with
%

```

```

% gconv = conv2(gaussian_kernel',gaussian_kernel,image_array,'same');
% bconv = conv2(boxcar_kernel', boxcar_kernel,image_array,'same');
%
% but for some reason, this is slow. The whole operation could be reduced
% to a single step using the associative and distributive properties of
% convolution:
%
% filtered = conv2(image_array,...
% gaussian_kernel'*gaussian_kernel - boxcar_kernel'*boxcar_kernel,...
% 'same');
%
% But this is also comparatively slow (though inexplicably faster than the
% above). It turns out that convolving with a column vector is faster than
% convolving with a row vector, so instead of transposing the kernel, the
% image is transposed twice.

gconv = conv2(image_array',gaussian_kernel','same');
gconv = conv2(gconv',gaussian_kernel','same');

if lobject
    bconv = conv2(image_array',boxcar_kernel','same');
    bconv = conv2(bconv',boxcar_kernel','same');

    filtered = gconv - bconv;
else
    filtered = gconv;
end

% Zero out the values on the edges to signal that they're not useful.
lzero = max(lobject,ceil(5*lnoise));

filtered(1:(round(lzero)),:) = 0;
filtered((end - lzero + 1):end,:) = 0;
filtered(:,1:(round(lzero))) = 0;
filtered(:,(end - lzero + 1):end) = 0;

% JWM: I question the value of zeroing out negative pixels. It's a
% nonlinear operation which could potentially mess up our expectations
% about statistics. Is there data on 'Now centroid gets subpixel accuracy
% much more robustly'? To choose which approach to take, uncomment one of
% the following two lines.
% ERD: The negative values shift the peak if the center of the cntrd mask
% is not centered on the particle.

% res = filtered;
filtered(filtered < threshold) = 0;
res = filtered;
end

function out=pkfnd(im,th,sz,fillFactor)
% finds local maxima in an image to pixel level accuracy.
% this provides a rough guess of particle
% centers to be used by cntrd.m. Inspired by the lmx subroutine of Grier
% and Crocker's feature.pro
% INPUTS:
% im: image to process, particle should be bright spots on dark background
% with little noise
% ofen an bandpass filtered brightfield image (fbps.m, fft.m or bpass.m) or a nice

```

```

% fluorescent image
% th: the minimum brightness of a pixel that might be local maxima.
% (NOTE: Make it big and the code runs faster
% but you might miss some particles. Make it small and you'll get
% everything and it'll be slow.)
% sz: if your data's noisy, (e.g. a single particle has multiple local
% maxima), then set this optional keyword to a value slightly larger than
% the diameter of your blob. if
% multiple peaks are found within a radius of sz/2 then the code will keep
% only the brightest. Also gets rid of all peaks within sz of boundary
% OUTPUT: a N x 2 array containing, [row column] coordinates of local maxima
% CREATED: Eric R. Dufresne, Yale University, Feb 4 2005
% MODIFIED: ERD, 5/2005, got rid of ind2rc.m to reduce overhead on tip by
% Dan Blair; added sz keyword
% ERD, 6/2005: modified to work with one and zero peaks, removed automatic
% normalization of image
% ERD, 6/2005: due to popular demand, altered output to give x and y
% instead of row and column
% ERD, 8/24/2005: pkfnd now exits politely if there's nothing above
% threshold instead of crashing rudely
% ERD, 6/14/2006: now exits politely if no maxima found
% ERD, 10/5/2006: fixed bug that threw away particles with maxima
% consisting of more than two adjacent points

%find all the pixels above threshold
%im=im./max(max(im));
ind=find(im > th);
[nr,nc]=size(im);
tst=zeros(nr,nc);
n=length(ind);
if n==0
    out=[];
    display('nothing above threshold');
    return;
end
mx=[];
%convert index from find to row and column
rc=[mod(ind,nr),floor(ind/nr)+1];
for i=1:n
    r=rc(i,1);c=rc(i,2);
    %check each pixel above threshold to see if it's brighter than it's neighbors
    % THERE'S GOT TO BE A FASTER WAY OF DOING THIS. I'M CHECKING SOME
MULTIPLE TIMES,
    % BUT THIS DOESN'T SEEM THAT SLOW COMPARED TO THE OTHER ROUTINES,
ANYWAY.
    if r>1 & r<nr & c>1 & c<nc
        if im(r,c)>=im(r-1,c-1) & im(r,c)>=im(r,c-1) & im(r,c)>=im(r+1,c-1) & ...
            im(r,c)>=im(r-1,c) & im(r,c)>=im(r+1,c) & ...
            im(r,c)>=im(r-1,c+1) & im(r,c)>=im(r,c+1) & im(r,c)>=im(r+1,c+1)
            mx=[mx,[r,c]'];
            %tst(ind(i))=im(ind(i));
        end
    end
end
%out=tst;
mx=mx';

```

```

[npks,crap]=size(mx);

%if size is specified, then get rid of pks within size of boundary + border
%from bpass filter (15 points from the edge)
if nargin>=3 & npks>0
    %throw out all pks within sz of boundary;
    ind=find(mx(:,1)>(sz+15) & mx(:,1)<(nr-(sz+15)) & mx(:,2)>(sz+15) & ...
        mx(:,2)<(nc-(sz+15)));
    mx=mx(ind,:);
end

%prevent from finding peaks within size of each other
[npks,crap]=size(mx);
if npks > 1
    %CREATE AN IMAGE WITH ONLY PEAKS
    nmx=npks;
    tmp=0.*im; %variable declaration, similar to zeros()
    for i=1:nmx
        %tmp is zero everywhere except where the peaks are
        tmp(mx(i,1),mx(i,2))=im(mx(i,1),mx(i,2));
    end
    %LOOK IN NEIGHBORHOOD AROUND EACH PEAK, PICK THE BRIGHTEST
    for i=1:nmx
        %defines the region of interest around the peak
        roi=tmp( (mx(i,1)-floor(sz/2)):(mx(i,1)+(floor(sz/2)+1)),(mx(i,2)-...
            floor(sz/2)):(mx(i,2)+(floor(sz/2)+1)));
        [mv,indi]=max(roi); %save max value of the columns in the roi and the column position
        [mv,indj]=max(mv); %save the max value of the rows and the row position
        tmp( (mx(i,1)-floor(sz/2)):(mx(i,1)+(floor(sz/2)+1)),...
            (mx(i,2)-floor(sz/2)):(mx(i,2)+(floor(sz/2)+1)))=0;
        tmp(mx(i,1)-floor(sz/2)+indi(indj)-1,mx(i,2)-floor(sz/2)+indj-1)=mv;
    end
    ind=find(tmp>0);
    mx=[mod(ind,nr),floor(ind/nr)+1];
end

[npks,crap]=size(mx);
if npks > 1
    %CREATE AN IMAGE WITH ONLY PEAKS
    nmx=npks;
    tmp=0.*im; %variable declaration, similar to zeros()
    for i=1:nmx
        %tmp is zero everywhere except where the peaks are
        tmp(mx(i,1),mx(i,2))=im(mx(i,1),mx(i,2));
    end
    %LOOK IN NEIGHBORHOOD AROUND EACH PEAK, PICK THE BRIGHTEST
    for i=1:nmx
        %added by Andrew Fisher 20190219
        %remove peaks in areas where it does not fill at least *fillFactor* of the circle of
        %interest
        imROI = im( (mx(i,1)-floor(sz/2)):(mx(i,1)+(floor(sz/2)+1)),...
            (mx(i,2)-floor(sz/2)):(mx(i,2)+(floor(sz/2)+1)));
        emptyness = find(imROI(:) < th);
        if length(emptyness) >= floor(fillFactor*length(imROI(:)))
            tmp( (mx(i,1)-floor(sz/2)):(mx(i,1)+(floor(sz/2)+1)),...
                (mx(i,2)-floor(sz/2)):(mx(i,2)+(floor(sz/2)+1)))=0;
        end
        %make sure there is a local contrast larger than one pixel, look at
        %the maximum third of pixels compared to the minimum third.

```

```

mid = logical([0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0;...
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0;...
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0;...
0,0,0,1,1,1,1,1,1,1,1,0,0,0;...
0,0,0,1,1,1,1,1,1,1,1,0,0,0;...
0,0,0,1,1,1,1,1,1,1,1,0,0,0;...
0,0,0,1,1,1,1,1,1,1,1,0,0,0;...
0,0,0,1,1,1,1,1,1,1,1,0,0,0;...
0,0,0,1,1,1,1,1,1,1,1,0,0,0;...
0,0,0,1,1,1,1,1,1,1,1,0,0,0;...
0,0,0,1,1,1,1,1,1,1,1,0,0,0;...
0,0,0,0,0,0,0,0,0,0,0,0,0,0;...
0,0,0,0,0,0,0,0,0,0,0,0,0,0;...
0,0,0,0,0,0,0,0,0,0,0,0,0,0]);
edge = logical([1,1,1,1,1,1,1,1,1,1,1,1,1,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,0,0,0,0,0,0,0,0,0,0,0,0,1;...
1,1,1,1,1,1,1,1,1,1,1,1,1,1]);
if mean(imROI(mid)) < 3*mean(imROI(edge))
    tmp( (mx(i,1)-floor(sz/2)):(mx(i,1)+(floor(sz/2)+1)),...
    (mx(i,2)-floor(sz/2)):(mx(i,2)+(floor(sz/2)+1)))=0;
end
end
ind=find(tmp>0);
mx=[mod(ind,nr),floor(ind/nr)+1];
end

if size(mx)=[0,0]
    out=[];
else
    out = mx;
end
end

```

```

function centers3D = findCenters3D(imageSpecs,planecenters,cellDiameterPxl,numPlanes)
%findCenters3D Finds cells that are at least numPlanes in depth
% and no longer than numPlanes+1 in depth

```

```

frameWidth = imageSpecs(1);
frameLength = imageSpecs(2);
planes = imageSpecs(3);
centers3D = false(frameWidth, frameLength, planes);
for i = planes:-1:2 %loop through the planes
    j = 1;
    if ~isempty(planecenters{:,i})
        %loop through the center points in the i-th plane
        while j <= length(planecenters{:,i}(:,1))
            loopNum = 1;

```



```

        below = 0;
        [loopNum, below, planeCenters] = ...
            findMatchingCenter(planeCenters,i,j,loopNum,below,...
                cellDiameterPx1,numPlanes);
        if loopNum < numPlanes+1 && below
            %create a binary matrix of center points in 3D space
            centers3D(planeCenters{1,i}(j,1),planeCenters{1,i}(j,2),i) = 1;
        end
        j = j+1;
    end
end
end
end
end
end

```

```

function [loopNum, anotherOne, planeCenters] =...
    findMatchingCenter(planeCenters,i,j,loopNum,below,cellDiameterPx1,numPlanes)
%findMatchingCenter Summary of this function goes here
% Detailed explanation goes here
n=0;
anotherOne = 0;
if i-loopNum > 0 && ~isempty(planeCenters{1,i}) && ~isempty(planeCenters{1,i-loopNum})
    %loop through the centerpoints in the k-th plane
    for k = 1:length(planeCenters{:,i-loopNum}(:,1))
        if pdist([planeCenters{1,i}(j,:); planeCenters{1,i-loopNum}(k,:)])...
            < cellDiameterPx1/4
            anotherOne=1;
            n=k;
        end
    end
end
if anotherOne == 1
    below = below + anotherOne;
    currentLoop = loopNum;
    loopNum=loopNum+1;
    [loopNum,~,planeCenters] =...
        findMatchingCenter(planeCenters,i,j,loopNum,below,cellDiameterPx1,numPlanes);
end
%once we find more than numPlanes points that connect to a single xy point,
%we delete all future points to prevent multiple counts of the same cell
if loopNum > numPlanes && anotherOne
    planeCenters{1,i-currentLoop}(n,:) = [];
end
end
end
end
end

```

```

function [marker_size, numCells] =...
    plot3DCenters(centers3D, imageSpecs, cellDiameterPx1,...
        marker_size, toPlot, color)
%plot3DCenters creates a 3D representation of the cells

frameWidth = imageSpecs(1);
frameLength = imageSpecs(2);
centerIndex = find(centers3D==1);
numCells = length(centerIndex);
centerX = zeros(numCells,1);
centerY = zeros(numCells,1);

```

```

centerZ = zeros(numCells,1);

for m = 1:numCells
    centerZ(m) = ceil(centerIndex(m)/(frameWidth*frameLength));
    centerX(m) = ceil((centerIndex(m)-(centerZ(m)-1)*frameWidth*frameLength)/frameLength);
    centerY(m) = centerIndex(m)-(centerZ(m)-1)*frameWidth*frameLength-...
        (centerX(m)-1)*frameLength;
    centerZ(m) = centerZ(m)*10; %goes from plane number, to um. 10 um per plane step
end
if marker_size == 0 && toPlot
    figure;
    xlim([0 2000]);
    ylim([0 2000]);
    zlim([0 2000]);

    ax = gca;
    AR = get(gca, 'dataaspectratio');
    if ~isequal(AR(1:3), [1 1 1])
        error(['Units are not equal on X, Y, and Z, cannot create marker'...
            'size that is one unit on both']);
    end
    oldunits = get(ax, 'Units');
    set(ax, 'Units', 'points');
    pos = get(ax, 'Position'); % [X Y Z width height depth]
    set(ax, 'Units', oldunits);
    XL = xlim(ax);
    points_per_unit = pos(3) / (XL(2) - XL(1));
    marker_size = (points_per_unit*cellDiameterPxl).^2 * pi / 4;
end
if toPlot
    scatter3(centerX, centerY, centerZ, marker_size, 'filled',...
        'MarkerEdgeColor','k', 'MarkerFaceColor', color);
    xlabel('X (\mum)');
    ylabel('Y (\mum)');
    zlabel('Z (\mum)');
    xlim([0 2000]);
    ylim([0 2000]);
    zlim([0 2000]);
    hold on;
end
end

```

```

function plotVessels(skelFITC,imageSpecs,toPlot)
%plotVessels creates a 3D representation of the vessels
%
    frameWidth = imageSpecs(1);
    frameLength = imageSpecs(2);
    vesselIndex = find(skelFITC==1);
    numCells = length(vesselIndex);
    for m = 1:numCells
        vesselZ(m) = ceil(vesselIndex(m)/(frameWidth*frameLength));
        vesselX(m) = ceil((vesselIndex(m)-(vesselZ(m)-1)*frameWidth*frameLength)/frameLength);
        vesselY(m) = vesselIndex(m)-(vesselZ(m)-1)*frameWidth*frameLength-(vesselX(m)-
1)*frameLength;
        vesselZ(m) = vesselZ(m)*10; %goes from plane number, to um, to pixels
    end
    if toPlot

```

```

        plot3(vesselY,vesselX,vesselZ,','Color','g'); %row,col,z
    end
end

```

```

function conc = boxConcentrations(centers,boxDim)
%boxConcentrations Counts the number of cells in each volume
% Volume defined by boxDim
imageDim = size(centers);
conc = zeros(imageDim(1)-(boxDim(1)-1),imageDim(2)-(boxDim(2)-1),...
    imageDim(3)-(boxDim(3)-1));
if max(max(max(centers))) ~= 0
    for i = 1:size(conc,3) %:floor(boxDim(3)/10):
        for j = 1:floor(boxDim(2)/10):size(conc,2)
            for k = 1:floor(boxDim(1)/10):size(conc,1)
                conc(k,j,i) = sum(sum(centers(k:k+(boxDim(1)-1),...
                    j:j+(boxDim(2)-1),i:i+(boxDim(3)-1))));
            end
        end
    end
end
end
end
end

```

```

function printExcel(sample, filename, boxDim,totalTRITCCells,totalCy5Cells,...
    totalFITCCells,mmVolume,boxNumTRITC,boxNumCy5,boxNumFITC,centers3DTRITC,...
    centers3DCy5,skelFITC, toPlot)
%printExcel Summary of this function goes here
% FITC is divided by 10X more because it is a single line so each cell
% with approx 10um diameter would have 10 pixels across it.

```

```

[index210, filteredHotSpots210] =
findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,2,10,50);

```

```

[index220, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,2,20,50);

```

```

[index230, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,2,30,50);

```

```

[index22, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,2,2,50);

```

```

[index25, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,2,5,50);

```

```

[index510, filteredHotSpots510] =
findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,5,10,50);

```

```

[index520, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,5,20,50);

```

```

[index530, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,5,30,50);

```

```

[index52, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,5,2,50);

```

```

[index55, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,5,5,50);

```

```

[index1010, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,10,10,50);

```

```

[index1020, filteredHotSpots1020] =
findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,10,20,50);

```

```

[index1030, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,10,30,50);

```

```

[index102, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,10,2,50);

[index105, ~] = findHotSpots(boxNumTRITC,boxNumCy5,boxNumFITC,10,5,50);

totalNumCells = sum([totalTRITCCells,totalCy5Cells,totalFITCCells]);

hotSpotTRITCCells210 = zeros(1,height(filteredHotSpots210));
hotSpotCy5Cells210 = zeros(1,height(filteredHotSpots210));
hotSpotFITCCells210 = zeros(1,height(filteredHotSpots210));
for n = 1:height(filteredHotSpots210)
    if toPlot
        h = figure;
    end
    hotSpotTRITCCells210(n) = plotHotSpot(filteredHotSpots210(n,:), centers3DTRITC,...
        'r', 0, toPlot);
    hotSpotCy5Cells210(n) = plotHotSpot(filteredHotSpots210(n,:), centers3DCy5, 'm',...
        0, toPlot);
    hotSpotFITCCells210(n) = plotHotSpot(filteredHotSpots210(n,:), skelFITC, 'g', 10, toPlot);
    if toPlot
        pause;
        close(h);
    end
end

hotSpotTRITCCells510 = zeros(1,height(filteredHotSpots510));
hotSpotCy5Cells510 = zeros(1,height(filteredHotSpots510));
hotSpotFITCCells510 = zeros(1,height(filteredHotSpots510));
for n = 1:height(filteredHotSpots510)
    if toPlot
        h = figure;
    end
    hotSpotTRITCCells510(n) = plotHotSpot(filteredHotSpots510(n,:), centers3DTRITC,...
        'r', 0, toPlot);
    hotSpotCy5Cells510(n) = plotHotSpot(filteredHotSpots510(n,:), centers3DCy5, 'm',...
        0, toPlot);
    hotSpotFITCCells510(n) = plotHotSpot(filteredHotSpots510(n,:), skelFITC, 'g', 10, toPlot);
    if toPlot
        pause;
        close(h);
    end
end

hotSpotTRITCCells1020 = zeros(1,height(filteredHotSpots1020));
hotSpotCy5Cells1020 = zeros(1,height(filteredHotSpots1020));
hotSpotFITCCells1020 = zeros(1,height(filteredHotSpots1020));
for n = 1:height(filteredHotSpots1020)
    if toPlot
        h = figure;
    end
    hotSpotTRITCCells1020(n) = plotHotSpot(filteredHotSpots1020(n,:), centers3DTRITC,...
        'r', 0, toPlot);
    hotSpotCy5Cells1020(n) = plotHotSpot(filteredHotSpots1020(n,:), centers3DCy5, 'm',...
        0, toPlot);
    hotSpotFITCCells1020(n) = plotHotSpot(filteredHotSpots1020(n,:), skelFITC, 'g', 10, toPlot);
    if toPlot
        pause;
        close(h);
    end
end

```



```

while j <= height(filteredHotSpots)
    % 400000 because 200x200x100 um / 10um/step in the z direction
    if filteredHotSpots{j,1} < 200000
        filteredHotSpots(j,:) = [];
        j = j-1;
    end
    j = j+1;
end

%% adjust edges of the bounding box so that the hot spot can be plotted
index = length(filteredHotSpots.Volume);
for i=1:index
    for j = 1:3
        if j < 3
            filteredHotSpots.BoundingBox(i,j) = filteredHotSpots.BoundingBox(i,j)+...
                dilStep/2;% shift the dilation from center to corner
            if 100-dilStep>0
                filteredHotSpots.BoundingBox(i,j+3) = filteredHotSpots.BoundingBox(i,j+3)...
                    +(100-dilStep);% correct for smaller dilation sizes than counting box sizes
            end
            if filteredHotSpots.BoundingBox(i,j)+filteredHotSpots.BoundingBox(i,j+3)>...
                size(boxNumTRITC,j)+95 %make sure it doesn't go out of bounds
                filteredHotSpots.BoundingBox(i,j+3) = floor(size(boxNumTRITC,j)+95-...
                    filteredHotSpots.BoundingBox(i,j));
            end
        else
            % shift the dilation from center to corner
            filteredHotSpots.BoundingBox(i,j) = filteredHotSpots.BoundingBox(i,j)+dilStep/20;
            if 100-dilStep>0
                filteredHotSpots.BoundingBox(i,j+3) = filteredHotSpots.BoundingBox(i,j+3)+...
                    ((100-dilStep)/10);% correct for smaller dilation sizes than counting box sizes
            end
            if filteredHotSpots.BoundingBox(i,j)+filteredHotSpots.BoundingBox(i,j+3)>...
                size(boxNumTRITC,j)+9 %make sure it doesn't go out of bounds
                filteredHotSpots.BoundingBox(i,j+3) = floor(size(boxNumTRITC,j)+9-...
                    filteredHotSpots.BoundingBox(i,j));
            end
        end
    end
end
end
end
end

```

```

function tlsNumCells = plotHotSpot(filteredHotSpot, centers3D, color, markerSize, toPlot)
%plotHotSpot creates a 3D representation of the TLS

```

```

    box = [filteredHotSpot.BoundingBox];
    tlsNumCells = sum(sum(sum(centers3D(floor(box(2)):ceil(box(2))+ceil(box(5)),...
        floor(box(1)):ceil(box(1))+ceil(box(4)), floor(box(3)):ceil(box(3))+ceil(box(6))))));
    if toPlot

        frameWidth = length(centers3D(1, :, 1));
        frameLength = length(centers3D(:, 1, 1));

        centerIndex = find(centers3D==1);
        totalNumCells = length(centerIndex);
        centerX = zeros(totalNumCells, 1);
        centerY = zeros(totalNumCells, 1);
        centerZ = zeros(totalNumCells, 1);
    end

```

```

centerZpxl = zeros(totalNumCells,1);

for m = 1:totalNumCells
    centerZpxl(m) = ceil(centerIndex(m)/(frameWidth*frameLength));
    centerX(m) = ceil((centerIndex(m)-(centerZpxl(m)-1)*frameWidth*frameLength)...
        /frameLength);
    centerY(m) = centerIndex(m)-(centerZpxl(m)-1)*frameWidth*frameLength-...
        (centerX(m)-1)*frameLength;
    centerZ(m) = centerZpxl(m)*10; %goes from plane number, to um, to pixels
end

cubeMaxSide = max([box(4) box(5) box(6)*10]);
xlim([floor(box(1)) floor(box(1))+cubeMaxSide+1]);
ylim([floor(box(2)) floor(box(2))+cubeMaxSide+1]);
zlim([floor(box(3))*10 floor(box(3))*10+cubeMaxSide+1]);
if markerSize == 0
    ax = gca;
    AR = get(gca, 'dataaspectratio');
    if ~isequal(AR(1:3), [1 1 1])
        error(['Units are not equal on X, Y, and Z, cannot create marker size'...
            'that is one unit on both']);
    end
    oldunits = get(ax, 'Units');
    set(ax, 'Units', 'points');
    pos = get(ax, 'Position'); % [X Y Z width height depth]
    set(ax, 'Units', oldunits);
    XL = xlim(ax);
    points_per_unit = pos(3) / (XL(2) - XL(1));
    markerSize = (points_per_unit*10).^2 * pi / 4;
end
scatter3(centerX, centerY, centerZ, markerSize, 'filled',...
    'MarkerEdgeColor','k', 'MarkerFaceColor', color);

xlim([floor(box(1)) floor(box(1))+cubeMaxSide+1]);
ylim([floor(box(2)) floor(box(2))+cubeMaxSide+1]);
zlim([floor(box(3))*10 floor(box(3))*10+cubeMaxSide+1]);
xlabel('X (\mum)');
ylabel('Y (\mum)');
zlabel('Z (\mum)');

volume = filteredHotSpot.Volume;
title(['TLS Volume: ' num2str(volume) '\mum^3'])
hold on;
end
end

```


APPENDIX D

Appendix D shows the data analysis software for the Raman/fluorescence extraction studies written in MATLAB (2015b). Full code downloadable from

<https://github.com/afishtex/APPLICATION-OF-OPTICAL-ANALYSES-TO-CANCER-THERAPEUTICS-AND-DIAGNOSTICS>.

```
function varargout = Raman_process_v5(varargin)
% RAMAN_PROCESS_V5 MATLAB code for Raman_process_v5.fig
%   RAMAN_PROCESS_V5, by itself, creates a new RAMAN_PROCESS_V5 or raises
%   the existing singleton*.
%
%   H = RAMAN_PROCESS_V5 returns the handle to a new RAMAN_PROCESS_V5 or the
%   handle to the existing singleton*.
%
%   RAMAN_PROCESS_V5('CALLBACK',hObject,eventData,handles,...) calls the local
%   function named CALLBACK in RAMAN_PROCESS_V5.M with the given input arguments.
%
%   RAMAN_PROCESS_V5('Property','Value',...) creates a new RAMAN_PROCESS_V5 or
%   raises the existing singleton*. Starting from the left, property value pairs
%   are applied to the GUI before Raman_process_v5_OpeningFcn gets called. An
%   unrecognized property name or invalid value makes property application
%   stop. All inputs are passed to Raman_process_v5_OpeningFcn via varargin.
%
%   *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
%   instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help Raman_process_v5

% Last Modified by GUIDE v2.5 11-Jan-2018 10:51:53

% Author(s): A. Fisher, K. Meissner
% Centre of NanoHealth, College of Science,
% Swansea University, Singleton Park, Swansea, SA2 8PP, United Kingdom

% Begin initialization code - DO NOT EDIT

gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @Raman_process_v5_OpeningFcn, ...
                  'gui_OutputFcn', @Raman_process_v5_OutputFcn, ...
                  'gui_LayoutFcn', [] , ...
                  'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
```

```

    end
% End initialization code - DO NOT EDIT
end

% --- Executes just before Raman_process_v5 is made visible.
function Raman_process_v5_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Raman_process_v5 (see VARARGIN)

% Choose default command line output for Raman_process_v5
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
xlabel('Raman Shift (cm^{-1})');
ylabel('Signal');
title('Input Data');
handles.lt_bl = [.871,.922,.98];
handles.dk_bl = [.729,.831,.957];
handles.lt_rd = [.961,.922,.922];
handles.dk_rd = [.925,.839,.839];
guidata(hObject,handles);

% UIWAIT makes Raman_process_v5 wait for user response (see UIRESUME)
% uiwait(handles.figure1);
end

% --- Outputs from this function are returned to the command line.
function varargout = Raman_process_v5_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;
end

% --- Executes on button press in export.
function export_Callback(hObject, eventdata, handles)
% hObject    handle to export (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
if get(handles.exp_fitparams,'Value') == 1
    assignin('base','fit_param',handles.fit_array);
end
if get(handles.exp_fitgood,'Value') == 1
    assignin('base','fit_goodness',handles.goodness_array);
end
if get(handles.exp_fluor_fitparams,'Value') == 1
    assignin('base','fit_param',handles.fluor_fit_array);
end
if get(handles.exp_fluor_fitgood,'Value') == 1

```

```

        assignin('base','fit_goodness',handles.fluor_goodness_array);
    end
    if get(handles.exp_mag,'Value') == 1
        assignin('base','image_mag',handles.image_mag);
    end
    if get(handles.exp_ctr,'Value') == 1
        assignin('base','image_ctr',handles.image_ctr);
    end
    if get(handles.exp_width,'Value') == 1
        assignin('base','image_width',handles.image_width);
    end
    if get(handles.exp_r2,'Value') == 1
        assignin('base','image_r2',handles.image_r2);
    end
    if get(handles.exp_fluor_mag,'Value') == 1
        assignin('base','image_mag',handles.fluor_mag);
    end
    if get(handles.exp_fluor_asym,'Value') == 1
        assignin('base','image_ctr',handles.fluor_asym);
    end
    if get(handles.exp_fluor_width,'Value') == 1
        assignin('base','image_width',handles.fluor_width);
    end
    if get(handles.exp_fluor_r2,'Value') == 1
        assignin('base','image_r2',handles.fluor_r2);
    end
    if get(handles.exp_raw,'Value') == 1
        assignin('base','data',handles.data);
        assignin('base','shift_axis',handles.shift_axis);
    end
    if get(handles.exp_raw_dec,'Value') == 1
        assignin('base','data_dec',handles.data_dec);
    end
    if get(handles.exp_f_dec,'Value') == 1
        assignin('base','data_f_dec',handles.data_f_dec);
    end
    if get(handles.exp_f_b_dec,'Value') == 1
        assignin('base','data_f_b_dec',handles.data_f_b_dec);
    end
    if (get(handles.exp_raw_dec,'Value') == 1) || (get(handles.exp_f_b_dec,'Value')...
        == 1) || (get(handles.exp_f_b_dec,'Value') == 1)
        assignin('base','shift_axis_dec',handles.shift_axis_dec);
    end
end

```

% --- Executes on button press in clear.

function clear_Callback(hObject, eventdata, handles)

% hObject handle to clear (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

cla(handles.input_plot);

handles.data=[];handles.shift_axis=[];

handles.data_dec=[];handles.data_f_dec=[];handles.data_f_b_dec=[];...

handles.shift_axis_dec=[];

handles.image_mag=[];handles.image_ctr=[];handles.image_width=[];...

handles.image_r2=[];

handles.rows=0;handles.cols=0;handles.spec=0;

axis auto;

```

xlabel('Raman Shift (cm^{-1})');
ylabel('Signal');
title('Input Data');
s = sprintf('Rows: %.0f',handles.rows);
set(handles.row_label,'String',s);
s = sprintf('Cols: %.0f',handles.cols);
set(handles.col_label,'String',s);
set(handles.file_info,'String','No File');
set(handles.decimate,'BackgroundColor',handles.dk_rd);
set(handles.average,'BackgroundColor',handles.dk_rd);
set(handles.process,'BackgroundColor',handles.dk_rd);
set(handles.file_load,'BackgroundColor',handles.dk_bl);
set(handles.file_info,'BackgroundColor',[.94,.94,.94]);
set(handles.sav_raw,'Value',0);
set(handles.exp_raw,'Value',0);
set(handles.sav_raw_dec,'Value',0);
set(handles.exp_raw_dec,'Value',0);
set(handles.sav_f_dec,'Value',0);
set(handles.exp_f_dec,'Value',0);
set(handles.sav_f_b_dec,'Value',0);
set(handles.exp_f_b_dec,'Value',0);
set(handles.sav_mag,'Value',0);
set(handles.sav_ctr,'Value',0);
set(handles.sav_width,'Value',0);
set(handles.sav_r2,'Value',0);
set(handles.exp_mag,'Value',0);
set(handles.exp_ctr,'Value',0);
set(handles.exp_width,'Value',0);
set(handles.exp_r2,'Value',0);
set(handles.exp_fitgood,'Value',0);
set(handles.exp_fitparams,'Value',0);
set(handles.exp_fluor_fitgood,'Value',0);
set(handles.exp_fluor_fitparams,'Value',0);
set(handles.exp_fluor_mag,'Value',0);
set(handles.exp_fluor_asym,'Value',0);
set(handles.exp_fluor_width,'Value',0);
set(handles.exp_fluor_r2,'Value',0);
set(handles.sav_fluor_mag,'Value',0);
set(handles.sav_fluor_asym,'Value',0);
set(handles.sav_fluor_width,'Value',0);
set(handles.sav_fluor_r2,'Value',0);
pause(.25);
set(handles.file_info,'String','No File');
set(handles.file_info,'BackgroundColor',[.8,.8,.8]);
guidata(hObject,handles);
end

% --- Executes on button press in exp_mag.
function exp_mag_Callback(hObject, eventdata, handles)
% hObject    handle to exp_mag (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in exp_ctr.
function exp_ctr_Callback(hObject, eventdata, handles)
% hObject    handle to exp_ctr (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB

```

```

% handles structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in exp_width.
function exp_width_Callback(hObject, eventdata, handles)
% hObject handle to exp_width (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in exp_r2.
function exp_r2_Callback(hObject, eventdata, handles)
% hObject handle to exp_r2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in exp_raw.
function exp_raw_Callback(hObject, eventdata, handles)
% hObject handle to exp_raw (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in exp_f_dec.
function exp_filt_Callback(hObject, eventdata, handles)
% hObject handle to exp_f_dec (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in save.
function save_Callback(hObject, eventdata, handles)
% hObject handle to save (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

%%%%%%Store data as files%%%%%%%%
%Get filename and path
[FileName,PathName] = uinputfile('*.txt');
%Write image cubes to 2D arrays
[rows cols spec] = size(handles.data);
tmpraw = zeros(rows*cols+1,spec+2);
tmpraw(1,1)=0;
tmpraw(1,2)=0;
tmpraw(1,3:end)=handles.shift_axis(1,:);
cntc = 0;
cntr = 1;
for r = 1:rows
    for c = 1:cols
        cntc = cntc+1;
        cntr = cntr+1;
        tmpraw(cntr,1) = r;
        tmpraw(cntr,2) = cntc;
    end
end

```

```

        for sp = 1:spec
            tmpraw(cntr,sp+2) = handles.data(r,c,sp);
        end
    end
    cntc = 0;
end

%run the filtered data separately since spatial filtering may have altered
%dimensions of array
[rows cols spec] = size(handles.data_f_dec);
tmpfilt_dec = zeros(rows*cols+1,spec+2);
tmpfiltb_dec = zeros(rows*cols+1,spec+2);
tmpfilt_dec(1,1)=0;
tmpfilt_dec(1,2)=0;
tmpfilt_dec(1,3:end)=handles.shift_axis_dec(1,:);
tmpfiltb_dec(1,1)=0;
tmpfiltb_dec(1,2)=0;
tmpfiltb_dec(1,3:end)=handles.shift_axis_dec(1,:);
cntc = 0;
cntr = 1;
for r = 1:rows
    for c = 1:cols
        cntc = cntc+1;
        cntr = cntr+1;
        tmpfilt_dec(cntr,1) = r;
        tmpfilt_dec(cntr,2) = cntc;
        tmpfiltb_dec(cntr,1) = r;
        tmpfiltb_dec(cntr,2) = cntc;
        for sp = 1:spec
            tmpfilt_dec(cntr,sp+2) = handles.data_f_dec(r,c,sp);
            tmpfiltb_dec(cntr,sp+2) = handles.data_f_b_dec(r,c,sp);
        end
    end
    cntc = 0;
end

[rows cols spec] = size(handles.data_dec);
tmpraw_dec = zeros(rows*cols+1,spec+2);
cntc = 0;
cntr = 1;
for r = 1:rows
    for c = 1:cols
        cntc = cntc+1;
        cntr = cntr+1;
        tmpraw_dec(cntr,1) = r;
        tmpraw_dec(cntr,2) = cntc;
        for sp = 1:spec
            tmpraw_dec(cntr,sp+2) = handles.data_dec(r,c,sp);
        end
    end
    cntc = 0;
end

%Write files to path/filename with extension
if get(handles.sav_raw,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...
        '_Data_Raw','.txt');
    save(s,'tmpraw','-ascii');
end

```

```

if get(handles.sav_raw_dec,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...
        '_Data_Raw_Dec','.txt');
    save(s,'tmpraw_dec','-ascii');
end
if get(handles.sav_f_dec,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...
        '_Data_Filt_Dec','.txt');
    save(s,'tmpfilt_dec','-ascii');
end
if get(handles.sav_f_b_dec,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...
        '_Data_Filt_Back_Dec','.txt');
    save(s,'tmpfiltb_dec','-ascii');
end
if get(handles.sav_mag,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...
        '_Raman_Image_Magnitude','.txt');
    tmp = handles.image_mag;
    save(s,'tmp','-ascii');
end
if get(handles.sav_ctr,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...
        '_Raman_Image_Center','.txt');
    tmp = handles.image_ctr;
    save(s,'tmp','-ascii');
end
if get(handles.sav_width,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...
        '_Raman_Image_Width','.txt');
    tmp = handles.image_width;
    save(s,'tmp','-ascii');
end
if get(handles.sav_r2,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...
        '_Raman_Image_RSquared','.txt');
    tmp = handles.image_r2;
    save(s,'tmp','-ascii');
end
if get(handles.exp_fluor_mag,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...
        '_Fluorescence_Image_Magnitude','.txt');
    tmp = handles.fluor_mag;
    save(s,'tmp','-ascii');
end
if get(handles.exp_fluor_asym,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...
        '_Fluorescence_Image_Asymmetry','.txt');
    tmp = handles.fluor_mag;
    save(s,'tmp','-ascii');
end
if get(handles.exp_fluor_width,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...
        '_Fluorescence_Image_Width','.txt');
    tmp = handles.fluor_width;
    save(s,'tmp','-ascii');
end
if get(handles.exp_fluor_r2,'Value') == 1
    s = sprintf('%s%s%s%s',PathName,FileName(1:end-4),...

```

```

        '_Fluorescence_Image_RSquared','.txt');
    tmp = handles.fluor_r2;
    save(s,tmp,'-ascii');
end
end

% --- Executes on button press in sav_mag.
function sav_mag_Callback(hObject, eventdata, handles)
% hObject    handle to sav_mag (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in sav_ctr.
function sav_ctr_Callback(hObject, eventdata, handles)
% hObject    handle to sav_ctr (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in sav_width.
function sav_width_Callback(hObject, eventdata, handles)
% hObject    handle to sav_width (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in sav_r2.
function sav_r2_Callback(hObject, eventdata, handles)
% hObject    handle to sav_r2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in sav_raw.
function sav_raw_Callback(hObject, eventdata, handles)
% hObject    handle to sav_raw (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in sav_f_dec.
function sav_filt_Callback(hObject, eventdata, handles)
% hObject    handle to sav_f_dec (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in decimate.
function decimate_Callback(hObject, eventdata, handles)
% hObject    handle to decimate (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB

```



```

% handles structure with handles and user data (see GUIDATA)
start_shift = str2num(get(handles.lam_start,'String'));
stop_shift = str2num(get(handles.lam_stop,'String'));
bkg_size = str2num(get(handles.bkg_pts,'String'));

%determine indices of start and stop
tmp = abs(handles.shift_axis-start_shift);
[min_val min_index] = min(tmp); % value and index of closest value
start = min_index; % start index
clear tmp;
tmp = abs(handles.shift_axis-stop_shift);
[min_val min_index] = min(tmp); % value and index of closest value
stop = min_index; % stop index
%add the background on either side of the fit to remove
handles.bkg1 = start - bkg_size;
handles.bkg2 = stop + bkg_size;

handles.data_dec = handles.data(:, :, handles.bkg1:handles.bkg2);
handles.shift_axis_dec = handles.shift_axis(1, handles.bkg1:handles.bkg2);
set(handles.decimate, 'BackgroundColor', handles.lt_rd);
s = sprintf('Spec: %.0f', handles.bkg2 - handles.bkg1 + 1);
set(handles.spec_label, 'String', s);
set(handles.sav_raw_dec, 'Value', 1);
set(handles.exp_raw_dec, 'Value', 1);
guidata(hObject, handles);
end

% --- Executes on button press in average.
function average_Callback(hObject, eventdata, handles)
% hObject handle to average (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% Remove noise
if get(handles.ave_none, 'Value')==1
    handles.label_spectral = 'None';
    handles.data_f = handles.data;
    handles.data_f_dec = handles.data_dec;
elseif get(handles.ave_box, 'Value')==1
    handles.label_spectral = 'Boxcar';
    box_size = str2num(get(handles.box_win, 'String'));
    handles.data_f = filter(ones(1, box_size)/box_size, 1, handles.data, [], 3);
    handles.data_f_dec = handles.data_f(:, :, handles.bkg1:handles.bkg2);
elseif get(handles.ave_sg, 'Value')==1
    handles.label_spectral = sprintf('Sovitzky-Golay Ord: %s Win: %s', ...
        get(handles.sg_order, 'String'), get(handles.sg_win, 'String'));
    order = str2num(get(handles.sg_order, 'String'));
    box_size = str2num(get(handles.sg_win, 'String'));
    handles.data_f = sgolayfilt_3D(handles.data, order, box_size, [], 3);
    handles.data_f_dec = handles.data_f(:, :, handles.bkg1:handles.bkg2);
end
if get(handles.ave_spatial, 'Value')==1
    handles.data_f_dec = spatial_filter(handles.data_f_dec, handles.spatial_array);
    handles.data_f = spatial_filter(handles.data_f, handles.spatial_array);
else
    handles.label_spatial = 'No Spatial';
end
set(handles.average, 'BackgroundColor', handles.lt_rd);
set(handles.sav_f_dec, 'Value', 1);
set(handles.exp_f_dec, 'Value', 1);

```

```

    guidata(hObject,handles);
end

% --- Executes on button press in process.
function process_Callback(hObject, eventdata, handles)
% hObject    handle to process (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

%Set limits for determining poor fit
mag_lim = 0;
width_lim = 1;
r2_lim = 0.2;

%Remove background and fit the data
if get(handles.bkg_detrend,'Value')==1
    handles.label_bkg = 'Detrend';
    [handles.image_mag,handles.image_ctr,handles.image_width,handles.image_r2,...
    handles.data_f_b_dec,handles.fit_array,handles.goodness_array] =...
    process_detrend(handles.data_f_dec,handles.shift_axis_dec',...
    str2num(get(handles.bkg_pts,'String')),mag_lim,width_lim,r2_lim,...
    get(handles.gauss_fit,'Value'));

else get(handles.bkg_poly,'Value')==1
    handles.label_bkg = 'Poly';
    if get(handles.poly_fit,'Value')==1
        poly = 'poly1';
    elseif get(handles.poly_fit,'Value')==2
        poly = 'poly2';
    end
    [handles.image_mag,handles.image_ctr,handles.image_width,handles.image_r2,...
    handles.data_f_b_dec,handles.fit_array,handles.goodness_array] =...
    process_poly(handles.data_f_dec,handles.shift_axis_dec',...
    str2num(get(handles.bkg_pts,'String')),mag_lim,width_lim,r2_lim,poly,...
    get(handles.gauss_fit,'Value'));
end
if get(handles.gauss_fit,'Value')==1
    handles.label_fit = 'Gaussian';
else
    handles.label_fit = 'Lorentzian';
end
figure('Name',get(handles.file_display,'String'));
subplot(2,2,1);
X = -5:5/8:5;
Y = 5:-5/8:-5;
imagesc(X, Y, handles.image_mag/max(max(handles.image_mag)));
colormap('jet');
caxis([0,1]);
colorbar;
title('Normalized Magnitude');
xlabel('X (\mum)');
ylabel('Y (\mum)');
text(min(X)-3.5, min(Y),'(b)','FontWeight','bold','FontSize',16,'Color','k',...
    'BackgroundColor','w');
fprintf('Avg Mag: %.3f\n', mean(handles.image_mag(:)));
subplot(2,2,2);
imagesc(X, Y, handles.image_ctr);
caxis([str2num(get(handles.lam_start,'String')),...

```

```

    str2num(get(handles.lam_stop,'String'))]);
colorbar;
title('Center');
xlabel('X (\mum)');
ylabel('Y (\mum)');
text(min(X)-3.5, min(Y),'(d)', 'FontWeight','bold','FontSize',16,'Color','k',...
     'BackgroundColor','w');
fprintf('Avg Ctr: %.3f\n', mean(handles.image_ctr(:)));
subplot(2,2,3);
imagesc(X, Y, handles.image_width);
caxis([0,(str2num(get(handles.lam_stop,'String'))-...
     str2num(get(handles.lam_start,'String'))/2]);
colorbar;
title('Width');
xlabel('X (\mum)');
ylabel('Y (\mum)');
text(min(X)-3.5, min(Y),'(c)', 'FontWeight','bold','FontSize',16,'Color','k',...
     'BackgroundColor','w');
fprintf('Avg Width: %.3f\n', mean(handles.image_width(:)));
subplot(2,2,4);
imagesc(X, Y, handles.image_r2);
caxis([0,1]);
colorbar;
title('R^{2}');
xlabel('X (\mum)');
ylabel('Y (\mum)');
text(min(X)-3.5, min(Y),'(e)', 'FontWeight','bold','FontSize',16,'Color','k',...
     'BackgroundColor','w');
fprintf('Avg R-Squared: %.3f\n', mean(handles.image_r2(:)));
suptitle(sprintf(['Raman Data: Spectral Ave: %s\nSpatial Ave: %s',...
     'Background: %s, Fit: %s, Range: %.0f - %.0f cm^{-1}'],...
     handles.label_spectral,handles.label_spatial,handles.label_bkg,...
     handles.label_fit,str2num(get(handles.lam_start,'String')),...
     str2num(get(handles.lam_stop,'String'))));
set(handles.process,'BackgroundColor',handles.lt_rd);
set(handles.sav_f_b_dec,'Value',1);
set(handles.exp_f_b_dec,'Value',1);
set(handles.sav_mag,'Value',1);
set(handles.sav_ctr,'Value',1);
set(handles.sav_width,'Value',1);
set(handles.sav_r2,'Value',1);
set(handles.exp_mag,'Value',1);
set(handles.exp_ctr,'Value',1);
set(handles.exp_width,'Value',1);
set(handles.exp_r2,'Value',1);
set(handles.exp_fitgood,'Value',1);
set(handles.exp_fitparams,'Value',1);
guidata(hObject,handles);
end

```

```

% --- Executes on button press in fluor_fit.
function fluor_fit_Callback(hObject, eventdata, handles)
% hObject    handle to fluor_fit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

```

```

%Set limits for determining poor fit
mag_lim = 0;
r2_lim = 0.2;

```

```

(handles.fluor_mag,handles.fluor_asym,handles.fluor_width,handles.fluor_r2,...
 handles.fluor_fit_array,handles.fluor_goodness_array) =...
 process_fluor(handles.data_f,handles.shift_axis,mag_lim,r2_lim);
figure('Name',get(handles.file_display,'String'));
X = -5:5/8:5;
Y = 5:-5/8:-5;
colormap('jet');
subplot(2,2,1);
imagesc(X, Y, handles.fluor_mag/max(max(handles.fluor_mag)));
caxis([0,1]);
colorbar;
title('Normalized Fluorescence');
xlabel('X (\mum)');
ylabel('Y (\mum)');
subplot(2,2,2);
imagesc(X, Y, handles.fluor_asym);
colorbar;
title('Asymmetry');
xlabel('X (\mum)');
ylabel('Y (\mum)');
subplot(2,2,3);
imagesc(X, Y, handles.fluor_width);
colorbar;
title('Width');
xlabel('X (\mum)');
ylabel('Y (\mum)');
subplot(2,2,4);
imagesc(X, Y, handles.fluor_r2);
caxis([0,1]);
colorbar;
title('R^{2}');
xlabel('X (\mum)');
ylabel('Y (\mum)');
suptitle(sprintf(['Fluorescence Data: Spectral Ave: %s\nSpatial Ave: %s','...
 ' Fit: Lognormal'],handles.label_spectral,handles.label_spatial));
set(handles.process,'BackgroundColor',handles.lt_rd);
set(handles.exp_fluor_fitgood,'Value',1);
set(handles.exp_fluor_fitparams,'Value',1);
set(handles.exp_fluor_mag,'Value',1);
set(handles.exp_fluor_asym,'Value',1);
set(handles.exp_fluor_width,'Value',1);
set(handles.exp_fluor_r2,'Value',1);
set(handles.sav_fluor_mag,'Value',1);
set(handles.sav_fluor_asym,'Value',1);
set(handles.sav_fluor_width,'Value',1);
set(handles.sav_fluor_r2,'Value',1);
guidata(hObject,handles);
end

```

```

function file_display_Callback(hObject, eventdata, handles)
% hObject handle to file_display (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

```

% --- Executes during object creation, after setting all properties.

```

function file_display_CreateFcn(hObject, eventdata, handles)
% hObject handle to file_display (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

    if ispc && isequal(get(hObject,'BackgroundColor'), ...
        get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end
end

% --- Executes on button press in file_load.
function file_load_Callback(hObject, eventdata, handles)
% hObject handle to file_load (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
[FileName,PathName,FilterIndex] = uigetfile('*.txt');
set(handles.file_display,'String',FileName);
handles.file = sprintf('%s%s',PathName,FileName);
A = importdata(handles.file,'\t');
%Determine the number of row_label and columns - assume row 1 is Raman shift
% row 2 begins data, col 1 is row_label and col 2 is columns
handles.cols = length(find(A(:,1)==A(2,1))); %find # cols in first row
Adim = size(A);
handles.rows = (Adim(1)-1)/handles.cols; %ignore first row and calc # of rows
% put data in an image cube with (row,column,spectrum)
handles.data =...
    zeros(handles.rows,handles.cols,Adim(2)-2); %spectra ignore first two cols
handles.shift_axis = A(1,3:end); %establish the Raman shift axis
for r = 1:handles.rows
    for c = 1:handles.cols
        handles.data(r,c,:) = A((r-1)*handles.cols+c+1,3:end);
    end
end
end
s = sprintf('Rows: %.0f',handles.rows);
set(handles.row_label,'String',s);
s = sprintf('Cols: %.0f',handles.cols);
set(handles.col_label,'String',s);
s = sprintf('Spec: %.0f',Adim(2)-2);
set(handles.spec_label,'String',s);
set(handles.file_info,'BackgroundColor',[.94,.94,.94]);
pause(.25);
set(handles.file_info,'String','File Loaded');
set(handles.file_info,'BackgroundColor',[.8,.8,.8]);
set(handles.file_load,'BackgroundColor',handles.lt_bl);
handles.input_plot;
for r = 1:handles.rows
    for c = 1:handles.cols
        plot(handles.shift_axis(1,:),squeeze(handles.data(r,c,:)));
        hold on;
    end
end
end
set(handles.file_info,'BackgroundColor',[.94,.94,.94]);
pause(.25);
set(handles.file_info,'BackgroundColor',[.8,.8,.8]);
set(handles.file_info,'String','File Loaded');
set(handles.file_info,'String','Raw Data');
set(handles.file_plot,'BackgroundColor',handles.lt_bl);

```

```

    set(handles.sav_raw,'Value',1);
    set(handles.exp_raw,'Value',1);
    guidata(hObject,handles);
end

% --- Executes on button press in file_plot.
function file_plot_Callback(hObject, eventdata, handles)
% hObject    handle to file_plot (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
    handles.input_plot;
    for r = 1:handles.rows
        for c = 1:handles.cols
            plot(handles.shift_axis(1,:),squeeze(handles.data(r,c,:)));
            hold on;
        end
    end
    set(handles.file_info,'BackgroundColor',[.94,.94,.94]);
    pause(.25);
    set(handles.file_info,'BackgroundColor',[.8,.8,.8]);
    set(handles.file_info,'String','File Loaded');
    set(handles.file_info,'String','Raw Data');
end

function lam_start_Callback(hObject, eventdata, handles)
% hObject    handle to lam_start (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes during object creation, after setting all properties.
function lam_start_CreateFcn(hObject, eventdata, handles)
% hObject    handle to lam_start (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

    if ispc && isequal(get(hObject,'BackgroundColor'), ...
        get(0,'defaultUiControlBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end
end

function lam_stop_Callback(hObject, eventdata, handles)
% hObject    handle to lam_stop (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes during object creation, after setting all properties.
function lam_stop_CreateFcn(hObject, eventdata, handles)
% hObject    handle to lam_stop (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

```

```

    if ispc && isequal(get(hObject,'BackgroundColor'), ...
        get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end
end

function bkg_pts_Callback(hObject, eventdata, handles)
% hObject    handle to bkg_pts (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes during object creation, after setting all properties.
function bkg_pts_CreateFcn(hObject, eventdata, handles)
% hObject    handle to bkg_pts (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

    if ispc && isequal(get(hObject,'BackgroundColor'), ...
        get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end
end

% --- Executes on selection change in poly_fit.
function poly_fit_Callback(hObject, eventdata, handles)
% hObject    handle to poly_fit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes during object creation, after setting all properties.
function poly_fit_CreateFcn(hObject, eventdata, handles)
% hObject    handle to poly_fit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

    if ispc && isequal(get(hObject,'BackgroundColor'), ...
        get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end
end

function sg_order_Callback(hObject, eventdata, handles)
% hObject    handle to sg_order (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes during object creation, after setting all properties.
function sg_order_CreateFcn(hObject, eventdata, handles)
% hObject    handle to sg_order (see GCBO)

```

```
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called
```

```
    if ispc && isequal(get(hObject,'BackgroundColor'), ...
        get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end
end
```

```
function sg_win_Callback(hObject, eventdata, handles)
% hObject handle to sg_win (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end
```

```
% --- Executes during object creation, after setting all properties.
```

```
function sg_win_CreateFcn(hObject, eventdata, handles)
% hObject handle to sg_win (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called
```

```
    if ispc && isequal(get(hObject,'BackgroundColor'), ...
        get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end
end
```

```
function box_win_Callback(hObject, eventdata, handles)
% hObject handle to box_win (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end
```

```
% --- Executes during object creation, after setting all properties.
```

```
function box_win_CreateFcn(hObject, eventdata, handles)
% hObject handle to box_win (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called
```

```
    if ispc && isequal(get(hObject,'BackgroundColor'), ...
        get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end
end
```

```
% --- Executes on button press in exp_raw_dec.
```

```
function exp_raw_dec_Callback(hObject, eventdata, handles)
% hObject handle to exp_raw_dec (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end
```

```
% --- Executes on button press in exp_f_dec.
```



```

function exp_f_dec_Callback(hObject, eventdata, handles)
% hObject handle to exp_f_dec (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in exp_f_b_dec.
function exp_f_b_dec_Callback(hObject, eventdata, handles)
% hObject handle to exp_f_b_dec (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in sav_raw_dec.
function sav_raw_dec_Callback(hObject, eventdata, handles)
% hObject handle to sav_raw_dec (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in sav_f_dec.
function sav_f_dec_Callback(hObject, eventdata, handles)
% hObject handle to sav_f_dec (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in sav_f_b_dec.
function sav_f_b_dec_Callback(hObject, eventdata, handles)
% hObject handle to sav_f_b_dec (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in roi_plot.
function roi_plot_Callback(hObject, eventdata, handles)
% hObject handle to roi_plot (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
cla(handles.input_plot);
axis auto;
xlabel('Raman Shift (cm^{-1})');
ylabel('Signal');
title('Input Data');
[rows cols spec] = size(handles.data_dec);
for r = 1:rows
    for c = 1:cols
        plot(handles.shift_axis_dec(1,:),squeeze(handles.data_dec(r,c,:)));
        hold on;
    end
end
set(handles.file_info,'BackgroundColor',[.94,.94,.94]);
pause(.25);
set(handles.file_info,'String','ROI');

```

```

    set(handles.file_info,'BackgroundColor',[.8,.8,.8]);
end

% --- Executes on button press in proc_plot.
function proc_plot_Callback(hObject, eventdata, handles)
% hObject    handle to proc_plot (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
    cla(handles.input_plot);
    axis auto;
    xlabel('Raman Shift (cm-1));
    ylabel('Signal');
    title('Input Data');
    [rows cols spec] = size(handles.data_f_b_dec);
    for r = 1:rows
        for c = 1:cols
            plot(handles.shift_axis_dec(1,:),squeeze(handles.data_f_b_dec(r,c,:)));
            hold on;
        end
    end
    set(handles.file_info,'BackgroundColor',[.94,.94,.94]);
    pause(.25);
    set(handles.file_info,'String','Processed');
    set(handles.file_info,'BackgroundColor',[.8,.8,.8]);
end

% --- Executes on button press in ave_plot.
function ave_plot_Callback(hObject, eventdata, handles)
% hObject    handle to ave_plot (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
    cla(handles.input_plot);
    axis auto;
    xlabel('Raman Shift (cm-1));
    ylabel('Signal');
    title('Input Data');
    [rows cols spec] = size(handles.data_f_dec);
    for r = 1:rows
        for c = 1:cols
            plot(handles.shift_axis_dec(1,:),squeeze(handles.data_f_dec(r,c,:)));
            hold on;
        end
    end
    set(handles.file_info,'BackgroundColor',[.94,.94,.94]);
    pause(.25);
    set(handles.file_info,'String','Averaged');
    set(handles.file_info,'BackgroundColor',[.8,.8,.8]);
end

% --- Executes on button press in ave_spatial.
function ave_spatial_Callback(hObject, eventdata, handles)
% hObject    handle to ave_spatial (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

```

```

% --- Executes on button press in exp_fitparams.
function exp_fitparams_Callback(hObject, eventdata, handles)
% hObject handle to exp_fitparams (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

```

```

% --- Executes on button press in exp_fitgood.
function exp_fitgood_Callback(hObject, eventdata, handles)
% hObject handle to exp_fitgood (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end

```

```

% --- Executes on selection change in ave_spatial_array.
function ave_spatial_array_Callback(hObject, eventdata, handles)
% hObject handle to ave_spatial_array (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

```

```

switch(get(handles.ave_spatial_array,'Value'))
case 1
    handles.spatial_array = [0,.2,0;.2,.2,.2;0,.2,0];
    handles.label_spatial = 'NN Uniform';
case 2
    handles.spatial_array = [0,.15,0;.15,.4,.15;0,.15,0];
    handles.label_spatial = 'NN .40, 4x.15';
case 3
    handles.spatial_array = [0,.10,0;.10,.6,.10;0,.10,0];
    handles.label_spatial = 'NN .60, 4x.10';
case 4
    handles.spatial_array = [0,.05,0;.05,.8,.05;0,.05,0];
    handles.label_spatial = 'NN .80, 4x.05';
case 5
    handles.spatial_array = (1/9)*ones(3,3);
    handles.label_spatial = 'NNN Uniform';
case 6
    handles.spatial_array = [.1,.1,.1;.1,.2,.1;.1,.1,.1];
    handles.label_spatial = 'NNN .20, 8x.10';
case 7
    handles.spatial_array = [.05,.15,.05;.15,.2,.15;.05,.15,.05];
    handles.label_spatial = 'NNN .20, 4x.15, 4x.05';
case 8
    handles.spatial_array = [.05,.10,.05;.10,.4,.10;.05,.10,.05];
    handles.label_spatial = 'NNN .40, 4x.10, 4x.05';
case 9
    handles.spatial_array = [.05,.05,.05;.05,.6,.05;.05,.05,.05];
    handles.label_spatial = 'NNN .60, 8x.05';
case 10
    handles.spatial_array = [.025,.075,.025;.075,.6,.075;.025,.075,.025];
    handles.label_spatial = 'NNN .60, 4x.075, 4x.025';
case 11
    handles.spatial_array = [.025,.025,.025;.025,.8,.025;.025,.025,.025];
    handles.label_spatial = 'NNN .80, 8x.025';
otherwise
    handles.spatial_array = zeros(3,3);

```

```

        handles.label_spatial = 'Error: all zeros';
    end
    guidata(hObject,handles);
end

% --- Executes during object creation, after setting all properties.
function ave_spatial_array_CreateFcn(hObject, eventdata, handles)
% hObject    handle to ave_spatial_array (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

    if ispc && isequal(get(hObject,'BackgroundColor'), ...
        get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end

    handles.spatial_array = [0,.2,0;.2,.2,.2;0,.2,0];
    handles.label_spatial = 'NN Uniform';
    guidata(hObject,handles);
end

% --- Executes on button press in exp_fluor_fitparams.
function exp_fluor_fitparams_Callback(hObject, eventdata, handles)
% hObject    handle to exp_fluor_fitparams (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in exp_fluor_fitgood.
function exp_fluor_fitgood_Callback(hObject, eventdata, handles)
% hObject    handle to exp_fluor_fitgood (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes during object creation, after setting all properties.
function average_CreateFcn(hObject, eventdata, handles)
% hObject    handle to average (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
end

% --- Executes on button press in exp_fluor_mag.
function exp_fluor_mag_Callback(hObject, eventdata, handles)
% hObject    handle to exp_fluor_mag (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
end

% --- Executes on button press in exp_fluor_asym.
function exp_fluor_asym_Callback(hObject, eventdata, handles)
% hObject    handle to exp_fluor_asym (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB

```

```
% handles structure with handles and user data (see GUIDATA)
end
```

```
% --- Executes on button press in exp_fluor_width.
function exp_fluor_width_Callback(hObject, eventdata, handles)
% hObject handle to exp_fluor_width (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end
```

```
% --- Executes on button press in exp_fluor_r2.
function exp_fluor_r2_Callback(hObject, eventdata, handles)
% hObject handle to exp_fluor_r2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end
```

```
% --- Executes on button press in sav_fluor_mag.
function sav_fluor_mag_Callback(hObject, eventdata, handles)
% hObject handle to sav_fluor_mag (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end
```

```
% --- Executes on button press in sav_fluor_asym.
function sav_fluor_asym_Callback(hObject, eventdata, handles)
% hObject handle to sav_fluor_asym (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end
```

```
% --- Executes on button press in sav_fluor_width.
function sav_fluor_width_Callback(hObject, eventdata, handles)
% hObject handle to sav_fluor_width (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end
```

```
% --- Executes on button press in sav_fluor_r2.
function sav_fluor_r2_Callback(hObject, eventdata, handles)
% hObject handle to sav_fluor_r2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
end
```

```
function [ arrayout ] = spatial_filter( arrayin,filter_template )
%SPATIAL_FILTER Spatial averages an array by Nearest Neighbor (NN) or Next-
%Nearest Neighbor (NNN)
% filter in is a 3x3 array showing the weight for each plane. Each plane
% in arrayin should be multiplied by filter and summed.
% output array will be reduced by 2 rows and 2 columns
```

```

% Author(s): A. Fisher, K. Meissner
% Centre of NanoHealth, College of Science,
% Swansea University, Singleton Park, Swansea, SA2 8PP, United Kingdom

```

```

[rows,cols,spec] = size(arrayin);
arrayout = zeros(rows-2,cols-2,spec);
filtmat = repmat(filter_template,1,1,spec);
for r = 2:rows-1
    for c = 2:cols-1
        tmp = arrayin(r-1:r+1,c-1:c+1,:).*filtmat;
        arrayout(r-1,c-1,:) = sum(sum(tmp,1),2);
    end
end
end
end

```

```

function y=sgolayfilt_3D(x,order,framelen,weights,dim)
%SGOLAYFILT Savitzky-Golay Filtering.
% SGOLAYFILT(X,ORDER,FRAMELEN) smoothes the signal X using a
% Savitzky-Golay (polynomial) smoothing filter. The polynomial order,
% ORDER, must be less than the frame length, FRAMELEN, and FRAMELEN must
% be odd. The length of the input X must be >= FRAMELEN. If X is a
% matrix, the filtering is done on the columns of X.
%
% Note that if the polynomial order ORDER equals FRAMELEN-1, no smoothing
% will occur.
%
% SGOLAYFILT(X,ORDER,FRAMELEN,WEIGHTS) specifies a weighting vector
% WEIGHTS with length FRAMELEN containing real, positive valued weights
% employed during the least-squares minimization. If not specified, or if
% specified as empty, WEIGHTS defaults to an identity matrix.
%
% SGOLAYFILT(X,ORDER,FRAMELEN,[],DIM) and
% SGOLAYFILT(X,ORDER,FRAMELEN,WEIGHTS,DIM) operate along the dimension DIM.
%
% % Example:
% % Smooth the mtlb signal by applying a cubic Savitzky-Golay filter
% % to data frames of length 41.
% load mtlb % Load data
% smtlb = sgolayfilt(mtlb,3,41); % Apply 3rd-order filter
% plot([mtlb smtlb]);
% legend('Original Data','Filtered Data');
%
% See also SGOLAY, MEDFIL1, FILTER

% References:
% [1] Sophocles J. Orfanidis, INTRODUCTION TO SIGNAL PROCESSING,
% Prentice-Hall, 1995, Chapter 8.

% Author(s): R. Losada
% Copyright 1988-2016 The MathWorks, Inc.

```

```
narginchk(3,5);
```

```

% Check if the input arguments are valid
if round(framelen) ~= framelen
    error(message('signal:sgolayfilt:MustBeIntegerFrameLength'))
end
if rem(framelen,2) ~= 1

```

```

    error(message('signal:sgolayfilt:SignalErr'))
end
if round(order) ~= order
    error(message('signal:sgolayfilt:MustBeIntegerPolyDegree'))
end
if order > framelen-1
    error(message('signal:sgolayfilt:InvalidRangeDegree'))
end

if nargin < 4
    weights = [];
elseif ~isempty(weights)
    % Check for right length of WEIGHTS
    if length(weights) ~= framelen
        error(message('signal:sgolayfilt:InvalidDimensionsWeight'))
    end
    % Check to see if all elements are positive
    if min(weights) <= 0
        error(message('signal:sgolayfilt:InvalidRangeWeight'))
    end
end

if nargin < 5, dim = []; end

% Check the input data type. Single precision is not supported.
%chkinputdatatype(x,order,framelen,weights,dim);

% Compute the projection matrix B
B = sgolay(order,framelen,weights);

if ~isempty(dim) && dim > ndims(x)
    error(message('signal:sgolayfilt:InvalidDimensionsInput', 'X'))
end

% Reshape X into the right dimension.
if isempty(dim)
    % Work along the first non-singleton dimension
    [x, nshifts] = shiftdim(x);
else
    % Put DIM in the first dimension (this matches the order
    % that the built-in filter function uses)
    perm = [dim,1:dim-1,dim+1:ndims(x)];
    x = permute(x,perm);
end

if size(x,1) < framelen
    error(message('signal:sgolayfilt:InvalidDimensionsTooSmall'))
end
for i = 1:length(x(1,1,:))
    % Compute the transient on
    ybegin = B(end:-1:(framelen-1)/2+2,:) * x(framelen:-1:1,.,i);

    % Compute the steady state output
    ycenter = filter(B((framelen-1)/2+1,:), 1, x(:,.,i));

    % Compute the transient off
    yend = B((framelen-1)/2:-1:1,:) * x(end:-1:end-(framelen-1),.,i);

    % Concatenate

```

```

        y(:,i) = [ybegin; ycenter(frameLen:end,:); yend];
    end

    % Convert Y to the original shape of X
    if isempty(dim)
        y = shiftdim(y, -nshifts);
    else
        y = ipermute(y,perm);
    end
end

function [imag,ictr,iwidth,ir2,dout,foutf,foutg ] = ...
    process_detrend( d,x,b_size,ml,wl,r2l,f)
%DATA_DETREND remove any background drift and fit Raman peaks
% background drift removed with MATLAB's detrend function.
% raman peaks fit to user's choice of gaussian or lorentzian functions.

% Author(s): A. Fisher, K. Meissner
% Centre of NanoHealth, College of Science,
% Swansea University, Singleton Park, Swansea, SA2 8PP, United Kingdom

[rmax, cmax, spec] = size(d);
shift = x;
dout = zeros(rmax,cmax,spec);
foutf = cell(rmax,cmax);
foutg = cell(rmax,cmax);
if f ~= 1 %1 means Gaussian fit, otherwise Lorentzian
    lzt = fitype( @(a1,b1,c1,x)...
        (a1)*((c1^2)./(((x-b1).^2)+c1^2 ))); % create Lorentzian fit
end
wb = waitbar(0,'Processing Spectra...');
for r = 1:rmax
    for c = 1:cmax
        data_sh = squeeze(d(r,c,:));

        %use detrend to remove background
        data_sh_detr = detrend(data_sh);
        data_sh_detr = data_sh_detr - mean(data_sh_detr(1:b_size));
        dout(r,c,:) = data_sh_detr;
        if f == 1
            [F1,G1] = fit(shift,data_sh_detr,'Gauss1');
        else
            [F1,G1] = fit(shift,data_sh_detr,lzt,'StartPoint',...
                [max(data_sh_detr)/2,mean(shift),shift(2)-shift(1)]);
        end
        foutf{r,c} = F1;
        foutg{r,c} = G1;
        if F1.a1 <= ml
            imag(r,c) = 0; %if mag < 0, make all values 0 = bad fit
            ictr(r,c) = 0;
            iwidth(r,c) = 0;
            ir2(r,c) = 0;
        elseif F1.c1 < wl
            imag(r,c) = 0; %if width too small, bad fit
            ictr(r,c) = 0;
            iwidth(r,c) = 0;
            ir2(r,c) = 0;
        elseif G1.rsquare < r2l

```



```

    imag(r,c) = 0; %if rsquared too small, bad fit
    ictr(r,c) = 0;
    iwidth(r,c) = 0;
    ir2(r,c) = 0;
else
    imag(r,c) = F1.a1; %make an image from the magnitude of fit
    ictr(r,c) = F1.b1; %make an image from the center of fit
    iwidth(r,c) = F1.c1; %make an image from the width of fit
    ir2(r,c) = G1.rsquare; %make an image from the r^2 of fit
end

end
waitbar(r/rmax);
end
close(wb);
end

function [ imag,ictr,iwidth,ir2,dout,foutf,foutg ] = ...
    process_poly(d,x,b_size,ml,wl,r2l,p,f )
%DATA_POLY remove any background drift and fit Raman peaks
% background drift removed by subtracting a polynomial fit of the first
% b_size points.
% raman peaks fit to user's choice of gaussian or lorentzian functions.

% Author(s): A. Fisher, K. Meissner
% Centre of NanoHealth, College of Science,
% Swansea University, Singleton Park, Swansea, SA2 8PP, United Kingdom

[rmax, cmax, spec] = size(d);
shift = x;
dout = zeros(rmax,cmax,spec);
foutf = cell(rmax,cmax);
foutg = cell(rmax,cmax);
if f ~= 1 % 1 means Gaussian fit, otherwise Lorentzian
    lzt = fitttype( @(a1,b1,c1,x) ...
        (a1)*((c1^2)/(((x-b1).^2)+c1^2))); % create Lorentzian fit
end
wb = waitbar(0,'Processing Spectra...');
for r = 1:rmax
    for c = 1:cmax
        data_sh = squeeze(d(r,c,:));

        %use polynomial to remove background
        num = b_size*2; %number of background points
        data_bkgnd = zeros(num,2);
        data_bkgnd(:,1) = [shift(1:b_size);shift(end-b_size+1:end)];
        data_bkgnd(:,2) = [data_sh(1:b_size);data_sh(end-b_size+1:end)];
        [F2,~] = fit(data_bkgnd(:,1),data_bkgnd(:,2),p);
        data_sh_poly(:,1) = data_sh;
        background(:,1) = F2(shift);
        data_sh_poly(:,2) = data_sh_poly(:,1) - background(:,1);
        dout(r,c,:) = data_sh_poly(:,2);
        tmp = sort(d,'descend');
        fit_max = tmp(3)*2;
        if f == 1
            [F3,G3] = fit(shift,data_sh_poly(:,2),'Gauss1',...
                'Lower',[0,min(shift),eps],...
                'Upper',[fit_max,max(shift),max(shift)-min(shift)]);

```

```

else
    [F3,G3] = fit(shift,data_sh_poly(:,2),lzt,...
        'StartPoint',[max(data_sh_poly(:,2))/2,mean(shift),...
            shift(2)-shift(1)],'Lower',[0,min(shift),eps],'Upper',...
            [fit_max,max(shift),max(shift)-min(shift)]);
    end
    data_sh_poly(:,3) = F3(shift);
    foutf{r,c} = F3;
    foutg{r,c} = G3;
    if F3.a1 <= ml
        imag(r,c) = 0; %if mag < 0, make all values 0 = bad fit
        ictr(r,c) = 0;
        iwidth(r,c) = 0;
        ir2(r,c) = 0;
    elseif F3.c1 < wl
        imag(r,c) = 0; %if width too large, make all values 0 = bad fit
        ictr(r,c) = 0;
        iwidth(r,c) = 0;
        ir2(r,c) = 0;
    elseif G3.rsquare < r2l
        imag(r,c) = 0; %if r^2 too small, make all values 0 = bad fit
        ictr(r,c) = 0;
        iwidth(r,c) = 0;
        ir2(r,c) = 0;
    else
        imag(r,c) = F3.a1; %make an image from the magnitude of fit
        ictr(r,c) = F3.b1; %make an image from the center of fit
        iwidth(r,c) = F3.c1; %make an image from the width of fit
        ir2(r,c) = G3.rsquare; %make an image from the r^2 of fit
    end
end

end
waitbar(r/rmax);
end
close(wb);
end

```

```

function [ imag,iasym,iwidth,ir2,foutf,foutg ] = process_fluor(d,x,ml,r2l)
%PROCESS_FLUOR Fits the fluorescence data
% fits to a lognormal function

% Author(s): A. Fisher, K. Meissner
% Centre of NanoHealth, College of Science,
% Swansea University, Singleton Park, Swansea, SA2 8PP, United Kingdom

lognorm2 = fitype(@ (a,b,c,I0,offset,x) offset+(I0*b./(x-a)).*exp(-c^2).* ...
    exp(-(1/(2*c^2))*(log((x-a)/b)).^2));
[rmax, cmax, ~] = size(d);
foutf = cell(rmax,cmax);
foutg = cell(rmax,cmax);
wb = waitbar(0,'Processing Spectra...');
for r = 1:rmax
    for c = 1:cmax
        [F, G] = fit(x',squeeze(d(r,c,:)),lognorm2,'StartPoint',...
            [mean([min(x)-eps -max(x)]),500,.5,...
                max(d(r,c,:))-mean(d(r,c,end-20:end)),mean(d(r,c,end-20:end))],...
            'Upper',[min(x)-eps,2000,10,max(d(r,c,:)),max(d(r,c,:))],...

```

```

    'Lower',[-max(x),eps,eps,0,0]);

foutf{r,c} = F;
foutg{r,c} = G;

if F.I0 <= ml
    imag(r,c) = 0; %if mag < 0, make all values 0 = bad fit
    iasym(r,c) = 0;
    iwidth(r,c) = 0;
    ir2(r,c) = 0;
elseif G.rsquare < r2l
    imag(r,c) = 0; %if r^2 too small, make all values 0 = bad fit
    iasym(r,c) = 0;
    iwidth(r,c) = 0;
    ir2(r,c) = 0;
else
    imag(r,c) = F.I0; %make an image from the magnitude of fit
    iasym(r,c) = F.c; %make an image from the asymmetry of fit
    iwidth(r,c) = F.b; %make an image from the width of fit
    ir2(r,c) = G.rsquare;
end
end
waitbar(r/rmax);
end
close(wb)
% figure;
%
% subplot(2,2,1);
% imagesc(imag/max(max(imag)));
% caxis([0,1]);
% colorbar;
% title('Normalized Magnitude');
% subplot(2,2,2);
% imagesc(iasym);
% colorbar;
% title('Assymetry');
% subplot(2,2,3);
% imagesc(iwidth);
% colorbar;
% title('Width');
% subplot(2,2,4);
% imagesc(ir2);
% caxis([0,1]);
% colorbar;
% title('R Squared');
end

```