

Hardware, software and applications of super-resolution microscopy



Marcus Fantham

Supervisor: Prof. Clemens F. Kaminski

Department of Chemical Engineering and Biotechnology
University of Cambridge

This dissertation is submitted for the degree of
Doctor of Philosophy

St. Catharine's College

June 2019

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Marcus Fantham

June 2019

Acknowledgements

This PhD would not have been possible without:

the excellent guidance and supervision of Clemens Kaminski;
the enthusiasm, dedication, and help of all the members of the LAG group;
the biological and computational advice from members of the MNG and QI group, and particularly Miranda Robbins for her friendship;
the incredible work of my collaborators, especially Michelle Teplensky and Edward Avezov;
the support of my parents and family;
and all the friends who have looked after me during the last 3 years, particularly Helene Mobbs and Ellie Chan, members of the St. Catharine's College choir, and the top floor crew.

For these groups and individuals I am extremely thankful.

Abstract

This work details the development of two tools, and demonstrates their applications to answer biological questions.

Chapter 2 describes LAG SIM, a versatile and user-friendly structured illumination microscope, capable of imaging in multiple colours at 11 Hz with a resolution of 80 nm. Data can be captured and reconstructed with minimal training, allowing the microscope to be used for a wide variety of biological investigations. A collection of case-study experiments is presented, which can be used as a resource for new users to find the optimal set-up to answer their biological question.

Chapter 3 presents my new volumetric rendering program, FPBioimage, which runs in a web browser. In combination with a suite of additional software built to complement the main tool, FPBioimage makes sharing and publishing 3D volumetric data a one-click process. With a focus on creating a tool that is intuitive and easy to use, researchers around the world can now immediately view their colleagues' experimental results, even when separated by thousands of miles.

Chapter 4 shows an application of these two tools for novel cancer treatments based on metal organic frameworks (MOFs). The large pore size of these crystalline materials allows them to be loaded with drugs which would otherwise be degraded in extracellular space, for example siRNA. Experiments show that MOF complexes are successfully endocytosed by the cell, and can therefore be used as effective drug delivery systems, with the potential of treating a wide range of cancers.

Chapter 5 utilises fast, high resolution TIRF SIM to reveal a new phenomenon in the endoplasmic reticulum (ER). Evidence is presented for tubule pinching, which generates non-diffusive directed flow. Computational modelling reveals that pinching reduces the time taken for molecules to travel through the network up to 3 \times compared to Brownian particle motion. This contribution to the community's understanding of fundamental cell biology will further assist the effort to find cures for the wide range of diseases associated with ER malfunction.

Table of contents

List of figures	xiii
List of tables	xvii
List of code snippets	xix
List of publications	xxi
List of symbols and abbreviations	xxiii
1 Introduction	1
1.1 Breaking the diffraction limit reveals structure below 200 nm	1
1.2 Microscopy in 3D provides a deeper biological understanding	3
1.3 Cell biology for microscopy	4
1.3.1 Labelling organelles to visualise cell location	4
1.3.2 Labelling membrane structures to observe endocytosis	6
1.3.3 Labelling mitochondria to investigate energy-dependent processes and cancer	6
1.3.4 Labelling the cytoskeleton shows cellular structure	7
1.4 Aims and structure of this thesis	9
2 LAG SIM: A versatile, user-friendly structured illumination microscope	11
2.1 Introduction	11
2.1.1 Background to SIM	11
2.1.2 SIM for resolution doubling	16
2.1.3 Refining the reconstruction algorithm reduces artefacts	19
2.1.4 Aims for LAG SIM	21
2.2 SIM hardware	22
2.2.1 Optical path design to generate a SIM pattern	22

2.2.2	A Pockels cell provides fast polarisation rotation	25
2.2.3	Pockels cell alignment procedure	27
2.2.4	Measuring Pockels cell voltages for optimal pattern contrast	29
2.3	LabVIEW hardware control	32
2.3.1	Control requirements for SIM imaging	32
2.3.2	A user-friendly LAG SIM interface enables operation by non-expert users	33
2.3.3	Live tab for fast image preview	33
2.3.4	Acquire tab facilitates multi-dimensional imaging	35
2.3.5	Laser tab provides safe power control	36
2.3.6	Bookmark tab for saving positions and creating mosaics	37
2.3.7	Focus tab controls the autofocus	37
2.3.8	Pockels tab to find optimal voltages for maximum pattern contrast .	39
2.3.9	Debug tab contains alignment options and other advanced features .	39
2.3.10	Log tab provides access to an online logbook	40
2.4	A comparison of reconstruction algorithms reveals their pros and cons .	41
2.4.1	Lin Shao SIM reconstruction runs on CUDA GPUs	42
2.4.2	SROS-SIM facilitates optical sectioning	43
2.4.3	fairSIM allows reconstruction in ImageJ	43
2.4.4	LAG SIM is specifically designed for our microscope	44
2.5	Reconstruction with LAG SIM for artefact-free images	44
2.5.1	The LAG SIM ImageJ interface simplifies parameter-finding for non-expert users	44
2.5.2	Illumination detection provides user feedback about pattern modulation contrast	47
2.5.3	Wiener filtering and apodisation reduces noise, but can introduce artefacts	48
2.5.4	Richardson-Lucy filtering introduces less artefacts when optical sectioning is not required	49
2.5.5	OTF attenuation facilitates resolution-enhanced SIM with optical sectioning	52
2.6	Results and Discussion	54
2.6.1	Multicolour beads for chromatic alignment	57
2.6.2	Resolution-enhanced optical sectioning of HeLa cells shows MOF colocalised with endosomes	59

2.6.3	Fast, multicolour imaging of COS-7 cells reveals dynamic colocalisation between ER and lysosomes	61
2.6.4	TIRF imaging of MEF cells resolves dynamic actin structure on the cell membrane	63
2.6.5	Resolution-enhanced optical sectioning of brain-slice tissue imaging for measuring the Node of Ranvier	67
2.7	Conclusion	73
3	FPBioimage: 3D visualisation on the web	75
3.1	Introduction	75
3.1.1	Publishers require new methods of providing research data	75
3.1.2	Volumetric rendering reveals features inside 3D objects	76
3.1.3	WebGL enables volume rendering in a web browser	78
3.1.4	Aims for FPBioimage	80
3.2	Method: software design	80
3.2.1	The fragment shader calculates the colour of each screen pixel efficiently	81
3.2.2	A texture atlas scheme enables efficient loading of 3D data	83
3.2.3	Movement provides personal exploration of the data	85
3.2.4	Bookmarking is implemented with a state machine to share perspectives across the world	86
3.2.5	Screenshots can be captured at a higher resolution than the display .	88
3.3	Results and discussion	89
3.3.1	An intuitive interface requires no user training	90
3.3.2	Measuring performance demonstrates compatibility with consumer-grade computers	97
3.4	FPBioimage Suite	102
3.4.1	Plugins are available for popular image analysis programs	102
3.4.2	OMERO-FPBioimage integrates volume rendering with organised image repositories	103
3.4.3	Mobile apps bring volumetric rendering to smartphones with virtual reality support	104
3.5	Conclusion	106
4	MOFs: Metal organic frameworks for drug delivery	109
4.1	Introduction	109
4.1.1	siRNA for cancer treatment	109

4.1.2	MOFs as delivery vehicles	110
4.1.3	Structure of this chapter	111
4.2	Results and methods	112
4.2.1	Temperature treatment of MOFs delays the release time	112
4.2.2	Loading MOFs with siRNA for therapeutic effects	115
4.2.3	MOFs deliver DCA to restart the Krebs cycle, with TPP as a cofactor to target mitochondria	121
4.3	Conclusion	125
4.4	Future work	126
5	ER: Endoplasmic reticulum network dynamics	127
5.1	Introduction	127
5.1.1	ER malfunction is linked to a variety of diseases	127
5.1.2	Flow in the ER is an active process	128
5.1.3	Structure of this chapter	129
5.2	Results and methods	130
5.2.1	High-speed SIM and image analysis reveals a dynamic ER network	130
5.2.2	Active tubule pinching correlates with luminal flow velocity . . .	134
5.2.3	Graphical simulation shows pinching accelerates luminal mixing .	136
5.3	Conclusion	140
5.4	Future research	141
6	Conclusion	143
References		147
Appendix A	LAG SIM Software Implementations	165
A.1	LabVIEW control software	165
A.2	SROS-SIM reconstruction on GPU	169
A.3	ImageJ/Fiji plugin	173
Appendix B	Directed flow in the Endoplasmic Reticulum	179
B.1	Flow calculations	179
B.2	FRET measurements	181

List of figures

1.1	Introduction: Animal cell labelled with organelles relevant to this thesis	5
1.2	Introduction: Cells import external contents through endocytosis	7
1.3	Introduction: The cytoskeleton gives the cell structure and strength	8
2.1	LAG SIM: Both TIRF microscopy and structured illumination patterns can be used to remove out-of-focus light	12
2.2	LAG SIM: Structured illumination can be used to computationally remove out-of-focus light	14
2.3	LAG SIM: A sinusoidal illumination pattern is visible as delta peaks in Fourier space	17
2.4	LAG SIM: Reconstruction of SIM images takes place in Fourier space	18
2.5	LAG SIM: Wiener filtering of the SIM OTF is required for artefact-free reconstruction	20
2.6	LAG SIM: Various optics work together to pattern the laser light with a SIM pattern and apply polarisation rotation for optical sectioning and resolution enhancement	23
2.7	LAG SIM: Laser polarisation must be perpendicular to the sinusoidal illumination for high pattern contrast	26
2.8	LAG SIM: Alignment of the Pockels cell and quarter-wave plate to facilitate fast linear polarisation rotation	28
2.9	LAG SIM: A Pockels cell is used to rotate the polarisation of laser light for maximum SIM pattern contrast	29
2.10	LAG SIM: Measurements of a bead sample reveal the optimal Pockels cell voltages for maximum pattern contrast	31
2.11	LAG SIM: The LabVIEW user interface for controlling LAG SIM is designed for operation by non-expert users	34
2.12	LAG SIM: An image of a full mouse brain can be captured as a mosaic of images	38

2.13 LAG SIM: Displaying specially designed patterns on the SLM assists with alignment of LAG SIM	40
2.14 LAG SIM: A Fiji interface makes artefact-free reconstruction quick and simple for non-expert users	46
2.15 LAG SIM: The Wiener parameter and apodisation strength must be chosen to minimise artefacts	50
2.16 LAG SIM: Richardson-Lucy filtering can further reduce SIM reconstruction artefacts	51
2.17 LAG SIM: OTF attenuation as part of the SIM reconstruction process removes out-of-focus light	53
2.18 LAG SIM: Optical sectioning through OTF attenuation removes out-of-focus light	55
2.19 LAG SIM: Removing out-of-focus reveals 3D structure	56
2.20 LAG SIM: Multicolour alignment beads are used for correcting chromatic offset	58
2.21 LAG SIM: 3D SIM reconstruction reveals that MOFs are endocytosed by HeLa cells	62
2.22 LAG SIM: Fast, multicolour imaging of ER in cells reveals co-localisation between lysosomes and ER tubules	64
2.23 LAG SIM: TIRF-SIM imaging of actin in MEF cells removes out-of-focus light allowing actin development to be studied	66
2.24 LAG SIM: Diagram of a neuron showing an axon protected by the myelin sheath	68
2.25 LAG SIM: Imaging in tissue degrades the SIM pattern due to photon scattering and out-of-focus light	70
2.26 LAG SIM: Multicolour optical sectioning SIM to measure the Node of Ranvier on ventrally- and dorsally-derived oligodendrocytes	72
3.1 FPBioimage: Raymarching facilitates volumetric rendering of 3D data	77
3.2 FPBioimage: WebGL is supported by 93% of users browsing the web	79
3.3 FPBioimage: Bookmark creation and restoration is controlled by a state machine	88
3.4 FPBioimage: Users are presented with a simple interface which is intuitive to use	90
3.5 FPBioimage: Side panels provide more advanced rendering options	91
3.6 FPBioimage: Advanced rendering features provide unique views of volumetric data	92

3.7 FPBioimage: Four projection methods highlight different details in volumetric data	94
3.8 FPBioimage: Screenshots allow views to be captured at a higher resolution than the user's display	95
3.9 FPBioimage: Keyboard controls provide shortcuts to all FPBioimage functions, allowing mouse-free operation	96
3.10 FPBioimage: Bookmarks can be shared as a URL for another user to open	98
3.11 FPBioimage: Adjustable quality settings allow FPBioimage to be used on a range of devices	101
3.12 FPBioimage: OMERO.web uses FPBioimage as its default renderer for 3D data	104
3.13 FPBioimage: The FPBioimage mobile app provides volumetric rendering in virtual reality	105
3.14 FPBioimage: LiveSlides in Powerpoint brings interactive FPBioimage rendering to presentations	107
4.1 MOFs: The NU-1000 MOF has a large pore size to store other molecules	112
4.2 MOFs: PXRD confirms calcein enters NU-1000 and temperature treatment causes partial pore collapse	113
4.3 MOFs: Payload release is delayed by temperature treatment	114
4.4 MOFs: Calcein-loaded NU-1000 is taken up by cells over a 24 h period	116
4.5 MOFs: An endocytosis study shows NU-1000 MOF is taken up by HeLa cells through the clathrin-mediated pathway	117
4.6 MOFs: NU-1000 protects siRNA from degradation by enzymes in extracellular space	118
4.7 MOFs: siRNA-loaded NU-1000 is endocytosed by HEK-293 cells, and released to the cytoplasm with an endosome release factor	119
4.8 MOFs: siRNA suppresses mCherry expression when an endosome release factor is loaded to MOFs	120
4.9 MOFs: UiO-66 MOF loaded with DCA changes mitochondria's shape	123
4.10 MOFs: Mitochondria segmentation in CellProfiler allow statistical shape analysis	124
4.11 MOFs: Mitochondria become more circular when treated with DCA and TPP delivered by UiO-66 MOF	124
4.12 MOFs: Loading MOFs with TPP as a cofactor facilitates lower drug concentration	125

5.1	ER: Single particle tracking reveals directed flow in ER tubules	128
5.2	ER: Flow velocity is reduced upon ATP depletion; a FRET probe reveals this is not due to increased crowding	129
5.3	ER: Growing and shrinking tubules show the ER network is in a state of dynamic equilibrium	131
5.4	ER: Weka segmentation allows the ER network to be extracted from a non-uniform background	132
5.5	ER: Distribution of tubule length remains constant despite growing and shrinking of individual tubules	133
5.6	ER: Tubule pinching can be observed with the ‘Edges’ colourmap	135
5.7	ER: Phototoxicity confirms the ability of ER to undergo reversible pinching	136
5.8	ER simulation: After one minute of mixing by Brownian motion the simulated ER mixture is not homogeneous	138
5.9	ER simulation: A homogeneous mixture is achieved after one minute in the ER simulation with pinching	139
5.10	ER simulation: Tubule pinching provides faster mixing than Brownian motion alone	140
5.11	ER simulation: The simulated model shows similar particle velocity tracks to experimental data	141
5.12	ER: Lysosomes form strong contacts with ER tubule endpoints to rearrange the network	142
A.1	Logging user activity with Labstep allows any problems to be identified and fixed quickly	166
A.2	SubVIs for uploading experiment information to Labstep	168
A.3	Top level functions for uploading to Labstep	169
B.1	Photoconversion allows luminal ER proteins to be tracked	180
B.2	Measuring fluorescent lifetime of a FRET-based probe is used to assess ER crowdedness	183

List of tables

2.1	LAG SIM: Different combinations of lenses, wavelengths, and SLM grating patterns provide different resolution enhancement factors	48
3.1	FPBioimage: Adjustable quality settings allow high performance on consumer-grade computers	100
A.1	SROS-SIM reconstruction is 32X faster on the GPU than the CPU	171

List of code snippets

2.1	Lin Shao's CUDA reconstruction program is executed as a command line program through a MATLAB interface	42
3.1	Fragment shader code for volumetric ray marching	82
3.2	Java code for 3D texture atlas arrangement	84
3.3	C# code using built-in Unity functions for rotating the volumetric data . . .	85
3.4	C# code using built-in Unity functions for moving the camera in a first-person manner	86
3.5	C#-JavaScript interface code for interacting with the browser's local storage to save and share bookmarks	87
3.6	C# code for capturing a high-resolution screenshot and presenting it to the user as a download	89
A.1	A check must be performed at the start of the main program to detect if a suitable graphics card is installed for GPU computation	171
A.2	A simple helper function, <code>gpuThis()</code> , sends an array to graphics memory if a GPU is available, but leaves the array in CPU memory if not	172
A.3	The function <code>gpuThis()</code> should be wrapped around new arrays when they are preallocated	172
A.4	Extracts from the Beanshell script used for the LAG SIM parameter tester .	175

List of publications

David Holcman, Pierre Parutto, Joseph E Chambers, **Marcus Fantham**, Laurence J Young, Stefan J Marciniak, Clemens F Kaminski, David Ron, and Edward Avezov. Single particle trajectories reveal active endoplasmic reticulum luminal flow. *Nature cell biology*, 20:1118–1125, 2018.

Janin Lautenschläger, Amberley D Stephens, Giuliana Fusco, Florian Ströhl, Nathan Curry, Maria Zacharopoulou, Claire H Michel, Romain Laine, Nadezhda Nesporitaya, **Marcus Fantham**, et al. C-terminal calcium binding of α -synuclein modulates synaptic vesicle interaction. *Nature communications*, 9(1):712, 2018.

Peyman Z Moghadam, Timur Islamoglu, Subhadip Goswami, Jason Exley, **Marcus Fantham**, Clemens F Kaminski, Randall Q Snurr, Omar K Farha, and David Fairen-Jimenez. Computer-aided discovery of a metal–organic framework with superior oxygen uptake. *Nature communications*, 9(1):1378, 2018.

Michelle H Teplensky, **Marcus Fantham**, Peng Li, Timothy C Wang, Joshua P Mehta, Laurence J Young, Peyman Z Moghadam, Joseph T Hupp, Omar K Farha, Clemens F Kaminski, and David Fairen-Jimenez. Temperature treatment of highly porous zirconium-containing metal–organic frameworks extends drug delivery release. *Journal of the American Chemical Society*, 139(22):7522–7532, 2017.

Marcus Fantham and Clemens F Kaminski. A new online tool for visualization of volumetric data. *Nature Photonics*, 11(2):69, 2017.

List of symbols and abbreviations

The International System of Units (SI) has been used for all units of measurement, except for any additions noted in the following tables.

Abbreviations	
2D	2-dimensional
3D	3-dimensional
ANOVA	Analysis of variance
API	Application programming interface
ASI	Applied Scientific Instrumentation (<i>company</i>)
ATP	Adenosine triphosphate
AWS	Amazon web services
<i>C. elegans</i>	<i>Caenorhabditis elegans</i>
COS-7	Monkey kidney fibroblasts
CT	Computed tomography
CUDA	Compute unified device architecture (<i>programming language</i>)
DAC	Digital-to-analog converter
DAPI	4',6-diamidino-2-phenylindole (<i>fluorescent stain</i>)
DM	Dichroic mirror
DNA	Deoxyribonucleic acid
ER	Endoplasmic reticulum
FACS	Fluorescence-activated cell sorting
FPBioimage	First Person Bioimage
FPS	Frames per second
FRAP	Fluorescence recovery after photo-bleaching
FWHM	Full-width half-maximum
GPU	Graphics processing unit
HEK-293	Human embryonic kidney 293 cells
HeLa	Henrietta Lacks (<i>cell line</i>)

HFF	Human skin fibroblast
HILO	Highly inclined and laminated optical sheet
HTML	Hypertext markup language
HTTP	Hypertext transfer protocol
HSV-1	Herpes Simplex Virus Type 1
IDR	Image data resource
JSON	JavaScript object notation
LAG	Laser Analytics Group
LCVR	Liquid crystal variable retarder
LED	Light emitting diode
MEF	Mouse embryonic fibroblasts
MRI	Magnetic resonance imaging
mRNA	Messenger RNA
MT	Michelle Teplensky
NA	Numerical aperture
OPC	Oligodendrocyte progenitor cell
OPT	Optical projection tomography
OTF	Optical transfer function
PALM	Photoactivated localisation microscopy
PNG	Portable network graphics
PS	Proton-Sponge®
PSF	Point spread function
PXRD	Powder x-ray diffraction
QWP	Quarter-wave plate
RFP	Red fluorescent protein
RGB	Red-green-blue
RL	Richardson-Lucy
RNA	Ribonucleic acid
RNAi	RNA interference
sCMOS	Scientific complementary metal-oxide-semiconductor
SH	Salame Haddad
SIM	Structured illumination microscopy
SiR-DNA	Silicon-rhodamine DNA (<i>fluorescent stain</i>)
siRNA	Small interfering RNA
SLM	Spatial light modulator
SPIM	Single plane illumination microscopy

SROS	Super-resolution optical-sectioning
STED	Stimulated emission depletion microscopy
STORM	Stochastic optical reconstruction microscopy
TIRF	Total internal reflection fluorescence
TMR	Tetramethylrhodamine
TPP	Triphenylphosphonium
TTL	Transistor-transistor logic
UI	User interface
URL	Uniform resource locator
UV	Ultra-violet
VR	Virtual reality
W-4PiSMSN	Whole-cell 4Pi single-molecule switching nanoscopy
WebGL	Web graphics library

Symbols	
\Leftrightarrow	Fourier transform operation
\otimes	Convolution operation
$\hat{\cdot}$	Fourier-transformed variable
\in	Is an element of the set
α	Acceptance angle
δ	Mean free path
ζ	Evanescence field exponential decay constant
θ	Rotation angle
λ	Wavelength of light
v	Wave number
Σ	Sum
τ	Time between collisions
ϕ	Phase
a	Apodisation strength
$A(\mathbf{k})$	Apodisation function
c	Speed of light
d	Minimum separation distance
D	Diffusion coefficient
D_i	Captured image data
D_R	Reconstructed image data
E	Energy

h	Plank's constant
H	Point spread function
\hat{H}	Optical transfer function
h	Hours - unit
I	Illumination pattern
i	Incrementing integer
j	imaginary unit = $\sqrt{-1}$
k_x	Spatial frequency in the x -direction
\mathbf{k}	Vector of spatial frequencies
m	Modulation depth
M	Molar concentration, mol/L
min	Minutes - unit
n	Refractive index
N	Number of bookmarks
\mathbf{p}	Vector in the direction of sinusoidal pattern
S	Distribution of sample fluorescence
t	Modulation frequency
V/V	Amplification factor (output volt per input volt) - unit
v	Velocity
W	Widefield image
w	Wiener coefficient
x	Spatial variable in the lateral direction
z	Spatial variable in the axial direction
\mathbb{Z}	Set of all integers

Chapter 1

Introduction

1.1 Breaking the diffraction limit reveals structure below 200 nm

Our understanding of the biology of life is limited to what we can observe. Naked human vision is limited to a spatial resolution on the order of 100 µm [1]. The invention of high power microscopes, often attributed to van Leeuwenhoek in 1660 [2], and pioneering experiments by Robert Hooke [3] and Swammerdam [4] around the same time, revealed phenomena previously invisible to the naked human eye.

Lens technology continuously improved through the 17th and 18th Centuries [5, 6], revealing smaller and smaller objects, until in 1873 Abbe showed that the diffraction of light as it passes through a lens places a fundamental limit on the size of objects which can be resolved [7]. The famous Abbe diffraction limit, given in Equation 1.1, states that the minimum separation distance, d , at which two objects can be resolved decreases with the wavelength of light, λ , but increases with the lens' acceptance angle, α , and refractive index of the medium between the lens and the object, n . The wavelength range of visible light is 400 nm–700 nm, and the maximum acceptance angle of a lens approaches 90°. The refractive index of air is 1.0, although special immersion oils can be used to increase n to ~1.5. Using Equation 1.1, we then calculate that the maximum resolution of a microscope lens is ~200 nm.

$$d = \frac{\lambda}{2n \sin(\alpha)} \quad (1.1)$$

A number of technologies which are not based on visible light exist to greatly surpass the resolution of optical microscopes, including transmission electron microscopy and scanning

electron microscopy [8–10]. However these techniques require sample preparation which includes a combination of freezing, fixation, or toxic staining [11]; must image the sample under vacuum conditions at cryogenic temperatures; and use a minimum electron dose which is lethal to cells [12]. Electron microscopy is therefore incompatible with live cell biology.

Furthermore, the invention of fluorescent labelling gives optical microscopy a distinct advantage over any other microscopy technique in terms of specificity. Sophisticated biochemistry, based on antibody chemistry, genetic expression, and other biotechnologies, can be used to label specific cellular compartments or organelles with fluorescent molecules [13]. When these fluorescent molecules are illuminated with a certain wavelength of light, they absorb photons, exciting electrons to a higher electronic energy state [14, ch. 1]. The electrons lose some energy through so-called vibrational states, then as the electron falls back to the ground electronic state photons are emitted at a red-shifted wavelength - where the wavelength shift is proportional to the energy lost through the vibrational states, as per the Plank-Einstein relation $E = h\nu = hc/\lambda$ [15, ch. 39].

Fluorescent labelling has a number of unique applications. Firstly, since the light used to excite fluorescence is blocked by filters before it reaches the observer, only fluorescence emission light is seen through the microscope [16]. This creates a high-contrast image showing the object of interest against a black background. Furthermore the specificity of labelling means that other parts of the cell which are not of interest for a given experiment are invisible, further enhancing the observability of the labelled target [13]. Finally, if two or more fluorescent labels are used with non-overlapping fluorescent spectra, then imaging in multiple channels can be used for colocalisation studies, for example to confirm that a certain protein interacts with a certain organelle [17].

The unique advantages of fluorescent labelling have motivated researchers in the last two decades to invent optical methods to image at resolutions below the diffraction limit [18]. Whilst it remains impossible to capture an individual image beyond the diffraction limit, digital cameras and intensive computational reconstruction have enabled techniques which sacrifice temporal resolution for enhanced spatial resolution. This is the principle behind techniques based on photoactivated localisation microscopy (PALM) [19] and stochastic optical reconstruction microscopy (STORM) [20]: in each image, only a small fraction - typically less than 1 % - of fluorophores are emitting light. Multiple images are captured so that the molecules blink in a video of the raw data; assuming that two adjacent fluorophores do not emit simultaneously, the centre of the diffraction pattern is calculated as the true location of the molecule. When enough frames have been captured for every fluorescent molecule to have a high probability of emitting - for example, 1000 frames for a 1 % duty cycle - the

localisation algorithm is applied to every image in the dataset, and a super-resolution image can be constructed.

In STORM microscopy, there is a trade-off between spatial resolution and acquisition time; a lower duty cycle with more frames captured will produce a higher spatial resolution, as there is a lower probability of two adjacent fluorescent molecules emitting simultaneously. In practice, STORM has been shown to achieve a spatial resolution of ~10 nm–20 nm; however resolutions in this range typically require 1 min–5 min of imaging to capture the data for a single reconstructed super-resolution image [21]. Dynamic events in live cell biology cannot be captured, and so a faster technique must be used for such experiments.

Structured illumination microscopy (SIM) can produce a resolution-enhanced image beyond the diffraction limit with just 9 raw images [22]. The technique is described in detail in Chapter 2, which explains how computational reconstruction produces an image with twice the resolution of a widefield image captured with the same lens. Using state-of-the-art lenses and cameras, we have shown SIM datasets with a spatial resolution of <90 nm, at a video rate of 11 Hz [23].

1.2 Microscopy in 3D provides a deeper biological understanding

Capturing 2-dimensional (2D) images of microscopic biology has enhanced our understanding of life and disease, enabling development of new medicines and diagnosis techniques. However human biology is made up of inherently 3-dimensional (3D) structures, from the sub-cellular scale to the full body.

Other fields of medicine have shown the advantages offered by capturing 3D information; for example, compare a 2D x-ray image with a 3D computed tomography (CT) dataset. Whilst the former can be used for detecting breaks in bones or confirming the presence of foreign objects in the body [24], the extra information provided by 3D data significantly improves diagnosis and treatment planning in oncology and trauma by accurately modelling tumours [25] or fractures [26].

In widefield epifluorescent microscopy, out-of-focus light makes visualising 3D reconstructions futile, as light emitting from a given axial position obscures structures in other focal planes of the lens. To accurately represent fluorescent objects in 3 dimensions, an optical section must be captured - that is, a clear image deep in the sample without out-of-focus light. Several optical sections can then be gathered at a number of depths to build up a 3-dimensional dataset.

A number of modern microscopy techniques are able to perform optical sectioning. Confocal microscopy is a point-scanning technique which uses a pinhole in the emission path to remove out-of-focus light, building up an optically-sectioned image pixel-by-pixel [27, 28]. Single plane illumination microscopy (SPIM) uses one objective to generate a thin sheet of light in the sample which is imaged by a second orthogonal objective, providing optical sectioning physically by not generating out-of-focus light in the first place [29]. Optical projection tomography (OPT) can be thought of as the optical analogue of x-ray computerised tomography, capturing images at several rotations of the sample and using the inverse Radon transform to reconstruct a 3-dimension dataset [30]. Structured illumination microscopy (SIM) uses pattered illumination light to computationally remove out-of-focus light, which can also provide resolution enhancement, detailed in Chapter 2. Several techniques exist for 3D localisation microscopy, utilising one or more of astigmatism [31], double-helix point-spread function (PSF) engineering [32], dual-lens interferometry [33], multiple focus planes [34], and Airy beams [35]; these techniques report typical resolutions of 10 nm–30 nm in 3 dimensions.

Resolving microscopic biology in 3D reveals structures and events that would otherwise be invisible; for example, observing the development of zebrafish embryo from 64 cells to 20000 cells [36], or the 3D interaction of mitochondria with microtubules [37]. Presenting and publishing these information-rich datasets in an informative way presents a challenge to authors, who are restricted to showing 2D slices or projections of the data or videos from one perspective. If readers were able to interact directly with the 3D volume they would gain a deeper understanding of the volumetric arrangement of structures. To address this problem, a new web-based tool for visualising volumetric 3D data is presented in Chapter 3.

1.3 Cell biology for microscopy

The tools described in Chapters 2 and 3 of this document have valuable applications in cell biology, which are detailed in Chapters 4 and 5. Since this thesis covers a broad range of themes, from engineering through computer science to biology, this section is included to provide a background of cell biology necessary for appreciating the findings detailed in the application chapters.

1.3.1 Labelling organelles to visualise cell location

Figure 1.1 shows a generic animal cell [38]. Highlighted in colour are 8 organelles which are investigated in work presented throughout this thesis. The scale and quoted sizes are

approximate, included to convey which organelles are resolvable in standard widefield microscopy, and which are smaller than the diffraction limit requiring super-resolution to observe clearly.

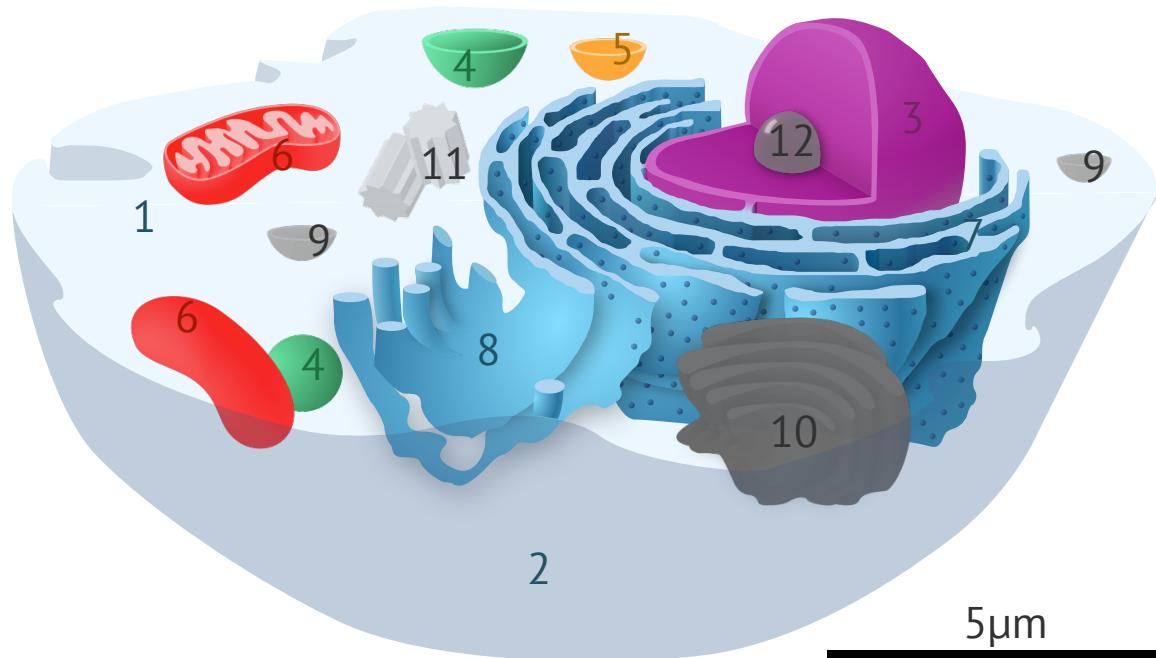


Figure 1.1: A generic animal cell, adapted from Wikipedia [38], but with the organelles discussed in this thesis highlighted in colour. Scalebar is approximate: most animal cells are 10 μm –30 μm in diameter. Numbered annotations with approximate sizes from [39] are as follows:

- | | |
|---------------------|--|
| 1. Cytosol | 10 μm –30 μm diameter |
| 2. Cell membrane | 1 nm thick |
| 3. Nucleus | 6 μm diameter (average) |
| 4. Lysosome | 0.1 μm –1.2 μm |
| 5. Endosome | 0.05 μm –1 μm |
| 6. Mitochondria | 250 nm wide, 2 μm –7 μm long |
| 7. ER network | 30 nm–100 nm diameter |
| 8. ER sheets | 40 nm–70 nm thick |
| 9. Vesicle | 200 nm–300 nm diameter |
| 10. Golgi apparatus | 1 μm –2 μm wide |
| 11. Centrosome | 200 nm diameter |
| 12. Nucleolus | 2 μm –4 μm diameter |

Each of the organelles highlighted in Figure 1.1 can be labelled with a fluorescent marker. Fluorescent labelling can provide a variety of information about the cell, depending on the function of the organelles of interest [13].

The cell nucleus contains DNA, the genetic code of life [40], and can be visualised under a microscope by labelling with dyes such as DAPI or SiR-DNA [41, 42]. Labelling the nucleus provides an easy method of automated cell counting, or an easy method for locating cells if it is not clear from other labels [43].

The contents of the cell are contained by the cell membrane, which acts as a barrier between the contents of the cell and extra-cellular space [44, *ch. 1*]. Membrane proteins can be specifically labelled to visualise the outline of cells, to confirm whether another fluorescent object is located inside or outside of the cell [45–47].

1.3.2 Labelling membrane structures to observe endocytosis

The cell can import useful raw materials through the membrane by a process called endocytosis [40]. To enter the cell, an intracellular compartment is created from the plasma membrane to contain the foreign material. Once inside the cell, there are many pathways for processing the endocytosed contents [48–50]. A typical pathway, shown in Figure 1.2, imports contents into small vesicles called early endosomes, which are selectively sorted to either recycle the contents to the cell membrane or develop into late endosomes. Late endosomes have an acidic environment, about pH 5.5, and begin hydrolytic digestion of their contents [51]. Late endosomes mature into lysosomes with a more acidic environment and enzymes which further break down their contents [40]. Labelling endosomes or lysosomes alongside other objects can be used for colocalisation studies, as performed in Chapter 4, which show whether another fluorescent substance is contained within the organelle or is moving freely in the cytosol [52].

In addition to endosomes and lysosomes, other internal membrane structures include the endoplasmic reticulum (ER) and the Golgi apparatus. Protein folding and synthesis occurs in the ER; conversely, the Golgi apparatus is involved in further protein processing and secretion of substances from the cell [53]. Content is transferred between all intracellular compartments, either through vesicles or by direct organelle-organelle interaction. To visualise the ER, fluorescent labels can be applied to either the ER membrane or to luminal proteins contained within the ER network [54]; both labelling techniques are utilised in Chapter 5.

1.3.3 Labelling mitochondria to investigate energy-dependent processes and cancer

Mitochondria generate energy for the cell by converting food to adenosine triphosphate (ATP), an energy-rich molecule that is used as the basic chemical fuel to power cellular

activity [44]. This process, known as the Krebs cycle, consumes oxygen and produces carbon dioxide and water as waste products, so is a type of cellular respiration. Furthermore, mitochondria are responsible for triggering apoptosis - that is, controlled cell death [55].

In cancerous cells the Krebs cycle is bypassed, and ATP is produced by a more direct process known as glycolysis [56]. By avoiding the Krebs cycle, cancer cells stop the mitochondria from triggering apoptosis, such that the cells become immortal and continuously reproduce to form tumours [55]. Chapter 4 uses fluorescently labelled mitochondria to assess their interaction with a drug designed to restart the Krebs cycle in cancerous cells.

All the organelles discussed so far are located in the cytosol, a crowded solution of many different types of molecules filling the volume of the cell [57]. This solution is so dense that the cytosol behaves as a water-based gel rather than a liquid [44].

1.3.4 Labelling the cytoskeleton shows cellular structure

The cellular cytoskeleton is composed of long, fine protein filaments forming a system of girders, ropes, and motors to give the cell mechanical strength, control its shape, and drive movement [44]. There are three types of filament: microtubules, actin, and intermediate

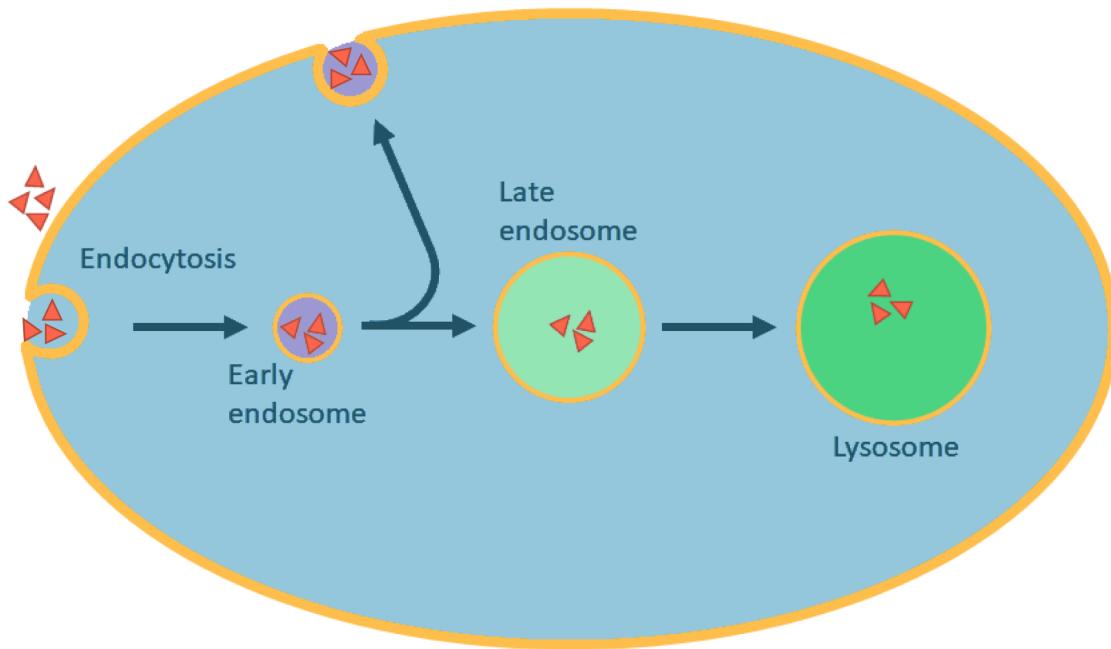


Figure 1.2: A typical endocytosis pathway shows the formation of various organelles, each of which contains a different environment for ingestion and processing of the external contents. Other endocytosis pathways also occur, for example to pass the contents to ER or release it directly into the cytosol. Figure adapted from [40].

filaments, as seen in Figure 1.3. Labelling these components with fluorescent markers provides a clear visualisation of the cell boundary, and shows strong colocalisation between filaments and other organelles [58]. They are also often used as test samples to assess the performance of a microscope, for example determining resolution by measuring the minimum resolvable separation distance between two fibres [59, 60].

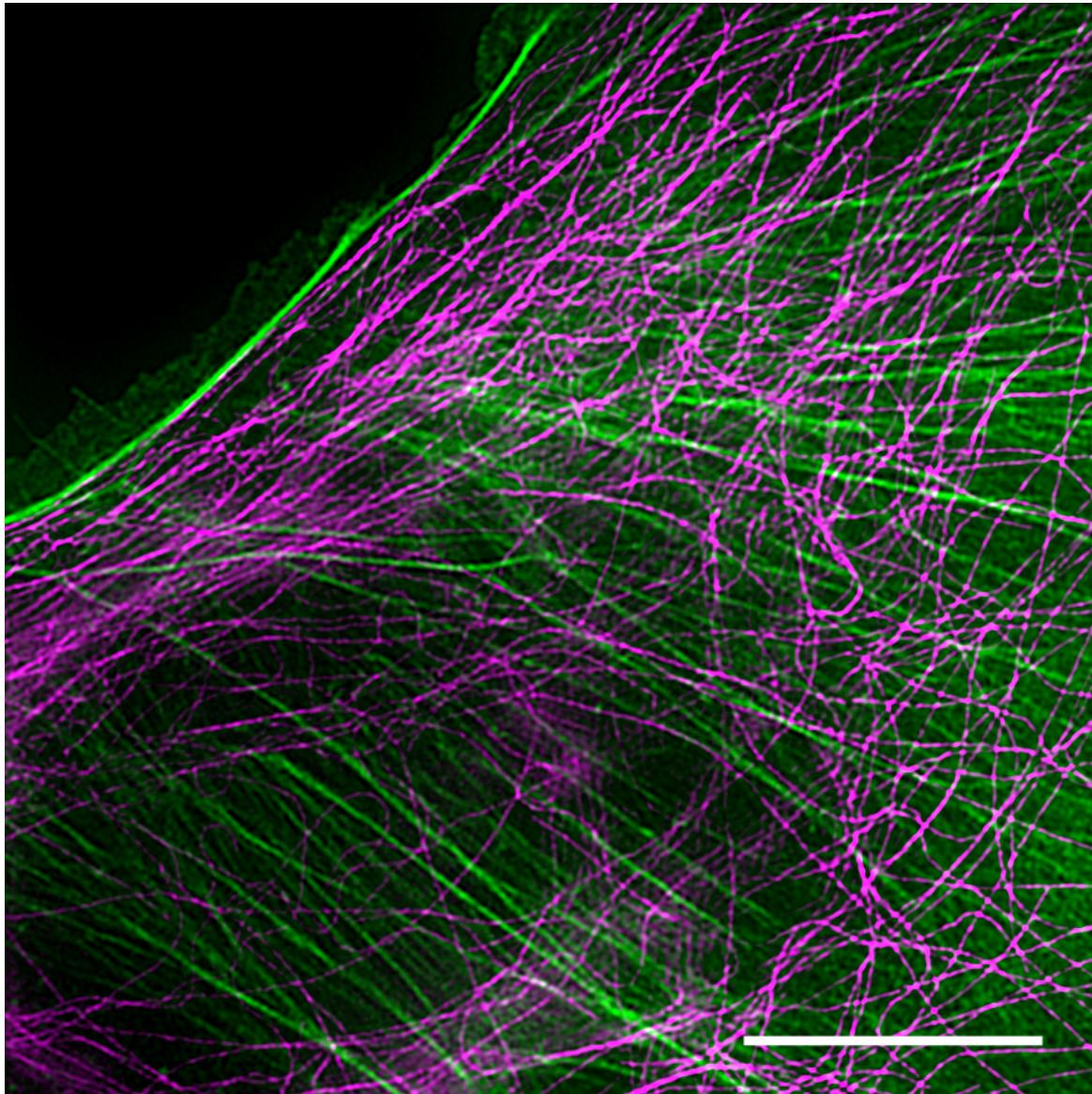


Figure 1.3: The cytoskeleton is composed of actin filaments, shown in green, microtubules, shown in magenta, and intermediate filaments, not shown, which give the cell structure and strength. The sample shown is of commercial GATTA Cells 4C, imaged on LAG SIM. Scalebar is 10 μm .

1.4 Aims and structure of this thesis

Since October 2015, I have been working in the Laser Analytics Group building tools and applying them to answer biological questions.

Chapter 2 describes the development a structured illumination microscope. SIM is a microscopy technique which generates images in the niche between STORM, which provides 20 nm resolution but takes 5 min or more per acquisition; and widefield microscopy, which can image at 100 frames per second but at a diffraction limited resolution of about 200 nm. The Laser Analytics Group's SIM (LAG SIM) achieves 80 nm resolution at 11 frames per second.

As a relatively new invention, super-resolution microscopes are often difficult to use and require expert skill to achieve their optimal performance. An important aim for LAG SIM was that any researcher should be able to use it, with minimal training required for capturing and reconstructing SIM data. In Chapter 2, a showcase of reconstructed SIM data from a wide variety of experiments is provided as a reference for capturing high-quality images.

Many questions in biology can only be answered by observing the specimen in 3D. LAG SIM is able to capture 3D data using a capability known as optical sectioning. Whilst this data is information-rich and exciting to explore in detail, transferring it for remote colleagues to evaluate is problematic due to large file sizes and complicated, expensive analysis software.

To address this issue, Chapter 3 presents FPBioimage, an online tool for sharing and publishing 3D volumetric data. A key aim again was that the software was easy to operate, so that non-expert users could effortlessly view or share data from and to anywhere in the world. The chapter describes the development of the software, and presents a wide assortment of data from different 3D imaging technologies highlighting the unique features of the tool.

Chapter 4 demonstrates the capability of LAG SIM for facilitating hypothesis-driven research, introducing the use of metal organic frameworks (MOFs) as drug delivery vehicles for treating cancer. Many potentially therapeutic drugs degrade quickly in extra-cellular media, either making them unsuitable for use as medicines or requiring a high dosage with unwelcome side-effects. Model drugs, including small-interfering ribonucleic acid (siRNA) and dichloroacetic acid (DCA), were loaded into the porous structure of MOFs, with the aim of protecting the contents for delivery into cells.

The LAG SIM was used to image the uptake of the treatment over a time period of 24-hours, and 3-dimensional images have been published online with FPBioimage to allow other researchers to explore the data themselves and verify the results. Three biological experiments are described showing the therapeutic effects offered by this newly-developed technology.

Chapter 5 utilises the fast, high-resolution imaging provided by LAG SIM to present pioneering discoveries about the endoplasmic reticulum (ER). It has recently been discovered that proteins within the ER move with directed flow, rather than diffusive motion which was previously assumed. Working in collaboration with the researchers who first discovered this phenomenon, LAG SIM's unique spatio-temporal resolution was utilised with the aim of finding the origin of directed luminal flow.

High-speed videos reveal the ER membrane pinching at a high frequency, which was hypothesised to cause a peristaltic flow. Computational analysis is presented to understand and interpret experimental results, and compelling evidence is shown which advances the understanding of fundamental cell biology.

Each chapter begins with its own detailed introduction to the topic, including background information and a review of relevant literature.

Finally, a concluding Chapter 6 highlights the key discoveries and contributions made as part of this work.

Chapter 2

LAG SIM: A versatile, user-friendly structured illumination microscope

2.1 Introduction

2.1.1 Background to SIM

Widefield epifluorescent microscopes use a uniform field of light to illuminate a labelled sample [14, *ch. 2*]. Fluorescent emission light is then emitted from any fluorescent molecule located within the microscope's field of view [61]. In this type of microscope, fluorescent molecules above and below the focal plane of the lens will also receive illumination light and fluoresce, causing an out-of-focus blur on the image [62].

Out-of-focus light can be removed by placing a pinhole in the excitation and emission path, and scanning the small spot of light across the sample to build up an image one pixel at a time. This technique, invented in 1955 and called confocal microscopy, physically blocks out-of-focus light, making 3D reconstructions of samples possible [28]. The cost of this optical sectioning is a slow acquisition speed due to the nature of a point-scanning technique.

A faster way of removing out-of-focus light can be achieved with total internal reflection fluorescence (TIRF) [63, *ch. 21*]. In this scheme light is projected into the outer edge of the back-aperture of a specialised lens, such that it emerges from the lens steeper than the critical angle required for reflection, rather than refraction, from a glass interface. Although this means that no light passes through the microscope slide, as shown in Figure 2.1a, energy from the evanescent field is able to couple into molecules close to the coverglass, inducing fluorescence [64]. The energy in the evanescent field decays exponentially above the interface, as described by Equation 2.1, where E_0 is the energy of the field at the interface, z is the distance from the interface, and ζ is given by Equation 2.2. In practice, the penetration depth

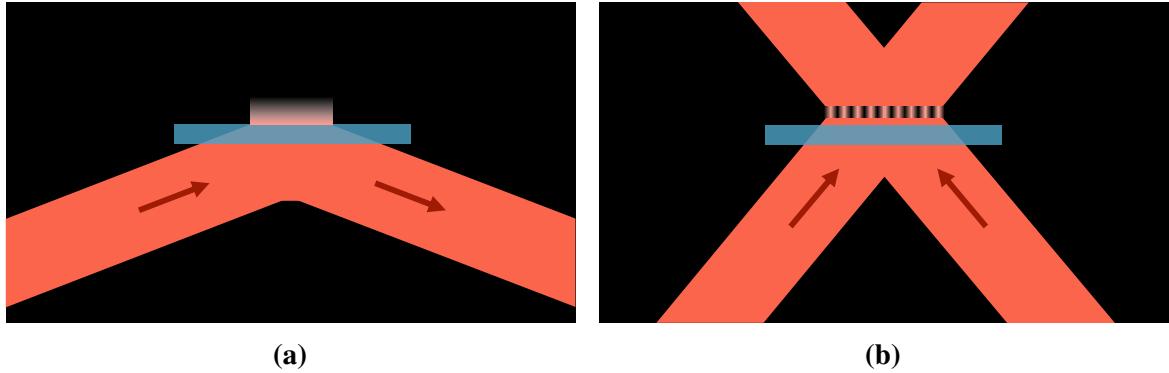


Figure 2.1: (a) shows a TIRF illumination scheme, where light intercepts the cover glass at an angle steeper than the critical angle for total internal reflection. The evanescent wave created decays exponentially in intensity, exciting molecules in the first 30 nm–300 nm as described by Equations 2.1 and 2.2. (b) shows the generation of a structured illumination pattern by the interference of two beams. This can be used to computationally reconstruct an optically sectioned image, as shown in Figure 2.2. Diagrams are not to scale.

of the evanescent field useful for imaging extends 30 nm–300 nm into the sample, depending on the wavelength λ , the illumination angle θ , and the refractive indices of the cover slide and sample, n_1 and n_2 respectively. Fluorophores beyond this region are not excited, allowing molecules at the coverglass surface to be imaged without obscuration by out-of-focus light, with the disadvantage that details deeper into the sample cannot be observed.

$$E = E_0 \exp\left(-\frac{z}{\zeta}\right) \quad (2.1)$$

$$\zeta = \frac{\lambda}{4\pi\sqrt{n_1^2 \sin^2(\theta) - n_2^2}} \quad (2.2)$$

A technique related to TIRF is highly inclined and laminated optical sheet (HILO) microscopy [65]. For HILO, the illumination angle θ is set to slightly below the critical angle, so that the light refracts at a very steep angle from the coverglass into the sample. This generates a thin beam in the sample, so that less fluorescence is induced outside the focal plane of the lens and an image can be captured with optical sectioning. The HILO beam, with a full-width at half-maximum below 7 μm , shows in practice a signal-to-background ratio up to 7.6 times greater than epifluorescent microscopy [65].

Sheppard and Wilson showed that optical sectioning can be achieved computationally if the illumination light is modulated with a structured pattern [61, 66]. Sinusoidal illumination can be produced by two coherent beams of light generated by a binary grating, which interfere at the focal plane of the objective lens as shown in Figure 2.1b. The resulting illumination I

is described by Equation 2.3, where m is the modulation depth, t the modulation frequency, and x a lateral spatial dimension in the sample plane.

$$I_i = (1 + m \cos(tx + \phi_i)) \quad (2.3)$$

The illumination pattern is stepped through three equally-spaced phases $\phi_i = \{0, \frac{2\pi}{3}, \frac{4\pi}{3}\}$, and image data D_i is captured at each step $i = \{1, 2, 3\}$. These images can be described by Equation 2.4, where S is the distribution of underlying sample fluorescence, and H is the microscope's point spread function (PSF).

$$D_i = (S I_i) \otimes H \quad (2.4)$$

If the acquired images are simply summed together, the result $D_1 + D_2 + D_3$ is $S \otimes H$, the image that would be obtained from a widefield microscope. Figure 2.2a shows this sum for an image of the Herpes Simplex Virus Type 1 (HSV-1) infecting a human foreskin fibroblast (HFF) cell, where out-of-focus light obscures details in the focal plane. When the raw images I_i are reconstructed with the pixel-by-pixel square-law computation of Equation 2.5, however, Figure 2.2b shows the effective removal of out-of-focus light.

$$D_R = \left((D_1 - D_2)^2 + (D_2 - D_3)^2 + (D_1 - D_3)^2 \right)^{\frac{1}{2}} \quad (2.5)$$

To understand this mathematically, consider the captured image data to consist of $D_{(if)}$, perfectly in-focus data, and $D_{(of)}$, completely out-of-focus data. Under structured illumination, only the in-focus light is modulated by the sinusoidal illumination scheme, resulting in captured image data described by Equation 2.6. Proof 2.1 shows that unmodulated light is computationally removed from the reconstructed image data D_R , and in-focus light is retained with intensity proportional to the modulation depth m . An identical result is given by Equation 2.7, which uses complex numbers and the modulus operation to obtain the optically-sectioned reconstruction [67].

$$D_i = D_{(of)} + D_{(if)} (1 + m \cos(tx + \phi_i)) \quad (2.6)$$

$$D_R = \left| D_1 + D_2 \exp\left(\frac{2\pi j}{3}\right) + D_3 \exp\left(\frac{4\pi j}{3}\right) \right| \quad (2.7)$$

This provides optical sectioning because the support of the PSF in frequency space drops sharply with defocus in z [67, 68], so that the contrast of the modulation pattern quickly decreases with defocus and the weakly modulated light is suppressed by Equation 2.5 or 2.7 [69]. The optimal spatial frequency of the structured illumination is found to be $\frac{N_A}{\lambda}$,

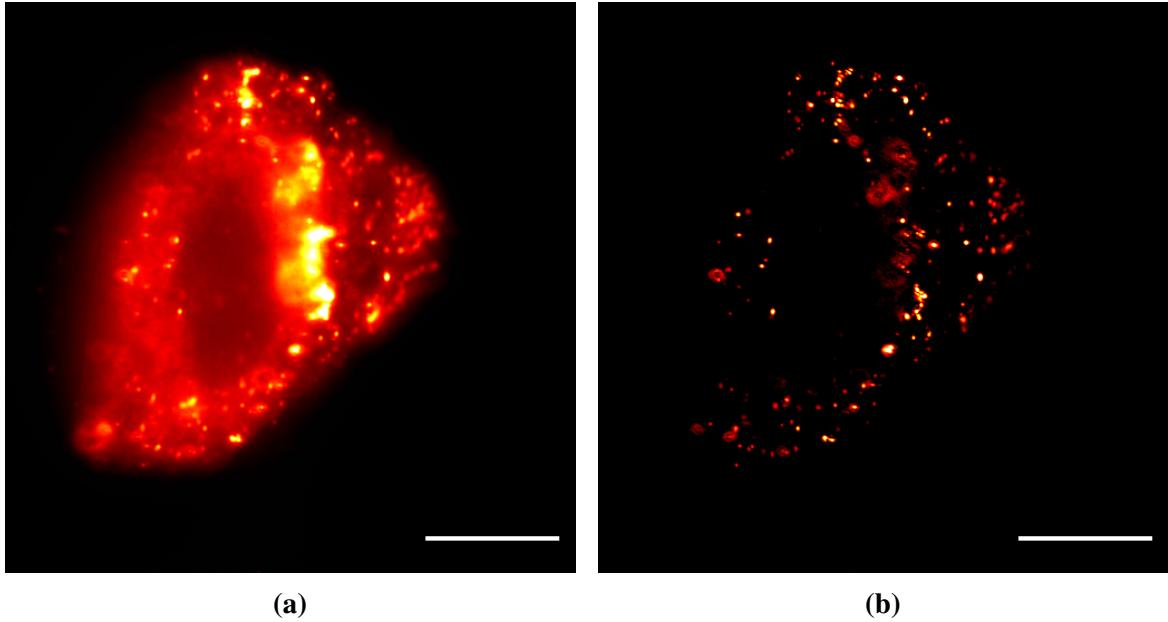


Figure 2.2: (a) shows a widefield image, $S \otimes H$, of HSV-1 infecting a human foreskin fibroblast (HFF) cell. (b) shows the same image after illumination with 3 phase-stepped sinusoidal patterns, which are only generated at the focal plane of the objective lens. Combining the images with square-law detection shown in Equation 2.5 leaves just the $mD_{(if)}$ term derived in Proof 2.1, effectively removing the out-of-focus light. Human foreskin fibroblast (HFF) cells were prepared and infected by Katharina Scherer. Scalebar is 10 μm .

where N_A is the numerical aperture of the lens and λ is the wavelength of fluorescent emission light [67, 70]. Furthermore, the non-linearity caused by squaring removes negative amplitudes in frequency space, causing discontinuities which introduce higher-order harmonics into the optical transfer function (OTF). These harmonics artificially enhance contrast of the underlying fluorescent structure, particularly when spatial frequencies in the sample are close to the frequency of the optimal illumination pattern [70].

The result of Proof 2.1 shows that the signal in the reconstructed image data is directly proportional to the modulation depth m of the sinusoidal illumination pattern. Since the noise power is constant with respect to modulation depth, the signal-to-noise ratio of the reconstructed image is also directly proportional to the modulation depth. It is therefore important for structured illumination microscopy to maximise the modulation depth to reconstruct an optically-sectioned image with as high a signal-to-noise ratio as possible.

$$\begin{aligned}
D_1 &= D_{(of)} + D_{(if)}(1 + m \cos(tx)), \quad D_2 = D_{(of)} + D_{(if)}(1 + m \cos(tx + \frac{2\pi}{3})), \\
D_3 &= D_{(of)} + D_{(if)}(1 + m \cos(tx - \frac{2\pi}{3})), \\
D_r^2 &= (D_1 - D_2)^2 + (D_1 - D_3)^2 + (D_2 - D_3)^2
\end{aligned}$$

Substitute D_1 , D_2 , and D_3 into the expression for D_r^2 :

$$\begin{aligned}
&= m^2 D_{(if)}^2 \left(\left(\cos(tx) - \cos(tx + \frac{2\pi}{3}) \right)^2 + \left(\cos(tx) - \cos(tx - \frac{2\pi}{3}) \right)^2 \right. \\
&\quad \left. + \left(\cos(tx + \frac{2\pi}{3}) - \cos(tx - \frac{2\pi}{3}) \right)^2 \right)
\end{aligned}$$

Expand squared brackets:

$$\begin{aligned}
&= m^2 D_{(if)}^2 \left(2 \cos^2(tx) + 2 \cos^2(tx + \frac{2\pi}{3}) + 2 \cos^2(tx - \frac{2\pi}{3}) \right. \\
&\quad \left. - 2 \cos(tx) \cos(tx + \frac{2\pi}{3}) - 2 \cos(tx) \cos(tx - \frac{2\pi}{3}) - 2 \cos(tx + \frac{2\pi}{3}) \cos(tx - \frac{2\pi}{3}) \right)
\end{aligned}$$

Apply double angle formulae to \cos^2 and $\cos(A) \cos(B)$ terms:

$$\begin{aligned}
&= m^2 D_{(if)}^2 \left(3 + \cos(2tx) + \cos(2tx + \frac{4\pi}{3}) + \cos(2tx - \frac{4\pi}{3}) - \cos(2tx + \frac{2\pi}{3}) \right. \\
&\quad \left. - \cos(-\frac{2\pi}{3}) - \cos(2tx - \frac{2\pi}{3}) - \cos(\frac{2\pi}{3}) - \cos(2tx) - \cos(0) \right)
\end{aligned}$$

Note that $\cos(x + \phi) = \cos(x + 2\pi n\phi)$, $n \in \mathbb{Z}$:

$$= m^2 D_{(if)}^2 (3 + 0.5 + 0.5 - 1)$$

$$= 3m^2 D_{(if)}^2$$

Take square root:

$$D_R = \sqrt{3}mD_{(if)}$$

Proof 2.1: Square law detection removes unmodulated out-of-focus light from 3 SIM images.

2.1.2 SIM for resolution doubling

The widefield image data D produced by a fluorescent microscope is made up of the underlying sample fluorescence, S , convolved with the microscope point spread function (PSF), H . When light from the sample plane passes through a lens, the finite aperture means that a small point of light is spread into a larger spot in the image plane [71, ch. 11]. This means that two independent points of light which are too close together cannot be resolved; that is, the lens acts as a low pass filter for spatial frequencies. This can be seen in the diffraction-limited image of 100 nm beads in Figure 2.3a, where the beads' fluorescence emission light spreads into a spot with a full-width half-maximum (FWHM) of 200 nm. The corresponding Fourier-space image in Figure 2.3b shows that spatial frequencies above the diffraction limit are not transmitted through the lens.

The same structured illumination microscopy (SIM) used for optical sectioning can be re-purposed for surpassing the diffraction limit. First discussed by Lucosz [72], and then more explicitly described by Heintzmann for sinusoidal illumination patterns [73], the first experimental result showing 2D isotropic resolution doubling with 9 raw image acquisitions was published by Gustafsson in the year 2000 [22]. Heintzmann shows that if we take the Fourier transform of the acquired images shown in Equation 2.3, we obtain the set of images shown in Equation 2.8; where $\hat{\cdot}$ -symbols represent 2D Fourier transforms of their equivalent variables, and k_x is the Fourier variable of x , that is $f(x) \Leftrightarrow \hat{f}(k_x)$. Note that W has been replaced with its underlying components $H \otimes S \Leftrightarrow \hat{H}\hat{S}$, where \hat{H} , the Fourier transform of the PSF, is known as the optical transfer function (OTF).

$$\hat{D}_i = \hat{H}\hat{S} \otimes \left(\delta(k_x) + \frac{m}{2} e^{j\phi_i} \delta(k_x + t) + \frac{m}{2} e^{-j\phi_i} \delta(k_x - t) \right) \quad (2.8)$$

Each acquired raw image contains contributions from each of 3 shifted Fourier components. Using $\phi_i = \{0, \frac{2\pi}{3}, \frac{4\pi}{3}\}$, these Fourier components can be separated into individual images by solving the set of simultaneous equations for $\hat{S} \otimes \delta(k_x)$, $\hat{S} \otimes \delta(k_x + t)$, and $\hat{S} \otimes \delta(k_x - t)$ with matrix inversion, as shown in Figure 2.3e and Equation 2.9 [74]. Assuming phase steps are all equal, performing a 3D Fourier transform of the phase-stepped raw images stacked in the 3rd dimension produces the same result [75].

$$\begin{bmatrix} \hat{H}\hat{S}(k_x) \\ \hat{H}\hat{S}(k_x + t) \\ \hat{H}\hat{S}(k_x - t) \end{bmatrix} = \begin{bmatrix} 1 & \frac{m}{2} \exp(0j) & \frac{m}{2} \exp(-0j) \\ 1 & \frac{m}{2} \exp\left(\frac{2\pi j}{3}\right) & \frac{m}{2} \exp\left(-\frac{2\pi j}{3}\right) \\ 1 & \frac{m}{2} \exp\left(\frac{4\pi j}{3}\right) & \frac{m}{2} \exp\left(-\frac{4\pi j}{3}\right) \end{bmatrix}^{-1} \begin{bmatrix} \hat{D}_1 \\ \hat{D}_2 \\ \hat{D}_3 \end{bmatrix} \quad (2.9)$$

When the matrix of Equation 2.9 is inverted, a factor of m can be extracted as a scaling factor in front of the high frequency Fourier components, shown in Equation 2.10. In the

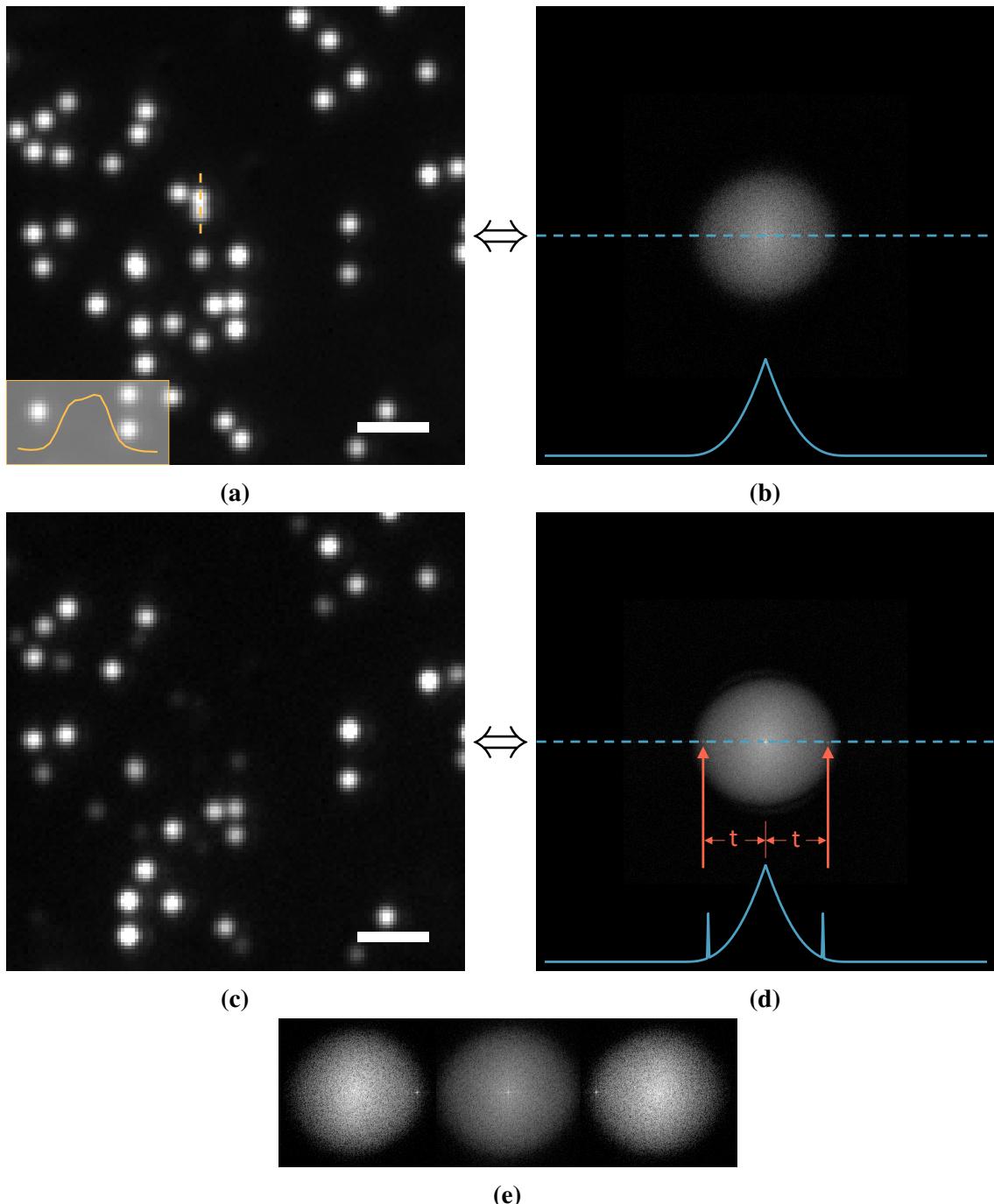


Figure 2.3: (a) shows a diffraction-limited image of 100 nm beads. (b) shows that spatial frequencies above the diffraction limit of the lens are not transmitted, causing a point-spread function which increases the apparent size of the beads and preventing two beads which are close together from being resolved, shown by the line profile inset in (a). (c) shows the same field of view under structured illumination; note that some beads are less intense than in (a) if they fall within a trough of the sinusoidal illumination pattern. (d) clearly shows delta peaks appearing in the Fourier transform of (c) due to the sinusoidal illumination pattern. (e) shows the Fourier components $\hat{H}\hat{S}(k_x)$, $\hat{H}\hat{S}(k_x + t)$, and $\hat{H}\hat{S}(k_x - t)$ extracted from the raw images by Equation 2.9. Scalebars are 1 μm .

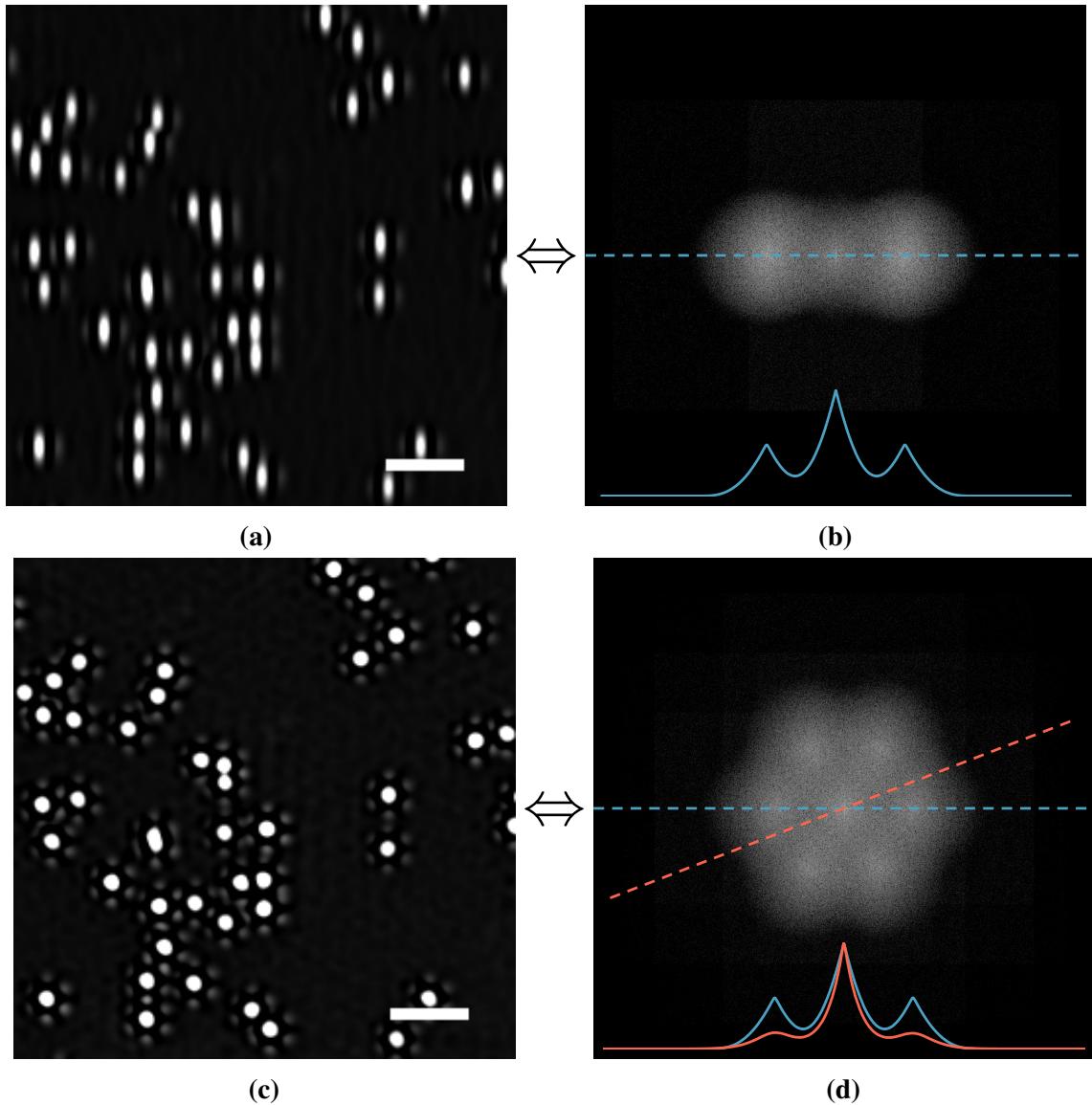


Figure 2.4: (a) shows beads after SIM reconstruction in the x -direction only, where the increase in x -resolution makes the circular beads appear as elongated ellipses. (b) shows the corresponding Fourier-space image, where the width of the support confirms that resolution has doubled in the x -direction compared to widefield. (c) shows an isotropic increase in resolution, achieved by performing the procedure used to generate (a) at two more angles, 60° and 120° to the x -axis. (d) shows the Fourier transform of (c), confirming isotropic resolution increase. Notably, this reconstruction scheme produces peaks of signal in Fourier space and the OTF is not rotationally symmetric, causing the characteristic hexagonal artefacts visible in (c). Scalebars are $1\text{ }\mu\text{m}$.

same way that a smaller modulation depth m reduces the signal-to-noise ratio in optical sectioning SIM, described in Section 2.1.1, this factor shows that the signal-to-noise ratio of the high frequency components is directly proportional to the modulation depth of the illumination pattern.

$$\begin{bmatrix} \hat{H}\hat{S}(k_x) \\ m\hat{H}\hat{S}(k_x+t) \\ m\hat{H}\hat{S}(k_x-t) \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 2 & (-1 - \sqrt{3}j) & (-1 + \sqrt{3}j) \\ 2 & (-1 + \sqrt{3}j) & (-1 - \sqrt{3}j) \end{bmatrix} \begin{bmatrix} \hat{D}_1 \\ \hat{D}_2 \\ \hat{D}_3 \end{bmatrix} \quad (2.10)$$

To complete the reconstruction, the separated Fourier components must be placed at the correct position in Fourier space, as determined by t . Assuming the sinusoidal illumination pattern is generated by the same lens used for imaging the sample, the highest frequency sinusoid which can be generated will correspond to delta peaks at the edge of the microscope's support in Fourier space [76], visible in Figure 2.3d. When the components are shifted into the appropriate location, the support of the OTF doubles in the x -direction, as shown in Figure 2.4b. Figure 2.4a confirms that resolution is doubled in the x -direction, which makes the circular beads appear as elongated ellipses.

To achieve isotropic 2D resolution doubling, the sinusoidal illumination pattern must be rotated to cover more area in Fourier space. Typically a total of three rotations are used, at 60° and 120° to the original pattern orientation [22, 77]. Performing the reconstruction procedure on all 9 images produces the Fourier space reconstruction shown in Figure 2.4d, showing an isotropic increase in the resolution support.

The final step for this reconstruction procedure is to inverse Fourier transform the reconstructed Fourier space image of Figure 2.4d to produce Figure 2.4c, an image with double the equivalent widefield resolution. Comparing Figure 2.4c to Figure 2.3a, the beads appear smaller and two close beads which could not be resolved before can now be distinguished. However, the reconstruction procedure must be refined to reduce the hexagonal artefacts introduced by this reconstruction method.

2.1.3 Refining the reconstruction algorithm reduces artefacts

The 2D OTF of a focussed lens is a rotationally symmetric function which gradually reduces to zero as spatial frequency increases [78]. The line profile through the OTF of an ideal lens in Figure 2.3b shows that the higher the spatial frequency, the less contrast the microscope produces. In a practical set-up, noise further limits the resolvable contrast at high frequencies, such that signal-to-noise ratio decreases as spatial frequency increases.

The signal-to-noise ratio's relationship with spatial frequency has a particularly dramatic effect in SIM. Equation 2.9 shows that each shifted frequency component, \hat{S}_i , is apodised

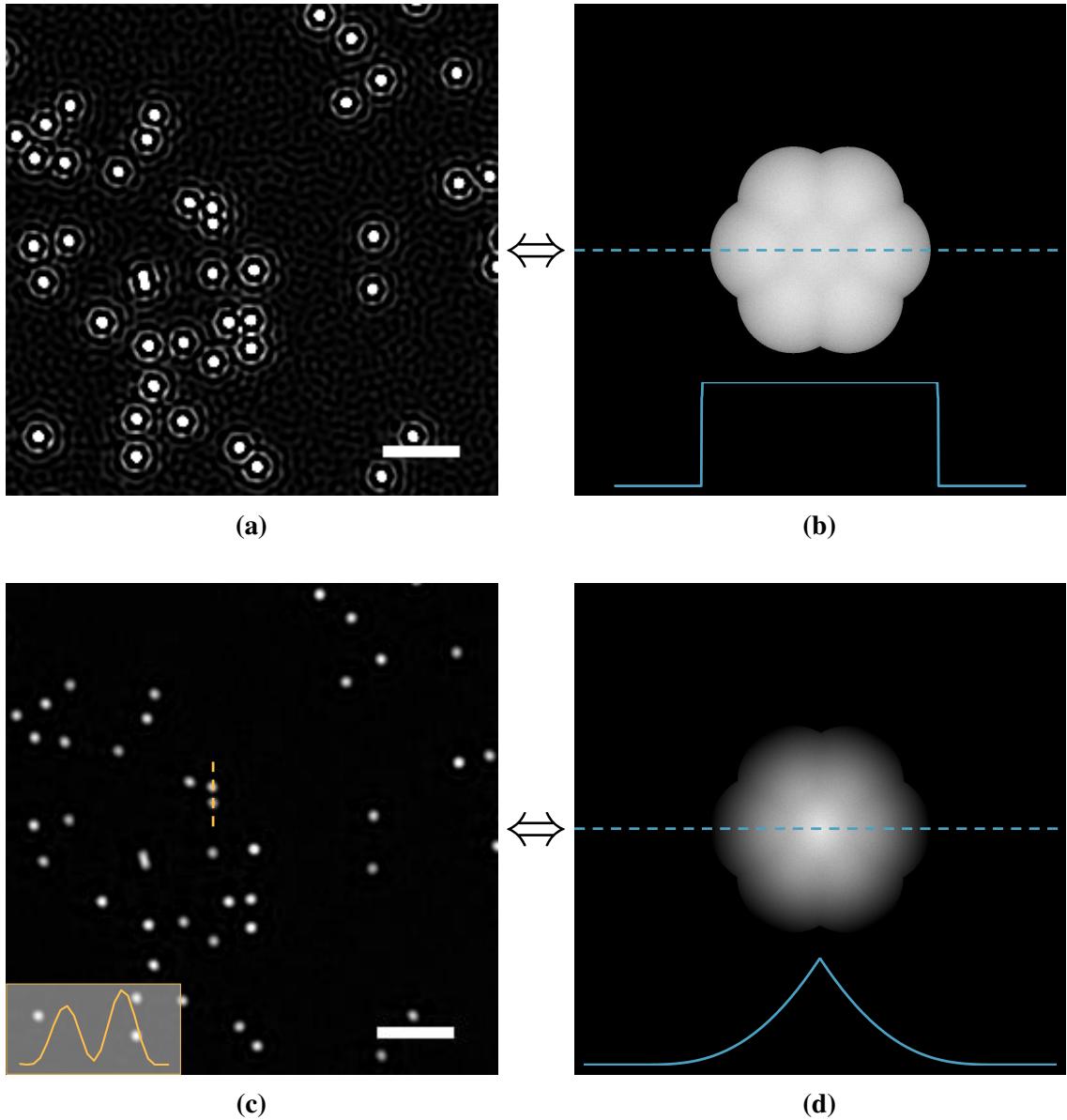


Figure 2.5: (a) shows a reconstructed SIM image after Wiener filtering described in Equation 2.11, with $A(k) = 1$. (b) shows that this filtering produces a flat OTF in Fourier space, which has more rotational symmetry than Figure 2.4d due to the removal of peaks. However, the sharp cutoff of the Fourier space support causes ringing in the reconstructed image, clearly visible around the beads in (a); although compared to Figure 2.4c the artefacts are now rotationally symmetric, rather than honeycombed. To reduce ringing artefacts, and produce the high-resolution bead image shown in (c), an apodisation filter which gradually decreases is applied to (b), as shown in (d). Comparing (d) to Figure 2.3b shows an isotropic doubling in the OTF's support; consequently, comparing the beads in (c) to Figure 2.3a shows a smaller point spread function, allowing two beads to be distinguished which were previously too close to resolve. Scalebars are $1 \mu\text{m}$.

by the microscope OTF, \hat{H} . It can be clearly seen in Figure 2.4b that this results in an overall OTF which does not gradually decrease, but rather has peaks at certain frequencies. These peaks translate directly to frequency peaks in the signal-to-noise ratio, therefore the noise pattern in the reconstructed image is no longer white noise with equal power at all frequencies. Furthermore, it can be seen in Figure 2.4d that the reconstructed SIM OTF is not rotationally symmetric. These two aspects combine to cause characteristic hexagonal honeycomb artefacts in reconstructed SIM images [74], clearly visible in Figure 2.4c.

To reduce these artefacts, Fourier components are reconstructed through a de-noising algorithm. The most common practice is to combine the frequency components through a generalised Wiener filter [79], as shown in Equation 2.11, to produce the reconstructed image \hat{D}_R . Note that this equation now describes the general 2D or 3D case, where \mathbf{k} is a vector of Fourier variables and \mathbf{p} is a vector in the direction of the sinusoidal illumination pattern for each direction d . w^2 is a constant, adjusted empirically depending on the level of noise in the image, and $A(\mathbf{k})$ is an apodisation function.

$$\hat{D}_R = \frac{\sum_{d,i} \hat{H}^*(\mathbf{k} + \phi_i \mathbf{p}_d) \hat{S}_{d,i}(\mathbf{k} + \phi_i \mathbf{p}_d)}{\sum_{d,i} |\hat{H}(\mathbf{k} + \phi_i \mathbf{p}_d)|^2 + w^2} A(\mathbf{k}) \quad (2.11)$$

If $A(\mathbf{k}) = 1$, this reconstruction scheme removes the decaying characteristic of the OTF, instead making a flat top-hat function shown in Figure 2.5b which suddenly cuts off at the doubled resolution limit. However, because the Fourier transform of a top-hat function is a sinc function, this causes ringing artefacts in the reconstructed image [80, ch. 10]. Ringing is clearly visible in Figure 2.5a, although compared to Figure 2.4c the artefact is now more rotationally symmetric rather than honeycombed. To remove ringing, the apodisation function $A(\mathbf{k})$ is chosen as a filter which decays from the 0 frequency to the new resolution limit, for example a Gaussian filter shown in Figure 2.5d [81].

More recent research suggests improvements to generalised Wiener filtering [82–84], and a detailed discussion of alternative filtering schemes follows in Section 2.4 of this chapter.

2.1.4 Aims for LAG SIM

When I arrived in the Laser Analytics Group (LAG) a SIM microscope designed by Laurie Young was reaching the end of its practical life [23]. Issues with device degradation, detailed in Section 2.2.2, limited the imaging speed to 0.1 Hz. Furthermore, a complicated ensemble of software required an expert user to operate the microscope.

A physical relocation of the laboratory in early 2017 [85] provided the ideal opportunity to rebuild the microscope and the associated control software. The first priority was to restore

the microscope to 11 Hz imaging speed. Once this was achieved, a list of several other aims were devised:

1. Provide an easy method to switch between optical sectioning SIM and resolution-doubling SIM in TIRF (Sections 2.2 and 2.3).
2. Split the fluorescence emission light at the output of the microscope to facilitate simultaneous capture of multiple colour channels (Section 2.2.1).
3. Re-write the control software to make the microscope user-friendly for a non-expert user to operate unsupervised (Section 2.3).
4. Design reconstruction software to allow users to quickly reconstruct artefact-free images without detailed knowledge of reconstruction algorithms (Sections 2.4 and 2.5).

As well as detailing solutions to these challenges, this chapter also presents a showcase of various biological experiments performed with LAG SIM. Section 2.6 can therefore be used a collection of case-study examples for anyone wishing to run similar investigations.

2.2 SIM hardware

2.2.1 Optical path design to generate a SIM pattern

As part of his PhD work from 2012-2016, Dr. Laurie Young designed and built a SIM microscope in the Laser Analytics Group [23]. The rebuilt SIM setup, ‘LAG SIM,’ is based heavily on this original design, but with several modifications which were implemented by me when the Group moved location in February 2017. The microscope is built around a commercial Olympus frame, with lasers from Cobolt, Coherent, and Topica Photonics; a Pockels cell from Conoptics; a spatial light modulator from Forth Dimension Displays; a stage from ASI; a filter wheel from Prior; a spectral beam splitter from Cairn Research; and a camera from Hamamatsu. These components were combined in a custom optical path for SIM imaging.

The optical path diagram is shown in Figure 2.6, and works as follows: laser light is generated by one of three lasers at a wavelength of 488 nm, 561 nm, or 640 nm; its polarisation is aligned with the direction of the structured illumination pattern by a Pockels cell and achromatic quarter-wave plate; it passes through a beam expander so that it fully covers the spatial light modulator (SLM) display; it reflects and diffracts off the SLM display, which is displaying a binary diffraction grating; it passes through a ‘beam minimiser’ (a beam expander in reverse) so that it is an appropriate size for the microscope lenses; it passes through a spatial filter at a Fourier plane, which removes all but the ± 1 diffraction orders from

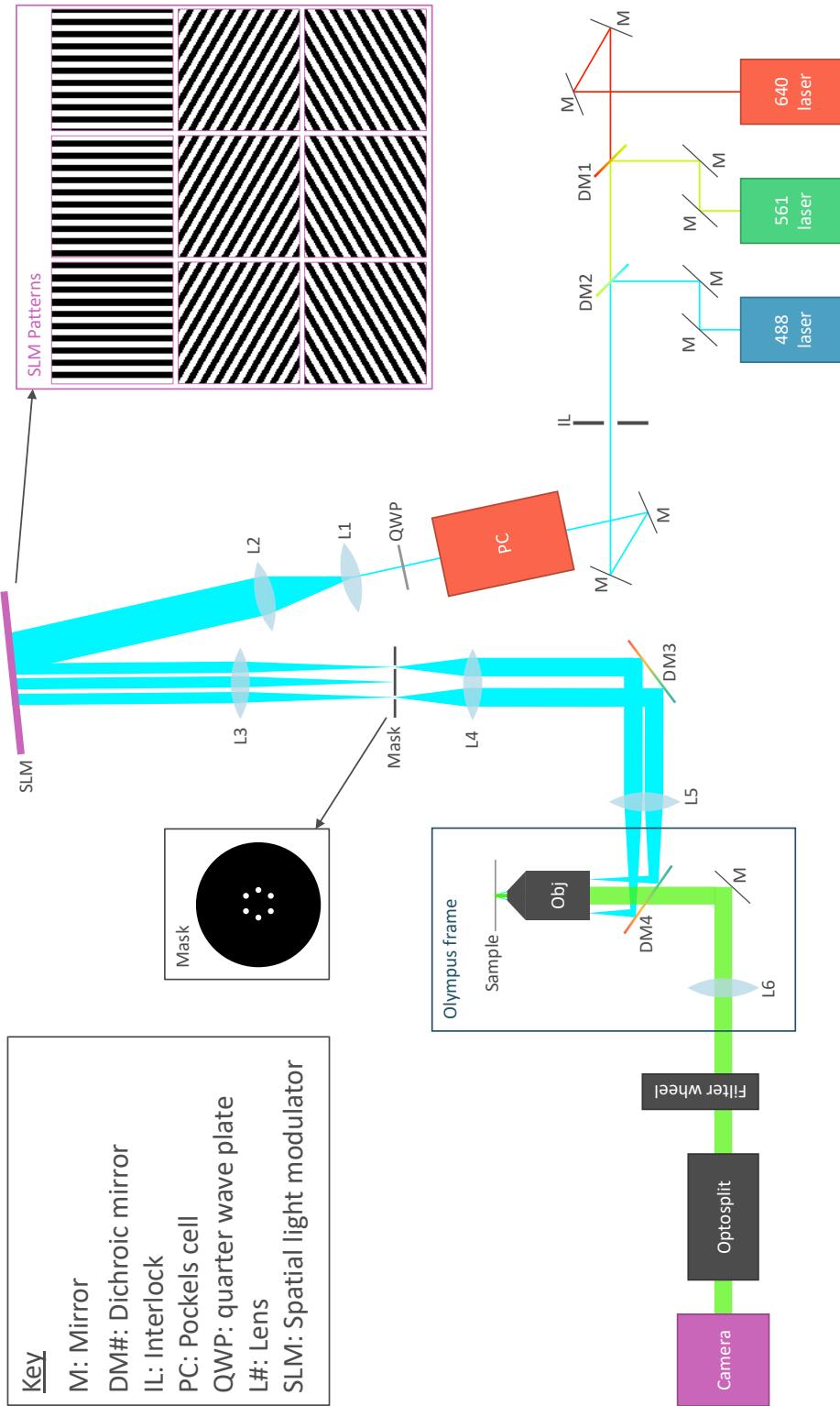


Figure 2.6: The optical layout of the SIM aligns light from one of 3 lasers onto an optical path. The light passes through a Pockels cell and quarter wave plate for polarisation rotation, before a square-wave pattern is applied by a spatial light modulator. The patterned light is passed through a spatial mask in the Fourier plane to produce a sinusoidal pattern, which is relayed onto the sample. The objective lens collects fluorescent emission light, which is filtered to remove any reflected excitation light before reaching the sCMOS camera.

the SLM diffraction; it reflects off of a dichroic mirror; it passes into the microscope's tube lens; it reflects off of a second dichroic mirror; and finally it passes into the objective lens and is focussed onto the sample. DM3 and DM4 are sourced from the same manufacturing batch, so that any aberrations introduced by non-uniformity of DM3 are undone upon reflection from DM4.

This effectively images a low-pass-filtered version of the SLM's diffraction grating onto the sample. The spatial filter means that the binary grating becomes a sinusoidal pattern, with the distance between peaks set by (a) the period of the diffraction grating shown on the SLM display and (b) the demagnification of the system from SLM to sample. An equivalent way of conceptualising this is to picture the two filtered beams of light entering the rear focal plane of the objective lens, such that they produce an interference pattern when they meet at the sample plane [71, ch. 9]. The result is a sinusoidal illumination pattern at the sample plane, the orientation and period of which can be set by binary patterns shown on the SLM.

This SIM is a fluorescence microscope, so that, assuming linear fluorophore behaviour [75, 86], the amount of light emitted from a point in the sample is proportional to the amount of light illuminating that point, but the emission light is at a different wavelength to the illumination light. In the LAG SIM microscope, fluorescent emission light travels back through the objective lens, through the dichroic mirror and towards the camera to record a 2D image. Before it reaches the camera, it is filtered in one of two ways to remove any reflected illumination light. The first option is to use a filter wheel, which contains three filters appropriate for removing the illumination light and can be switched via a serial command from the computer. The second option is to use the Optosplit III from Cairn Research, which uses filter cubes to split the emission light into three paths directed onto separate areas of the camera, facilitating simultaneous imaging of three colour channels.

The Optosplit provides a significant increase in speed for multicolour imaging. In this setup, the fastest speed at which the camera can reliably record images without dropping frames is 10 ms. A SIM reconstruction requires 9 images, giving a total exposure time of 90 ms per channel; however switching channels with the filter wheel adds an additional 100 ms as the wheel rotates to the next filter and settles. For a 3-colour image using the filter wheel, the fastest SIM imaging rate is therefore 570 ms per frame, or 1.75 Hz. Using the Optosplit is over 6 times faster, providing a maximum 11.1 Hz imaging rate even for multicolour images.

Finally, an ASI autofocus system between the microscope frame and filter wheel works with the z-control of the microscope stage to maintain a constant focus on the sample. This is useful in two scenarios. Firstly, when a timelapse sequence is captured over a number of hours, the microscope lens drifts downwards away from the sample. Without the autofocus

system, this would result in a loss of focus; however by automatically moving the stage as the lens drifts, focus can be maintained over a period of days. Secondly, if a large field of view is captured to create a mosaic of images, focus can be lost due to the cover glass not being perfectly flat. Again, the autofocus system holds focus over this full field of view, allowing large areas to be imaged.

2.2.2 A Pockels cell provides fast polarisation rotation

In the SIM excitation path, a Pockels cell is used to rotate the polarisation of the illumination light so that it is aligned with the sinusoidal SIM pattern.

In SIM, it is important that the modulation contrast is as large as possible, that is, that the troughs of the sinusoidal illumination pattern are as dark as possible. When the 9 raw images are computationally recombined, the signal-to-noise ratio of high-spatial-frequency components is directly proportional to the modulation contrast [87], as described in Section 2.1.2.

Modulation contrast of fluorescence emission light can be degraded by a number of sources, including out-of-focus light and aberrations in the imaging system, which is unavoidable in thick samples. Furthermore, if the polarisation of the illumination light is not rotated to align with the rotations of the SIM pattern, modulation contrast is degraded due to the effects of a high numerical aperture lens.

Two waves with orthogonal polarisation will not interfere [88]. As light passes through a high numerical aperture lens, the polarisation state which is parallel to the meridional plane (p-polarised) is rotated; whereas the polarisation state perpendicular (s-polarised) is unaffected [89]. To ensure that light from the two beams entering the back-aperture of the lens arrive at sample plane with the same polarisation, it is therefore necessary that the light is entirely s-polarised [87].

If the meridional plane is defined at an angle of $\theta = 0^\circ$ around the optical axis, aligned with the back-aperture spots when the SLM is showing a vertical SIM grating pattern, the incident light should be linearly polarised at $\theta = 90^\circ$ to ensure s-polarised light, as shown in Figure 2.7. However, for 2D isotropic resolution enhancement, the SIM pattern must also be rotated to $\pm 60^\circ$. For these patterns the $\theta = 90^\circ$ polarisation of the light does not represent s-polarised light. In order to maintain s-polarised light, thereby ensuring maximum pattern contrast, the polarisation must be rotated to align with each orientation of the SIM pattern.

The original SIM system, detailed in reference [23], used a liquid crystal variable retarder (LCVR) to rotate the polarisation to align with the SLM's grating orientation. This suffered an unexpected problem where the laser beam damaged the liquid crystal material leaving a visible spot on the LCVR and no longer retarded the light, so the system no longer rotated

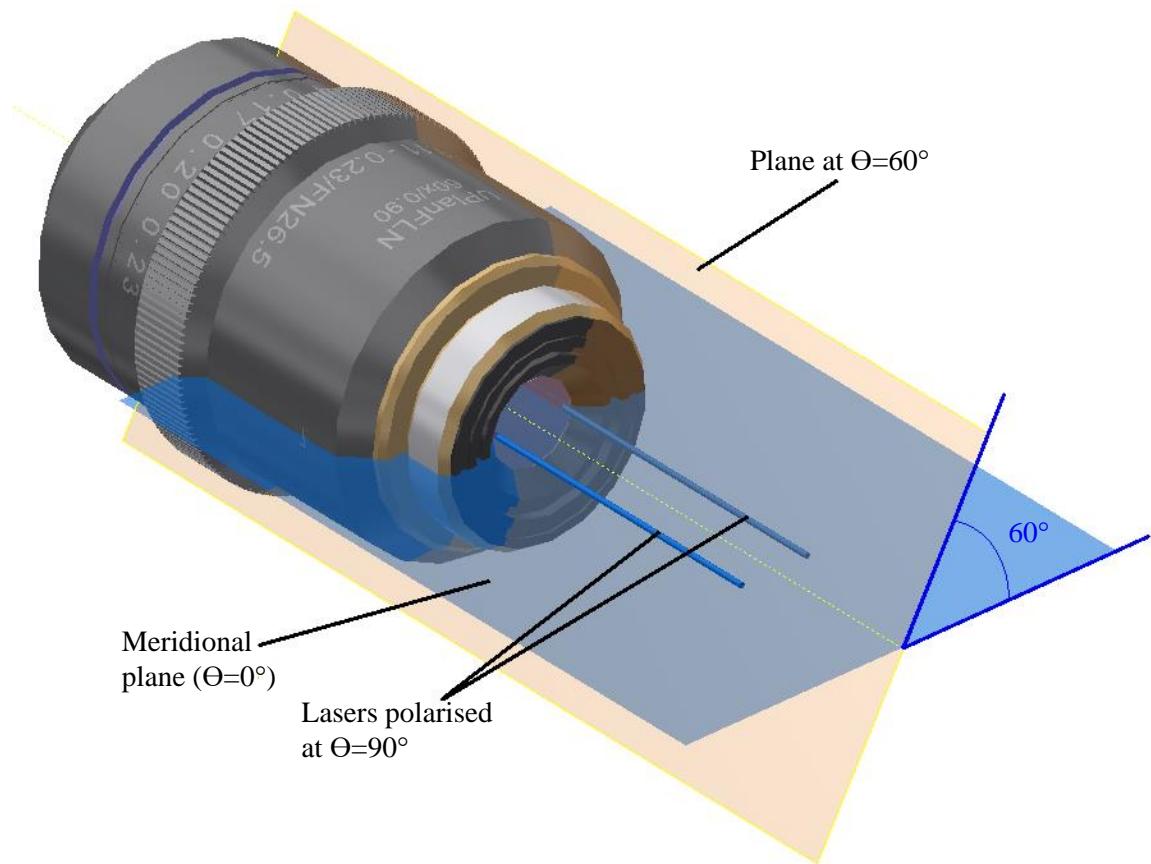


Figure 2.7: To ensure maximum pattern contrast, the excitation laser light must be s-polarised - that is, polarised at 90° to the meridional plane for the vertical sinusoidal illumination pattern. When the illumination pattern is rotated to $\pm 60^\circ$, $\theta = 90^\circ$ polarisation no longer produces s-polarised light. If the polarisation is not rotated to align with the rotated illumination pattern, some p-polarised light will enter the objective, which is rotated as it passes through the high NA lens. The p-polarised light will therefore not interfere at the sample plane, reducing the contrast of the illumination pattern.

polarisation. The product line has since been discontinued. As an interim solution, an achromatic half-wave plate was mounted in a serial-controlled rotation stage in place of the LCVR [71, ch. 8]. This setup successfully rotated the polarisation and ensured good modulation contrast, but was only able to image at 0.1 Hz.

For fast and reliable polarisation rotation, facilitating 11 Hz SIM imaging, we purchased a Pockels cell from Conoptics. Similarly to the LCVR, a Pockels cell provides voltage-controlled retardation of one polarisation axis which, in combination with a quarter-wave plate, creates rotation of linear polarisation [70]. The level of retardation is controlled by a voltage across the Pockels cell, restoring polarisation rotation with no mechanical movement.

2.2.3 Pockels cell alignment procedure

To use the Pockels cell as a polarisation rotator, light entering the cell must be linearly polarised and aligned at 45° to the fast axis of the device. The Pockels cell from Conoptics Inc. was delivered with a Glan-Taylor polariser appropriately aligned in the factory.

To align the Pockels cell to the rest of the system, the following procedure was devised, with the voltage across the Pockels cell set to 0 V:

1. Place the Pockels cell in 5-axis mount
2. Coarsely align so that the beam passes through the centre of the Pockels cell
3. Using a power meter after the Pockels cell, rotate the cell until power is maximised
4. Adjust the tip-tilt of the Pockels cell, maximising power again
5. Insert a linear polariser between the Pockels cell and power meter, and rotate the polariser until power is maximised
6. Insert the quarter-wave plate (QWP) into the beam path mounted in a rotation mount between the Pockels cell and linear polariser
7. With the power meter after the QWP, rotate the QWP until power is maximised, as shown in Figure 2.8
8. Remove the linear polariser

Although this alignment procedure could be completed with any laser, for LAG SIM it is best to use the 561 nm laser, as this is the closest to the centre wavelength of the achromatic quarter-wave plate.

The polarisation state of the light was measured to ensure the Pockels cell was behaving as expected by using a linear polariser in a rotation mount as an analyser before the power meter, as shown in Figure 2.8. The laser emission entering the Pockels cell was linearly polarised, with an extinction ratio of 100:1 for the 561 nm laser. The Glan-Taylor polariser theoretically increases the extinction ratio to >100000:1 [90], although because it is attached to the Pockels cell in the factory, as seen in Figure 2.8, this extinction ratio cannot be

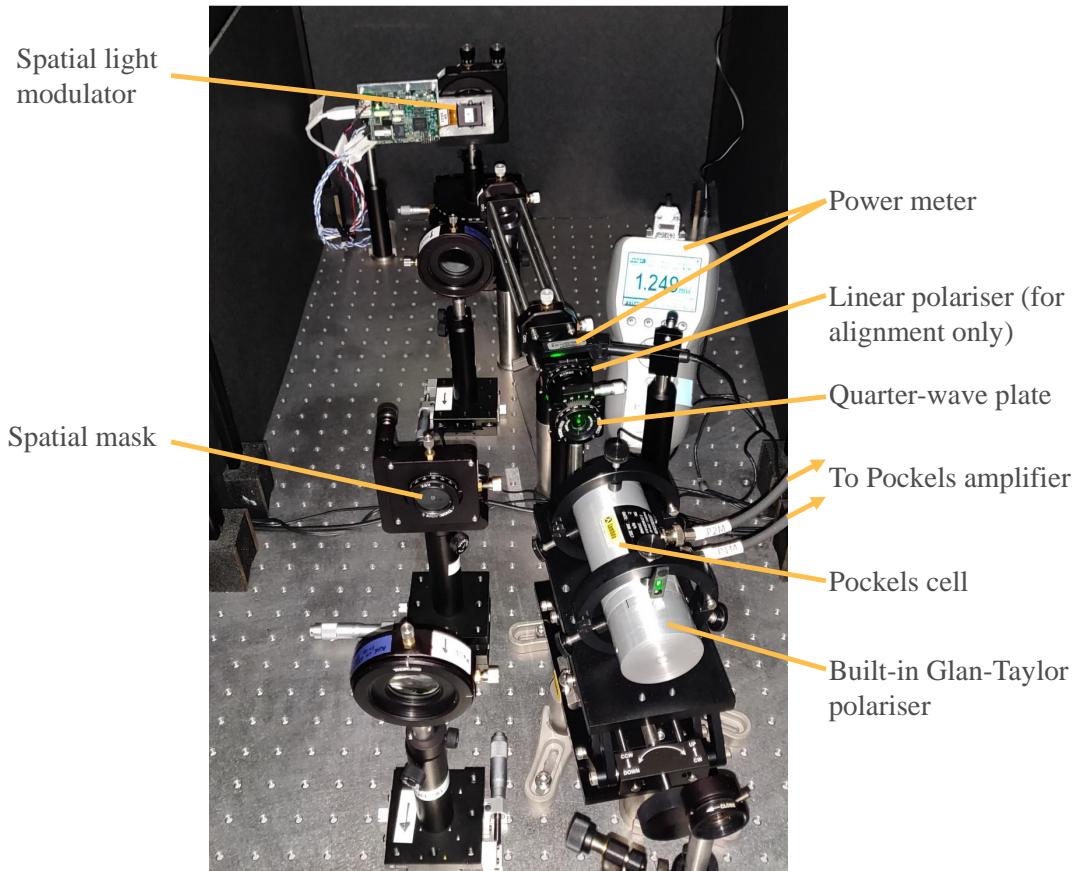


Figure 2.8: The photo shows the LAG SIM beam path during alignment of the Pockels cell. A power meter is used in combination with a linear polariser to measure the polarisation rotation of the beam after retardation of one polarisation axis by the Pockels cell and quarter wave plate. Alignment is performed as per the protocol described in Section 2.2.3. Also visible are the spatial light modulator and spatial mask, which are used in combination to generate the sinusoidal illumination pattern.

measured. Light exiting the Pockels cell is elliptically polarised, with the degree of ellipticity determined by the voltage. Finally, the QWP restores the elliptically polarised light to linear polarisation, but rotated compared to the input as determined by the voltage over the Pockels cell. The extinction ratio at the output of the QWP was 500:1.

2.2.4 Measuring Pockels cell voltages for optimal pattern contrast

The voltage across the Pockels cell determines the amount of optical anisotropy in the cell [71, ch. 8], which, in combination with a quarter wave plate, is directly proportional to the linear polarisation rotation. The voltages required to control anisotropy are of the order of 100 V. To achieve these voltages, an amplifier supplied by Conoptics takes a -1 V to 1 V input and provides 375 V/V gain. The amplifier input voltage is generated by a National Instruments digital to analog converter (DAC), which is controlled by a LabVIEW program.

Figure 2.9 shows the relationship between voltage and polarisation rotation for the three laser lines used on the LAG SIM. Two useful properties of the Pockels cell are evident. Firstly, the Pockels cell is able to provide a full 180° polarisation rotation for all wavelengths, more

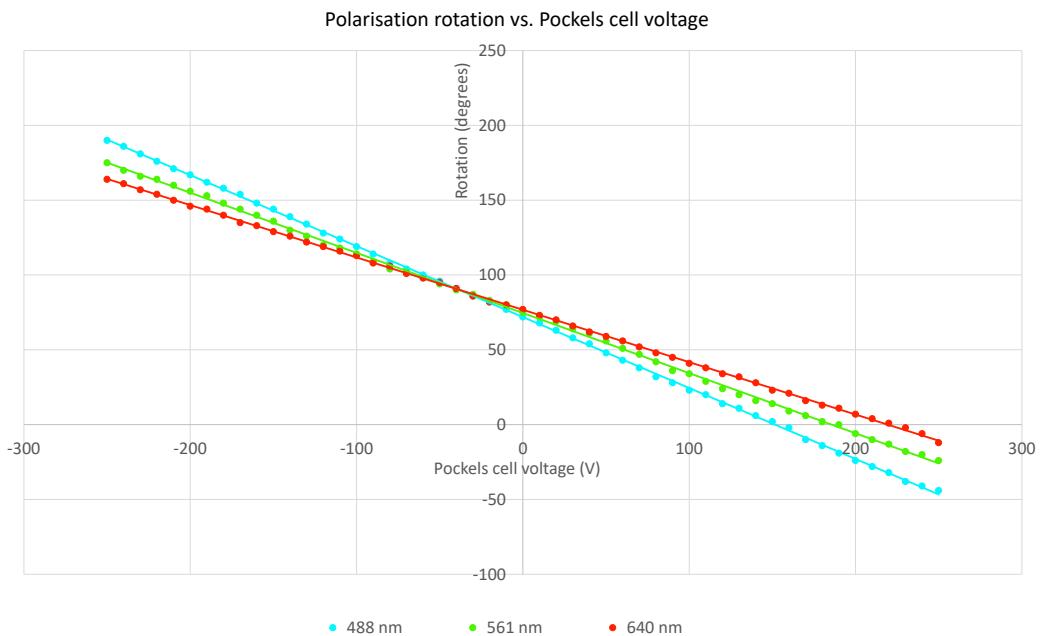


Figure 2.9: When used in combination with a quarter wave plate, the voltage across the Pockels cell is proportional to polarisation rotation for each wavelength of laser light. Though the relationship is slightly different for each wavelength, at low voltages the values are similar enough that the 561 nm voltages can be used when the Optosplit is employed to image multiple channels simultaneously.

than adequate for the $\pm 60^\circ$ required to align polarisation with the SIM pattern orientations. Secondly, to achieve a given rotation a similar voltage is required for all wavelengths.

To measure pattern contrast at a given voltage, images of sub-diffraction beads were recorded under sinusoidal SIM illumination. Beads located in a trough of the pattern were dark; conversely beads lying in a peak were bright. At each voltage step the phase of the illumination pattern was swept so that, at some point during the phase sweep, each bead passed through a peak and a trough. Calculating the difference between the maximum and minimum brightness for each bead provides a measure of pattern contrast for a given voltage.

To calculate the voltages required to achieve maximum pattern contrast at each wavelength and orientation, a LabVIEW program was written to sweep the Pockels cell voltage in steps of 25 V. Pattern contrast was measured at each voltage, shown in Figure 2.10a. Each orientation of the illumination pattern has a different voltage which gives a peak contrast. The optimal pattern contrast measurement was repeated for each pattern orientation at each wavelength, as shown in Figure 2.10b.

Without polarisation rotation, optimum pattern contrast cannot be achieved, and is almost at a minimum for the 120° pattern orientation. Performing SIM without polarisation rotation would introduce significant artefacts in the reconstruction, as low modulation contrast translates into low signal-to-noise ratio of high frequency components, as described in Section 2.1.2. Furthermore the signal-to-noise ratio would vary in each orientation, producing an OTF and corresponding PSF which is not rotationally symmetric, adding more artefacts to the reconstructed image.

When the Pockels cell is used to rotate polarisation, a high pattern contrast can be achieved in all orientations of the illumination pattern at each wavelength. This corresponds to a high modulation depth m in Equation 2.10, translating directly into high signal-to-noise ratio for high frequency components and facilitating artefact-free SIM reconstruction with $2\times$ resolution enhancement.

Furthermore, because the voltages to achieve a given polarisation rotation are similar for all wavelengths, the polarisation of all three laser wavelengths can be rotated simultaneously using the 561 nm optimal voltages. Figure 2.10b shows that a high pattern contrast is still maintained for each wavelength in this scheme. This facilitates use of the Optosplit for simultaneous 3-channel SIM imaging, providing multicolour resolution enhancement over $6\times$ faster than capturing channels separately and switching emission filters with a filter wheel.

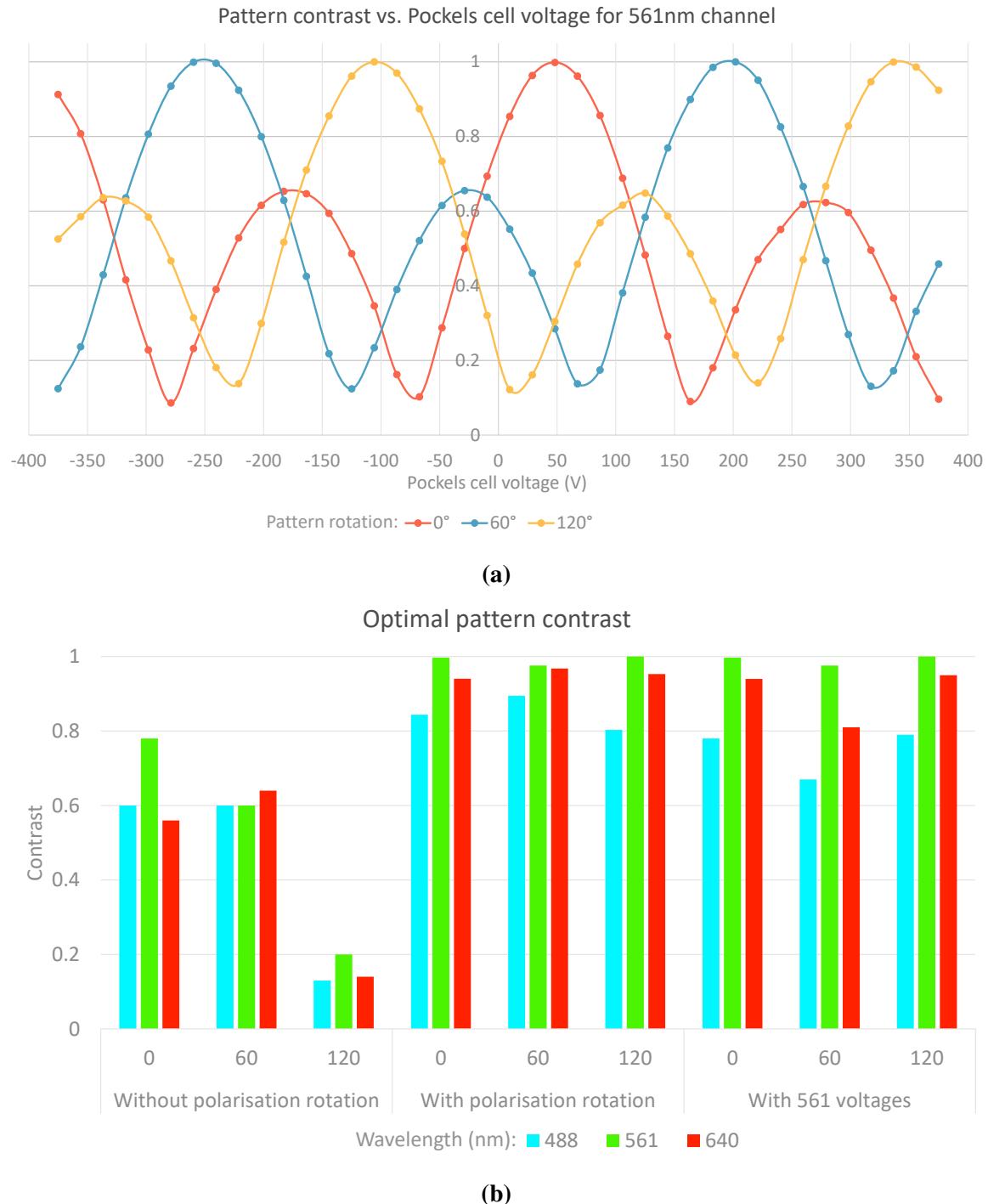


Figure 2.10: (a) shows the pattern contrast, measured on a 100 nm bead sample, as the voltage was swept from -375 V to 375 V to find the voltage which gave maximum contrast. The process was repeated for each rotation of the sinusoidal SIM pattern, and for each wavelength of light. (b) shows that rotating polarisation leads to high pattern contrast for all rotations and wavelengths. Furthermore, since the required voltage to achieve a given rotation is similar for all wavelengths, using the 561 nm voltages allows for imaging multiple channels simultaneously whilst maintaining a high pattern contrast.

2.3 LabVIEW hardware control

2.3.1 Control requirements for SIM imaging

The LAG SIM has a number of hardware components which must all work together to acquire a set of images appropriate for SIM reconstruction. Specifically, the following sequence of events must take place:

1. Set filter wheel to correct position for the excitation and emission wavelengths
2. Set voltage across Pockels cell for correct polarisation rotation
3. Set SLM pattern to the correct orientation and phase
4. Turn on the laser
5. Open the camera shutter for the exposure time
6. Close the camera shutter
7. Turn off the laser
8. Record the image

Steps 2 to 8 must be repeated for each of the 9 illumination patterns, and the entire process then repeated for additional colour channels if the Optosplit is not being utilised. For a time series, this entire process must then be repeated for the required number of frames; for a z-stack, the distance from the stage to the lens must also be adjusted before repeating the process.

The original SIM, detailed in [23], used a separate software program for controlling each piece of hardware. This caused a number of problems: firstly, setting up experiments was a complicated process prone to human errors, due to the number of parameters which had to be correctly entered in separate programs; and secondly, making modifications to the imaging parameters during an imaging session was difficult and time-consuming. When imaging live-cell biology, there is a real impetus to image quickly, as cells become unhealthy and soon die on the microscope stage.

Furthermore, the complicated system meant that only one person in the LAG had the knowledge to gather images from the SIM. This meant that any biological experiments which would benefit from SIM required my direct supervision, so that many projects were not able to use the SIM, and that I personally had little time for my other research aims. There was a clear need for a new, unified SIM interface, which could be used by non-expert users with minimal training.

2.3.2 A user-friendly LAG SIM interface enables operation by non-expert users

The new interface was built in LabVIEW. Although each hardware component came with its own control software, all components could also be controlled either through serial commands, analog voltages, or transistor-transistor logic (TTL) levels. As a graphical language with hardware control in mind, LabVIEW provides easy-to-use built-in blocks to fulfil each of these functions, as well as a clear visual representation of a sequence of control events. Furthermore it makes designing user interfaces a straightforward process, allowing a programmer to create a user-friendly program for the end user. This made LabVIEW an ideal candidate for building an accessible program for controlling the LAG SIM.

The LAG SIM interface is shown in Figure 2.11. The key functions are located in the orange Live Mode area and the blue Acquire tab, although for the sake of completeness each tab is detailed in this section.

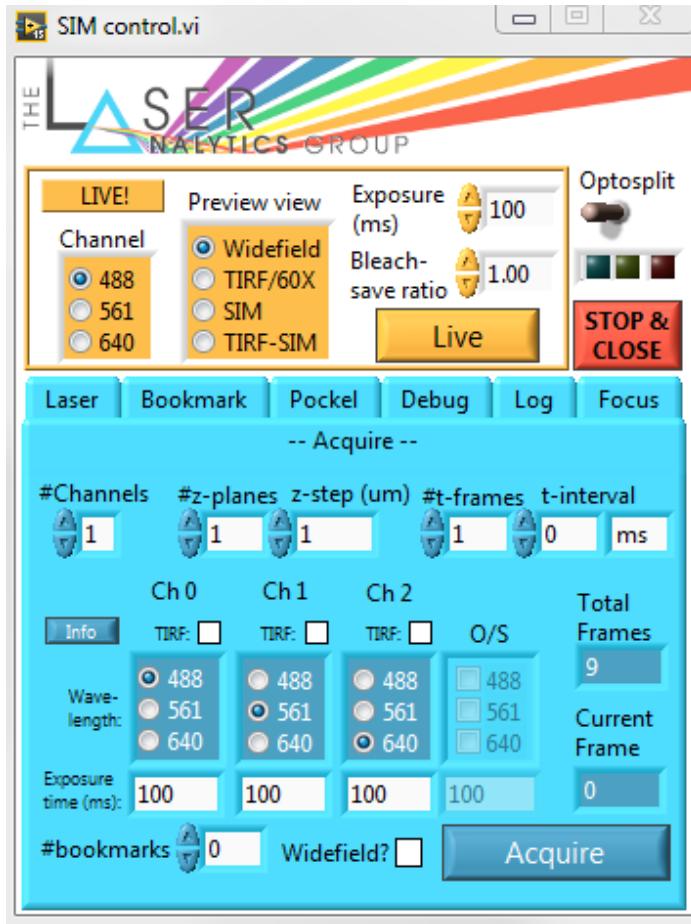
2.3.3 Live tab for fast image preview

The live tab is always visible whenever the LAG SIM control software is open. It works in conjunction with the HClImage camera software provided by Hamamatsu, which provides a live display of the camera chip.

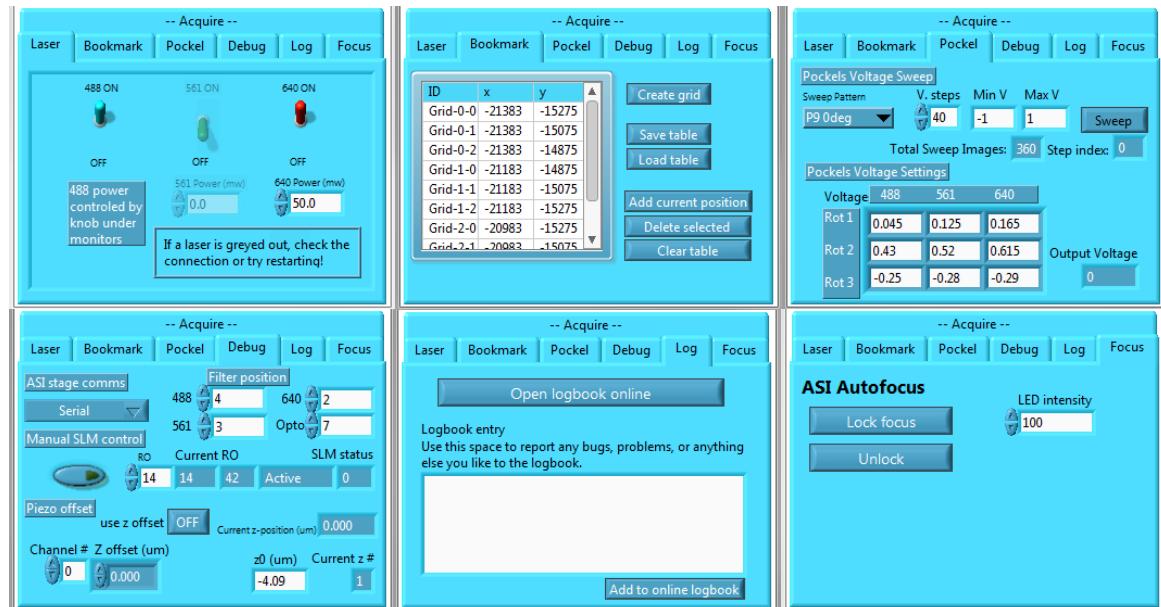
In live mode, the illumination wavelength can be changed with the radio buttons or with the F2, F3 and F4 keys for fast, mouse-free operation. This allows users to quickly identify cells in one channel, for example with a fluorescent nucleus marker, and confirm presence of another marker in the second channel. This fast channel switching minimises photobleaching when panning for healthy cells to image in detail.

Another live-mode feature designed to reduce bleaching is the so-called ‘bleach-save ratio.’ This is a number between 0 and 1 which switches off the laser for a fraction of the exposure time. By setting the exposure time and bleach-save ratio as small as possible while still being able to find healthy cells, the level of light the sample is exposed to can be minimised. This is especially important in live-cell biology, where excess light can cause phototoxic effects bringing about cell death. The strobe effect provided by the bleach-save ratio is provided to switch quickly between low light exposure in live mode and high laser power for image acquisition; for fluorophores particularly sensitive to photobleaching, there is some evidence that reducing the laser power and using a longer exposure time may be more effective at minimising damage [91].

The illumination pattern used for live mode will usually be a pseudo-widefield mode, which emulates widefield illumination by dithering all three phases of a SIM pattern over one



(a)



(b)

Figure 2.11: The LabVIEW user interface used to control the LAG SIM has been designed for operation by a non-expert user, with advanced tasks hidden in the extra tabs shown in (b).

exposure time of the camera. True epi-illumination is not possible on the LAG SIM due to the spatial mask, which blocks the 0-order diffraction spot in the Fourier plane. The user can also choose to preview the SIM pattern by selecting the appropriate radio button; if the stripe pattern is visible, this is a good indication that later reconstruction into a resolution-enhanced image will be successful. If the 1.49 NA TIRF lens is used, then selecting the TIRF radio button sets the SLM pattern such that the ± 1 diffraction spots hit the TIRF ring on the objective back aperture, leading to TIRF illumination at the sample. This can be done with the SIM pattern or in a dithered mode, as with non-TIRF illumination - however, it should be noted that in TIRF the stripe pattern will not be visible by eye, since its spatial frequency is at the edge of the OTF's support and therefore has low contrast.

The Optosplit rocker in the control software sets the system up for imaging with the Cairn Optosplit III. The Optosplit is a device on the output port of the microscope which contains two filter cubes and a set of mirrors so that each channel can be imaged simultaneously onto a separate 512×512 area of the 2048×2048 camera chip. When the filter cubes are removed, and the Optosplit rocker is in the OFF position, the Optosplit has no effect and channels are imaged sequentially, with an appropriate filter for each channel loaded into position by the filter wheel. With Optosplit mode ON, however, the live-mode channel selection radio buttons become checkboxes, so that each colour channel can be switched on and off independently. The Optosplit provides much faster imaging speeds, at the potential cost of cross-talk between colour channels.

This upper section of the user interface contains two more important components. A laser status indicator shows which lasers are currently exposing, an important safety feature so that users know it is safe to remove their sample after an imaging session. The STOP & CLOSE button safely exits the control software, ensuring that the lasers are switched off, the camera is finished exposing, and the voltage across the Pockels cell is set to 0 V. To ensure that this exit sequence runs when a user has finished their imaging session, the \times close button in the window's title bar is disabled.

2.3.4 Acquire tab facilitates multi-dimensional imaging

Once a healthy cell with the required staining has been located from live mode, it can be captured as a SIM image with the Acquire tab. With the default settings shown in Figure 2.11a, this will capture a set of 9 raw SIM images following the steps described in Section 2.3.1. Additional settings in this tab allow a set of images to be captured for 3D z-stacks, timelapse imaging, and other multi-image acquisitions.

If the number of z-steps is greater than 1, then the ASI stage will move in the axial direction at the end of every SIM acquisition by the value given in the z-step box. Since

the axial resolution of the SIM is 440 nm, for a Nyquist-limited z-stack a z-step of 200 nm should be used.

To take a series of images over time, the t-frames parameter should be set to greater than 1. For observing fast, dynamic events, the t-delay parameter should be set to 0 ms. Timelapse imaging, for example capturing an image every minute for an hour, can be achieved by setting the t-delay to 60 s, which will pause the software for the given amount of time at the end of each acquisition.

Multi-channel imaging can be achieved either with or without the Optosplit. If the Optosplit switch is set OFF, and the number of channels is set to more than 1, the filter wheel will rotate the correct filter into position after each channel is captured. The exposure time of each raw SIM frame can be set per-channel, and the channel order can be changed with the radio buttons.

Faster imaging can be achieved with the Optosplit. This sets the filter wheel to an empty filter, and filtering is performed with filter cubes in the Optosplit. In this mode, the choice of channels is controlled with checkboxes.

The bookmarks number should be set to the number of bookmarks in the bookmarks table, or a lower number N if the user only wants to capture the first N bookmarked locations from the table. If bookmarks are not to be used, this value should be left at 0 - a value of 1 will move to the first bookmark listed in the bookmark table and capture an image there. Bookmarks are discussed in more detail in Section 2.3.6.

The LAG SIM microscope can be used to capture images in a pseudo-widefield mode, by dithering the SIM pattern as described in Section 2.3.3. This is enabled by ticking the Widefield checkbox. This will enable imaging with widefield resolution at the camera's maximum acquisition speed of 100 Hz.

Once the appropriate variables have been set for the acquisition, the total frames is displayed in the Total frames box. This number should be entered into the HCImage camera software, to prepare the camera for a high-speed image acquisition. Once the user has pressed Start in the HCImage software, the camera buffer is ready to receive images; then pressing Acquire in the LAG SIM software starts the acquisition process.

A future version of the software could automate this final step, eliminating the need for the HCImage software. This would require synchronised access to the camera data stream through LabVIEW.

2.3.5 Laser tab provides safe power control

Laser control is integrated into the LAG SIM control software.

At the launch of the software, the controls are greyed-out and disabled. When LabVIEW successfully connects to each laser, its controls become enabled, to switch the lasers on and off and change the power. Lasers are switched off automatically when the software is closed, for added safety when the microscope is unattended.

Having fast and straightforward access to laser control is particularly useful for reducing light dosage in live mode. Power can then be increased for acquisition to obtain the highest possible image quality.

2.3.6 Bookmark tab for saving positions and creating mosaics

The LAG SIM LabVIEW program is able to store locations on the microscope slide as bookmarks. If the number of bookmarks in the Acquire tab is set to greater than 0, the microscope stage will move before each acquisition to the next bookmarked position in the bookmark table. This is especially useful for imaging a collection of cells over a long period of time.

The SIM user can pan around the microscope slide in Live mode searching for healthy cells with good fluorescent labelling, and save them to the bookmark table by clicking Add bookmark. The location can be returned to by double-clicking on the table entry, or acquired sequentially and repeatedly in the Acquire tab.

Furthermore this tab contains an option for creating bookmarks in a grid pattern. This can be utilised for imaging large areas, which are then stitched together to form one large mosaic image, for example the whole mouse brain shown in Figure 2.12.

2.3.7 Focus tab controls the autofocus

Using the grid bookmark function to image large fields of view can fail if the coverglass or microscope stage is not perfectly flat, which is often the case in practical experiments. To keep the focus locked at the same distance from the sample, the ‘Lock focus’ button begins the autofocus procedure. Assuming the autofocus sensor receives enough reflected signal from the infrared autofocus light emitting diode (LED), the sample will be locked at a fixed distance from the objective lens. If not, an error message is returned requesting the user to increase the LED power for a stronger signal. The system can also be used to maintain focus over a time period of hours to days, even as the objective lens drifts away from the sample.

The autofocus is switched off with the ‘Unlock focus’ button, releasing z-control back to the user.

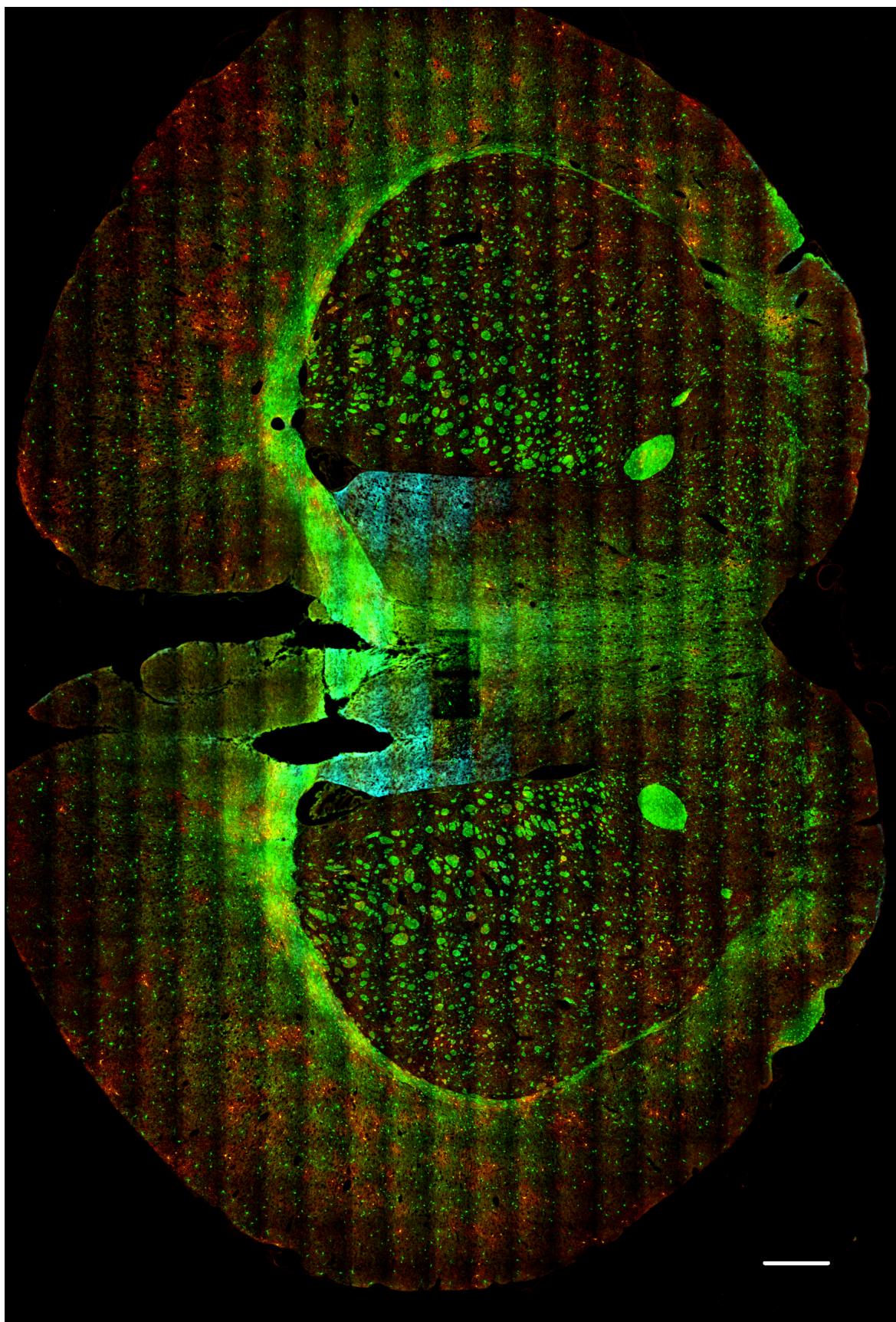


Figure 2.12: The bookmark mode offered by LAG SIM can be used to create tiled images, such as this 3-colour image of a mouse brain. Dissection and staining performed by Sarah Förster. Scalebar is 500 µm.

2.3.8 Pockels tab to find optimal voltages for maximum pattern contrast

The Pockels cell retards one polarisation axis by an amount dependent on the voltage across it. In combination with a quarter wave plate, this allows for linear polarisation rotation necessary for the high pattern contrast required for SIM, explained in Section 2.2.2. The amount of retardation depends on the wavelength of light, therefore different wavelengths require different voltages to achieve the same polarisation rotation.

To find the optimum voltage for each wavelength, functionality is implemented in the Sweep tab to sweep the voltage from -1 V to 1 V , in a number of voltage steps. At each voltage, the SIM pattern's phase is swept and pattern contrast can be measured using a diffraction-limited bead sample. Pattern contrast is calculated with a MATLAB script to produce the graph in Figure 2.10a, showing how pattern contrast changes with voltage.

This gives a coarse estimation of the voltage required to achieve maximum pattern contrast. The sweep limits can then be adjusted to a small amount either side of the coarse estimation's maximum contrast to obtain a more accurate voltage.

This process must be repeated for each wavelength, and each rotation of the SIM pattern. Once the process is complete, the optimum voltages can be entered into the table, and will be applied to the Pockels cell during a SIM acquisition.

2.3.9 Debug tab contains alignment options and other advanced features

A number of advanced user options are contained in the Debug tab. These extend the functionality of the microscope for non-conventional uses including alignment.

The emission filters in the filter wheel switch with excitation wavelength for use with commercially available fluorophores. However, some samples - such as semiconducting perovskites - emit around 750 nm when excited with 488 nm light. Unconventional filters can be selected in this tab to image these samples.

Manual SLM control gives the user direct control over the pattern displayed on the SLM. This means specialised patterns, such as a SIM pattern masked by a circle, or even 3 SIM patterns with different orientations overlayed, can be used. These types of patterns are particularly useful for alignment of the system: overlaying 3 orientations of the SIM pattern produces 6 spots at the Fourier plane, which must be centred around the optical axis; therefore a coarse alignment can be performed using a translucent pinhole alignment tool on the microscope, as shown in Figure 2.13. A fine alignment is then performed using the masked patterns, by ensuring the two circles overlap at the focal plane of the microscope [23].

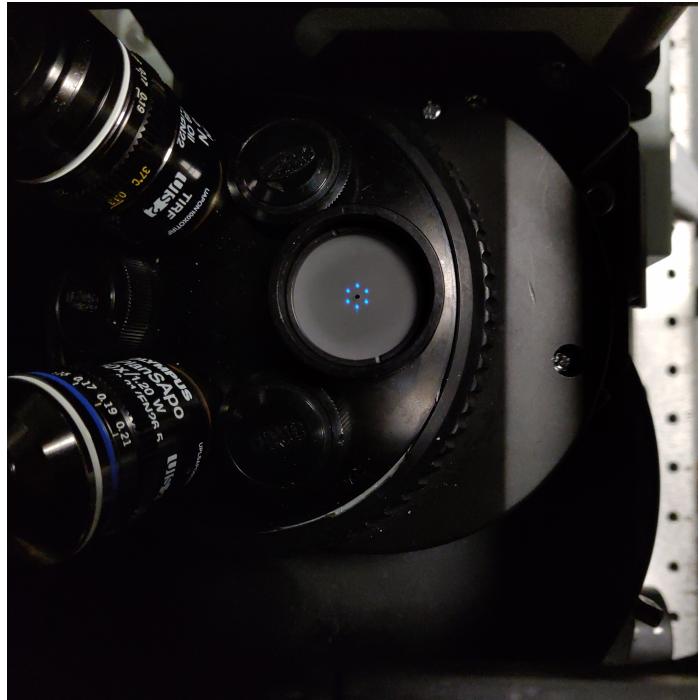


Figure 2.13: The 3 orientations of the SIM pattern are overlaid on the SLM, such that 6 spots are produced in the Fourier plane for alignment. The spots should be aligned such that the pinhole is at the centre of the spots.

Finally, the **Piezo Offset** section adjusts the z-position of the stage for each channel to account for chromatic offset. In practice this offset is less than 100 nm and **Piezo Offset** is only used for maintaining correct focus in TIRF imaging.

2.3.10 Log tab provides access to an online logbook

The LAG SIM control software logs all activity automatically to Labstep, an online laboratory management system [92] shown in Figure A.1.

When a user opens the LAG SIM software, they are asked to enter their name for the online logbook. When they press **Start**, the software sends an HTTP request to a Labstep timeline, recording the user's name and the start time. Similarly, when they **Stop & Close** the software, their name and finish time is logged to the timeline, along with any errors generated by the software during their use.

Furthermore, every acquisition is also logged to the timeline. This contains metadata relating to the acquisition, for example number of t-frames, z-steps, laser powers, and exposure times. This is especially helpful if the user is later unsure what acquisition

parameters were used for a particular image. The online Labstep logbook can be opened by clicking the Open logbook button.

Online logging benefits both the microscope user and the developer, allowing problems to be identified and fixed promptly. The implementation of the logging functionality is detailed in Appendix A.1, and the program can be reused in any other LabVIEW software to record and monitor instrument use.

2.4 A comparison of reconstruction algorithms reveals their pros and cons

Once SIM data is collected, it requires reconstruction to either enhance the resolution, remove out-of-focus light, or a mixture of both.

A basic SIM reconstruction algorithm implements the following steps:

1. Extract illumination pattern parameters (orientation angle and phase shift)
2. Separate Fourier space images into individual frequency components
3. Weight components in Fourier space by OTF and filter for noise removal
4. Shift frequency components into correct position in Fourier space
5. Perform inverse Fourier transform to obtain the reconstructed image

Additional steps are often added into the basic algorithm to reduce artefacts. Raw images may be corrected for consistent intensity and contrast across the dataset, and an additional filtering step for noise removal can be applied before the images are separated in Fourier space. Note also that step 3 can be performed after step 4, but this can lead to additional artefacts in the reconstructed image [79]. Artefacts can furthermore be suppressed by apodisation before the final inverse Fourier transform [79].

This basic algorithm will provide resolution enhancement, but does not implement optical sectioning for removing out-of-focus light. This requires an additional attenuation step, applied simultaneously with step 3.

Various implementations of the SIM reconstruction algorithm exist. During the course of my PhD, I have utilised and modified three different reconstruction algorithms, and written a plugin for ImageJ tailored specifically for LAG SIM and detailed in Section 2.5. Each implementation has its own merits and disadvantages, discussed in the remainder of this section.

2.4.1 Lin Shao SIM reconstruction runs on CUDA GPUs

The first implementation of the resolution-enhancing SIM algorithm used in the Laser Analytics Group was provided by Lin Shao, who worked on SIM with Mats Gustafsson [93]. It was developed during his time at Janelia Farm Research Campus, under the supervision of Gustafsson and Eric Betzig [94].

Shao's algorithm is written in native CUDA code, a programming model for parallel computation with Nvidia graphics cards [95]. This makes reconstruction very fast - around 1 frame per second on an Nvidia Quadro 2000. However, native CUDA must be recompiled from source on every new computer it is used on, and furthermore because the software has not been updated, it does not work with the latest version of CUDA.

The program has no graphical user interface, but is called from the command line. Command line calls can be made from most other programming languages, so a custom interface was built in MATLAB to make reconstruction a simpler process. The MATLAB interface firstly separates complex datasets such as multi-channel z-stacks into individual SIM frames, and sends these along with the correct reconstruction parameters, based on the lens used, to the command line reconstruction program. An example command-line call is shown in Snippet 2.1, showing the various input parameters sent as arguments to the compiled CUDA program.

Snippet 2.1: Lin Shao's CUDA reconstruction program is executed as a command line program through a MATLAB interface

```

1 command = ['C:\Users\user\Desktop\SIM\Reconstruction\cudaSIREcon\cudaSIREconDriver.exe' ...
2   ' --input-file=.\' outname '.mrc' ...
3   ' --output-file=.\' outname '_recon.mrc' ...
4   ' --otf-file=C:\Users\user\Desktop\SIM\Reconstruction\cudaSIREcon\' otf_file '.mrc'...
5   ' --nphases=3'...
6   ' --angle0=1.5708'...
7   ' --ls=' num2str(linespacing)...
8   ' --na=' num2str(na)...
9   ' --nimm=' num2str(n_im)...
10  ' --zoomfact=' num2str(zoom_factor)...
11  ' --wiener=' num2str(wiener)...
12  ' --otfRA=1'...
13  ' --dampenOrder0=0'...
14  ' --nosuppress=0'...
15  ' --nokz0=0'...
16  ' --k0searchAll'...
17  ' --background=100'...
18  ' --equalizet=1'];
19 [status,cmdout] = system(command,'-echo');
20 if status ~= 0
21   error('cudasirecon failed');
22 else
23   save([outname '_info.mat'], 'cmdout')
24 end

```

Shao's algorithm does not implement OTF attenuation, which is used for optical sectioning. This means it cannot remove out-of-focus light, so is not appropriate for reconstructing 3D image stacks.

The program was not built for widespread distribution, and as such there is no documentation or publication for it. Nor is the program publicly available; requests for a copy of the software have to be made directly to Lin Shao. This means the software is complicated to use, and has not been tested and verified by the wider community.

Lin Shao is now at Yale University; although he is contactable, he is not actively maintaining this piece of software. Since this implementation is fast becoming obsolete, it is now only installed on one computer in the Laser Analytics Group, which cannot have graphics updates installed.

2.4.2 SROS-SIM facilitates optical sectioning

SROS-SIM is a SIM reconstruction program by Florian Ströhl which implements both optical sectioning and resolution enhancement.

SROS-SIM is a MATLAB implementation of the super-resolution optical-sectioning (SROS) algorithm detailed by O'Holleran and Shaw in 2014 [96]. Optical sectioning removes out-of-focus light, so that 3D images can be reconstructed, for example of a whole cell with mitochondria labelled as shown in Section 2.6.2.

As a set of MATLAB scripts and functions, the code can be executed on any computer with a MATLAB license. Therefore the implementation is available for all major desktop operating systems, and furthermore is easy to modify. I have made modifications to the original code to perform operations on the graphics processing unit (GPU), bringing a $32\times$ increase in reconstruction speed. More detail on the implementation and profiling on different hardware is provided in Appendix A.2.

SROS-SIM has the disadvantage of not being publicly available or published and verified by the community. Furthermore the need for an expensive MATLAB license limits who can use the software. Finally, the important step of measuring the illumination pattern parameters is not fully automated, so that the software always requires some user input and can cannot be left to run unsupervised for a batch reconstruction task.

2.4.3 fairSIM allows reconstruction in ImageJ

fairSIM is an open-source plugin for ImageJ/Fiji, a popular image analysis suite which is also open source and free [97, 98]. The software is well-documented and hosted on Github, so it is continually updated and improved by the community.

The fairSIM reconstruction algorithm implements resolution enhancement and optical sectioning, with two different noise filters. It provides the user with complete control over the input parameters, and with debugging options enabled will show detailed visual output at each stage of the reconstruction. Although the number of options can be overwhelming for some users, it is powerful and useful software for advanced SIM users.

The fairSIM user interface is cumbersome, and reconstruction can only be performed on a single-colour, single-frame SIM image. This process requires around 20 clicks, and input parameters must be resubmitted for every reconstruction. This, coupled with the fact that fairSIM does not have GPU support, makes reconstruction very slow, especially for large, multicolour image sets.

2.4.4 LAG SIM is specifically designed for our microscope

LAG SIM is my personal implementation of the SIM reconstruction algorithm, built using the well-documented fairSIM Java library [99]. It is designed specifically for the Laser Analytic Group's SIM, simplifying the interface for non-advanced users, and implements a fast parameter-finding technique.

It is available publicly as an ImageJ/Fiji plugin [100]. This means that it is free, open source, and stays up-to-date for all users.

Snippets from the source code are given in Appendix A.3. The next section describes the LAG SIM plugin in detail for users of the software, providing a detailed resource to anyone reconstructing LAG SIM data in the future.

2.5 Reconstruction with LAG SIM for artefact-free images

2.5.1 The LAG SIM ImageJ interface simplifies parameter-finding for non-expert users

Setting SIM reconstruction parameters, particularly for noise filtering and optical sectioning, is a difficult task for new users. In LAG SIM, which uses the same reconstruction algorithm as fairSIM, there are 3 filter parameters; 2 apodisation parameters; 2 attenuation parameters; and a choice of 2 filters, which can be used in combination. This makes a high-dimensional parameter space to optimise over. Selecting appropriate values for these parameters requires understanding what each is for; setting optimum values requires experience and expert skill.

LAG SIM provides a number of tools designed to assist with common SIM reconstruction tasks, reducing the time required between acquisition and analysis of the data. The tools are accessed through the ImageJ menu shown in Figure 2.14a.

The “Fast widefield” function performs a simple pixel-by-pixel sum across the 9 raw images in each SIM acquisition. As described in Section 2.1.1, this simply reconstructs a widefield view of the data. This is a very fast computational process, and can be used to quickly generate a preview of the data before SIM reconstruction. If it later transpires that the acquisition was not good enough for artefact-free reconstruction, for example due to poor signal-to-noise ratio, the widefield reconstruction can be used as a “better than nothing” fallback so that the experimental data is not wasted. In this way, even in the worst case LAG SIM is as good as a widefield microscope.

The “Fast LAG SIM” function runs a SIM reconstruction on the active ImageJ image with reconstruction parameters which typically produce a good reconstruction, assuming a good signal-to-noise ratio. To find more optimal parameters, the “Parameter chooser” can be utilised, shown in Figure 2.14b. Every time the “Test!” button is pressed, a frame is reconstructed with the given parameters and placed side-by-side with previous test reconstructions. This allows a user to quickly compare the effect of different reconstruction parameters to produce artefact-free images. A detailed description of each parameter follows in Sections 2.5.2 through 2.5.5.

When investigating the effect of different reconstruction parameters on their data with the parameter tester, users can measure artefacts numerically by calculating Squirrel metrics. Squirrel is an imageJ plugin designed to assess super-resolution quality through the calculation of two metrics, resolution-scaled Pearson’s coefficient (RSP) and resolution-scaled error (RSE) [101]. To calculate these metrics, a widefield image is provided to Squirrel - for LAG SIM, simply the sum of the 9 raw frames - and this is compared to the super-resolution image convolved with a resolution scaling function (RSF), a kernel function designed to reduce the resolution of the SIM reconstruction into a widefield equivalent. The RSP is the Pearson’s correlation coefficient between the pixels of the widefield image and the resolution-scaled SIM reconstruction; the closer this value is to 1, the fewer artefacts have been introduced by the reconstruction algorithm. The RSE metric describes the root-mean-square error between the widefield image and resolution-scaled SIM image; the smaller this value the better. The RSE is intensity-dependent, so is useful for assessing non-linear intensity scaling applied by the reconstruction algorithm. As well as an average RSE for the whole image, the parameter tester will also output an error map showing the RSE for each pixel, which highlights areas of the image where the reconstruction algorithm is performing poorly. Figure 2.14d, generated

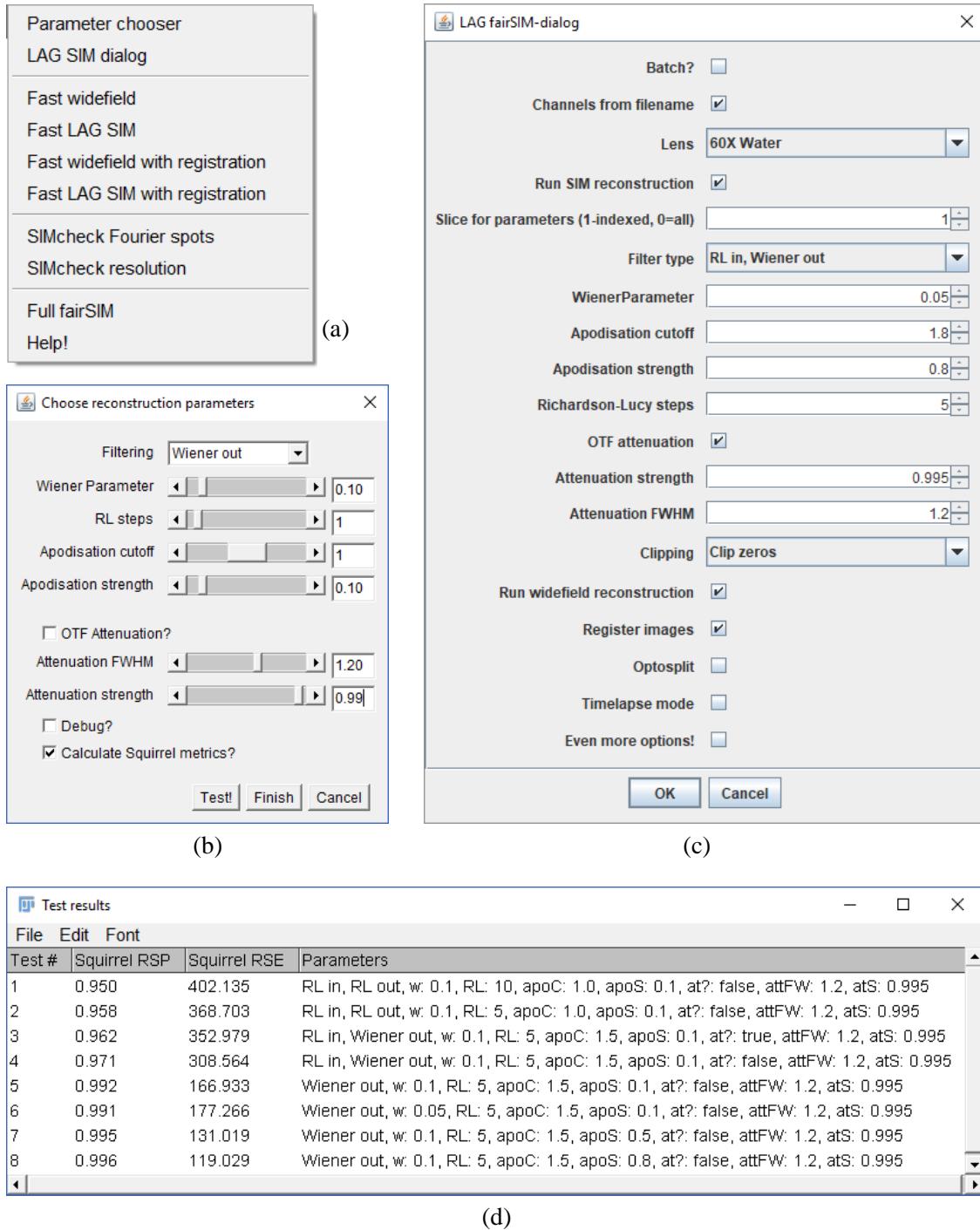


Figure 2.14: The LAG SIM ImageJ/Fiji user interface, utilising the fairSIM reconstruction algorithm, is specifically designed for the LAG SIM to make artefact-free reconstruction quick and straightforward. (a) shows the main LAG SIM menu, with the many reconstruction and SIMcheck tools created; (b) shows the quick parameter tester, for finding the optimal reconstruction parameters; (c) shows the main reconstruction interface, facilitating batch reconstruction; and (d) shows Squirrel metrics [101] which are calculated by the parameter tester for a numerical measure of artefacts.

by running the parameter tester 8 times, shows how the Squirrel metrics can be used to reduce artefacts in the reconstruction as different parameters are optimised.

Once appropriate parameters have been chosen for optimal artefact-free reconstruction, the “LAG SIM dialog,” shown in Figure 2.14c, is used to reconstruct the full image stack or time series. Parameters are saved to the ImageJ user profile, so that they are entered automatically from the parameter chooser and do not have to be entered every time the dialog is opened. When the batch checkbox is ticked, a folder-selection dialog window is displayed for reconstructing an entire folder of raw SIM data. Progress and estimated finish time are displayed to the user, allowing them to run the task unattended and return later for analysis of the reconstructed data.

The LAG SIM plugin makes SIM reconstruction a much less time-consuming task than with other reconstruction algorithms, and the convenient “Parameter chooser” enables non-expert users to create artefact-free SIM reconstructions.

2.5.2 Illumination detection provides user feedback about pattern modulation contrast

In order to produce an artefact-free reconstruction, the SIM algorithm requires measurement of the illumination pattern to a sub-pixel accuracy [97]. This is necessary for correctly performing the Fourier-space shift detailed in Section 2.1.2.

The LAG SIM reconstruction interface (Figure 2.14c) allows the user to choose which slice the illumination pattern is measured from. For z-stacks, it is best that the most in-focus slice is used, to obtain the most accurate measurement. Conversely, for a time-series it is best that slice 1 is used, since it will have the least amount of photobleaching.

Illumination detection with LAG SIM has been designed to require the least amount of user input necessary. The user simply needs to select the objective lens used, and the software will infer the necessary information such as pixel size and numerical aperture. Wavelength can be extracted directly from the filename if the file is saved using the correct naming convention.

After this process, the software provides feedback to the user about how well the parameters were detected, based on the modulation depth and the patterns’ delta peak locations in Fourier space. LAG SIM has well-defined resolution enhancement factors based on the period of the SLM grating patterns, shown in Table 2.1. The resolution enhancement factor is calculated as $1 + \frac{t}{O_r}$, where O_r is the radius of the lens’ OTF for the given emission wavelength and t is the spatial frequency of the illumination pattern. If the detected pattern

Table 2.1: The table shows the resolution enhancement factors for each combination of lens, laser, and SLM mode available for the LAG SIM. When the 100X lens is used in TIRF mode, the full $2\times$ resolution improvement can be achieved. Note that the 60X “tSIM*” mode uses the same SLM patterns as 100X tSIM, but because the 60X lens is not designed for TIRF this does not create total internal reflection at the coverglass.

Lens	Laser (nm)	Emission (nm)	SLM Mode	Resolution improvement
60X	488	525	SIM	1.59
60X	561	600	SIM	1.67
60X	640	676	SIM	1.61
60X	488	525	tSIM*	1.78
60X	640	676	tSIM*	1.76
100X	488	525	SIM	1.78
100X	561	600	SIM	1.89
100X	640	676	SIM	1.8
100X	488	525	tSIM	2.04
100X	640	676	tSIM	2.01

parameters do not correspond to the expected resolution enhancement factor for a given lens, wavelength, and SLM mode combination, then a warning is presented to the user.

If the delta peaks’ Fourier-space positions are correct but modulation contrast is low ($m < 0.5$), the software waits for user confirmation that they wish to continue with the reconstruction, and recommends that they need to capture higher quality SIM images. Note that feedback is not provided in batch-reconstruction mode.

2.5.3 Wiener filtering and apodisation reduces noise, but can introduce artefacts

When the Wiener filtering scheme is used for noise removal, the Wiener parameter and apodisation parameters are used in a complementary fashion.

Noise filtering is particularly important in SIM due to the unnatural relationship between signal-to-noise ratio and spatial frequency caused by Fourier-space shifting. A conventional 2D microscope OTF has a peak at 0, and decreases symmetrically, as shown in the widefield Fourier plot in Figure 2.15. As noise is assumed to be constant over all spatial frequencies, this causes a decrease in signal-to-noise ratio at high spatial frequencies. In SIM, a super-resolution image is built up by shifting the OTF in Fourier space. This results in local maxima in signal-to-noise ratio, which can be seen when a low value is chosen for the Wiener parameter. The extra noise in this region causes swirling hexagonal noise patterns, seen in the contrast-enhanced reconstruction for $w = 0.01$.

Although these patterns are simply how noise presents in a SIM image, they can easily be misinterpreted as features. The Wiener filter is a conventional noise filter used in image processing [102, ch. 4], which successfully removes hexagonal SIM noise as shown in Figure 2.15 for $w = 1.0$. The amount of noise filtering can be controlled by the user through LAG SIM’s ‘Wiener parameter’ input.

Aggressive noise removal, using high values for the Wiener parameter, can cause ringing in the reconstructed images, clearly visible as side-lobes around the $w = 1.0$ reconstruction in Figure 2.15 [82]. To reduce ringing artefacts, the reconstructed image is passed through an apodisation filter.

Apodisation applies a smooth low-pass filter, gradually reducing power in higher spatial frequencies avoiding a sharp cutoff. The radius of the apodising filter in Fourier space is controlled by the ‘apodisation cutoff’ parameter, where a value of 1 corresponds to the cutoff spatial frequency of the objective lens based on its numerical aperture and the emission wavelength of the image. The shape of the apodisation filter is controlled by the ‘apodisation strength’ parameter, where a value of 0 gives a tophat filter, $\frac{1}{\sqrt{2}}$ a triangular filter, and larger values further reduce the full-width half maximum.

The apodisation cutoff should be set to the same value as the resolution enhancement given in Table 2.1, with a maximum value of 2.04 for full SIM resolution enhancement. Smaller cutoff values reduce the radius of the reconstruction in Fourier space, directly reducing resolution; a value of 1 corresponds to widefield resolution. The apodisation strength should be as small as possible whilst still removing ringing artefacts for maximum contrast of high-frequency features.

2.5.4 Richardson-Lucy filtering introduces less artefacts when optical sectioning is not required

An alternative filtering scheme for noise removal is Richardson-Lucy deconvolution [83]. The Richardson-Lucy filtering scheme has several advantages over Wiener filtering: it guarantees non-negative pixel values, removing the need for an apodisation step; it does not cause ringing in the reconstructed image; and it only requires one parameter to control the level of noise removal, simplifying the reconstruction process for the LAG SIM user [83, 103].

Richardson-Lucy deconvolution is an iterative process, where each iteration of the algorithm converges on the maximum likelihood solution of an image corrupted with Poisson noise [104, 105]. The number of iterations is set by the user in LAG SIM. More iterations result in a narrower point spread function, and thus higher perceived resolution; but can also lead to noise amplification, causing speckle patterns to appear in the reconstructed image.

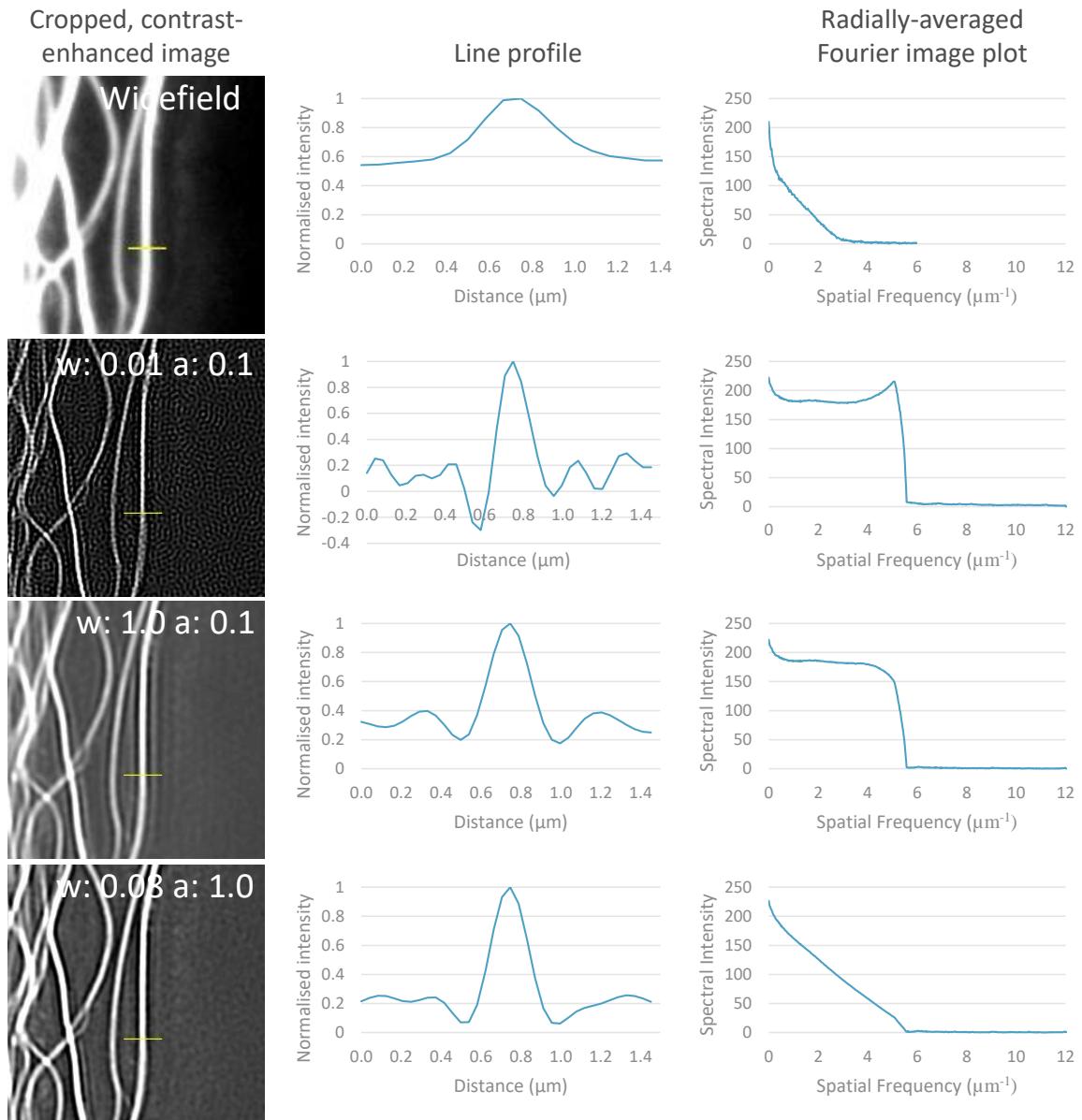


Figure 2.15: The left column shows crops from a SIM reconstruction with various values of Wiener parameter (w) and apodisation strength (a), and a widefield image for comparison. Note that image contrast has been greatly enhanced to make the noise pattern visible. The middle column shows the line profile through a tubule, showing the noise pattern with a low Wiener parameter, ringing with a high Wiener parameter, and suppression of noise and ringing using apodisation. The right hand column shows the radially averaged Fourier transform plot of each image; note the local peak when a low Wiener filter is used, which boosts noise at this frequency causing the unconventional noise pattern.

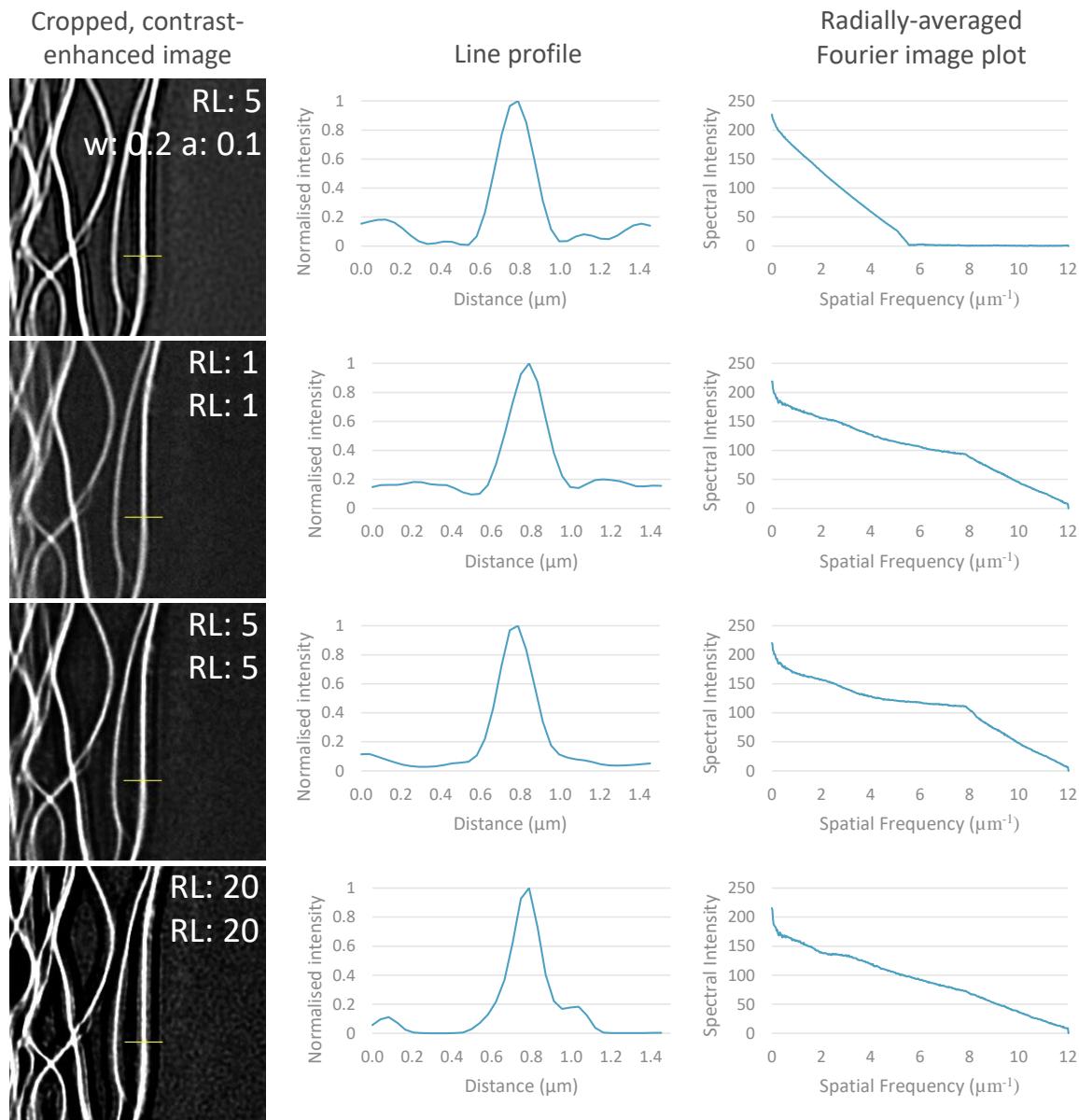


Figure 2.16: Richardson-Lucy deconvolution on the raw data reduces noise, leading to fewer artefacts in the Weiner-filter reconstruction. Furthermore using Richardson-Lucy filtering for reconstruction produces a more conventionally shaped OTF than Wiener filtering, which further reduces the number of reconstruction artefacts.

Row 1 in Figure 2.16 shows Richardson-Lucy deconvolution applied to the input images before conventional SIM reconstruction through the Wiener filter. This pre-filtering is effective at removing hexagonal SIM artefacts, since less noise is shifted and boosted in Fourier space.

Richardson-Lucy filtering can also be used on the final reconstructed image in place of Wiener filtering. Figure 2.16 shows that as the number of iterations is increased from 1 to 20, bumps in Fourier space caused by SIM reconstruction are smoothed out. This narrows the width of the reconstructed line, but can also lead to artificial amplification when the number of iterations is too high.

Richardson-Lucy input filtering can be used in combination with either the Wiener filter or with more Richardson-Lucy filtering on the reconstructed image. Therefore there are in total 4 filtering schemes offered by LAG SIM: Wiener-output; RL-output; RL-input, Wiener-output; and RL-input, RL-output.

2.5.5 OTF attenuation facilitates resolution-enhanced SIM with optical sectioning

In conventional widefield microscopy, light from areas above and below the focal plane are captured by the lens, causing out-of-focus light and reducing the axial resolving contrast of the microscope. When observing the 3D widefield OTF, shown in red in Figure 2.17, it is clear why this is the case. The OTF has no support in the axial direction z at low frequencies, which is known as the ‘missing cone’ problem [106].

In the Laser Analytics Group’s SIM setup, the illumination pattern is only projected onto the sample plane; therefore out-of-focus light does not show the SIM pattern. Intuitively, one would think this out-of-focus light can be removed by computationally rejecting any part of the image that does not have the SIM pattern, and only showing the in-focus parts of the image with SIM pattern on them. O’Holleran and Shaw show that this is achieved practically by attenuating certain parts of the OTF during reconstruction [96, 70].

If the center part of the OTF is computationally removed, out-of-focus light will be removed from the reconstructed image. In conventional widefield microscopy, this would have the side-effect of also removing low-frequency information from the image. In structured illumination microscopy, however, this information can be recovered from shifted components in Fourier space.

Figure 2.17 shows which parts of the OTF should be attenuated to achieve optical sectioning with no loss of resolution. The optical sectioning is controlled by a Gaussian notch in the first-order passbands, shown in blue, and a complementary Gaussian in the

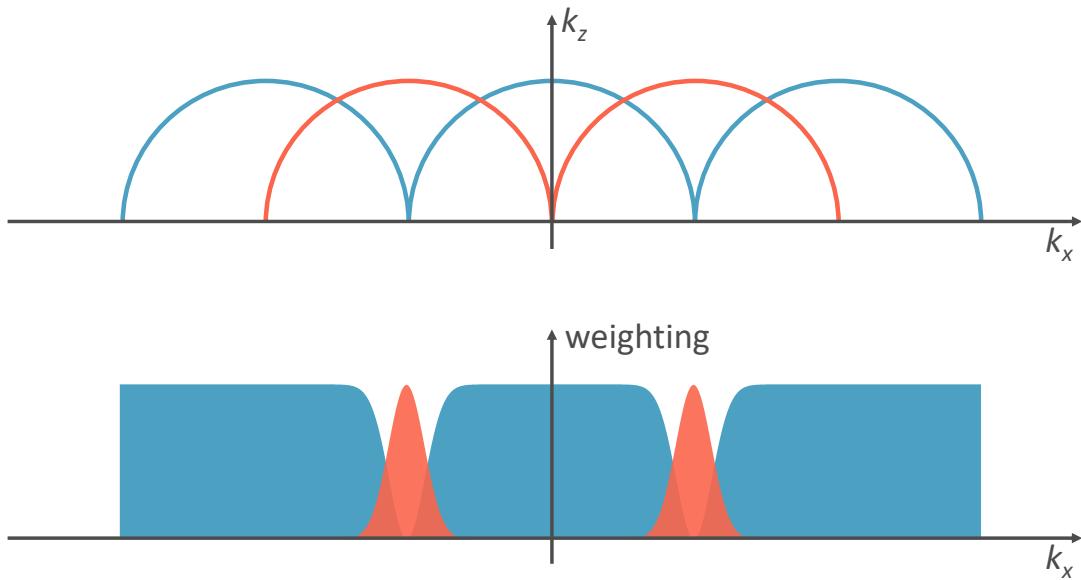


Figure 2.17: A widefield OTF, shown in red, does not have axial support at low lateral frequencies, leading to out-of-focus light in widefield images. Applying the red weighting as a Fourier-space filter removes this out-of-focus light; however in conventional epifluorescent microscopy, this would also remove low-frequency spatial information. The Fourier-space shifting inherent to SIM reconstruction recovers the attenuated information, as shown by the blue line, providing support at low lateral frequencies. When the shifted components are filtered by the blue weighting, which removes their own out-of-focus light, an optically-sectioned resolution-enhanced image can be reconstructed. Adapted from [96].

zero-order passband, shown in red. The width and depth of this passband is controlled in LAG SIM with the parameters ‘Attenuation FWHM’ and ‘Attenuation strength’ respectively, which control how much out-of-focus light is rejected from the image.

Out-of-focus light makes creating 3D reconstructions impossible. The widefield images of microtubules in Figures 2.18a and 2.19a show that out-of-focus light reduces the contrast of tubules in the focal plane, so that its position in the z -direction cannot be determined. Figures 2.18b and 2.19b show a standard SIM reconstruction using a Wiener filter as described in Section 2.5.3 - the out of focus information has caused artefacts to appear in the image, clearly visible as ringing in the z -projection image. By using OTF attenuation, the out-of-focus light and therefore the ringing artefacts can be removed, as shown in Figures 2.18c and 2.19c.

Note that OTF attenuation cannot be used with the Richardson-Lucy output filter. However, since it is a deconvolution algorithm, reconstruction by the Richardson-Lucy filter as described in Section 2.5.4 removes out-of-focus light, seen in Figures 2.18d and 2.19d. The disadvantage compared to using OTF attenuation, however, is the lack of control over how much out-of-focus light to reject, since the only input parameter to the Richardson-Lucy filter is the number of iterations. Therefore if optical sectioning is required, it is recommended to use the "RL-in, Wiener-out" filtering scheme with the OTF attenuation checkbox ticked.

Using optical sectioning to remove the out-of-focus light allows 3D reconstructions of the sample to be created. This can reveal information otherwise unobservable, such as whether a particle is located on the inside or outside of a cell membrane [107].

2.6 Results and Discussion

LAG SIM has been designed to be a versatile and user-friendly microscope, providing high-speed multicolour imaging surpassing the diffraction limit with optical sectioning provided computationally or with TIRF. With a number of useful automations, LAG SIM can be used to create timelapse videos, 3D reconstructions, large mosaic images, and for unsupervised imaging of multiple cells at different locations around the coverslip. As a result, the microscope has found applications in a wide variety of biological investigations.

Two such investigations are discussed in detail in Chapters 4 and 5. This section, however, presents a brief overview of a diverse selection of experiments demonstrating a wide range of LAG SIM’s capabilities. Acquisition and reconstruction methods are provided as a resource for researchers interested in conducting similar experiments with the microscope.

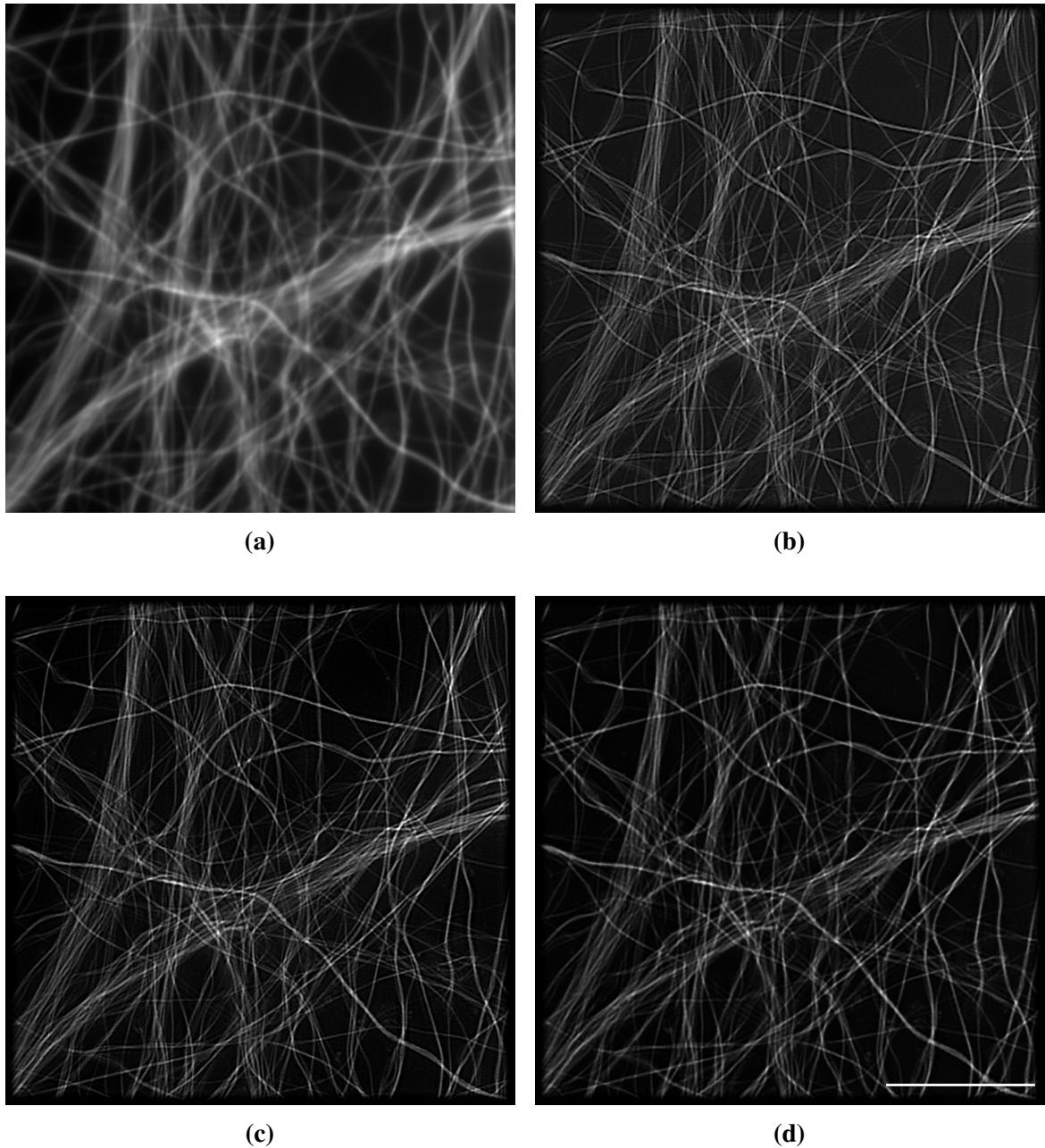


Figure 2.18: The images show a slice from a z-stack of microtubules. (a) Shows a widefield image, where out-of-focus light reduces the contrast of tubules in the focal plane. (b) shows a SIM reconstruction Wiener reconstruction, where the out-of-focus light causes ringing artefacts. (c) shows a SIM Wiener reconstruction with OTF attenuation applied, with a attenuation FWHM of 1.2 and an attenuation strength of 0.995. (d) shows a SIM reconstruction using the Richardson-Lucy filter, which removes out of light through deconvolution. Samples were prepared and labelled by Miranda Robbins. Scalebar is 10 μm .

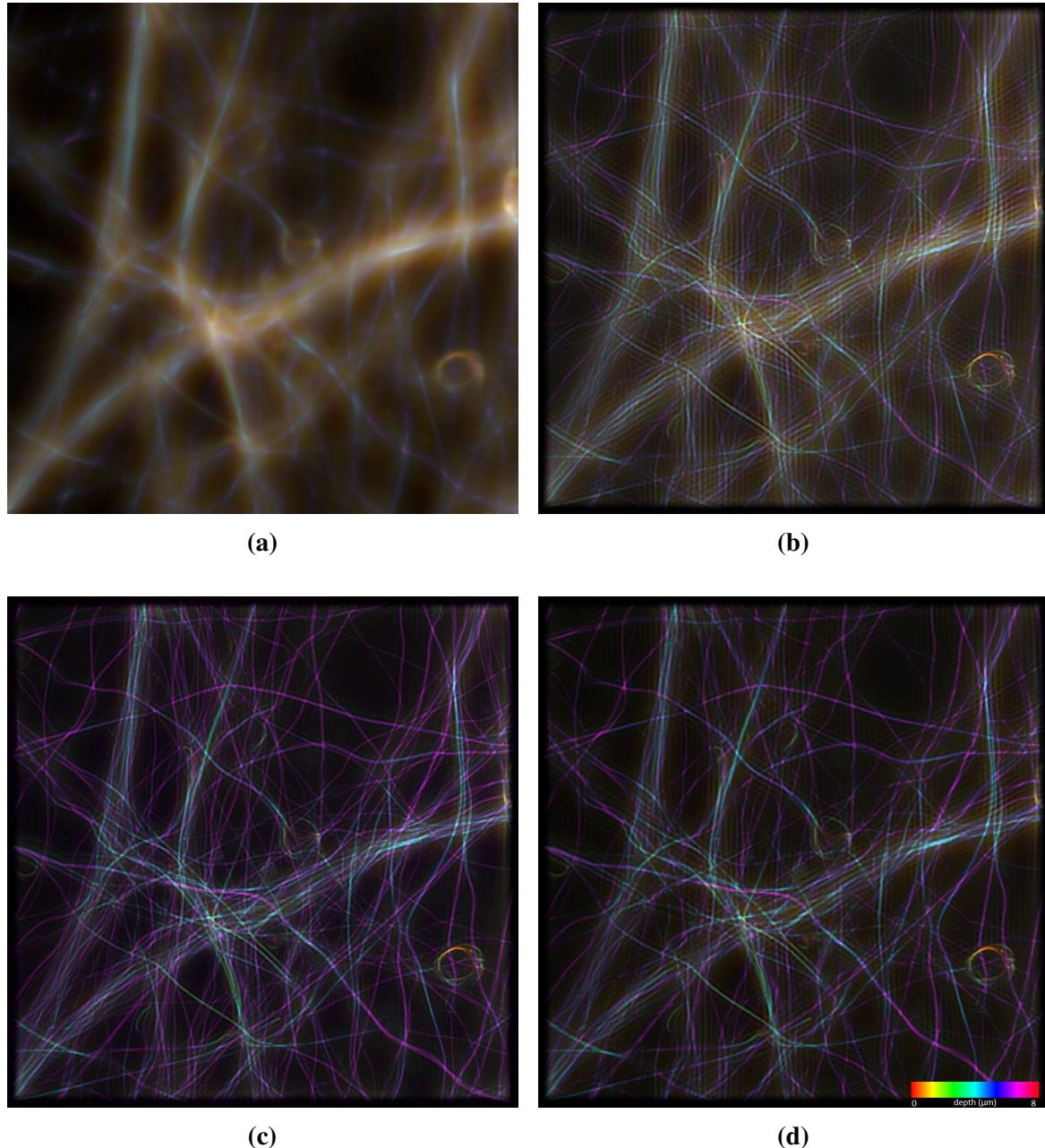


Figure 2.19: The figure shows a 3D stack of microtubules coloured by depth. The reconstruction methods shown are the same as for Figure 2.18: (a) is a widefield image; (b) a SIM reconstruction with Wiener filtering; (c) a SIM reconstruction with OTF attenuation applied; and (d) a SIM reconstruction using the Richardson-Lucy filter. Samples were prepared and labelled by Miranda Robbins. Colorbar represents relative depth from 0 μm to 8 μm .

2.6.1 Multicolour beads for chromatic alignment

For multicolour SIM experiments, small chromatic offsets between colour channels, shown in Figure 2.20a, require that image registration is performed. A slide of multicolour sub-diffraction beads can be used for measuring and correcting chromatic offset, because the well-defined points of light have the same spatial distribution in each colour channel.

For effective registration, the correct density of beads is required, and the beads should be well separated. A dense region of beads will form a homogeneous layer across the microscope slide, without well-defined bead coordinates, so cannot be used for registration. Conversely if the density of beads is too low then too few beads per field-of-view reduces the accuracy of registration.

A microscope slide or Labtek™ well with an appropriate density of beads can be prepared as follows:

1. Sonicate 0.1 µm beads for 10 min
2. Dilute beads 1:10 in distilled water, producing a 1.8×10^{10} particles/ml suspension density
3. Dispense 20 µl of the diluted beads onto a coverglass, or 50 µl into an 8-well Labtek
4. Wait for the solution to dry, about 2 h
5. Add mounting medium: immersion oil for use with the 100× oil lens, or water for use with the 60× water lens.
6. Mount the coverglass on a glass slide, or replace the lid on the Labtek™ and seal with Parafilm™ to prevent spillages

Following this protocol produces a bead sample with about 100 particles per field-of-view. The sonication step means that beads are well spread out, so that individual beads can be accurately localised.

Before capturing images, the bead sample can also be used for minimising spherical aberrations introduced by temperature changes and refractive index mismatches. When the bead sample is viewed on the microscope, adjusting the correction collar on the objective lens will change the shape of each bead's point-spread function. For minimal aberrations, the point-spread function should be symmetrical when defocussing above and below the bead layer.

To measure chromatic offset, a multicolour SIM acquisition should be captured in the same colour channels as the subsequent biological experiment. The single layer of beads will not have contribution from out-of-focus light, and should also have a high signal-to-noise ratio. Therefore the SIM reconstruction process does not require OTF attenuation for optical sectioning, so the Richardson-Lucy filter is recommended on the output stage of the reconstruction process. Section 2.5.4 shows that 5 iterations of filtering reduce noise without

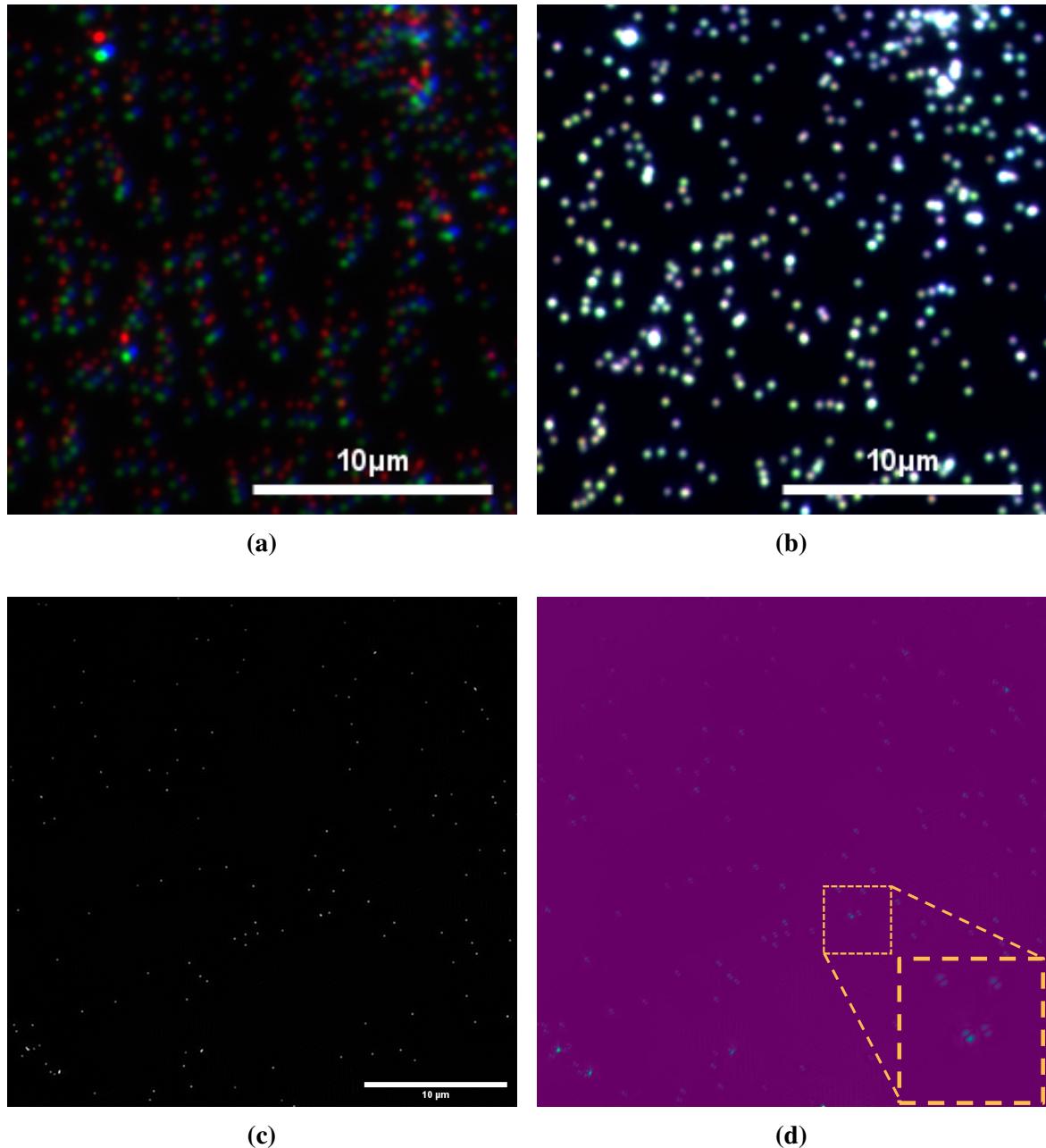


Figure 2.20: A small chromatic offset between the three SIM colour channels can be seen in (a). When the offset is corrected, as in (b), the beads appear white; the offset parameters can then be used to correct other multicolour images captured during the same imaging session. (c) shows a single channel of beads for the full microscope field of view, which is analysed in (d) with Squirrel [101] to assess microscope alignment and performance.

introducing artefacts. The suggested parameters to enter into the LAG SIM Fiji plugin are summarised as follows:

Filter	RL out
RL steps	5
OTF attenuation	false

After reconstruction, there will be some chromatic offset between each channel, as shown in Figure 2.20a. After registration and correction, Figure 2.20b shows that a red-green-blue (RGB) colour map will produce white beads. These same registration parameters can then be used for other multicolour experiments in the same imaging session.

Furthermore, the images of fluorescent beads can be used to assess the microscope's performance to ensure it is operating optimally. Squirrel metrics of a reconstructed full field-of-view were calculated using the LAG SIM parameter tester, and the resulting error map is shown in Figure 2.20d [101]. The map shows that errors are generally higher at the edges of the field-of-view, where the illumination is not as intense. Upon closer inspection, the Squirrel error map shows that the reconstructed PSF is not symmetrical, but has higher errors at 120° clockwise from horizontal which is not obvious in Figure 2.20c. This implies that the illumination pattern in this direction is not optimal, and perhaps needs realigning or the polarisation rotation needs to be checked.

2.6.2 Resolution-enhanced optical sectioning of HeLa cells shows MOF colocalised with endosomes

A frequent biological application of microscopy is imaging cultured cells under different treatments or with different genetic modifications [27, 108, 109].

Collaborators in the department had been developing a drug delivery system based on metal organic frameworks (MOFs), and wanted to observe the treatment to ensure MOFs were taken up by cells for delivery of their payload. In this particular application, which Chapter 4 describes in much greater detail, calcein was loaded into the porous structure of crystalline MOFs as a model drug. This MOF treatment was incubated with HeLa cells for 24 h, and uptake over time was captured with microscopy.

Widefield images had shown MOF located at the cell membrane; however widefield's inherent lack of optical sectioning meant that conclusions could not be drawn from these images about whether the MOF was inside or outside the cell [110].

One microscopy technique often used to provide optical sectioning is confocal imaging; this was unsuitable for this experiment for several reasons. Firstly, confocal microscopy has a

resolution limit of 200 nm. The MOF used in this experiment forms crystalline structures on the order of 100 nm, so a higher resolution was required to accurately capture their location. Secondly, acquiring a z-stack of confocal images to produce a 3D model of the cell, which is necessary for determining if the MOF is within the cell membrane, is a slow process, with a 3-channel acquisition taking at least 5 min per cell [111]. Since the cells could only survive on the microscope stage for 2 h, this limited the number of cells which could be captured per imaging session.

The LAG SIM was able to address both these issues. It provides optical sectioning with resolution enhancement, resolving the MOFs and accurately determining their location. It is also able to capture images faster, requiring just 9 images per plane, rather than point-scanning which takes longer to capture the same amount of fluorescence emission light [111].

To observe uptake using the laser lines available on the LAG SIM, the following labelling scheme was designed: calcein, which naturally is fluorescent when illuminated with 488 nm excitation light, was used as the model drug loaded into the MOFs; endosomes were labelled with CellLight Early Endosomes-RFP BacMam 2.0; and the nucleus was labelled with HCS NuclearMask™ Deep Red Stain to visualise the cell. The LAG SIM's laser lines, 488 nm, 561 nm, and 640 nm, could then be used to excite fluorescence for the MOFs, endosomes, and nucleus respectively.

Since fast dynamics were not observed, a 200 ms exposure time was used per raw frame. This relatively long exposure time ensured high signal-to-noise ratio. Similarly, to minimise cross-talk between imaging channels the Optosplit was not employed and filtering was performed by filters in the filter wheel.

To determine whether MOF had entered the cell or was simply sitting on the cell membrane, it was necessary to view the cells in 3D. The 60 \times water lens was selected, and operated in optical sectioning SIM mode. Images were captured at every 200 nm in the axial direction, to build up a Nyquist-limited z-stack.

To provide optical sectioning the z-stack was reconstructed in the LAG SIM Fiji plugin with OTF attenuation. The full-width half-maximum and depth of the attenuation Gaussian, defined in Section 2.5.5, were set to 1.2 and 0.995 respectively. These values generally work well to remove out-of-focus light from reconstructed images without introducing artefacts.

Since OTF attenuation was applied, the Richardson-Lucy filter could not be used on the output stage of the reconstruction; however 5 Richardson-Lucy iterations were applied to the input data to prevent noise-boosting in the reconstruction process, as described in Section 2.5.4. The Wiener parameter was set to the relatively low value of 0.03 by virtue of the high signal-to-noise ratio of the raw data. The apodisation cutoff was set to 1.6 for the 488 nm and 640 nm channels, and 1.67 for the 561 nm channel, since this is the factor of

resolution enhancement provided in 60X optical-sectioning SIM, listed in Table 2.1. Finally the attenuation strength was set to 0.8, to ensure that the OTF of the reconstructed image has the same shape as a conventional OTF, which prevents ringing artefacts.

The parameters required for the LAG SIM Fiji plugin are summarised as follows:

Filter	RL in, Wiener out	RL steps	5
Wiener parameter	0.03	OTF attenuation	true
Apodiation cutoff	1.6	Attenuation FWHM	1.2
Apodiation strength	0.8	Attenuation strength	0.995

Figure 2.21a shows a 3-colour reconstructed SIM image overlaid on a brightfield image, revealing green MOF particles within the magenta cell boundary. Furthermore, white spots in the image show colocalisation between green and magenta, implying that MOF has been taken up into the cell through an endocytosis pathway and is localised within endosomes.

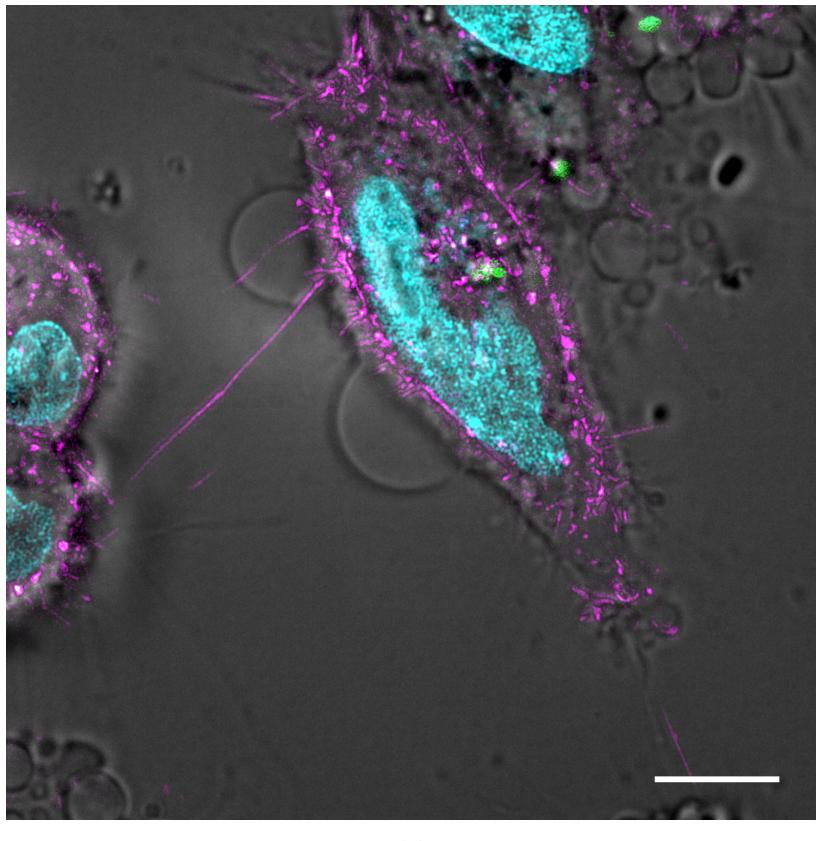
To confirm that MOF is truly within the cell boundary and is not sitting on top of the cell requires the 3D reconstructions provided by optical sectioning SIM. Figure 2.21b indeed verifies this. Whilst some MOF is on the outside of the cell boundary, other fragments of MOF have been endocytosed into the cell. White areas of the image again verify MOF colocalised with endosomes.

The 3D projection shown in Figure 2.21b highlights an issue with presenting 3D data on a static 2D medium. It is difficult for the reader to understand the orientation and perspective of the data. For this reason the 3D dataset is also available to explore interactively in a web browser at <https://fpb.ceb.cam.ac.uk/MOF>, using the new tool FPBioimage detailed in Chapter 3.

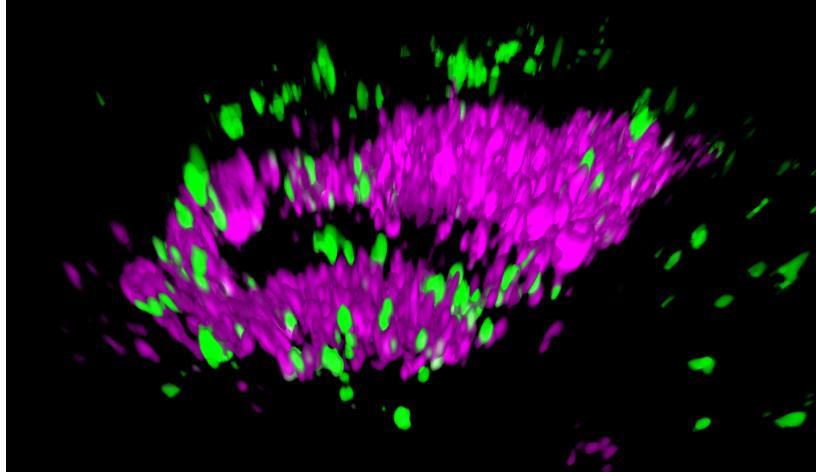
This drug-delivery experiment highlights the LAG SIM's capability of imaging live cells in 3D by utilising optical sectioning, and also providing resolution enhancement over other techniques. Details of the experiment are presented in Chapter 4, including timelapse imaging for observing uptake over a 24 h period, as well as similar experiments delivering therapeutic drugs to cells with different MOFs.

2.6.3 Fast, multicolour imaging of COS-7 cells reveals dynamic colocalisation between ER and lysosomes

During experiments on the dynamics of endoplasmic reticulum (ER), which are detailed in Chapter 5, we observed a strong interaction between ER and other organelles, including lysosomes and tubulin. To develop a deeper understanding of these relationships, we required fast, high-resolution imaging of live cells in multiple colour channels simultaneously.



(a)



(b)

Figure 2.21: (a) shows a 3-colour SIM reconstruction overlaid on a brightfield image of a cell, with MOF coloured in green, endosomes coloured in magenta and the nucleus coloured in cyan. MOF can be seen located within the cell boundary, and white spots showing colocalisation between green and magenta imply that MOF is taken into cells through an endocytosis pathway. (b) shows another cell with the same colour scheme as a 3D projection, confirming that MOF is truly located within the cell and is not simply sitting on the cell membrane. The 3D data set is available to explore interactively at <https://fpb.ceb.cam.ac.uk/MOF>. Cells were prepared for imaging by Michelle Teplensky. Scalebar in (a) is 20 μm .

To resolve the shape of lysosomes and the fine network structure of ER requires a microscope capable of resolving structures smaller than 100 nm. Whilst some super-resolution techniques are able to resolve structures as small as 20 nm, such as STORM and STED, they do not have the imaging speed required to observe fast dynamic events such as ER tubule growth. LAG SIM is able to provide resolution beyond 100 nm, at 11 frames per second, suitable for imaging these organelle dynamics.

For high resolution, the 100 \times oil lens was chosen, although, because we were looking deeper than 100 nm into the cell, it was not operated in TIRF mode. To facilitate simultaneous multicolour imaging, the Optosplit was employed. Cross-talk between colour channels was minimised by labelling just two organelles per experiment, in the 488 nm and 640 nm excitation channels.

A time-series of SIM images was acquired at a raw exposure time of 10 ms per frame, which equates to a reconstructed image rate of 11 Hz. Since optical sectioning was not required, the Richardson-Lucy filter was used on both the raw images and for the reconstruction output stage, as described in Section 2.5.4. Richardson-Lucy filtering was performed with 5 iterations, which Figure 2.16 shows gives a good balance of noise suppression without noise amplification, leading to artefact-free images. The parameters required for the LAG SIM Fiji plugin are summarised as follows:

Filter	RL in, RL out
RL steps	5
OTF attenuation	false

A frame of the reconstructed time series is shown in Figure 2.22, demonstrating artefact-free SIM imaging of ER and lysosomes. The figure shows lysosomes in cyan colocalised with ER in red. The timelapse video reveals a highly dynamic restructuring of the ER, which appears to be regulated by lysosome movement. Lysosomes located on the end of ER branches pull the ER into new shapes, and help attach it to other parts of the organelle to rearrange the network connections.

Work on these observations is ongoing, with a variety of genetic mutations now being applied to cell lines to understand the biochemistry of these organelle interactions.

2.6.4 TIRF imaging of MEF cells resolves dynamic actin structure on the cell membrane

Structure on the cell membrane is difficult to observe in epifluorescent microscopy due to out-of-focus light obscuring details in the focal plane. This is demonstrated in Figure 2.23a,

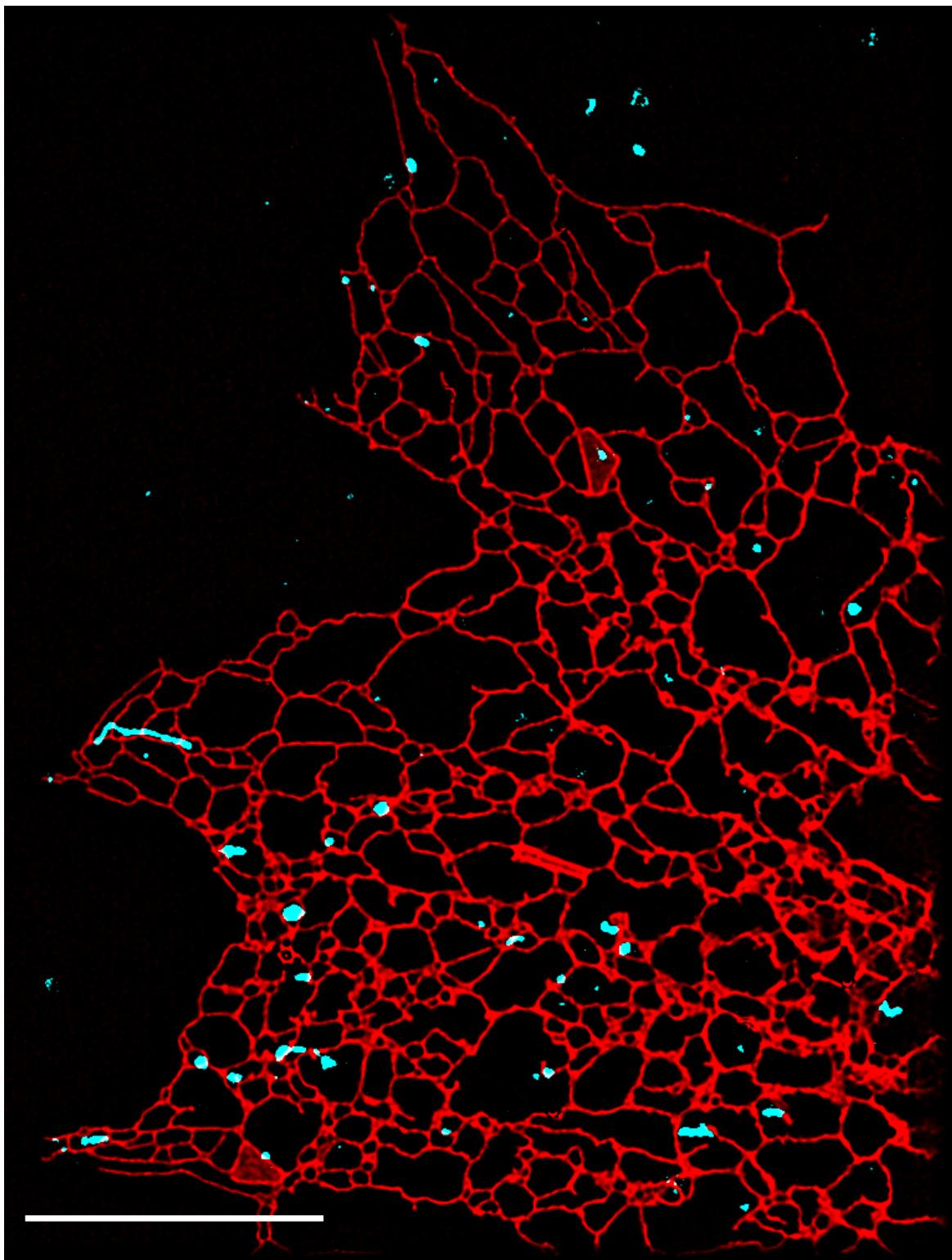


Figure 2.22: COS-7 cells with ER is coloured in red, and lysosomes are coloured in cyan. Capturing a timelapse utilising the Optosplit and optical sectioning SIM reveals a dynamic colocalisation between lysosomes and ER tubules, where lysosomes appear to pull the ER network to rearrange connections. Cells were cultured and labelled by Meng Lu. Scalebar is 10 μm .

a widefield image of mouse embryonic fibroblast (MEF) cells with fluorescently labelled actin coloured in cyan and the protein KRAS in orange. Out-of-focus light from areas above the imaging plane make it difficult to distinguish individual actin structures, particularly in areas with a high density of filaments. The out-of-focus light is even more problematic in the protein channel, producing a bright background blur in the focal plane so that the local distribution of protein cannot be determined.

As described in Section 1.3.4, actin is a component of a cell's cytoskeleton, responsible for maintaining the rigid shape of the cell [44]. The KRAS gene and the associated KRAS protein control cell proliferation [112] and are involved in the early stages of many signal transduction pathways [113, 114]. The aim of this experiment was to image the formation and growth of actin in the cell membrane and understand its dependence on the KRAS protein.

To perform this task required a microscope with TIRF capability to capture fluorescence only from proteins located at the cell membrane. Furthermore a resolution beyond the diffraction limit was required to image individual actin filaments in dense areas. LAG SIM was chosen for its fast, high-resolution TIRF-SIM capability.

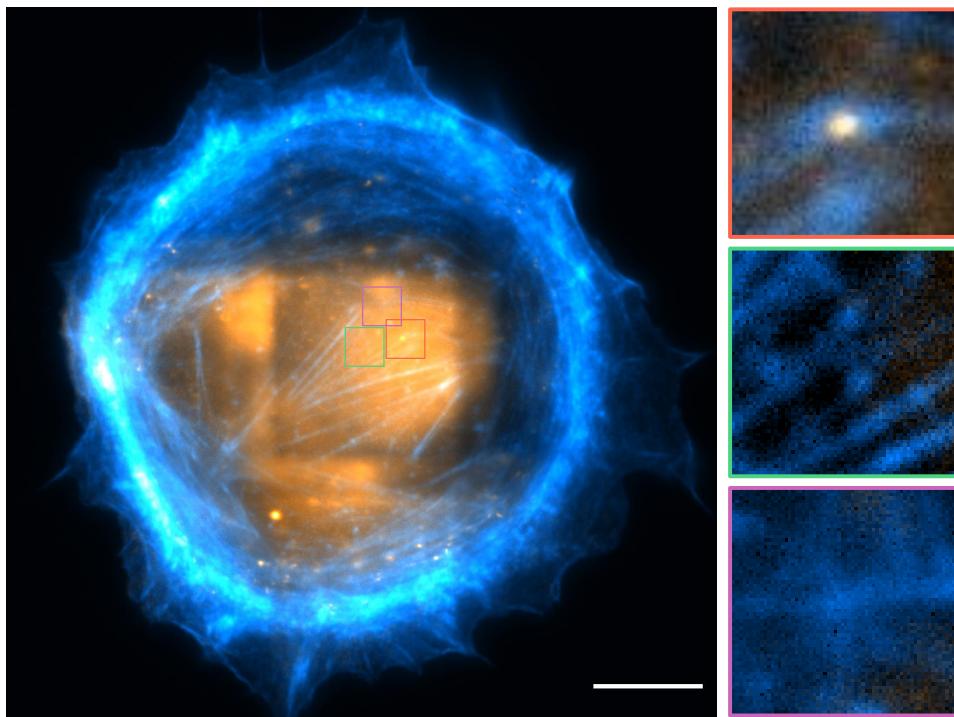
To image in TIRF at the highest resolution, the 100X lens was used. KRAS protein and actin were labelled with fluorescent probes with emission wavelengths of 488 nm and 640 nm respectively.

To capture the development of a large number of cells over a long time period, LAG SIM's bookmarking feature was used to store the position of cells as described in Section 2.3.6. This way time lapse imaging of the actin growth could be recorded for many cells. Furthermore, due to the large size of MEF cells, LAG SIM's mosaic feature was used to capture 3×3 field-of-view grids to show entire cells in a reconstructed image. Some artefacts from the non-uniform illumination field of the lens are visible in the widefield Figure 2.23a.

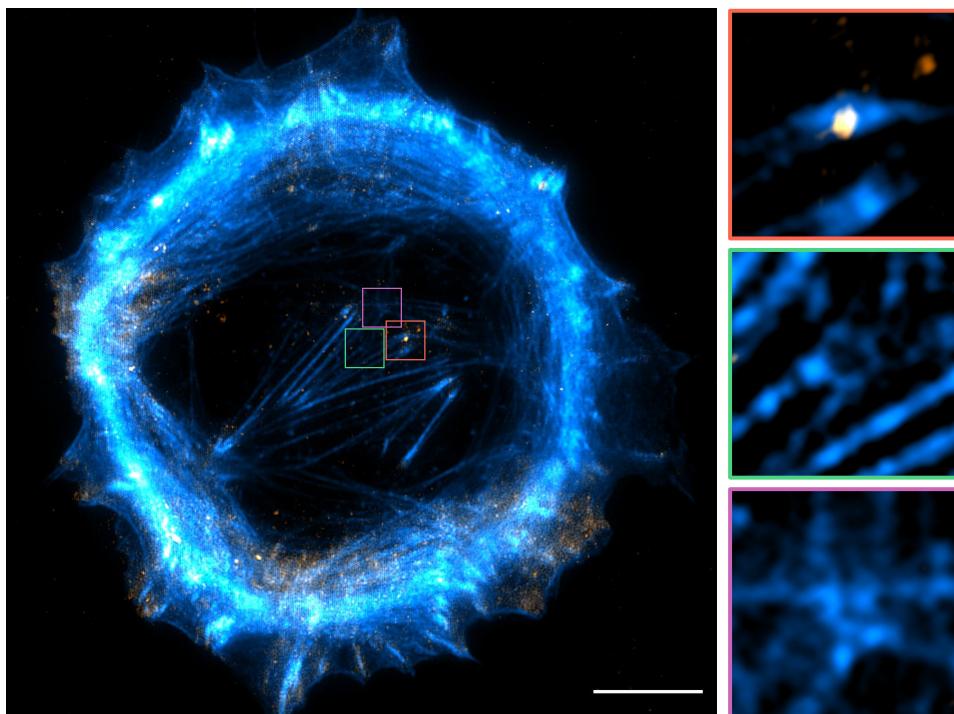
Out-of-focus light is physically removed by TIRF, so optical sectioning was not required as part of the SIM reconstruction process. Richardson-Lucy filtering was used on both the input data and in the reconstruction output stage with 5 iterations to remove noise without introducing artefacts, as described in Section 2.5.4. The parameters entered into the LAG SIM Fiji plugin are summarised as follows:

Filter	RL in, RL out
RL steps	5
OTF attenuation	false

A reconstructed TIRF-SIM image is shown in Figure 2.23b. In contrast to the widefield image, individual clusters of protein can now be clearly identified, with the out-of-focus



(a)



(b)

Figure 2.23: (a) shows actin labelled in MEF cells, where out-of-focus light blurs details in the actin. Utilising TIRF-SIM, shown in (b), enhances resolution and removes out-of-focus light. Inset images show various stages of actin development, with a dense cluster of actin (inset in red box) forming into a ring structure (inset in green box) which acts as a source for new filament growth (inset in purple box). Cells were prepared for imaging by Anchal Chandra. Images are stitched 3×3 mosaics; scalebars are $10 \mu\text{m}$ long; and inset images are $3.2 \mu\text{m}$ squares.

background light removed. Furthermore, the structure of individual actin filaments can now be clearly resolved.

During these experiments 3 distinct phases of actin development were observed, shown in Figure 2.23b's coloured boxes. The red box shows an early stage of development, with KRAS protein colocalised with a dense area of the actin label. As time progresses, the actin forms a ring structure, seen in the green inset box. Finally long actin filaments begin to grow out of the ring structure, as shown in the purple box.

Work is currently ongoing to optimise the labelling of the KRAS protein. Under the current labelling scheme only one frame of the 488 nm channel could be captured before photobleaching of the fluorescent label reduced the signal-to-noise ratio so that artefact-free SIM reconstruction could no longer be performed. Various fluorescent probes are now under investigation to design a scheme where the KRAS protein can be imaged for the full time-lapse duration. It is hypothesised that KRAS protein will be colocalised with the ring structure as it forms and as the actin filaments grow from it. LAG SIM's unique combination of high-speed, high resolution, multicolour TIRF imaging make it one of the only instruments in the world that can capture these events to test the hypothesis.

2.6.5 Resolution-enhanced optical sectioning of brain-slice tissue imaging for measuring the Node of Ranvier

Neurons in the brain communicate at junctions known as synapses, which are formed where an axon from one neuron meets a dendrite from another [115]. The axons, coloured yellow in Figure 2.24, form long, thin projections, and can be compared to electrical wires carrying signals across the brain to control bodily functions such as sensing, movement, and consciousness.

Figure 2.24 shows axons coated in a fatty sheath known as myelin [115]. This protects the axon from interference with other nearby axons, and increases the transmission speed of electrical signals as action potentials jump from one gap in the myelin sheath - known as a Node of Ranvier - to the next in a process called saltatory conduction [116]. Diseases which cause injury to the myelin sheath are associated with motor neuron disorders such as multiple sclerosis and leukodystrophy [117].

The myelin sheath is not part of the neuron cell, but is rather a component of another type of brain cell called an oligodendrocyte, shown in blue in Figure 2.24 [119]. Oligodendrocytes are created from cell-lineage parent cells called oligodendrocyte progenitor cells (OPCs), which themselves are derived from embryonic stem cells [120]. In mouse models two developmentally distinct OPC populations exist: a first wave of OPCs arise from precursors

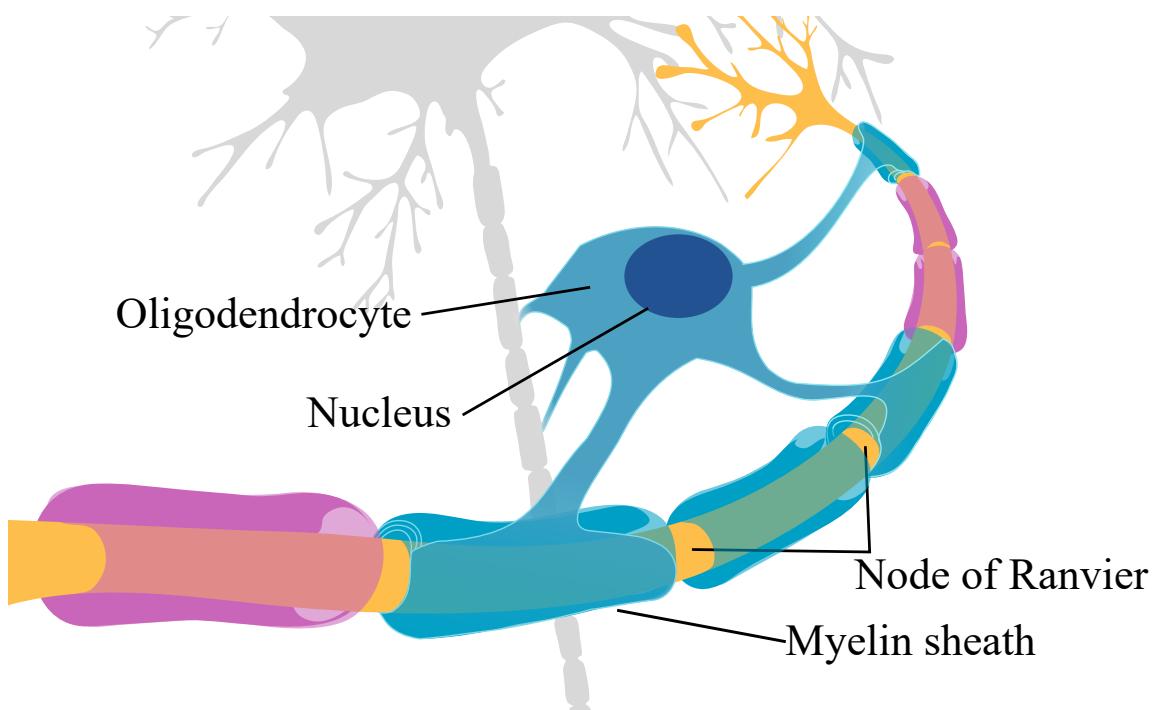


Figure 2.24: Diagram of a neuron, adapted from Wikipedia [118]. A neuron, coloured in yellow, has long wire-like projections called axons. Axons are covered by a fatty coating known as the myelin sheath, which protects it from interference from neighbouring axons and increases the transmission speed of signals. The myelin sheath is part of a different brain cell type called an oligodendrocyte, shown here in blue. The gap between two myelin sheaths is known as the Node of Ranvier.

in the ventral part of the brain at day 15 of embryo development; then a second wave of OPCs from dorsal precursors develop from birth [121]. Collaborators have shown that when one of these waves of OPCs is genetically suppressed mice are still able to develop to maturity, but have significant difficulties with fine motor control.

It was hypothesised that a physiological difference in the myelin sheath would be visible between oligodendrocytes derived from ventral OPCs and those derived from dorsal OPCs.

A fluorescent labelling scheme was devised by collaborators to distinguish between ventrally- and dorsally-derived oligodendrocytes. Also, the Node of Ranvier was made visible by labelling the end of myelin sheaths in a third colour; the distance between two adjacent labels was therefore the length of the Node of Ranvier. The collaborators were interested to find whether there was a difference in this length for the two oligodendrocyte lineages.

The SIM examples presented in this results section so far have shown data of cultured cells which form a single layer on the microscope slide, and do not stack on top of each other. However, the versatility of LAG SIM also allows it to perform tissue imaging beyond the diffraction limit.

Imaging deep in tissue is a notorious challenge in microscopy for two main reasons [122]. Firstly, multiple layers of fluorescent cells cause significant out-of-focus light, which obscures details in the focal plane. Secondly, scattering of photons as the light passes through multiple refractive index changes means that it becomes more and more difficult to focus light the deeper into the tissue one tries to observe [123]. Both these issues are particularly problematic in SIM, as the out-of-focus light degrades the modulation contrast of the sinusoidal illumination pattern, and scattering of photons as they travel to the focal plane leads to less pattern-generating interference, as shown in Figure 2.25.

Nevertheless, it has been found that LAG SIM can image up to 10 μm into tissue without degradation of the reconstructed images.

8 μm -thin slices of mouse brain were labelled to study the myelin sheath of oligodendrocytes. Myelin was labelled in one of two colours, depending on whether the cell originated from the ventral or dorsal area of the brain. The end of each myelin sheath was labelled with a third colour to measure the length of the Node of Ranvier, to discover whether there is an observable difference between the two oligodendrocyte types related to disease pathology.

To allow the greatest imaging depth, the lens with the closest refractive index matched to the tissue was chosen; in this case, the 60 \times water lens. A greater penetration depth could be achieved with a lens better matched to the tissue's refractive index, for example a silicon-oil immersion lens with a refractive index of 1.406, if one is available. A z-stack was captured



Figure 2.25: A false-colour diagram of SIM imaging in thick tissue samples. As light passes through tissue, scattering of photons causes the wavefront to become distorted. When the two beams meet at the focal plane, distortion of the wavefronts means that the sinusoidal interference pattern is degraded. Moreover, out-of-focus light from fluorescence above and below the focal plane further reduces the modulation contrast of the interference pattern. Due to these effects, LAG SIM can only produce reliable images about $10\text{ }\mu\text{m}$ into thick samples. Note that the diagram is not to scale, and that the green and yellow beams come from the same laser source but false-colour has been used for clarity.

with a step-size of $0.2\text{ }\mu\text{m}$, to build up a 3D image. Cells were fixed, so a raw frame exposure time of 200 ms was used to ensure a high signal-to-noise ratio.

To remove out-of-focus light in the reconstructed SIM images, OTF attenuation was used in the LAG SIM Fiji plugin. An attenuation FWHM of 1.2 and an attenuation strength of 0.995 were chosen, which Section 2.5.5 shows provide optical sectioning without introducing artefacts. Richardson-Lucy filtering was applied to the input images to suppress noise, and a Wiener parameter of 0.1 was chosen for output filtering. Table 2.1 shows that the resolution enhancement provided by the $60\times$ lens in optical sectioning mode is 1.59 to 1.67 - however, relatively weak pattern modulation contrast lead to high-frequency artefacts in reconstructed images, so an apodisation cutoff of 1.5 was chosen to suppress artefacts as much as possible, with an attenuation strength of 0.8 to approximate the shape of a widefield OTF.

The reconstruction parameters entered into the LAG SIM Fiji plugin are summarised as follows:

Filter	RL in, Wiener out	RL steps	5
Wiener parameter	0.1	OTF attenuation	true
Apodiation cutoff	1.5	Attenuation FWHM	1.2
Apodiation strength	0.8	Attenuation strength	0.995

Figure 2.26 shows a slice of a reconstructed z-stack. Orange dots marking the ends of myelin sheaths are located either on dorsally-derived oligodendrocytes, shown in magenta, or on ventrally-derived oligodendrocytes, shown in cyan. The length of the Node of Ranvier is measured as the distance between two adjacent orange markers. Lengths were measured automatically with a custom script in the image analysis suite Icy [124], and assigned to “dorsally-derived” or “ventrally-derived” based on the colour of the oligodendrocyte they were located on.

The signal-to-noise ratio in the reconstructed images is notably poor for the oligodendrocyte channels, and artefacts have been introduced by the SIM algorithm despite optimal reconstruction parameters. This can be explained by a high 3D density of oligodendrocytes, causing lots of out-of-focus light in the focal plane which reduces the modulation contrast of the SIM illumination pattern. As derived in Section 2.1.2, pattern contrast is directly proportional to signal-to-noise ratio for the high-frequency components in reconstructed SIM images, so excessive background light severely affects the ability to produce clear SIM images. The LAG SIM was chosen for this experiment because of its multi-colour capability and speed, allowing a large number of samples to be captured automatically to calculate reliable statistics. In the future, a multi-colour STED system may reveal more detail in the oligodendrocytes.

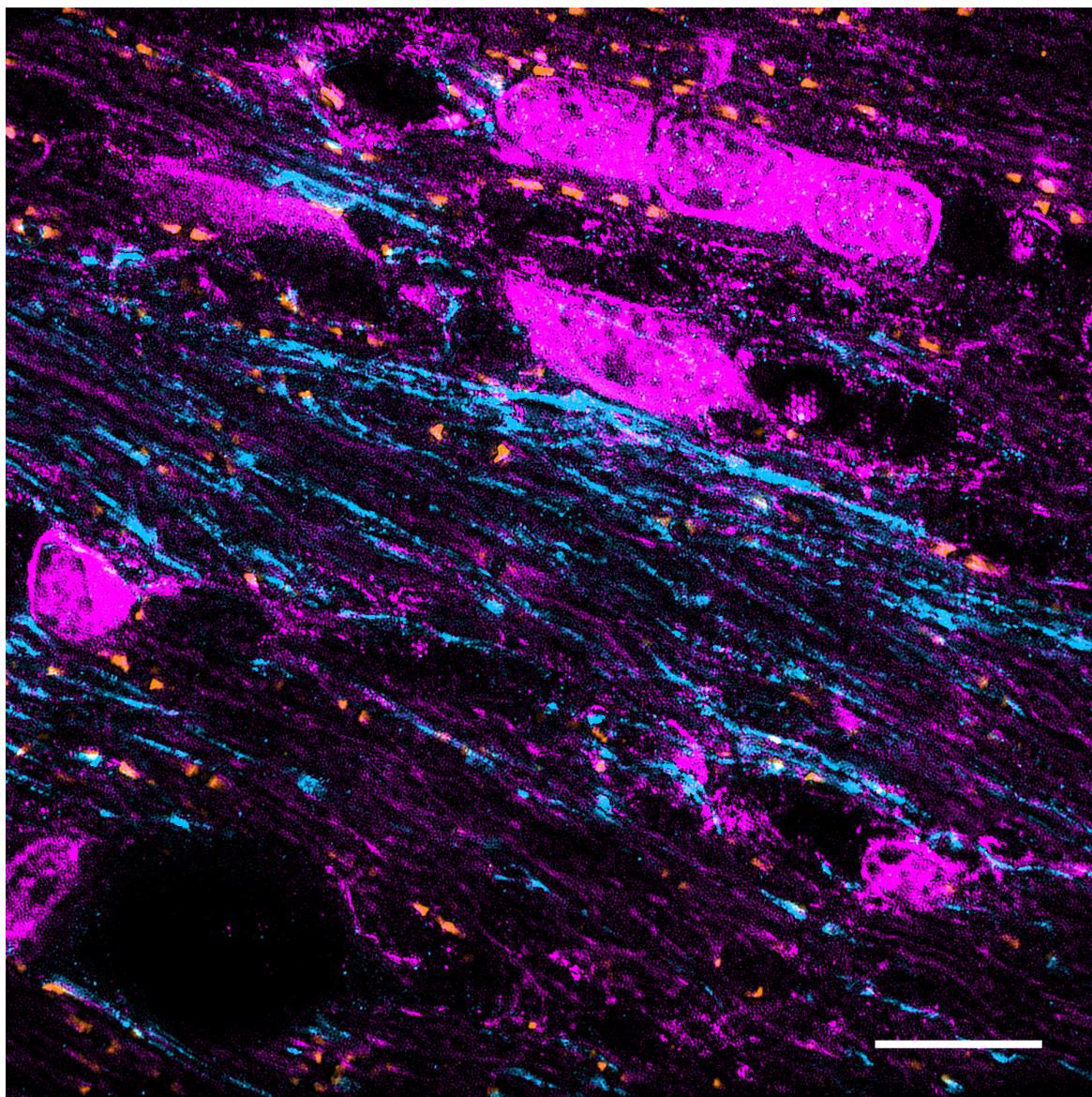


Figure 2.26: Dorsally-derived and ventrally-derived oligodendrocytes are coloured in magenta and cyan respectively. The end of the myelin sheath is coloured in orange, and can be used to measure the length of the Node of Ranvier. Optical sectioning SIM was used on 8 μm brain slices. Slices were dissected and labelled by Sarah Förster. Scalebar is 10 μm .

In contrast to the oligodendrocyte channel, the orange Node of Ranvier channel has a low density of fluorescent markers. The SIM pattern modulation contrast is therefore not degraded by out-of-focus light, leading to artefact-free SIM reconstructions with a high signal-to-noise ratio. This allows accurate length measurements of the Node of Ranvier. Since length was the important measurement for this experiment, and the oligodendrocyte type could still be determined despite the substandard reconstruction, the images were sufficient for testing the hypothesis that dorsally-derived oligodendrocytes have a different Node of Ranvier length from ventrally-derived oligodendrocytes.

Upon analysis of the data a statistically-significant difference was not calculated, suggesting that there is no difference in the Node of Ranvier length between the two oligodendrocyte types and that the movement problems observed at the organism level must arise from another characteristic of the cells, or a more complicated combination of properties. Nevertheless, this experiment has shown the versatility of the LAG SIM instrument and its ability to image tissue samples as well as cultured cells.

2.7 Conclusion

LAG SIM has been restored to its fast imaging speed of 11 super-resolution frames per second through the introduction of a Pockels cell for polarisation rotation. The Pockels cell is a robust, future-proof solution to ensure that the SIM illumination is generated with high pattern contrast. With this successfully in place, the other desirable features for LAG SIM could be addressed.

The instrument can now quickly switch between TIRF imaging, resolution-enhanced SIM imaging and SIM optimised for optical sectioning using user-friendly software which interacts seamlessly with the new hardware. This makes it suited to a wide variety of biological imaging experiments, as presented in Section 2.6.

The addition of an Optosplit, made possible by the implementation of the Pockels cell in lieu of an LCVR, further enhances LAG SIM's fast imaging capability. Rather than requiring the use of a filter wheel to switch between colour channels, 3 colours can be imaged simultaneously. This removes any mechanical movement from the system, reducing the fastest 3-channel acquisition from 570 ms to the headline 90 ms, a 5-fold improvement in frame rate.

Considerable care has been taken to ensure the microscope can be operated by non-expert users. The interface is clear and compact, and can be taught to new users in a single imaging session. This means that LAG SIM can be used unsupervised, saving time and allowing for the microscope to perform a diverse range of world-leading biological research.

Finally, the LAG SIM Fiji plugin complements the easy-to-use microscope to help users generate artefact-free images. As shown throughout this chapter, reconstruction is a complicated process involving complex mathematics and optical engineering. LAG SIM simplifies this process, so that a user can go from hypothesis to result without the need for a full-time SIM expert.

In addition to the biological investigations presented here, Chapters 4 and 5 describe in detail two successful experiments facilitated by LAG SIM. Before that, however, Chapter 3 reveals FPBioimage, a user-friendly tool for visualising and sharing 3D volumetric data, such as those obtained through optical sectioning on the LAG SIM.

Chapter 3

FPBioimage: 3D visualisation on the web

3.1 Introduction

3.1.1 Publishers require new methods of providing research data

Modern scientific experiments gather data at an ever-increasing rate. This is particularly apparent in 3D microscopy, where techniques such as confocal [28], optical-sectioning SIM, and light-sheet methods including selective plane illumination microscopy (SPIM) [29] generate terabytes of volumetric data per hour.

Reputable scientific journals require authors to make data available for reproducibility and further analysis. All *Nature* journals, for example, require “the availability of the minimal dataset that would be necessary to interpret, replicate and build upon the methods or findings reported” [125]; similarly *Science* mandate that large data sets must be deposited in public repositories and made available at the time of publication [126]. Furthermore the FAIR Guiding Principles for scientific data management and stewardship recommend “Findability, Accessibility, Interoperability, and Reusability” for all published data [127], which currently requires an urgent improvement to the infrastructure which supports this.

To give readers of a scientific publication a deeper understanding of 3D data, many authors present supplementary fly-through videos showing the data from a variety of perspectives, including from within the volume [37, 128–130]. Whilst this provides significantly more information than static 2D images can provide, the reader is nevertheless unable to interact with the data themselves. They are therefore restricted to just the perspective provided by the author, with no opportunity for personal exploration or analysis. To describe the data as open access should require a link to the volumetric data set so that readers can download it and examine it themselves.

It is perhaps understandable, however, that most published work does not contain links to full volumetric data sets; hosting and maintaining such large volumes of data online incurs prohibitively expensive costs. Repositories specifically designed for this type of data are emerging, such as the Image Data Resource (IDR) from OMERO [131], but have not been widely adopted. Many authors therefore simply make a statement that their data is “available upon reasonable request”.

After making such a request, however, receiving and examining 3D volumetric data is not trivial. The large file size means that the common practice is to temporarily upload data to a file storage repository, and share a link to the repository through an email. Once the large data file has been downloaded, specialist image analysis software must be used to render it, requiring expert skill and experience [132].

As a producer of volumetric data, I faced the data sharing issue myself when one of my collaborators moved to Chicago. We had gathered 3D data utilising optical sectioning on the LAG SIM, as described in Chapter 2, which now required their specialist biological interpretation. The issue was further complicated since the collaborator had little knowledge of volume rendering software, so could not easily examine the data.

To solve this, I searched for an online application where I could easily upload 3D data to share immediately with my collaborator. Finding that none existed, I created First Person Bioimage (FPBioimage), my own tool to fulfil these requirements.

3.1.2 Volumetric rendering reveals features inside 3D objects

The first demonstration of volumetric rendering was by Pixar in 1987 [133], a year after their acquisition from Lucasfilms by Steve Jobs for \$5 million. For the first time, a full volumetric rendering of a computerised tomography (CT) scan was displayed. Pixar was later sold to Disney in 2006 for \$7.4 billion [134, 135], demonstrating the value of advanced rendering techniques.

At SIGGRAPH '88 Alvy Ray Smith clarified the difference between volumetric rendering and traditional, geometric rendering [136]. In geometric rendering, an object is described by one or more geometrical shapes, such as a collection of spheres and cubes. To render the object, the surface of the shape is approximated by a mesh of triangular primitives, and the interaction of a virtual light ray with each triangle is calculated [137].

Volumetric rendering uses a fundamentally different technique. A virtual light ray, aimed at a pixel on the screen, progresses through a 3D array of voxels, as shown in Figure 3.1 [138]. At equally spaced steps along the ray, the value of the voxel is recorded to build up a ray profile [139].

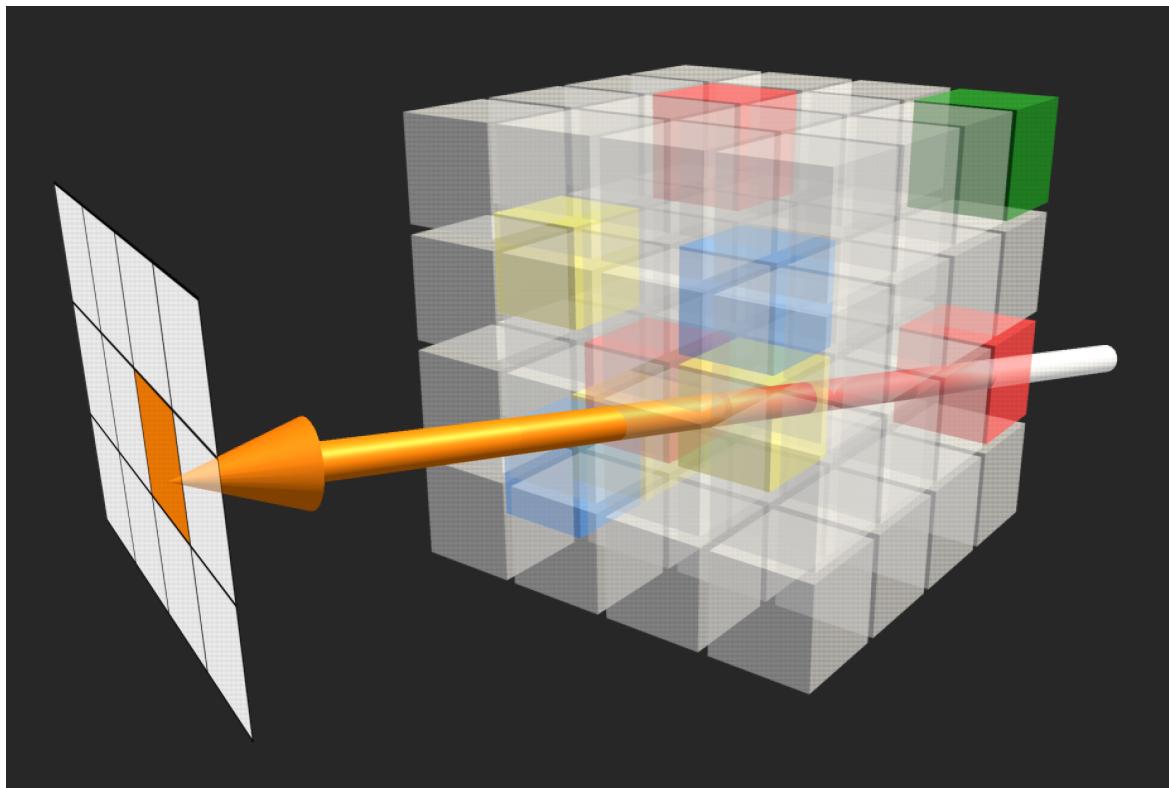


Figure 3.1: In a traditional ray marching volumetric renderer, a colourless ray starts at the back of the volume, and picks up colour and transparency at each voxel it passes through. After its journey through the volume, it paints its accumulated colour to the screen pixel it lands on.

Volumetric rendering allows detail within the volume to be visualised [140]. By tuning the opacity of each voxel, details which would be invisible in a surface rendering can be seen in context, through the outer voxels of the model.

The key drawback of volumetric rendering compared to geometric rendering is speed [141]. In the simple volumetric renderer shown in Figure 3.1, a ray is calculated for each pixel on the screen - 2×10^6 pixels for a 1080p high definition display. If each ray takes 512 samples as it passes through the volume, each rendering of the volume requires over 1×10^9 voxel calculations.

When rendering a single volumetric image, or even a video which is first rendered then presented at a later time, this is not an issue: the rays can be computed in parallel across a cluster of computers, and with some patience the images will build up. However, until recently displaying volumetric data at video rate in real time required specialised hardware optimised solely for volumetric rendering [142].

The gaming industry has led to a rapid development of devices optimised for performing thousands of simple calculations in parallel. A single graphics processing unit (GPU) contains as many as 2560 individual processing cores [143], which can be used to calculate volumetric rays simultaneously. Combined with intelligent optimisations to the rendering algorithm, GPUs found in devices ranging from laboratory workstations to smartphones now enable real-time volumetric visualisation on personal computers.

3.1.3 WebGL enables volume rendering in a web browser

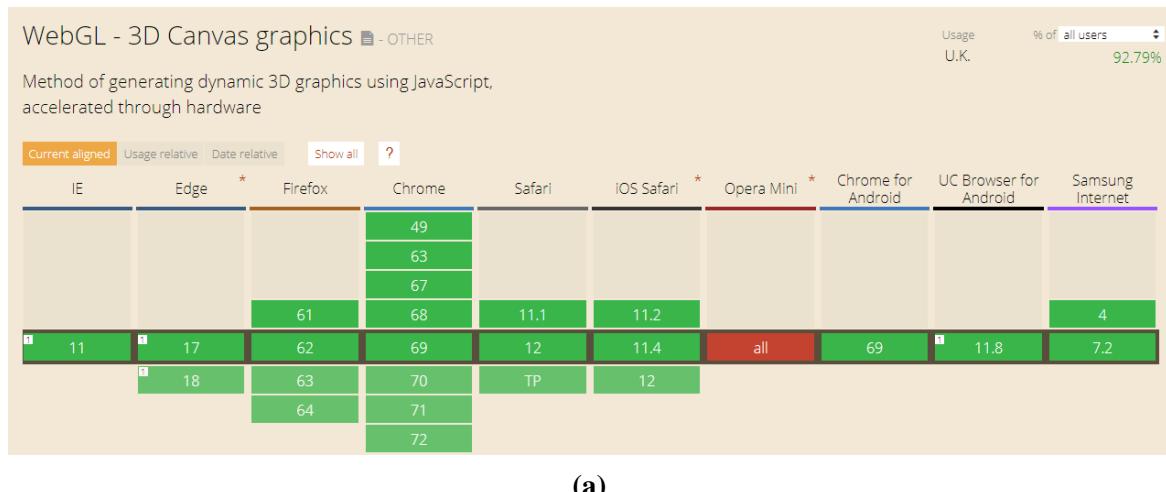
Since the development of highly parallel GPUs, a handful of free and commercial programs have been released which provide volumetric rendering capabilities. Popular software used in the microscope community includes ImageJ [98], Icy [124], and Imaris [144]; ITK-Snap [145], ParaView [146], and ImageVis3D [147] are also used in the medical community for visualising patient data [148]. These programs provide advanced offline volumetric rendering, and FPBioimage is not designed to replace them; instead, the key feature of FPBioimage is providing rendering on the web, to facilitate easy and interactive sharing of 3D data.

The internet was originally designed to display static information, limiting web browsers' access to computational resources [149]. However the relatively recent invention and widespread adoption of the Web Graphics Library (WebGL) now provides access to a GPU's processing power directly from the web browser [150]. This means that three-dimensional effects on webpages can be rendered to the screen in real time.

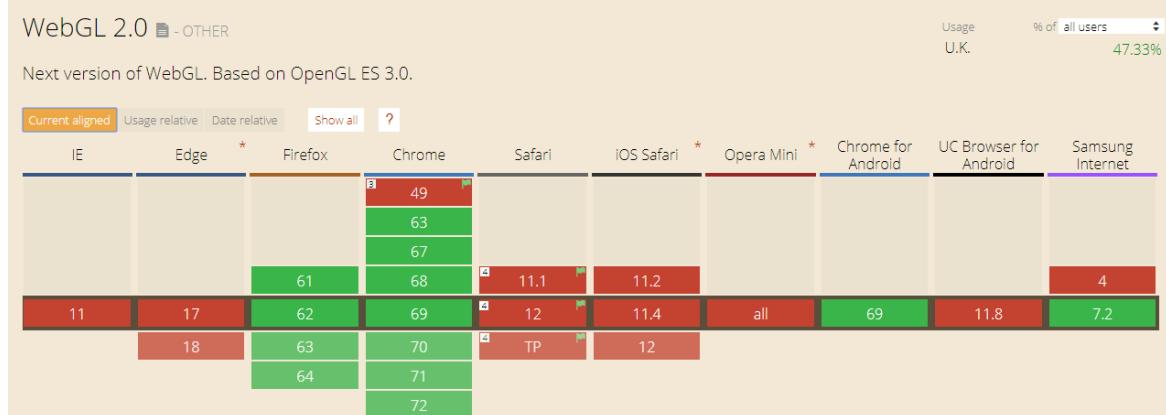
In technical terms, WebGL exposes the OpenGL ES 2.0 feature set [150]. This is a shader-based application programming interface (API). The API provides a limited number of simple functions which are optimised for running on GPUs in parallel.

Figure 3.2a shows that the support of WebGL is now widespread [151], available through HTML5 provided by all modern web browsers.

A more modern version, WebGL 2.0, exposes the OpenGL ES 3.0 API, which makes programming for 3D much easier [152]. However, at the time FPBioimage was published, support for WebGL 2.0 was highly limited. Even at this time of writing, Figure 3.2b shows



(a)



(b)

Figure 3.2: Screenshots taken from <https://caniuse.com>[151]. (a) shows that FPBioimage, which is built with WebGL, is supported by the web browsers of 93% of UK users. Although WebGL 2.0 makes 3D programming easier, (b) shows that support would drop to 47% of users.

that WebGL 2.0 is only supported by third-party browsers, and cannot be used in Microsoft’s Edge or Apple’s Safari, limiting it to just 47.33 % of UK web users [151].

To maintain maximum compatibility, volumetric rendering in FPBioimage is implemented entirely in WebGL 1.0. Various strategies for overcoming this restriction are detailed in Section 3.2.1.

3.1.4 Aims for FPBioimage

The primary goal for FPBioimage was to create an easy-to-use web application for visualising volumetric data in a web browser; therefore a requirement was set that it must run without requiring any other plugins or downloads, in order that viewing data was a simple one-click process.

After gathering user feedback, a full list of desired features was compiled.

1. Rendering options for more advanced users
2. Ability to cut volumetric data open at arbitrary angles and positions (sometimes known as *reslicing*)
3. Offer high resolution screenshots of the data
4. Ability to save a particular view as a bookmark, and share that exact view with others

For researchers sharing their own generated 3D data, more easy-to-use tools were proposed. These had to be compatible with popular image analysis programs, and facilitate fast sharing of volumetric data.

From the first prototype FPBioimage was designed to be user-friendly. The program had to be intuitive so that a user could visualise their data without any training.

3.2 Method: software design

This section describes, through the use of diagrams and code snippets, the method by which the volumetric visualisation software was developed.

The primary requirement of FPBioimage was that it must run in a web browser. As mentioned in Section 3.1.1, offline visualisation tools are already available, so the distinguishable feature of this software was the ability to share and publish volumetric data online. For fast development and future-proof compatibility with multiple devices, I chose to develop the software in a game engine called Unity [153]. This gives access to an extensive library of functions for creating dynamic 3D scenes, as well as a language for executing code on the graphics card. When the software is complete, Unity allows the designer to compile the ‘game’ into compressed HTML/JavaScript code to run in a web browser.

3.2.1 The fragment shader calculates the colour of each screen pixel efficiently

FPBioimage uses a ray marching technique similar to that presented in Section 3.1.2. The main ray marching loop implementing this is shown in lines Snippet 3.1, based on Nvidia’s “Render to 3D Texture” guide [154]. This loop is part of the ‘fragment shader’, which means it runs for every pixel rendered on the display. It is therefore the most important loop in the whole program for efficiency, and any efforts to increase the speed at which this loop executes will be amplified by the number of pixels. We will now examine the loop, and the efficiency modifications which have been made compared to traditional ray marching.

Lines 43 and 50 show that there are two different modes of rendering: a maximum intensity projection (`_RenderMode == 0`) and the volumetric ray marching mode (`_RenderMode == 1`) discussed in section 3.1.2. In the maximum intensity projection mode, the ray checks the mean colour of every voxel it passes through, and paints the colour of the voxel with the largest ‘mean color’ to the screen pixel. Note that mean colour is calculated as the sum of the red, green, and blue colour channels. Strictly we should divide by 3 to calculate the true mean colour value; however, this would require 2 extra divisions for every loop iteration for every pixel on the screen; removing this step has no effect on the comparison, but gives an improvement in frames rendered per second.

Before discussing the volumetric ray marching calculation, we should note that in this implementation the ray marches from the screen to the back of the cube - this is the opposite of traditional schemes shown in Figure 3.1. The reason for this is, of course, efficiency. As a ray progresses from a pixel through the cube, it becomes more and more opaque, denoted by the alpha value of the colour `ray_col.a`. Once the alpha value reaches 1, any voxels further along the ray’s path will not be visible, so there is no reason to look up their colour or add their contribution to the ray; line 50’s second condition is that `ray_col.a < 1.0f` to implement this behaviour. Note that in most programming languages we would break out of the loop at this point, but WebGL 1.0 loops must have a constant number of iterations.

The requirement to have a constant number of iterations also explains the very first condition in the loop on line 29. The variable `k_Steps` sets the depth resolution of the volume. We might expect the `for` loop to finish when `k < k_Steps`, but this would set the depth resolution as a constant, removing the option to increase it for powerful machines or decrease it for mobile devices. By looping for `MAX_STEPS` iterations (which is currently set to 512, but could be increased as hardware improves in the future) but performing no calculations if the loop variable `k` is greater than `k_STEPS`, we can set the depth resolution to any integer from 0 to `MAX_STEPS` and gain a performance benefit for low resolutions.

Snippet 3.1: Fragment shader code for volumetric ray marching

```

1 // Fragment shader
2 float4 frag(frag_input i) : COLOR
3 {
4     // Calculate eye ray (i) intersection with cube bounding box
5     float3 boxMin = { -0.5, -0.5, -0.5 };
6     float3 boxMax = { 0.5, 0.5, 0.5 };
7     float tNear, tFar;
8     bool hit = IntersectBox(i.ray_o, i.ray_d, boxMin, boxMax, tNear, tFar);
9     if (!hit) discard;
10    if (tNear < 0.0) tNear = 0.0;
11
12    // Calculate intersection points
13    float3 pNear = i.ray_o + i.ray_d*tNear;
14    float3 pFar = i.ray_o + i.ray_d*tFar;
15    // convert to texture space
16    pNear = pNear + 0.5;
17    pFar = pFar + 0.5;
18
19    // March along ray inside the cube, accumulating color
20    float3 ray_pos = pNear;
21    float3 ray_dir = pFar - pNear;
22    float3 ray_step = normalize(ray_dir) / _Steps;
23
24    float normalised_opacity = _Opacity * length(ray_step);
25    float4 ray_col = 0;
26    float mean_max_voxel = 0;
27
28    for(int k = 0; k < MAX_STEPS; k++){
29        if (k<_Steps){
30            // Get current position
31            ray_pos = pNear + k * ray_step;
32
33            // Clip if behind clip plane or outside unit cube
34            bool clipVoxel = dot(_ClipPlane, float4(ray_pos-0.5f, 1.0f)) > 0.0f
35            || !(ray_pos.x > 0.0f && ray_pos.y > 0.0f && ray_pos.z > 0.0f
36            && ray_pos.x < 1.0f && ray_pos.y < 1.0f && ray_pos.z < 1.0f);
37
38            if (!clipVoxel && ray_col.a < 1.0f){
39                // Get colour of the voxel we're passing through
40                float4 voxel_col = sample2D(ray_pos);
41                float mean_col = (voxel_col.r + voxel_col.g + voxel_col.b);
42
43                if (_RenderMode == 0 && mean_col > _Threshold){
44                    // Max Intensity Projection
45                    if (mean_col > mean_max_voxel){
46                        ray_col = voxel_col;
47                        mean_max_voxel = (voxel_col.r + voxel_col.g + voxel_col.b);
48                    }
49
50                } else if (_RenderMode == 1 && mean_col > _Threshold){
51                    // Volumetric ray marching with transparency
52                    voxel_col.a *= normalised_opacity;
53                    voxel_col.rgb *= voxel_col.a;
54                    ray_col += (1.0f-ray_col.a) * voxel_col;
55
56                }
57            }
58        }
59    }
60    ray_col.rgb *= _Intensity;
61    return ray_col;
62}

```

There are two additional conditions which must be met for the volume renderer to add a voxel to the ray. Firstly, shown in lines 34 to 38, the ray position must be within the unit cube and behind the clip plane. The clip plane can be used to cut open the volumetric model at any angle and position, and is described in more detail in Section 3.3. Secondly, lines 43 and 50 show the voxel's mean colour intensity must be above the value of `_Threshold` to contribute to the ray, ensuring that dark voxels around the volume are not visible.

If all the aforementioned conditions are met, then the code in lines 52 - 54 performs the volumetric ray marching step - that is, the colour of the current voxel mixes onto the ray that is passing through it. Firstly, the voxel's opacity is adjusted based on the input variable `_Opacity`, which allows the volume to be rendered with transparency. This new value of alpha is multiplied onto the voxel's colour so that more transparent voxels contribute less intensity to the ray. Finally, the adjusted voxel is added to the ray, with its contribution weighted by the current transparency of the ray.

3.2.2 A texture atlas scheme enables efficient loading of 3D data

To render volumetric images using the graphics card, the volumetric data must be loaded onto the graphics card in an appropriate format. WebGL 2.0 provides a `sample3D` class, which is able to store 3D images and return a voxel colour at a requested 3D coordinate. Unfortunately WebGL 1.0 does not have this functionality, and so an alternative scheme based on 2D textures has been devised.

The simplest way to picture volumetric imaging data is as a z-stack of 2D image slices. One option to obtain a voxel colour at a given xyz -coordinate would be to store all 2D image slices, and look up the xy value of the appropriate slice based on the z -coordinate. This is conceptually simple, but inefficient to implement on a graphics card, as each slice must be sent to the graphics card individually, which is not only slow but adds a significant memory overhead to the graphics card loading process. Instead, I devised a scheme where slices are stored on a large 2D texture atlas in a grid pattern. Then, a voxel's z -coordinate corresponds to a specific slice on the grid, and the xy components can be added to the bottom-left coordinates of that slice to obtain the xyz -coordinate voxel colour.

A restriction of WebGL 1.0 is a maximum texture size of 4096×4096 . Using the atlas technique, this would support a maximum voxel resolution of $256 \times 256 \times 256$. This was found to be inadequate for most researchers' data, so a target of $512 \times 512 \times 512$ was set. In order to meet the requirements of WebGL 1.0, this requires 8 2D texture atlases of size 4096×4096 .

The 8 texture atlases are filled according to the pseudo-code shown in Snippet 3.2. The code uses a mixture of the modulo operator (%) and division (/) to determine (a) which atlas

to place each volume image slice in, and (b) the location in that atlas where the slice should be placed. Complementary code runs in the WebGL rendering code to read a particular voxel's colour given an xyz -coordinate.

Snippet 3.2: Java code for 3D texture atlas arrangement

```

1 int sliceWidth = volumeImage.width;
2 int sliceHeight = volumeImage.height;
3 int numSlices = volumeImage.depth; // Number of z-slices
4
5 int atlasWidth; int atlasHeight;
6 int numberofAtlases = 8;
7
8 // Pad image size up to next power of 2
9 int paddedSliceWidth = ceil2(sliceWidth);
10 int paddedSliceHeight = ceil2(sliceHeight);
11
12 int xOffset = (int) Math.floor((paddedSliceWidth - sliceWidth)/2);
13 int yOffset = (int) Math.floor((paddedSliceHeight - sliceHeight)/2);
14
15 // Calculate the xy size of the atlases, making atlases as square as possible
16 int slicesPerAtlas = (int) Math.ceil((float)numSlices/(float)numberofAtlases);
17 atlasWidth = ceil2(paddedSliceWidth);
18 atlasHeight = ceil2(paddedSliceHeight * slicesPerAtlas);
19 while((atlasHeight > 2*atlasWidth) && (atlasHeight > sliceHeight)) {
20     atlasHeight /= 2;
21     atlasWidth *= 2;
22 }
23
24 // Initialise an array of 8 black 2D textures to become atlases
25 Image[] atlasArray = new Image[numberofAtlases];
26 for (int i=0; i<numberofAtlases; i++){
27     atlasArray[i] = new Image(atlasWidth, atlasHeight, black);
28 }
29
30 // Fill out atlases with image slices
31 int slicesPerRow = (int) Math.floor((float)atlasWidth/(float)paddedSliceWidth);
32 for (int i=0; i<numSlices; i++){
33     int atlasNumber = (int)((float)i % (float)numberofAtlases);
34     int locationIndex = (int) Math.floor((float)i/(float)numberofAtlases);
35
36     // Get slice
37     Image slice = volumeImage.getSlice(i);
38
39     // Put slice into atlas at the correct position
40     int xStartPixel = (int)((float)locationIndex % (float)slicesPerRow) * paddedSliceWidth +
41         xOffset;
41     int yStartPixel = (int) Math.floor((float)locationIndex / (float)slicesPerRow) *
42         paddedSliceHeight + yOffset;
43
44     // The following line should be uncommented for coordinate systems that start top-left
45     //yStartPixel = atlasHeight - yStartPixel - paddedSliceHeight + 2*yOffset;
46
47     copySubImageToAtlas(sliceTexture, atlasArray[atlasNumber], xStartPixel, yStartPixel);
47 }
```

3.2.3 Movement provides personal exploration of the data

FPBioimage would not be a particularly exciting real-time volumetric renderer if the user only had one view of the data. By utilising Unity library functions, implementing movement is a straightforward and resource-efficient process.

Firstly, as with all offline volumetric rendering programs, the 3D model can be rotated about its x -, y - and z -axes to view a different part of the data. This is achieved in Unity by mapping the volume rendering shader described in Section 3.2.1 onto a cube. The cube is then linked to both the arrow keys and to the mouse to provide intuitive manipulation of the volume, as shown in Snippet 3.3. For data whose combination of xyz resolution and xyz voxel size describes a cuboid, the xyz scale of the rendering cube can be adjusted to ensure distortion-free visualisation.

Snippet 3.3: C# code using built-in Unity functions for rotating the volumetric data

```

1 void Update () {
2     // This code is attached to the rendering cube, and called once per frame
3
4     // Keyboard arrow control
5     Vector3 vertRotAxis = transform.InverseTransformDirection(camera.TransformDirection(
6         Vector3.right)).normalized;
7
8     float horizontalRot = Input.GetAxis("H2");
9     float verticalRot = Input.GetAxis("V2");
10
11    transform.Rotate(vertRotAxis, verticalRot, Space.Self);
12    transform.Rotate (Vector3.up, horizontalRot, Space.World);
13
14    // Mouse click control
15    if(Input.GetMouseButton(0)){
16        // Rotate with the mouse
17        Vector3 vertRotAxis = transform.InverseTransformDirection(camera.TransformDirection(
18            Vector3.right)).normalized;
19
20        float horizontalRot = Input.GetAxis("Mouse X");
21        float verticalRot = Input.GetAxis("Mouse Y");
22
23        transform.Rotate(vertRotAxis, verticalRot, Space.Self);
24        transform.Rotate (Vector3.up, horizontalRot, Space.World);
25    }
}

```

The ‘FP’ in FPBioimage stands for ‘First Person,’ and refers to the control the user has over the scene’s camera. Rather than just viewing the rotatable render cube from one perspective, as is common in offline volumetric renderers [98, 124, 144], FPBioimage also allows the user to change the position and look-direction of the camera. This is implemented as shown in Snippet 3.4. The $H1$ axis is mapped to the A and D keys, and the $Z1$ axis to the W and S keys. This, combined with the functionality of using the mouse to change the camera’s look-direction, will make control of the camera very familiar to anyone who has

played first-person perspective computer games. Moving the camera allows the user to create in real time the type of dynamic fly-bys sometimes seen in publication videos, providing a highly immersive visualisation of the data.

Snippet 3.4: C# code using built-in Unity functions for moving the camera in a first-person manner

```

1 void Update(){
2     // Translation
3     transform.position += Input.GetAxis ("V1") * Camera.main.transform.up * Time.deltaTime;
4     transform.position += Input.GetAxis ("H1") * Camera.main.transform.right * Time.deltaTime
5         ;
6     transform.position += Input.GetAxis ("Z1") * Camera.main.transform.forward * Time.
7         deltaTime;
8     transform.position += Input.GetAxis ("Mouse ScrollWheel") * Camera.main.transform.forward
9         * Time.deltaTime;
10
11    // Rotation
12    if (!Input.GetMouseButton(1)) {
13        // This is the 'first person' mouse mode
14        rotationX += (Input.GetAxis ("Mouse X")) * mouseSpeed;
15        rotationY += (Input.GetAxis ("Mouse Y")) * mouseSpeed;
16        rotationY = ClampAngle (rotationY, minimumY, maximumY); // Stops camera doing backflips
17            !
18
19        Quaternion xQuaternion = Quaternion.AngleAxis (rotationX, Vector3.up);
20        Quaternion yQuaternion = Quaternion.AngleAxis (rotationY, -Vector3.right);
21
22        transform.localRotation = startRotation * xQuaternion * yQuaternion;
23    }
24
25    // Camera movement with mouse
26    if (Input.GetMouseButton (1)) {
27        transform.position -= Input.GetAxis("Mouse X") * Camera.main.transform.right * Time.
28            deltaTime / Screen.width;
29        transform.position -= Input.GetAxis("Mouse Y") * Camera.main.transform.up * Time.
30            deltaTime / Screen.height;
31    }
32}

```

3.2.4 Bookmarking is implemented with a state machine to share perspectives across the world

FPBioimage implements a bookmarking function to save a particular view with an annotation. The bookmarked view can then be shared with others as a web link, or restored on the same computer at a later time.

To implement the bookmark saving and restoring functionality requires several distinct actions, the order of which depends on the user's input. An elegant method for reliably moving the user between these stages is a state machine. The state machine controlling the bookmark

saving and restoration procedures is shown in Figure 3.3, and can be described as follows:

- State 0** The default state, where all states eventually return to. Nothing regarding bookmarking is shown on the screen. Pressing ‘B’ on the keyboard will start the bookmark creation process; pressing a number key will start the bookmark restoration process. Any other key does not change the state.
- State 1** The bookmarking textbox pops up requesting the user to enter a number key to save the new bookmark in that position. Pressing a number key will take the user to the next stage of bookmark creation; pressing ‘Escape’ will close the textbox and return to the default state.
- State 2** The user can enter a text annotation into the text box. Pressing ‘Escape’ will immediately close the textbox and return to the default state; pressing ‘Return’ (also known as ‘Enter’) will save the bookmark as described below and move to State 3. Note that an empty annotation is accepted.
- State 3** This state displays information or annotation text. If arriving from State 2, it will show a confirmation message; if arriving from State 4, it will show the annotation text saved alongside the bookmark being restored.
- State 4** This state simply checks that the volumetric model has loaded, to ensure no error is thrown if trying to restore a bookmark on an uninitialised volume. The state machine continues to wait in State 4 while the volume is loading, so any requested bookmark will be loaded immediately when the volume is ready.

Internally, the bookmarks are encoded as JavaScript Object Notation (JSON) strings, converted to a base-64 string, and then saved to the web browser’s ‘Local Storage’. Saving to Local Storage requires the C# program to interface with the web browser through JavaScript. Unity provides the function `ExternalEval(string jscode)` to complete this task, where `jscode` is a string containing the JavaScript code to be executed. Once the bookmark has been encoded, it is saved to the web browser as shown in lines 1 and 2 of Snippet 3.5. The bookmark is saved to Local Storage with a unique prefix followed by the bookmark’s number, so that it can be restored at a later time.

Snippet 3.5: C#-JavaScript interface code for interacting with the browser’s local storage to save and share bookmarks

```

1 string evalMe = "localStorage.setItem('fpb-' + variables.fpbJSON.uniqueName + "-bookmark" +
  bookmarkNumber + ',', '" + bookmarkString64ToSave + "')";  

2 Application.ExternalEval (evalMe);  

3 evalMe = "history.replaceState(null, null, '?b=' + bookmarkString64ToSave + '')";  

4 Application.ExternalEval (evalMe);

```

Lines 3 and 4 of Snippet 3.5 shows how the base-64 bookmark string is appended to the base url in the web browser using the JavaScript function `replaceState`. The user can then

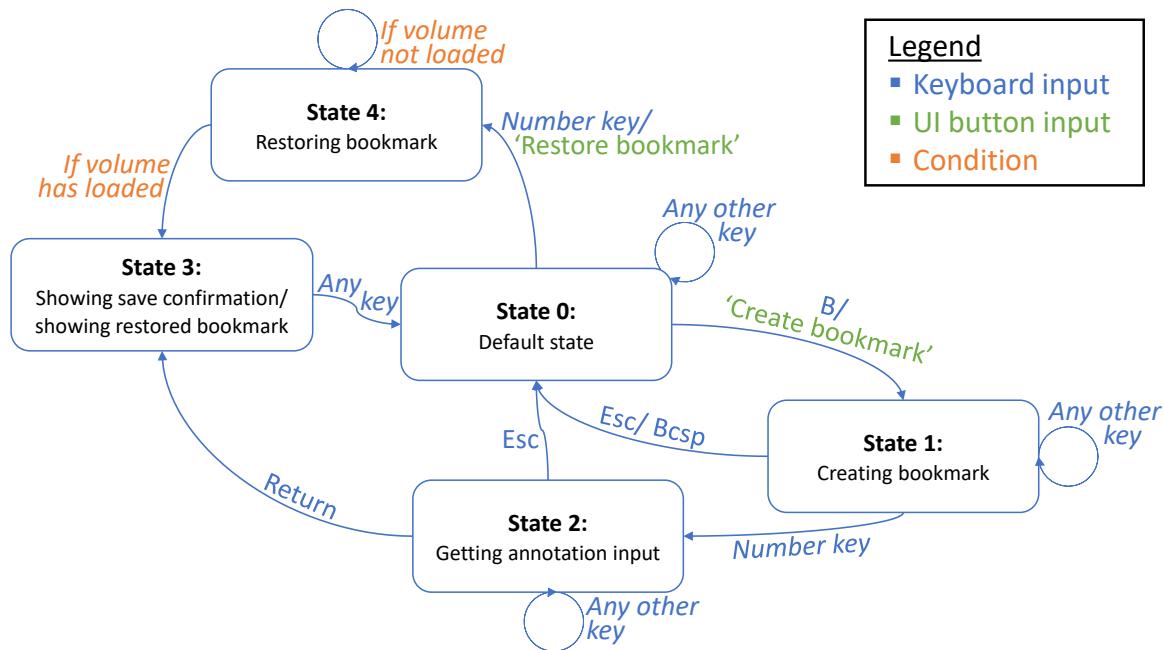


Figure 3.3: The state machine shown controls the creation and restoration of bookmarks.

simply copy the URL and send it to a friend or colleague to open on their own computer. When the volume is being loaded, the bookmarking C# script checks the URL of the webpage it is running on, and if it finds a "?b=" in the URL string it will begin restoring the bookmark. This puts the bookmarking state machine into State 4, ready to show the bookmark and annotation text as soon as the volumetric model has loaded. In this way users can easily share specific views of the data across the globe.

3.2.5 Screenshots can be captured at a higher resolution than the display

FPBioimage provides the capture of high resolution screenshots, shown in Snippet 3.6.

When the `takeScreenshot` function is called, a new `RenderTarget` is created to save the current view to. The size of this can be set either by the current pixel size of the main camera or by user input. In this way screenshots at a higher resolution than the current view can be created. This is advantageous, as the user can view the volumetric data in a small window for fast performance, but then take a screenshot at a higher resolution for presentation and publication.

The new render texture is converted to a PNG-encoded byte stream, and then offered to the user as a download through the JavaScript library `download.js`. `download.js` provides

a single function `download(data, strFileName, strMimeType)`, which is called from Unity using the C#-JavaScript interface function `ExternalCall`. This is shown in line 32 of Snippet 3.6.

Snippet 3.6: C# code for capturing a high-resolution screenshot and presenting it to the user as a download

```

1 public void takeScreenshot(bool hiRes){
2     if (hiRes) {
3         bool wOK = int.TryParse (wRes.text, out ssWidth);
4         bool hOK = int.TryParse (hRes.text, out ssHeight);
5
6         if (!wOK || ssWidth < 1) {
7             ssWidth = 1920;
8         }
9         if (!hOK || ssWidth < 1) {
10            ssWidth = 1080;
11        }
12    } else {
13        ssWidth = Camera.main.pixelWidth;
14        ssWidth = Camera.main.pixelHeight;
15    }
16
17    // Create screenshot-sized render texture to render to
18    RenderTexture rt = new RenderTexture(ssWidth, ssWidth, 24);
19    mainCamera.targetTexture = rt;
20    Texture2D snapShot = new Texture2D(ssWidth, ssWidth, TextureFormat.RGB24, false);
21    mainCamera.Render();
22    RenderTexture.active = rt;
23    snapShot.ReadPixels(new Rect(0, 0, ssWidth, ssWidth), 0, 0);
24    mainCamera.targetTexture = null;
25    RenderTexture.active = null;
26    Destroy(rt);
27
28    // Encode snapshot to PNG byte stream
29    byte[] bytes = snapShot.EncodeToPNG ();
30
31    // Offer screenshot as download
32    Application.ExternalCall ("download", "data:image/png;base64," + System.Convert.
33      ToBase64String (bytes), "Screenshot " + System.DateTime.Now.ToString ("yyyy-MM-dd")
      + " at " + System.DateTime.Now.ToString ("HH.mm.ss") + ".png", "image/png");
}

```

3.3 Results and discussion

The FPBioimage software, designed in Unity as detailed in Section 3.2, was compiled into JavaScript to run in a web browser. In this section, all screenshots and tests were performed in Google Chrome version 58 on the Windows 10 operating system. However FPBioimage also runs flawlessly on any modern web browser, including Mozilla Firefox, Microsoft Edge, and Apple Safari, and also mobile versions of these browsers.

3.3.1 An intuitive interface requires no user training

Great care has been taken to ensure FPBioimage is a user-friendly piece of software. Many offline software packages already exist for rendering volumetric data; FPBioimage's unique selling point is that it runs in a web browser, to make sharing data with colleagues across the world a simple process. For this reason it is important that a non-expert user can simply click on a link and use the software without any training. At the same time as being immediately accessible to new users, the software also contains a number of advanced rendering options for those wishing to extract more information from the data.

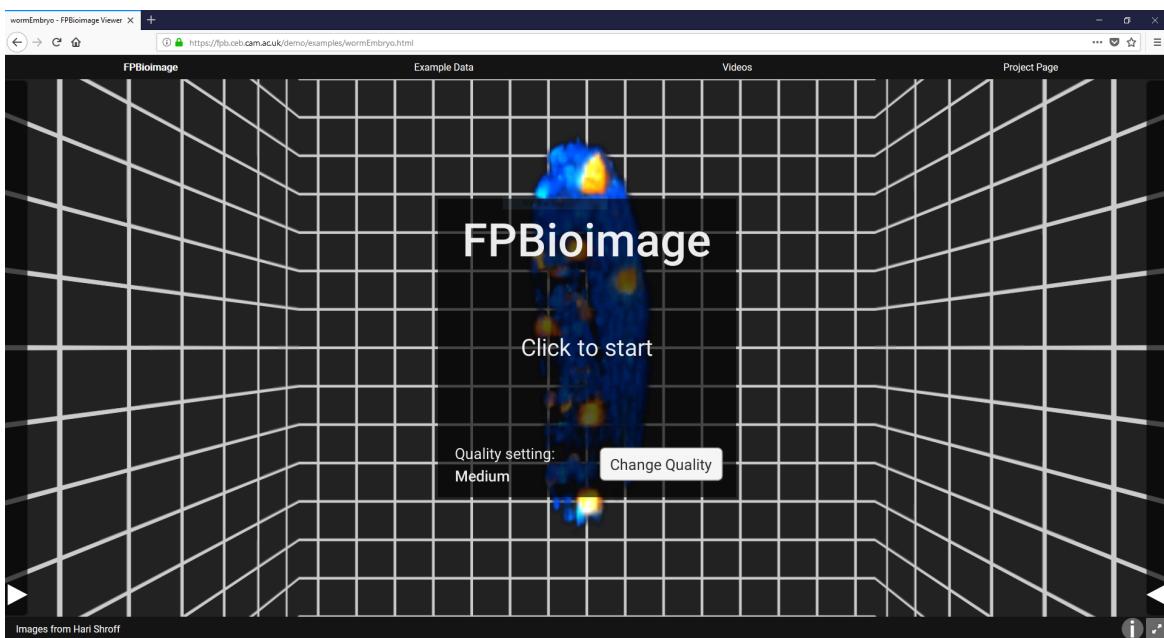


Figure 3.4: When a user opens an FPBioimage webpage, the data will be downloaded and the user will be presented with the simple interface shown. The model can be interacted with intuitively by clicking and dragging with the mouse, or finger on a touchscreen. The data shows a *C. elegans* embryo captured by single plane illumination microscopy (SPIM), provided by Hari Shroff [155].

When opening up a web link with FPBioimage on the page, the user is presented with the view shown in Figure 3.4. Interacting with the volumetric model is responsive and intuitive. Using the mouse, the user can rotate the volume by clicking and dragging, or on a touch screen by dragging with their finger. Alternatively the arrow keys can be used on a keyboard. Pinching on a touch screen, or using the scroll wheel on a mouse, zooms in and out of the model. This creates a natural interaction with the data that requires no additional instruction.

To access more advanced rendering features, Figure 3.5 shows a panel on the right-hand side of the screen which slides in. The user is presented with 3 rendering parameters: *Opacity*,

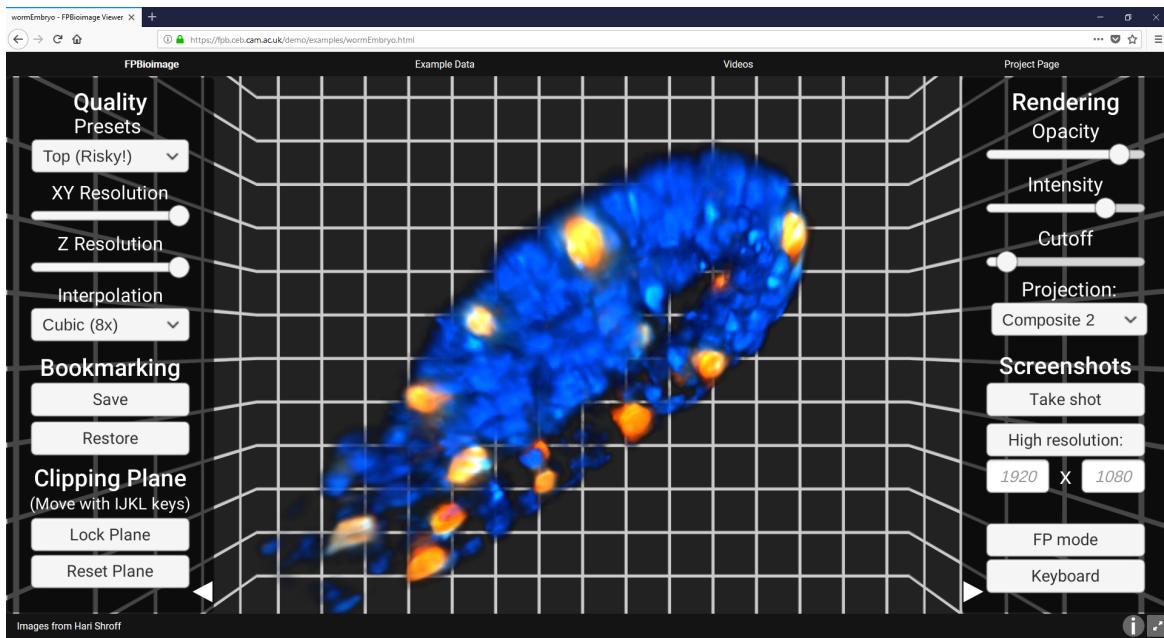


Figure 3.5: Clicking on the side panel arrows slides in advanced rendering options, clipping plane options, quality options, and bookmark and sharing options. Data from [155].

Intensity, and *Cutoff*. Matching these parameters up to the volume rendering code described in Section 3.2.2, it is easy to see how each parameter modifies the presentation of the data.

Lowering the *Opacity* value means that each voxel contributes less alpha to the ray marching through it; consequently, the volumetric data becomes more transparent. This makes features inside the model visible through the outer voxels, as shown in Figure 3.6. Combined with rotating the volume, this facilitates visualisation of how different structures are arranged in 3D space, providing answers that are not possible to obtain in any other way.

The *Intensity* value acts globally on the data at the end of the ray marching process. This simply increases or decreases the overall brightness of the model, to reveal contrast either at the bright end or the dark end of voxel values.

The *Cutoff* slider sets the value below which voxels are not rendered. Most microscopy images, as well as images from other modalities such as MRI, use black voxels around the imaging data to represent a lack of light or objects. Setting a small positive value for *Cutoff* ensures that the volume does not appear as a black box when *Opacity* is set high. Larger values for *Cutoff* can also be used to highlight bright objects which lie within the volume.

The next element in the right-hand UI tab is the *Projection* dropdown menu. As described in Section 3.2, FPBioimage provides several volumetric visualisation rendering pipelines. These include *Max. Intensity*, which renders a maximum intensity projection from any angle; and *Composite*, which performs volumetric ray tracing with support for transparency.

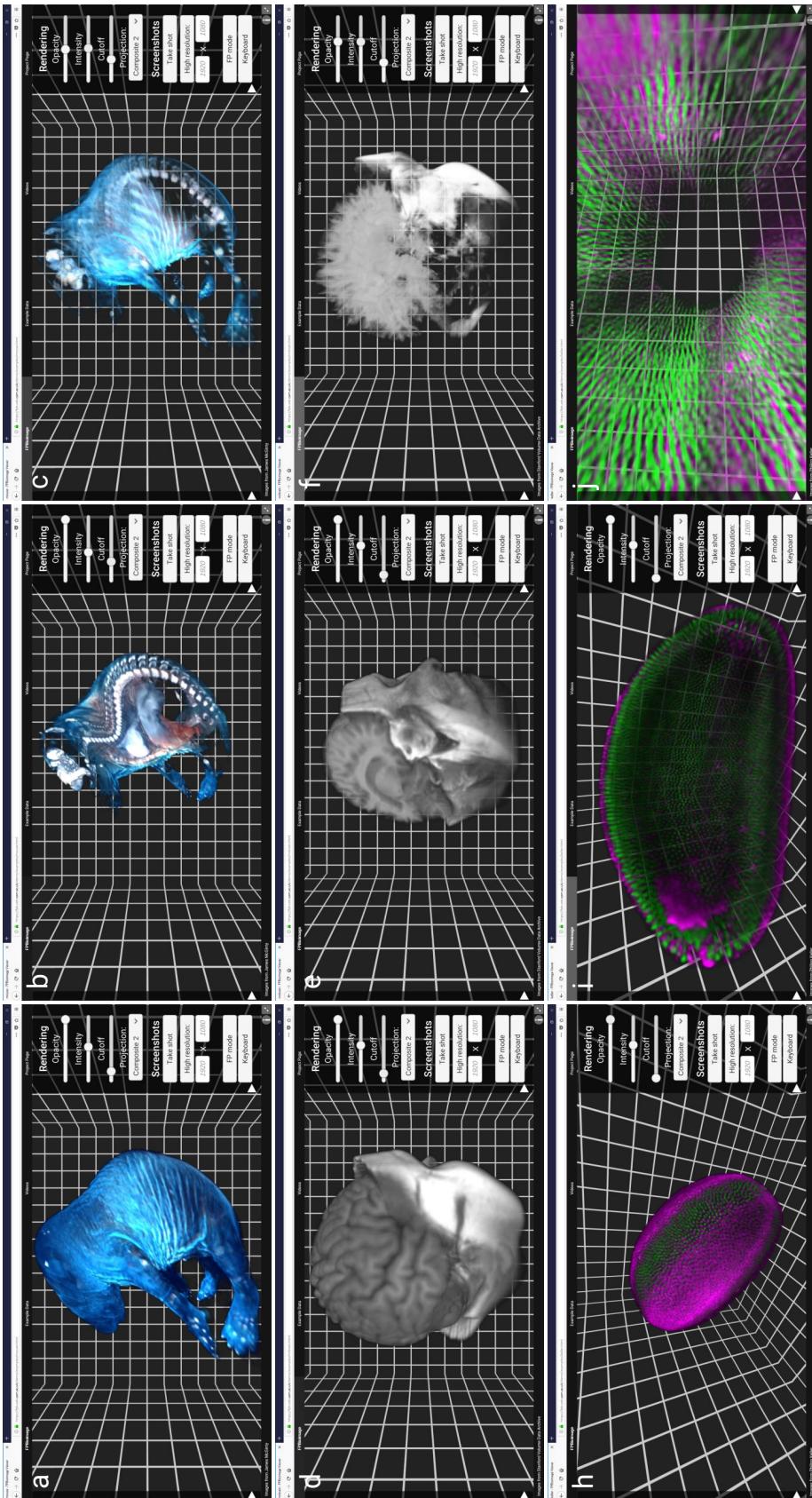


Figure 3.6: (a–c): Four-colour OPT data of a human head. (d–f): MRI data of a mouse embryo. (g–i): two-colour light-sheet microscopy data of a drosophila embryo. FPBioimage’s built-in transparency feature has been used to render (c) and (f), revealing data within the volume; the cutting tool has been used for (c), (f), (h) and (i), removing outer voxels from the data; and h shows a bookmark in creation, which can be shared with other users. Reproduced from [156]. Raw data provided by: (a–c), J. McGinty [30]; (d–f), Stanford Volume Data Archive [157]; (g–i), P. Keller [130].

Furthermore the *Iso-surface* projection reveals only voxels within a certain percentage of the *Cutoff* voxel value. Screenshots of the various *Projection* methods rendering a mouse embryo head are shown in Figure 3.7.

Further down the right-hand UI tab are two UI buttons for capturing screenshots. The first, *Take shot*, captures a screenshot at the current resolution of the drawing canvas. Perhaps more usefully, the *High resolution* screenshot button captures a screenshot at the resolution defined in the two text boxes below. This allows the user to capture screenshots at a higher resolution than their display, in case high-resolution images are required for presentation or publication. The advantage of this is exemplified in Figure 3.8.

The final two buttons in the right-hand UI tab are toggle buttons, labelled *FP mode* and *Keyboard*. *FP mode* toggles on the First Person mouse control that gives FPBioimage its full name. In this mode the user can use the mouse to look around, in a way that is familiar to anyone who has played a first-person perspective game. This can provide dramatic and immersive flights through the data controlled by the user, revealing much more information than can be provided on a pre-recorded video.

At all times (except when creating a bookmark), the user can use the WASD keys to move around, as shown when the *Keyboard* button is active. Pressing the *Keyboard* button brings up a full list of controls, presented in Figure 3.9. Every rendering option is available through a keyboard shortcut, for complete mouse-free operation.

The left-hand UI tab reveals yet more parameters which can be controlled by a more advanced user.

The first of these is *Quality*. Volume rendering in real time is a graphics-intensive operation which has only recently become possible, and there is a clear trade off between quality and performance on most computers or mobile devices. A dropdown list shows a selection of preset options with suggestions of the type of device they are suitable for. For more advanced users, the individual quality settings *XY Resolution*, *Z Resolution*, and *Interpolation* can be adjusted independently. The *Quality* settings are saved to the web browser's Local Storage upon modification and are automatically restored on the user's next visit to an FPBioimage webpage. The performance-quality trade-off is discussed in more detail in Section 3.3.2.

The bookmarking functionality can be accessed from the UI using the *Save* and *Restore* buttons. This begins the state-machine process shown in Figure 3.3. Creation of a bookmark is shown in Figure 3.10a. When a bookmark is created or restored, the URL in the browser bar changes to encode the bookmark as a web link. This link can then be shared with other users, by email, text, or on social media, who will see exactly the same view when

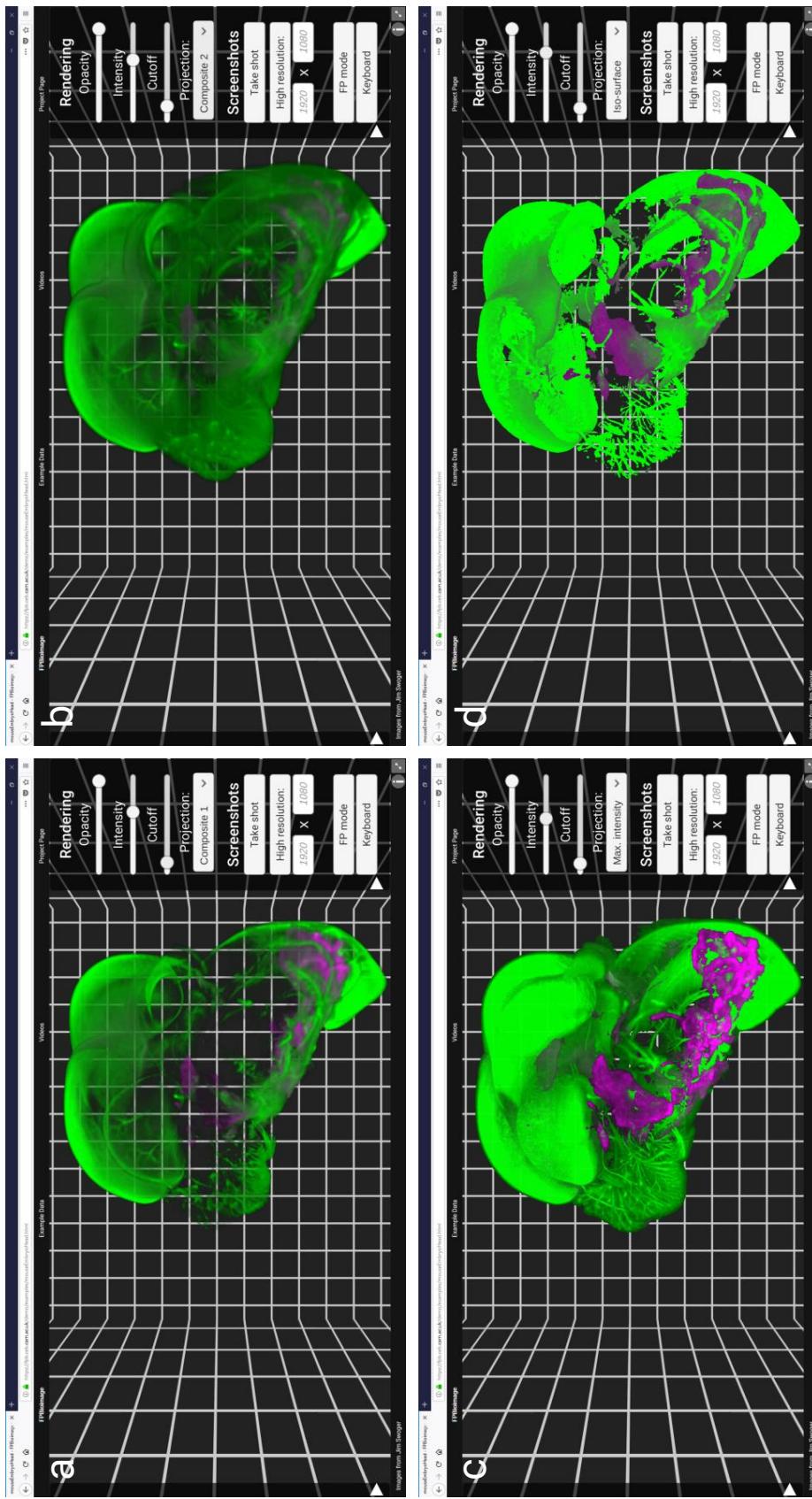


Figure 3.7: First Person Bioimage offers 4 different types of volumetric projection. (a) shows a maximum intensity projection, (b) a volumetric ray marching projection, (c) a variation on the volumetric ray marching projection, with alpha calculated from RGB voxel values, and (d) an iso-surface projection. Data is of a mouse embryo head captured with ‘OPTiSPIM’, provided by Jim Swoyer [158].

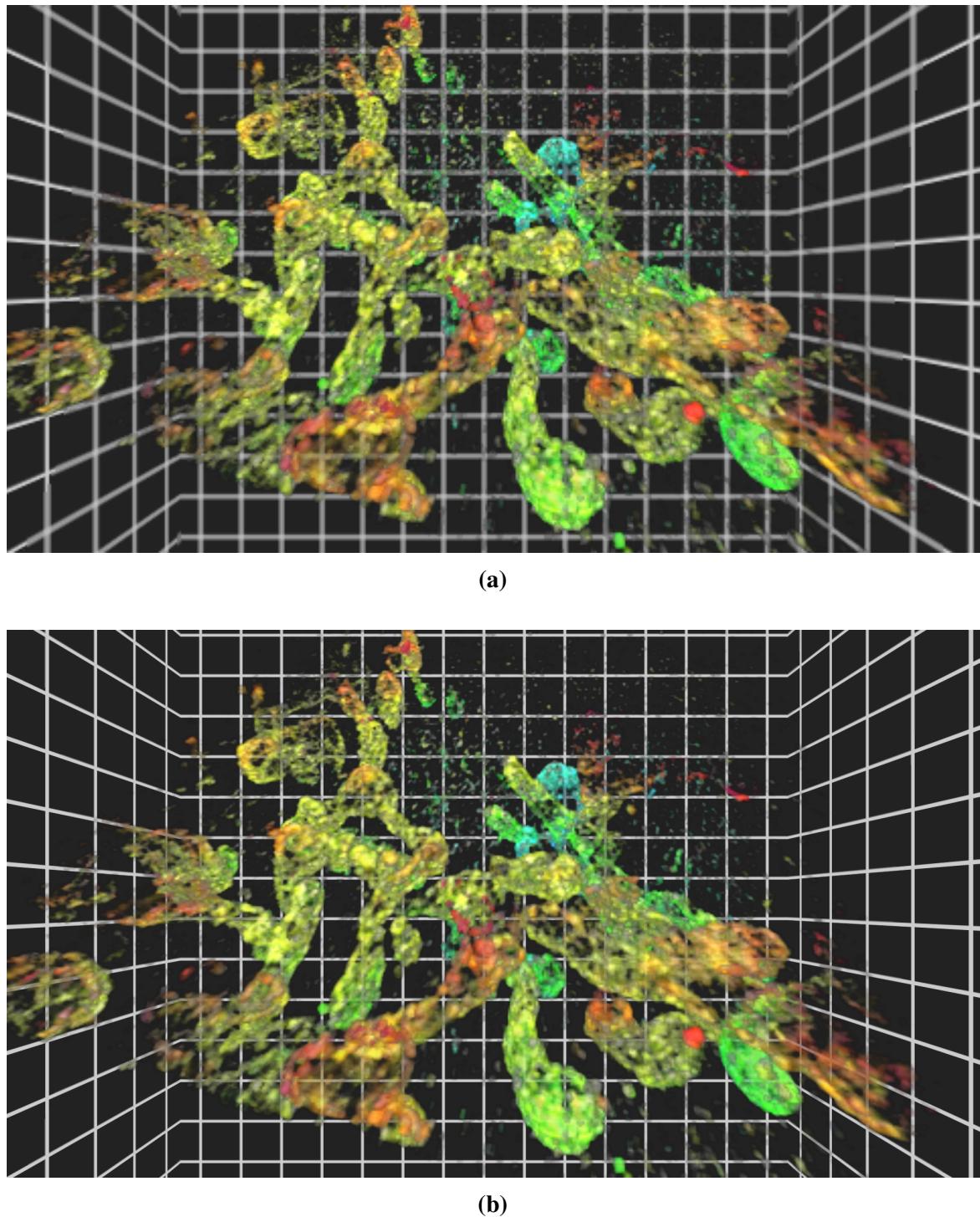


Figure 3.8: High resolution screenshots can be created even on a small display. (a) shows a screenshot at the native 768x1024 display resolution; (b) shows a 1080p screenshot of the same view, created using the *High resolution* button. Images are of mitochondria captured with a whole-cell 4Pi single-molecule switching nanoscopy (W-4PiSMSN) microscope, provided by Joerg Bewersdorf [37].

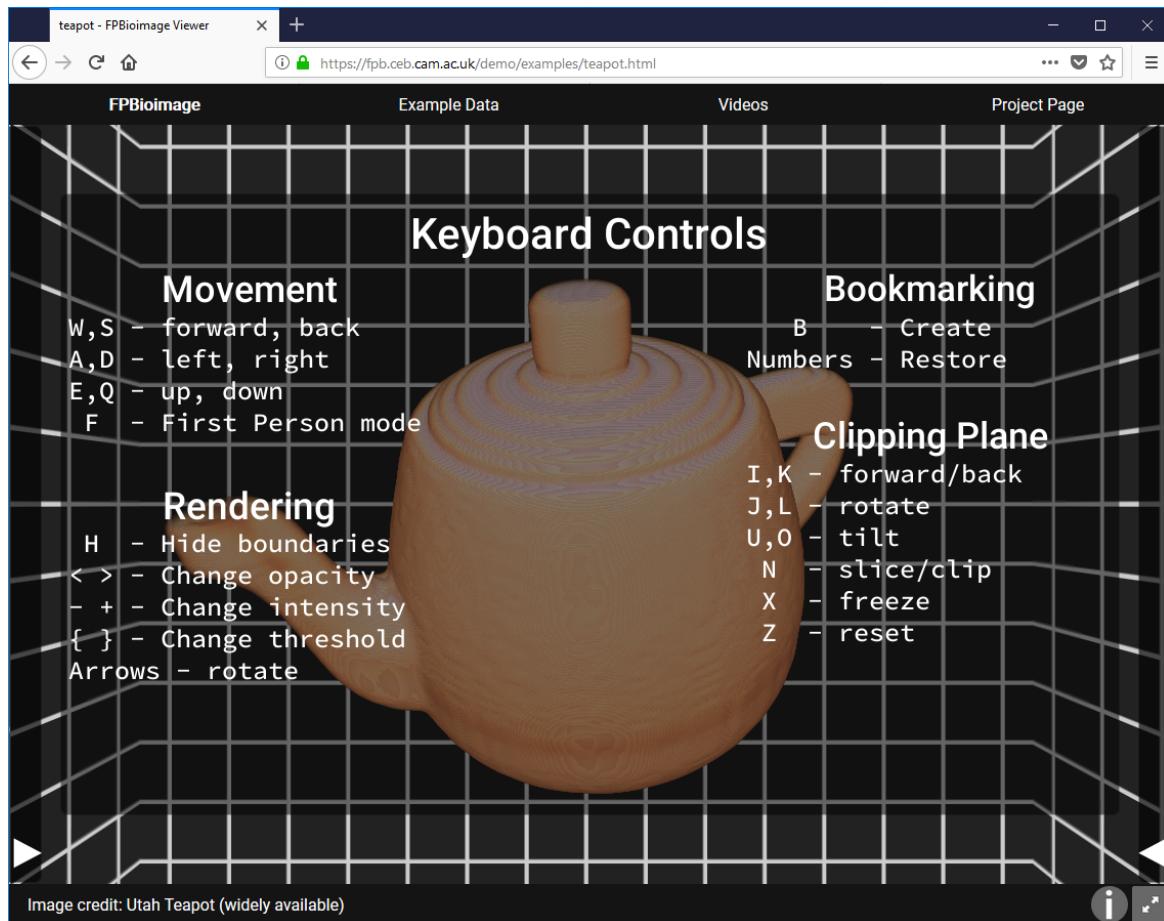


Figure 3.9: All rendering options in FPBioimage can be controlled by the keyboard, for mouse-free operation. The teapot in the background is the widely available Utah teapot created by Martin Newell [159].

FPBioimage loads. This is demonstrated in Figure 3.10b, where the shared bookmark is opened on another computer, with a different browser and operating system.

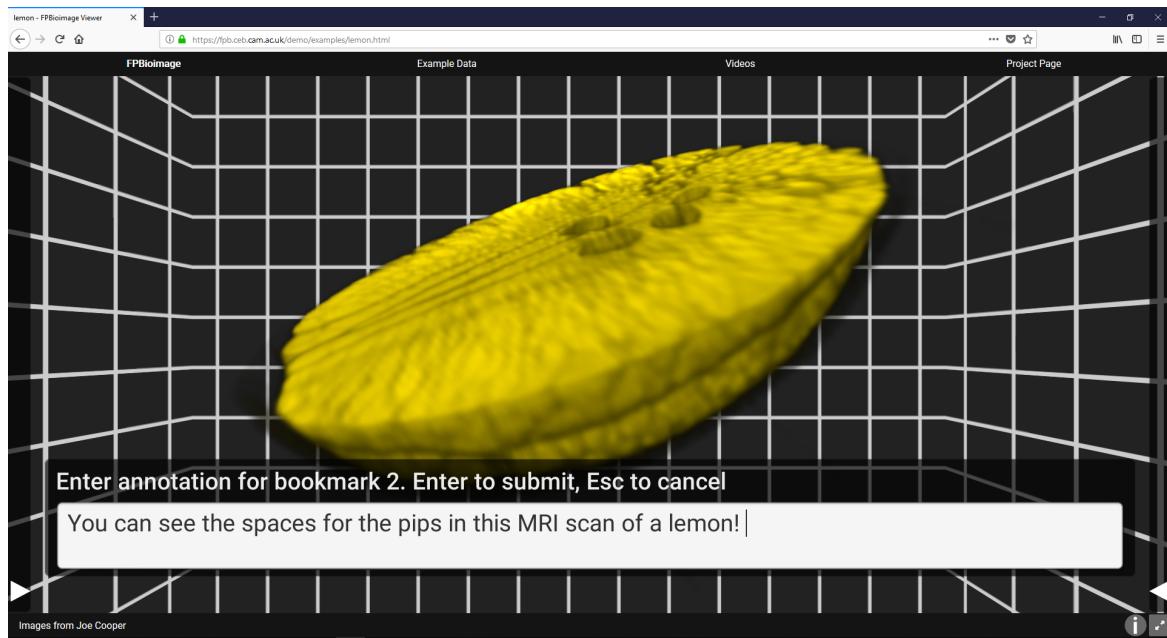
Finally, at the bottom of the left-hand UI tab are the UI buttons for the clipping plane. Mentioned briefly in Section 3.2.2, the clipping plane prevents voxels that are behind it from being rendered, revealing voxels inside the volumetric data that are otherwise obscured. The clipping plane's location is controlled with the IJKL keys, where I and K are used to move the clipping plane forward and back, and J and L rotate it around to cut the data at an arbitrary angle. Once an interesting cut has been made, the clipping plane can be locked in place with respect to the model by pressing the *Lock Plane* button. With the clipping plane locked, the user can move the camera or the model to view the cut volumetric data from a different angle. Use of the clipping plane can therefore provide insightful perspectives of the volumetric data that are not possible to obtain with any other method, as shown in Figures 3.6b, e, h, and i.

3.3.2 Measuring performance demonstrates compatibility with consumer-grade computers

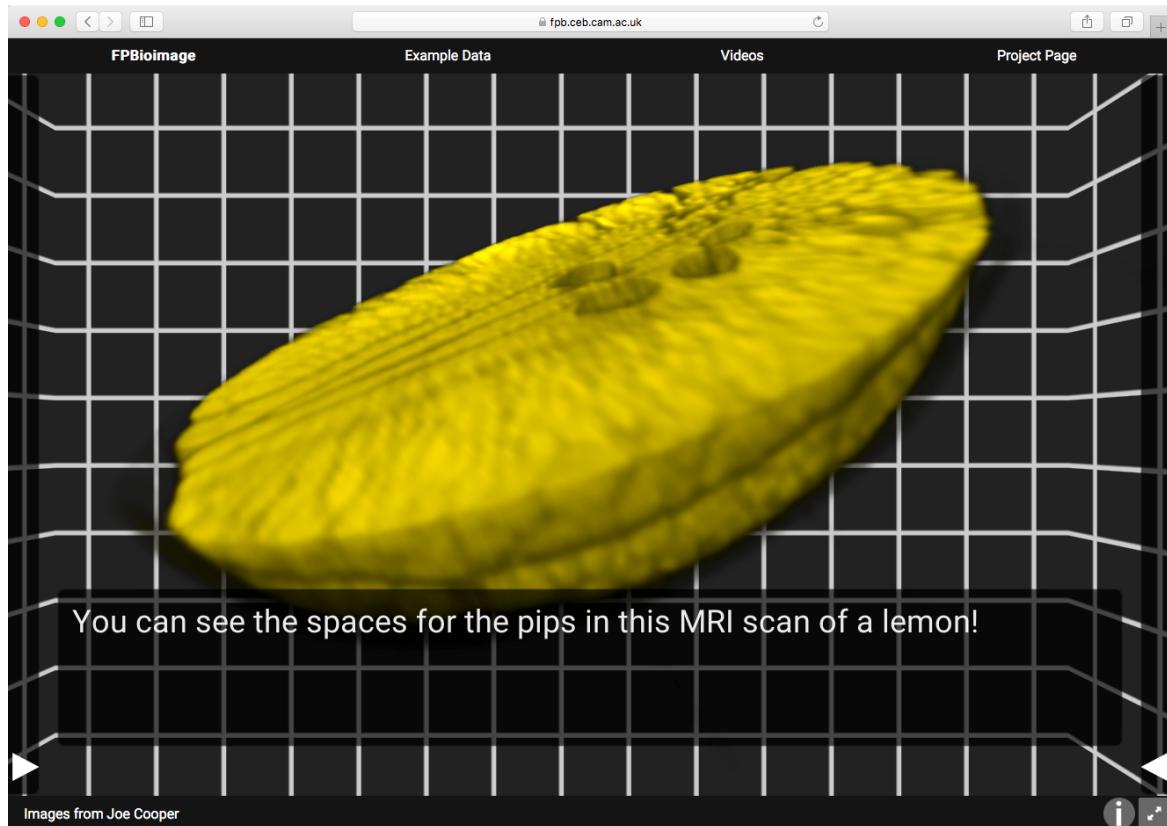
Until recently, volume rendering was not possible in real time, and would be calculated for a specific camera perspective in advance of presentation to render either a single image or video. In 2000, Ray and Silver produced the Race II Engine, designed with dedicated hardware to render $512 \times 512 \times 512$ volumetric images in real time. This was achieved by realising the volume rendering algorithm described in Section 3.1.2 with physical devices, with hundreds of calculations running in parallel to compute all rays at once [160]. Today, top-end graphics cards are able to perform the same calculations in real time on a graphics processing unit (GPU), a highly parallelised device optimised for computing lighting models [161].

A primary aim of FPBioimage was that it should be user-friendly, with a particular focus on ease-of-use for non-expert users viewing data. It was therefore expected that a majority of users would not have access to expensive, high-performance graphics cards required for rendering volumetric data in real time at full resolution. Furthermore FPBioimage was designed so that it could be run on any device - Statista statistics show that over 40% of web traffic in August 2018 was from mobile browsers [162]. In order to support the full range of users, it was important to devise a scheme where the user could reduce the quality of the volume rendering in exchange for an increase in performance.

In the computer games industry, “performance” essentially means how many frames are rendered to the screen per second - assuming the graphics card is working at full capacity. If the graphics rendering does not require the full capacity of the graphics card, performance can be measured as the percentage of the graphics card’s compute capabilities in use. In the



(a)



(b)

Figure 3.10: (a) shows creation of a bookmark, with an annotation, in the Firefox web browser running on Windows 10. The bookmark can be restored on the same computer, or shared as a URL to another user. (b) shows the same bookmarked view in Apple Safari running on macOS 10.13 High Sierra. The MRI scan of a lemon was provided by Joe Cooper.

tests presented in this chapter, performance is given in both frames per second (FPS) and graphics card usage percentage, measured using Windows Task Manager. Screenshots of the software under test are shown in Figure 3.11.

“Quality” is harder to define. For a scene of simple 2D geometry quality could be measured by the XY-resolution of the screen, and we would expect performance to decrease as XY-resolution increases. For our volumetric data, however, we also have a Z-resolution, defined by the number of _Steps in Snippet 3.1. Furthermore visual quality is improved if, instead of just taking the nearest voxel to the tip of the marching ray, interpolation is performed between either the 2- or 8-closest voxels.

The three quality parameters can be adjusted as user inputs, or preset values can be selected. A table of quality against performance for different combinations of the three quality metrics, and for the preset values, is shown in Table 3.1.

As expected, the table shows that as the quality settings are increased in value, performance decreases. To choose the most useful preset values, the visual impact of the quality settings was compared to the performance trade-off.

As XY-resolution is increased, performance decreases dramatically. This is because the number of pixels being rendered increases with the square of the XY-resolution number. It is interesting that XY-resolution does not have such a dramatic effect on the perceived visual quality, as seen in Figure 3.8. Since there is little visual quality gain for a relatively large performance impact, XY-resolution is kept to moderate values even for the higher quality presets.

Compared to the performance impact of XY-resolution, Z-resolution has a relatively small impact. From a Z-resolution of 256 steps to 768 steps the GPU use increases linearly. As shown in Figure 3.11, however, Z-resolution has a considerable impact on visual quality of 3D images. For an equivalent impact on performance, an increase in Z-resolution gives better visual quality than an increase in XY-resolution; therefore providing high Z-resolution is prioritised for the presets.

Interpolation of voxels has a big impact on visual quality, removing the staircase effect shown in Figure 3.11, but also has a big impact on performance. When $2\times$ interpolation is used, two voxels from the 2-nearest Z-planes to the marching ray’s current position are averaged (weighted by distance) to smooth the appearance of the volume. This requires an extra texture lookup at every z-step for every screen pixel, causing a significant performance impact particularly at high XY- or Z-resolutions. In $8\times$ interpolation, the 8-nearest voxels to the marching ray’s position are used, requiring 8 texture lookups at every z-step. The extra texture lookups explains the high impact on performance of interpolation.

Table 3.1: The table shows how performance, measured by FPS and GPU percentage use, is affected by the FPBioimage quality settings. This test was performed on a midrange laptop, which cost £580 in 2017, with the following specification: 2-core i5-7200U @2.50 GHz, 8 GB RAM, NVIDIA GeForce GTX 1050 with 4 GB graphics RAM.

Name	Device	Preset			Performance	
		XY-res.	Z-res.	Interp.	FPS	GPU use
XY-resolution test						
-		256	256	1X	60	14%
-		768	256	1X	60	41%
-		1024	256	1X	60	63%
-		2048	256	1X	41	88%
Z-resolution test						
-		450	64	1×	60	15%
-		450	256	1×	60	21%
-		450	512	1×	60	30%
-		450	768	1×	60	39%
Interpolation test						
-		768	256	1×	60	40%
-		768	256	2×	60	49%
-		768	256	8×	54	83%
Preset values						
Very low	Old laptop	256	64	1×	60	11%
Low	Mobile	400	100	1×	60	15%
Medium	Laptop	450	150	1×	60	18%
High	Desktop	768	350	1×	60	49%
Very high	Graphics card	1024	500	2×	38	87%
Top	3D workstation	2048	768	8×	9	97%

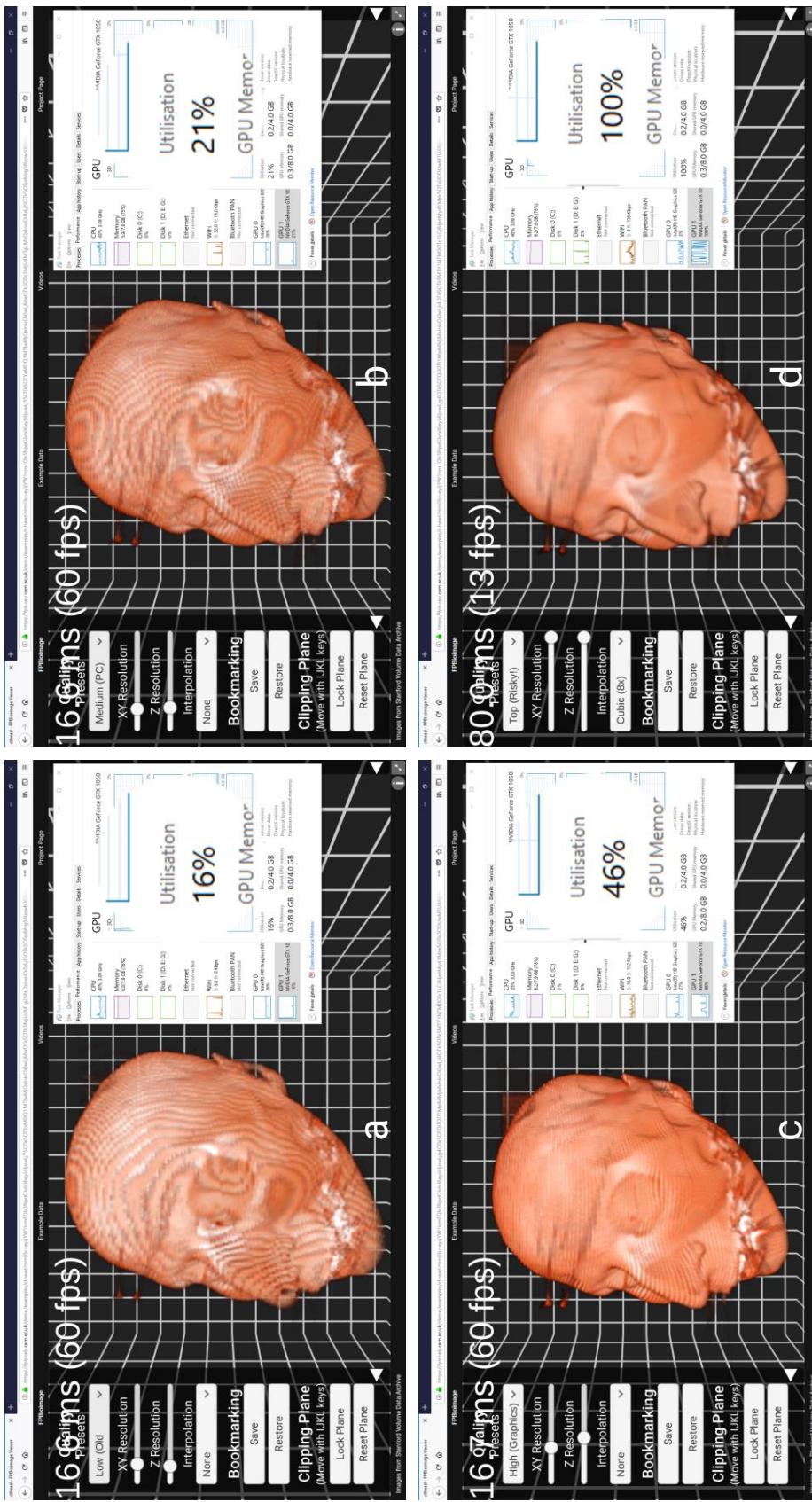


Figure 3.11: The figure shows FPBioimage rendering with different quality presets: (a) Low; (b) Medium; (c) High; and (d) Top. The highest quality levels are designed for use on a high-end desktop computer; lower quality settings are provided to allow FPBioimage to run on mobile devices. Performance was measured using frames per second (FPS) and graphics card percentage usage, shown on the screenshots. Results from these tests are presented in Table 3.1. The CT scan of a human head is from the Stanford Volumetric Data Archive [157].

The ‘Top’ preset setting sets all quality variables to maximum, and the user is warned that this setting is ‘risky’ because it can crash the web browser! This is an impractical setting on most personal machines - Table 3.1 shows that the GPU is working at full capacity, and only rendering 9 FPS. However, on dedicated graphics workstations where the GPU is able to render 60 FPS at top quality a fly-through video for publication can be recorded in real-time.

3.4 FPBioimage Suite

FPBioimage is a volumetric renderer designed to be used in a web browser. However, to assist users uploading their 3D data to share on the internet, plugins are available for the open-source image analysis platforms ImageJ, FIJI, and Icy. Furthermore, thanks to its open-source nature and an easy-to-use JSON interface, other groups have been able to integrate FPBioimage into their own software packages, for example OMERO-FPBioimage [163]. Finally, for users who want to view volumetric data on their mobile device, I have created dedicated Android and iOS apps, which support virtual reality for truly immersive interaction with the data. Together with FPBioimage, this collection of applications has been branded FPBioimage Suite.

3.4.1 Plugins are available for popular image analysis programs

After the initial release of FPBioimage [156], uploading data to share with others was a convoluted process. Users had to download a copy of FPBioimage, install it on their own personal web server, upload a stack of PNG files with a specific naming convention, edit some JavaScript on a template webpage, and finally share the web link. This is not an issue for any web developer, but was beyond the ability of many scientists who gather volumetric data. Whilst many users enjoyed viewing data with FPBioimage, very few used it to share their own data.

In order to simplify the upload process, I created the FPBioimage Helper plugin for the popular open-source image analysis programs ImageJ, FIJI, and Icy. These programs all support volumetric image data in a wide variety of formats thanks to OME Bioformats [164]. Users simply have to open their 3D data in the program, click one button, and the plugin will output a web link that the researcher can share with others.

By default, FPBioimage Helper will upload the data to FPBioimage-Host, a Firebase website hosted on Google Cloud Platform to make it easier for researchers to share volumetric image data [165]. Users log in to FPBioimage-Host with a Google account to manage their uploads, and can immediately copy the URL to allow colleagues and collaborators to view the

data from anywhere in the world. If the user would prefer to host their data on a private web server, for example to implement password protection for datasets or use their own website template, they can choose not to upload the data but simply save the formatted images and website to their local machine for their own purpose.

Using the plugin, rather than manually uploading PNG image slices, also provides faster download times for those viewing the data. Instead of saving every plane as a separate PNG file, the plugin arranges images into texture atlases as described in Section 3.2.2. A line of JSON ('atlasMode': true) tells FPBioimage to expect atlases, rather than individual image slices, reducing the number of downloads to 8. This reduces loading times in four ways: the overhead time associated with each download is reduced from number-of-slices to just 8; overall download size is smaller, because the PNG compression algorithm is able to exploit more informational redundancy [166, ch. 1]; the 8 texture atlases can be downloaded in parallel; and FPBioimage can skip the step of arranging slices into texture atlases, since it can send the downloaded atlases directly to the graphics card.

The plugin is distributed as an “update site” through the FIJI updater, and also through the Icy online plugin repository. This makes it easy to keep up-to-date, so that users get the access to the latest features with as little effort as possible.

3.4.2 OMERO-FPBioimage integrates volume rendering with organised image repositories

OMERO is a web-based platform for managing scientific images [167]. Created by OME, the same team behind Bioformats, it integrates with a number of other software packages to help users upload their imaging data to a remote server, so that it can be accessed from anywhere. Furthermore a web interface makes it easy for OMERO users to share their data with others, allowing them to perform further analysis or even meta-studies on independent image sets [131].

After reading about FPBioimage in *Nature Photonics*, an OMERO user noticed that the two programs have similar goals, and suggested integrating FPBioimage into OMERO as the default volume renderer [168]. Thanks to FPBioimage’s clear documentation and simple JSON interface, the OMERO team were able to incorporate an FPBioimage viewer into their software before any contact with me.

If a user opens a volumetric image in OMERO, an option appears in the viewer options to open in FPBioimage, as shown in Figure 3.12. When selecting this option, the OMERO server quickly creates a static webpage with an FPBioimage viewer and the relevant JSON

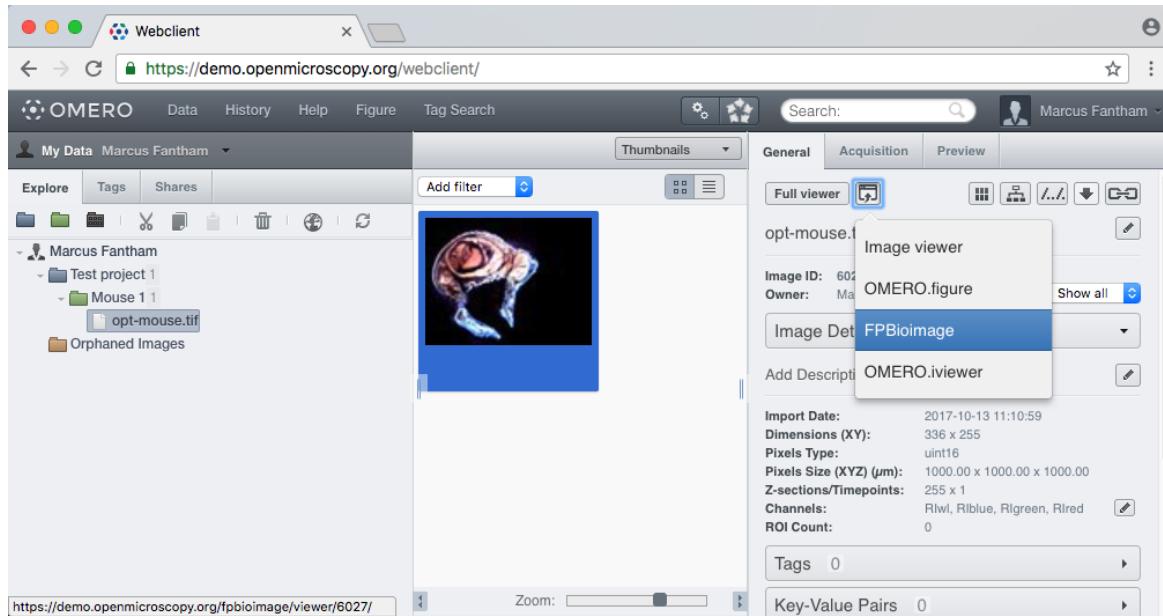


Figure 3.12: The screenshot shows OMERO.web, which uses FPBioimage as its default volumetric image viewer.

to load the data into FPBioimage. Close collaboration with the OMERO team means that OMERO now sends texture atlases directly to FPBioimage, reducing loading time for users.

The viewer has proved popular with OMERO users, finding diverse uses from plant sciences to pathology [169]. It has been adopted as OMERO’s default volumetric viewer, and is featured prominently on their webpage [163].

3.4.3 Mobile apps bring volumetric rendering to smartphones with virtual reality support

Since FPBioimage is a web-based application, it will run on any device with a web browser. This includes mobile devices, such as iPhones and Android smartphones, as well as tablet devices such as an iPad. However, mobile browsers have limited access to the device’s computational resources, causing long loading times and poor performance, characterised by a low frame rate.

To allow FPBioimage to run on mobiles with full performance, native apps were compiled for the Android and iOS operating systems, which are available through their respective app stores. Since mobiles usually only run one app at a time, the app has full access to all the computational and graphics processing capability of the device.

For the mobile apps, a universal linking scheme has been devised. This means that if a user visits a website with an FPBioimage viewer on it, they will get the option to open

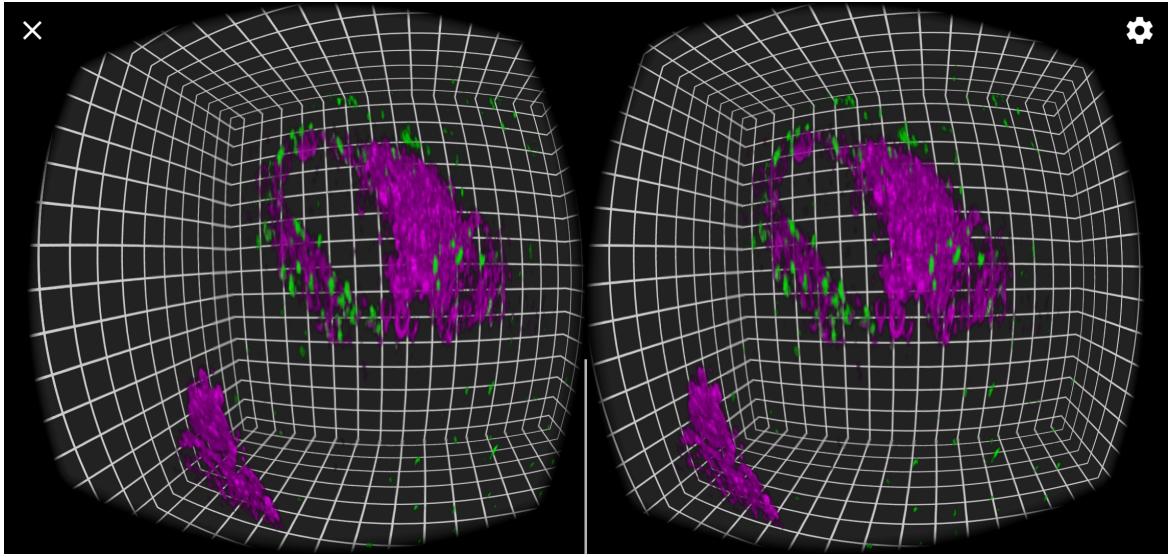


Figure 3.13: FPBioimage has native apps for Android and iOS. The screenshot shows a split-screen stereo view/, which can be used in conjunction with Google Cardboard [170] to support virtual reality. The data shows MOF (green) taken up by HeLa cells labelled with a lysosomal marker (magenta), captured on the LAG SIM [107].

the data in a native app on their phone instead of in the mobile browser. Since the mobile app uses the same rendering algorithm (presented in Section 3.2.2) as the online app, the visualisation will be exactly the same.

As part of the port to mobile, a new focus was put on FPBioimage touch controls. Users can intuitively rotate the model by touching and dragging. Users can also zoom in and out by a two-finger pinch, and move around the model using two fingers and dragging. As with the web app, the quality can be changed to accommodate a range of mobile chipsets, and the usual rendering options of opacity, intensity and cutoff are available, along with the various projection methods.

Running on mobile brings one more advantage, thanks to the low-cost virtual reality (VR) device Google Cardboard [170]. This is a VR viewer - originally made of cardboard but now available in more robust plastic models [171] - which utilises a smartphone's high-resolution screen and precise accelerometer to provide a low-cost VR experience. The user simply has to open their data in the mobile app, click the "Start VR" button, and put their phone into their Cardboard viewer. This option allows users to view their data in VR, providing a truly immersive experience and giving a perspective of the data not possible through any other means.

3.5 Conclusion

As an online volumetric rendering program, FPBioimage has a wide range of uses which have not been possible before.

As a volumetric renderer, FPBioimage offers a number of volumetric rendering modes found in most offline volume visualisation software, including composite ray marching and maximum intensity projections. However all other volume rendering programs have a complicated array of settings for changing transparency and colour maps, and no option for adjusting quality. With a focus on ease-of-use from its conception, FPBioimage provides an interface that anyone can use, even non-expert users.

Furthermore, because FPBioimage is designed to run on the web, or on devices with limited computational resources, the rendering algorithms have been optimised such that full-quality volume rendering can be performed in real time. This real-time performance has not been seen in other software. Indeed, the ease-of-use coupled with high performance have generated many requests that FPBioimage could be used as the default volumetric viewer in other image analysis software [172].

In combination with the ImageJ, FIJI, or Icy plugins, scientists collecting volume data can share it with anyone in the world in one click. Data is uploaded to a public repository hosted on Amazon Web Services, at a permanent link, or can be uploaded to a private server for password protection. This allows researchers to get unprecedented instant feedback from international collaborators.

Researchers can embed FPBioimage in a presentation by utilising a slideshow plugin such as LiveSlides [173], as shown in Figure 3.14. This allows an interactive flight through the data whilst talking about it, bringing a new dynamic to presentations and allowing detailed responses to questions which are not possible with a pre-recorded video.

Finally, for sharing volumetric data with the wider community FPBioimage can be used as a publication tool. A whole set of 3D images can be uploaded, such that readers can explore for themselves the findings of an experiment [107]. It is hoped that journals will adopt FPBioimage as a dynamic viewer for online publications, so that readers can interact with data alongside the text.

FPBioimage makes sharing of volumetric data with anyone in the world a one-click operation, and introduces a new paradigm for publishing 3D image data.

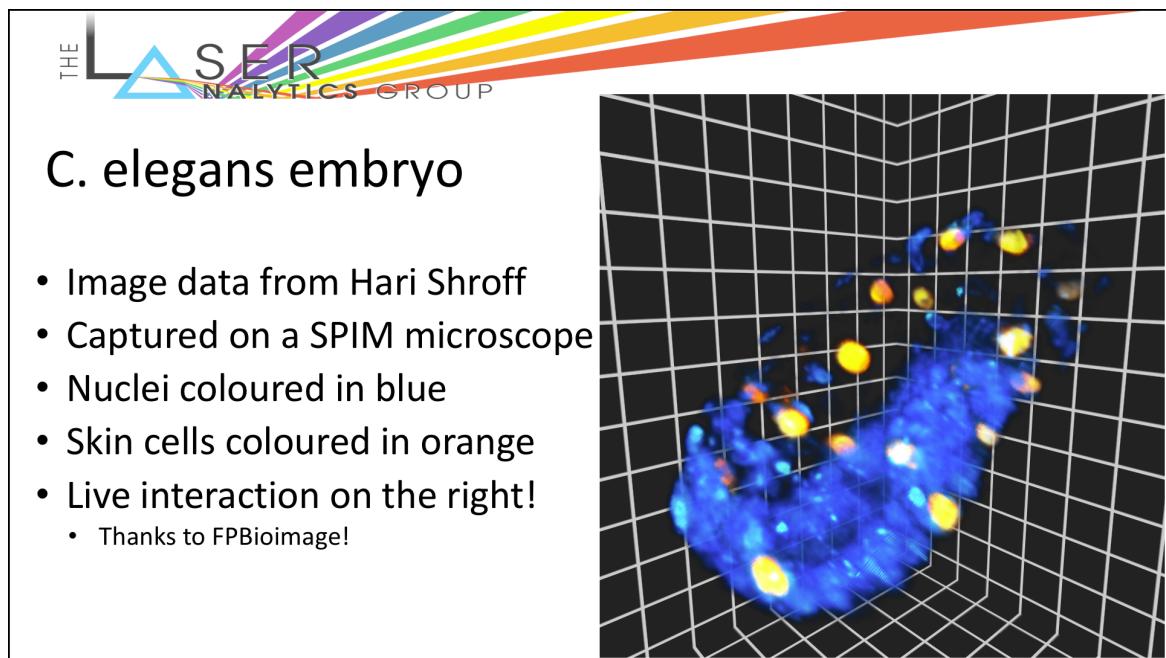


Figure 3.14: Using the Powerpoint plugin LiveSlides, FPBioimage can be embedded in a presentation, bringing an interactive dynamic to explaining volumetric data. Data from [155].

Chapter 4

MOFs: Metal organic frameworks for drug delivery

4.1 Introduction

4.1.1 siRNA for cancer treatment

When the latest statistics find that over half of people born in the United Kingdom since 1960 will develop some form of cancer in their lifetime [174], it hardly needs stating that cancer has affected almost everyone in the country in some way. Any successful efforts to discover new treatments for this set of diseases therefore have a significant impact reducing suffering and extending life expectancy [175]. In this chapter, I describe a novel method of delivering therapeutic drugs to cells, utilising the LAG SIM for imaging and FPBioimage for 3D visualisation and analysis.

The availability of the human genome sequence [176, 177] provides new opportunities for using DNA and other nucleic acids to target specific genes for therapeutic treatment. One such form of gene manipulation is RNA interference (RNAi) [178, 179]. RNAi prevents the expression of certain genes by degrading messenger RNA (mRNA) after transcription, thus preventing translation [180]. RNAi can be performed with small interfering RNA (siRNA) [181, 182], which silences mRNA molecules by cleavage of the mRNA strand into two pieces.

From a molecular biology perspective cancer is a highly complex disease, requiring the accumulation of around five independent genetic events for a malignant tumour to develop [55, *ch. 9*], [183, *ch. 6*]. However if a specific gene, or set of genes, can be linked to causing cancer in a given cell type, then appropriate siRNA can be synthesised to silence that gene [184]. The siRNA will be 100% complementary to the gene sequence it is targeting,

providing very high specificity. This means that, assuming the siRNA is inside the cell, it can prevent the cancer without other side effects. If the siRNA is inside a cell where the mRNA does not transcribe the cancer-causing gene, the siRNA will have nothing to bind to, and will simply degrade [184].

There are two difficulties with this potential treatment method. Firstly, discovery of cancer-causing oncogenes is not trivial; although some cancers, such as retinoblastoma, are caused by mutations in just the *RB1* gene [185], usually cancer is a multistep process where a series of genetic mutations collectively reduce the function of tumor-suppressor genes [55, ch. 9][186, ch. 24]. Nevertheless, it has now been 2 decades since the invention of siRNA, with billions of dollars of research funding invested by pharmaceutical companies [187]; accordingly, a worldwide patent search now gives 7200 hits for *siRNA*, with 1200 specifically for *siRNA and cancer* [188].

The other issue with gene silencing as a treatment method is that siRNA is degraded by enzymes in extracellular space [184]. Intuitively this makes sense - it would be dangerous for the body to allow arbitrary pieces of nucleic acid to enter the cell, since, as described, it can affect the cell's ability to correctly express genes. To allow siRNA to act as a treatment, we therefore require a system which both protects the siRNA in extra-cellular space and also allows it to enter cells. This process is known as delivery [189].

The literature describes several methods under investigation for use as an siRNA delivery agent. Lipofectamine, the gold standard for transfection efficacy [190], is a cationic lipid formulation which can encapsulate siRNA alongside other drugs for delivery to cells [191]. Alongside lipofectamine exist a large number of other lipid-based nanoparticles under development, with various functional coatings [192]. Similarly, polymer-based nanoparticles can encapsulate and deliver siRNA in much the same way [193, 194]. Finally, naked siRNA can be conjugated with other molecules, such as cholesterol [195] or cell-penetrating peptides [196] for efficient cell delivery.

Collaborators in the Department of Chemical Engineering at Biotechnology in Cambridge University are experts in metal organic frameworks (MOFs), and we therefore worked together to develop a method of drug delivery utilising MOFs.

4.1.2 MOFs as delivery vehicles

MOFs are a group of crystalline materials which self-assemble from a mixture of metal ions and organic linkers to form porous solids. They have a diverse use across many fields; because of their ability to contain other molecules within their pores, they have been applied to catalysis [197, 198], ion exchange [199, 200] and sensors [201, 202]. We recently published

an application of MOFs for oxygen storage for use in emergency healthcare [203]; the details of this work fall outside the scope of this thesis.

It was hypothesised that a MOF with the correct pore size could be loaded with siRNA, protecting the contents from extracellular space. If the MOF is subsequently able to enter cells and release its payload, then the system successfully delivers the therapeutic to the cell. To be an effective delivery system, however, the MOF must also be biocompatible. At all stages of its journey to the cell the MOF must not produce a toxic or immunological response. This requirement extends, of course, to any of the products the MOF is broken down into.

As an alternative to targeting specific cancer-causing genes, we can directly target the effect of cancer itself. Cancerous cells generate ATP through glycolysis, which bypasses the respiratory Krebs cycle and switches off normal mitochondria function [55, 56]. Since mitochondria are also responsible for triggering programmed cell death, cancer cells become immortal, continuously reproducing to form tumours. A potentially exciting drug called dichloroacetate (DCA) is a chemical similar to those involved in the Krebs cycle [204, 205]. Experiments on cultured cancer cells have shown that DCA can re-establish the Krebs cycle, preventing further cell division and initiating apoptosis [206].

Despite its potential, clinical trials have so far failed to provide sufficient evidence that DCA produces a therapeutic response in cancer patients [207]. Naked DCA has a half-life in the body of under 1 h [204]. Protecting the DCA in a biocompatible MOF as a delivery vehicle could be used to deliver it to cells more effectively, increasing the concentration delivered to cells without increasing the overall patient dose [208].

4.1.3 Structure of this chapter

Section 4.2 describes the methods and results of three experiments which were imaged on the LAG SIM.

The first experiment looks at whether drug release over time can be controlled by pre-treatment of the MOF-drug complex, using calcein as a model drug.

The second experiment investigates MOF loaded with siRNA, for use as a cancer therapeutic.

The third describes MOFs loaded with DCA and a mitochondria-targetting cofactor, a drug combination designed to destroy cancerous cells.

In all experiments, synthesis of the MOFs and culturing of cell lines was performed by collaborators Michelle Teplensky (*MT*) and Salame Haddad (*SH*). Microscope setup, image capture and reconstruction, and the subsequent analysis was performed by me.

4.2 Results and methods

4.2.1 Temperature treatment of MOFs delays the release time

A problem of drug delivery systems that has been noted in the literature is the so-called *burst release effect* [209]. This is an undesirable characteristic of delivery systems where high concentrations of the drug are released from the carrier soon after loading, so that after a short time there is no payload left to release. This either leads to toxic levels of drug, or low treatment concentrations which must be applied frequently to fulfill their therapeutic effect [210]. An ideal drug delivery system would release its payload slowly and steadily over an extended period of time, maintaining non-toxic concentrations.

When a MOF is used to encapsulate other molecules, for example as a drug delivery vehicle, the payload drug is stored in the pores of the MOF, as shown in Figure 4.1. It was hypothesised that, after loading the MOF with drugs, release could be delayed by collapsing

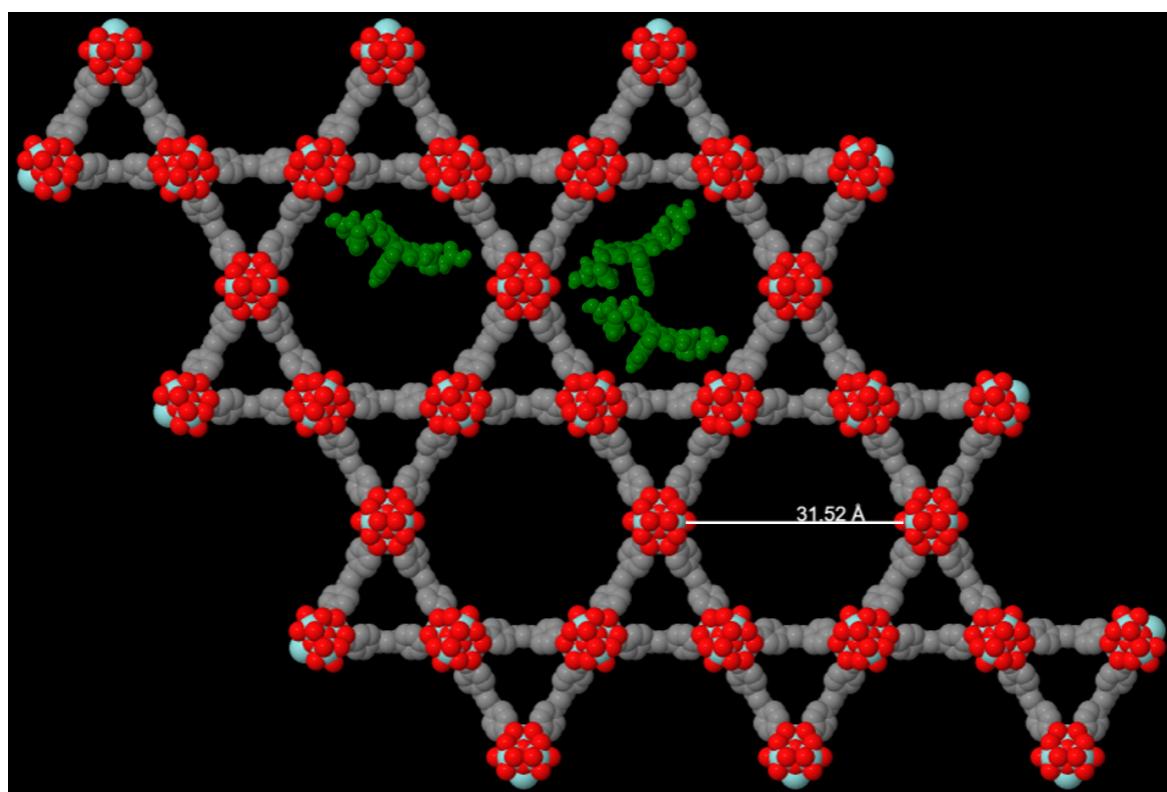


Figure 4.1: MOFs are crystalline structures of metal ions joined by organic linkers. The pore size of NU-1000 MOF, 31.5 \AA , is large enough to store calcein molecules shown here in green.

the pore structure. Pore collapse was achieved with a temperature treatment step after drug loading, before incubation with the cells and imaging.

In this experiment a naturally fluorescent dye, calcein, was used as a model drug. MOFs developed by Northwestern University, known as NU-1000 and NU-901, were chosen and synthesised for their large pore size. Calcein was loaded into the MOFs by soaking 10 mg of either NU-1000 or NU-901 in a 10 mg/ml calcein solution for 3 days in a 37 °C shaking incubator. After this period, the supernatant was removed through centrifuging, and dried at 37 °C for 24 h.

Temperature treatment was then performed on the calcein-loaded MOFs. Samples were placed in a high-temperature vacuum oven at 180 °C for 24 h to induce pore collapse, except for those set aside for control measurements.

Pore collapse was verified by powder X-ray diffraction patterns (PXRD). Figure 4.2 shows that Bragg peaks broaden on addition of calcein, but are still visible at the same θ , confirming the overall MOF structure is maintained. The slight broadening of loaded MOF is most likely a result of the loading process, which leads to some decrystallisation of the MOF. The decrease in peak intensity for temperature-treated MOF is due to a loss of crystalline

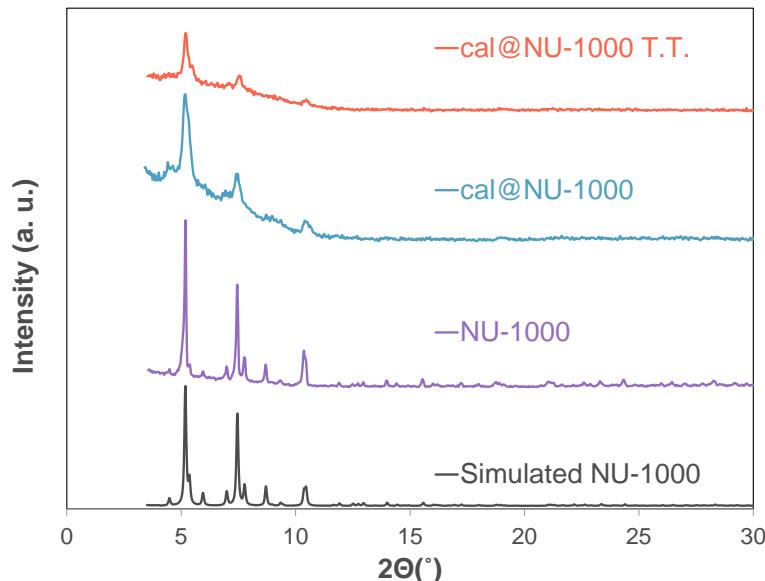


Figure 4.2: Experimental PXRD data (purple) matches well to simulated data (black). When calcein is loaded as a model drug (blue) the same Bragg peaks are still visible, showing that the crystalline structure is maintained; however the slight broadening of the peaks suggests calcein is successfully loaded into the MOF pores. After temperature treatment (red), pore collapse reduces the intensity of the peaks; however they are still visible, suggesting that long-range crystalline order is preserved. PXRD data was generated by *MT*.

structure from the pore collapse; however peaks are still visible, suggesting that long-range order was not completely destroyed. This is in contrast to previous work, which induced pore collapse by mechanical amorphisation, causing peaks to completely disappear [110].

The release profile of calcein from NU-1000 MOF is shown in Figure 4.3, highlighting the burst release effect and its delay by temperature treatment. The effect of temperature treatment is particularly notable at the 24 h time point in Figure 4.3c, where MOF has only released 30% of its payload, compared with 60% released by untreated MOF. This shows that the temperature treatment successfully suppresses the burst release effect.

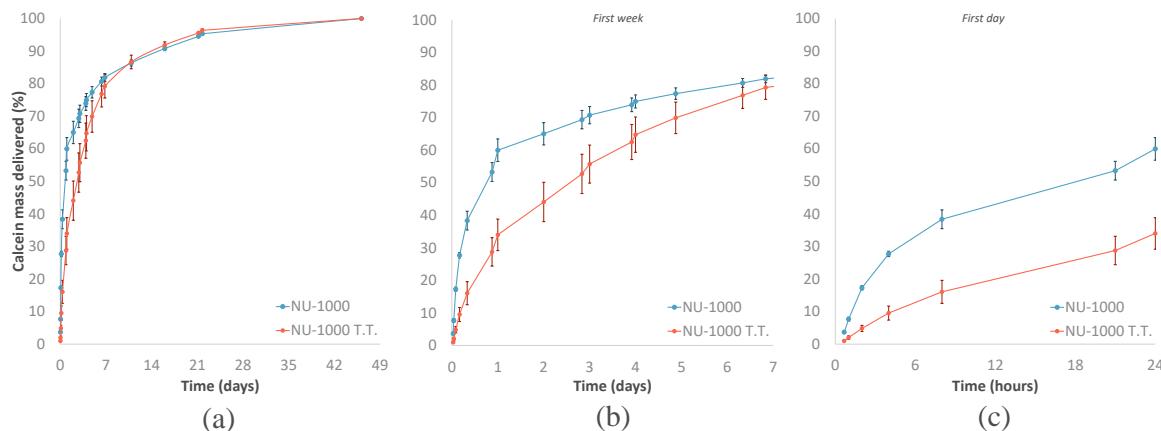


Figure 4.3: The figure shows three views of the same graph, with the time axis scaled differently to highlight the delay of payload release from MOF. The effect is particularly noticeable after 24 h, where just 30% of the payload is released from temperature-treated MOF compared to 60% released from untreated MOF. By delaying the release the MOF has longer to enter cells and will deliver a higher percentage of its payload, rather than releasing it in extracellular space. The release mass was measured through UV-vis spectroscopy by MT.

In order to visualise calcein release in a biological context, HeLa cells were labelled with HCS NuclearMask™ Deep Red Stain to view the nucleus, and CellLight Lysosomes-RFP BacMam 2.0 or CellLight Early Endosomes-RFP BacMam 2.0 to view lysosomes or endosomes respectively. Calcein itself is fluorescent at 495/515 nm for excitation and emission wavelengths respectively, so had good spectral separation for imaging in 3-colours.

Images were captured on the LAG SIM, utilising optical sectioning to remove out of focus light. Representative image slices over a 24 h period are shown in Figure 4.4. Note that the nucleus was not captured for most images to minimise imaging time and thus prolong cell viability; a 3-colour image is presented in the SIM showcase, Figure 2.21. Also notable is that the MOF is highly autofluorescent at the same wavelength as calcein, such that the

calcein released from the MOF cannot be seen on its own. In later experiments, detailed in Section 4.2.2, a drug with better spectral separation was used for more distinct imaging.

An issue with examining 2D image slices is that it is unclear whether MOF has entered the cell, or is simply sitting on the cell membrane with out-of-focus light making it appear localised within the cell. Optical sectioning allowed images to be reconstructed in 3D, so that the cell could be viewed from any angle, verifying that MOF had indeed been taken inside the cell. To allow readers of the publication to verify this themselves, full 3D reconstructions of the slices in Figure 4.4 are hosted online with FPBioimage, and can be viewed at <https://fpb.ceb.cam.ac.uk/MOF/>.

The slices in Figure 4.4, together with their online 3D visualisations, show that MOF is successfully taken up by HeLa cells over a 24 h period. The MOF appears to be taken up by endocytosis, evident from the colocalisation of endosomes with MOFs, seen as white areas in the figure. To verify this observation, an endocytosis study was performed. Inhibitors of various endocytosis pathways were added to the cells, and the fluorescence in the cytoplasm measured after a 24 h incubation period. Figure 4.5 firstly confirms that MOF uptake is an active process, as reducing the temperature to 4 °C causes a significant reduction in cytoplasmic fluorescence. Furthermore, the study shows that 0.3 M sucrose also causes a significant reduction in the amount of MOF uptake, implying MOF is endocytosed by the clathrin-mediated pathway.

Further evidence of endocytosis is provided by video-rate imaging of MOF uptake [107]. Timelapse videos show that MOF outside the cell moves with fast Brownian motion; once inside the cell, however, MOF is comparatively static. This provides further evidence that MOF is taken up by the cells to successfully deliver its payload.

The timescale over which MOF enters cells is important, and emphasises the need for the temperature treatment process. Figure 4.3 shows that, without temperature treatment, 30% of the payload is released within the first 4 h. The timelapse images in Figure 4.4 show that very little MOF has entered the cell by this time, indicating that the calcein payload has been dumped in extracellular space. For a real drug, this would significantly reduce the therapeutic effect.

4.2.2 Loading MOFs with siRNA for therapeutic effects

To quantify the therapeutic effect of delivering drugs to cells using MOFs, we loaded NU-1000 MOFs with small interfering ribonucleic acid (siRNA).

A HEK-293 cell line with the T-REx Flp-In™ system was used to express the fluorescent protein mCherry. Under normal conditions therefore, these cells are fluorescent at 587/610 nm for excitation and emission wavelengths respectively.

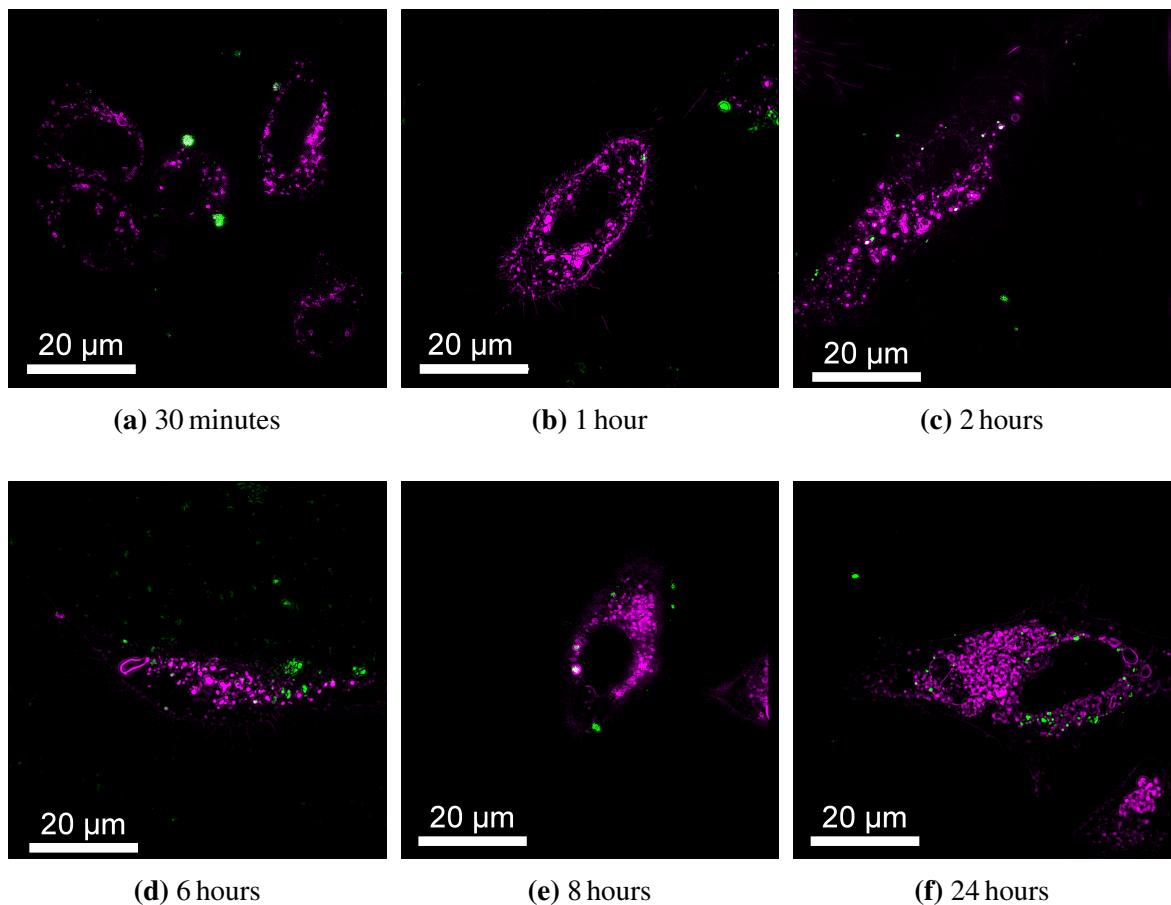


Figure 4.4: Images of HeLa cells, with endosomes coloured in magenta, captured over a 24 h time period show the successful uptake of calcein-loaded NU-1000, coloured in green. The low quantity of uptake in the first 2 h highlights the need for temperature treatment to delay payload release, avoiding the burst release effect. These images can be viewed interactively in 3D at <https://fpb.ceb.cam.ac.uk/MOF> to confirm MOF is located within the cells, and not simply sitting on the cell membrane. Cell culture and staining was performed by *MT*.

An siRNA sequence was designed to cleave the mRNA molecule and suppress expression of the mCherry protein. The 21-nucleotide-long sense strand which was most effective at suppressing fluorescent emission was 5'-AAGGAGTTCATGCGCTTCAAG-3'. Scrambled siRNA, used as a negative control, did not cause any suppression of fluorescence. A combination of mCherry-fluorescent HEK-293 cells and mCherry-targeting siRNA creates a model drug delivery system which is quantifiable: the lower the fluorescence, the more effective the siRNA is at suppressing gene expression.

Without MOF, Figure 4.6a shows that siRNA is broken down by enzymes present in extra-cellular space - in this case, RNase [211]. When the siRNA is loaded into MOF, it is protected from enzymes, demonstrated by the retention of a mark at 21 nt. The retention and broadening of low- θ peaks in Figure 4.6b confirms that the MOF is loaded inside the pores of the MOF, which protects it from enzymes.

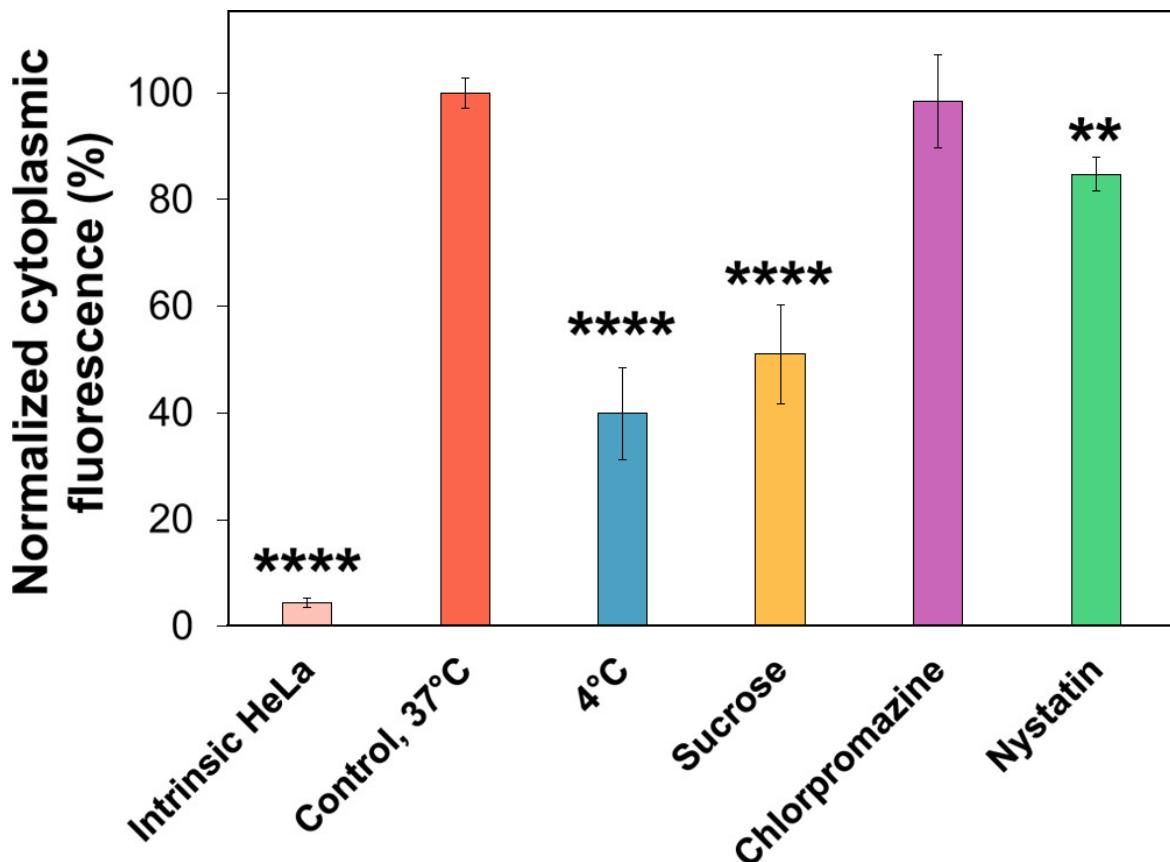


Figure 4.5: Cytoplasmic fluorescence was measured after incubating cells with NU-1000 MOF for 24 h under various conditions to measure how much MOF enters cells. The low bar after incubation at 4 °C shows that MOF uptake is an active endocytosis process. The low bar after incubation with sucrose suggests MOF is endocytosed by the clathrin-mediated pathway. Flow cytometry was performed by MT. *** : $p < 0.0001$, ** : $p < 0.01$.

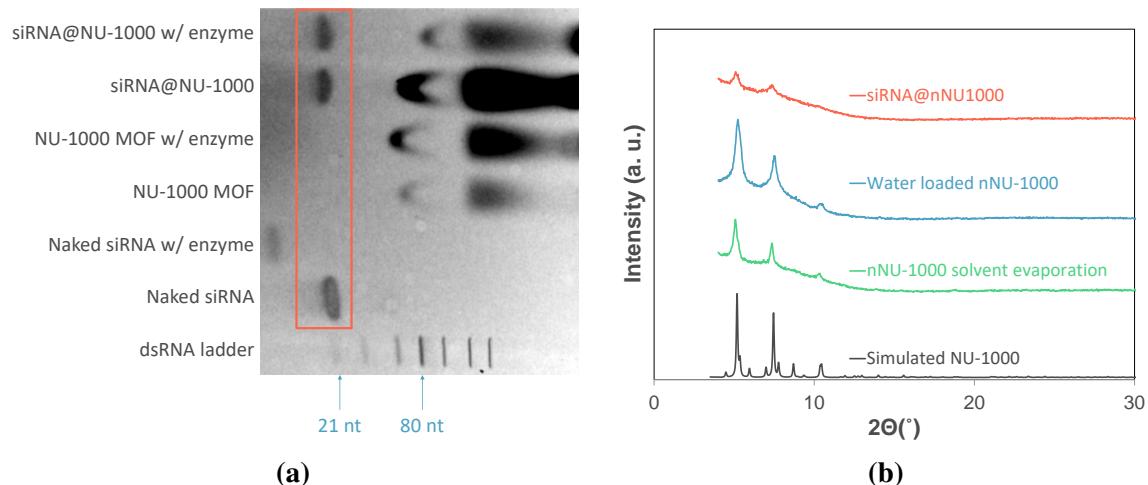


Figure 4.6: The gel in (a) has a mark at 21 nt when siRNA is present. When RNAase is added to the system, naked siRNA is broken down and no mark is present. Loading siRNA into NU-1000 protects the siRNA from degradation, and the mark at 21 nt is retained. PXRD plots shown in (b) confirm siRNA is contained within the pores of NU-1000, shown by broadening of the Bragg peaks. Measurements were performed by *MT*.

To confirm that siRNA-loaded MOF could enter cells, we used LAG SIM to image cells, MOF, and siRNA. As seen in Section 4.2.1, NU-1000 MOF is naturally fluorescent at 488 nm. The CellLight Early Endosomes-RFP BacMam 2.0 stain was used to visualise endosomes in cells, and Alexa Fluor 647 was tagged onto the 5' end of the siRNA sense strand. Multi-colour SIM imaging could then be used to assess cellular uptake of the MOF complex, and confirm that siRNA could enter the cell.

Figure 4.7 shows MOF uptake experiments under four conditions, after 4 h of incubation. MOF is coloured in green, endosomes in blue, and siRNA in red.

When naked siRNA was incubated with the cells, enzyme in the extracellular space broke down the siRNA. This can be seen in Figure 4.7a as an absence of red channel.

The current state-of-the-art for loading RNA into cells is using a transfection reagent such as Lipofectamine 2000. Figure 4.7b shows that this does indeed protect the siRNA from extracellular space, as red clumps of siRNA can be seen. However the siRNA does not appear to be located within the cell at the 4 h timepoint at which the cells were imaged.

Figure 4.7c shows MOF (green) and siRNA (red) colocalised and also within the cell boundary. Furthermore they are colocalised with the endosome stain (blue); the overlap of these three colours produces white. This corroborates with our previous findings from Section 4.2.1 that MOF is taken up by endocytosis.

Although it can be clearly seen in Figure 4.7c that MOF and siRNA are taken up by the cell, measuring the decrease in mCherry expression by flow cytometry revealed that simply

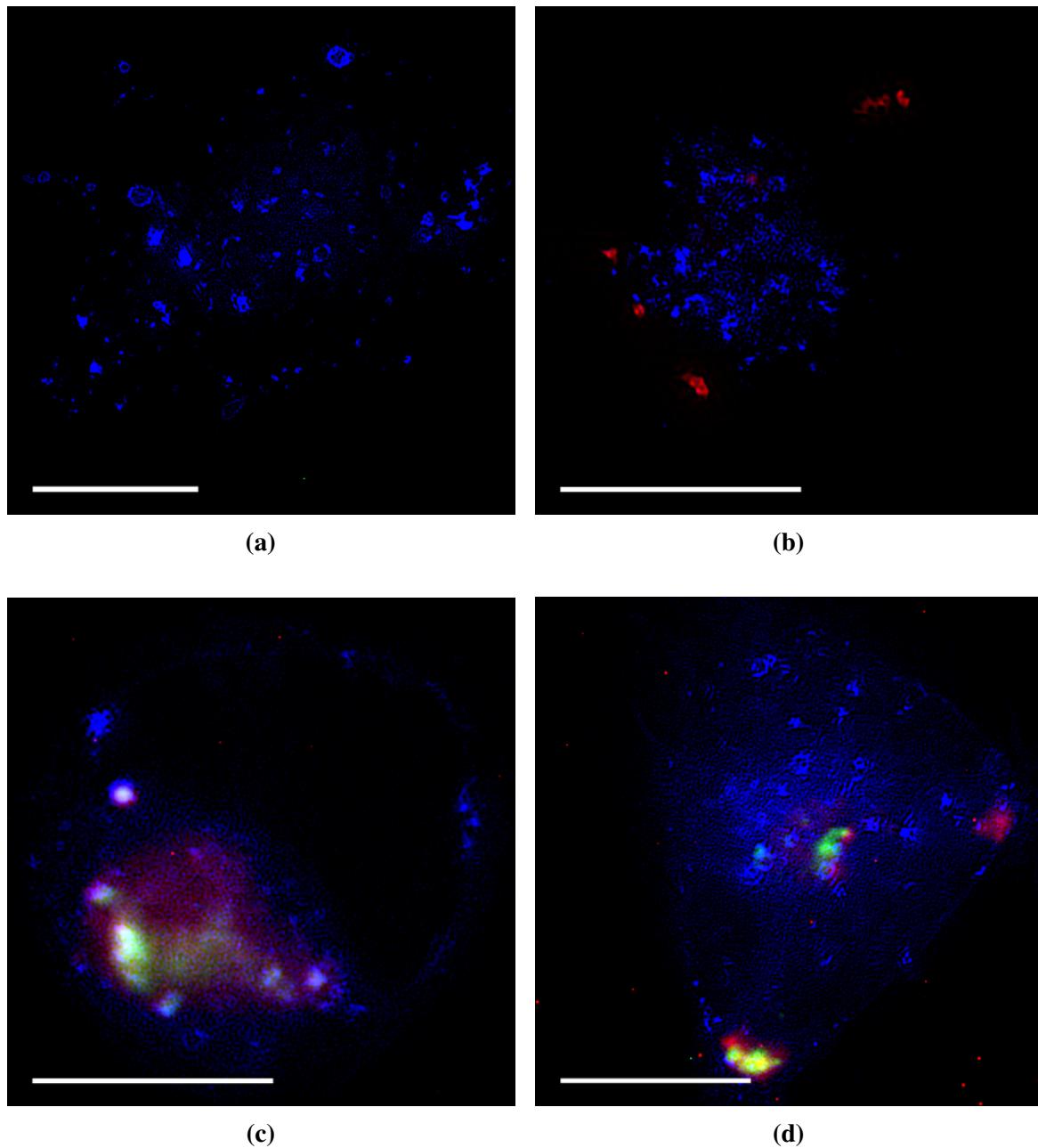


Figure 4.7: Images show endosomes coloured in blue, NU-1000 MOF in green, and siRNA in red, for HEK-293 cells incubated for 4 h under different experimental conditions. (a) shows cells incubated with naked siRNA; the absence of red signal suggests siRNA is degraded in extra-cellular space. (b) shows siRNA loaded into Lipofectamine 2000, which protects the siRNA from degradation, but does not appear to enter cells after 4 h. (c) shows MOF, siRNA, and endosomes colocalised, characterised by white spots; however despite being taken up by the cells, the siRNA has an insignificant therapeutic effect. Addition of an endosome release factor shown in (d) allows the siRNA-MOF complex to escape the endosome without being degraded, shown by yellow spots not colocalised with endosome. Cells were cultured and labelled by MT. Scalebars are 10 μ m.

entering the cell in this way is not enough to suppress gene expression, as shown in Figure 4.8. It was hypothesised that siRNA-MOF complexes were unable to escape the endosome when taken up by the cell, and so the siRNA was degraded by the acidic environment of the endosome [51]. To test this, endosome release factors were loaded into the MOF alongside the siRNA.

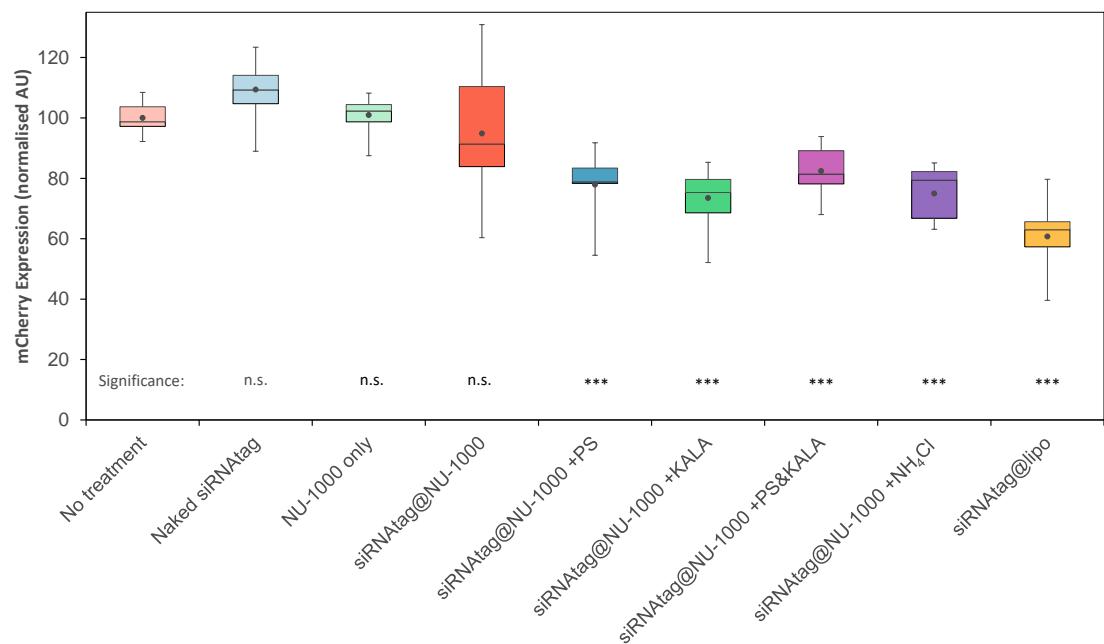


Figure 4.8: The expression of mCherry, measured with flow cytometry, was used to assess how effective different siRNA delivery methods are at gene silencing. Treatment with naked siRNA or NU-1000 MOF only causes no significant change in expression. Furthermore, using siRNA loaded in NU-1000 causes an insignificant decrease in the mCherry expression level. Under this condition it is hypothesised that siRNA is not able to escape the endosome to cleave the mCherry. Addition of endosome release factors, such as KALA, Proton-Sponge® (PS) or ammonium chloride (NH_4Cl) break down the endosome, allowing the siRNA to enter the cytoplasm and perform its intended function; although delivery with Lipofectamine (lipo), the current gold standard for transfection, is still the most effective method for gene silencing. Treatment and measurement was performed by *MT*.

Three endosome release factors were used: Proton-Sponge®, an effective H^+ scavenger from Sigma-Aldrich [212]; KALA, a cationic amphipathic cell-penetrating peptide which binds oligonucleotides and disrupts cell membrane; and ammonium chloride, which has previously been shown to induce release of molecules from sub-cellular vesicles [213]. siRNA was also delivered to cells with Lipofectamine, the current gold standard for transfection, for comparison with delivery by MOFs.

Figure 4.8 shows that with the addition of any release cofactor mCherry expression level was significantly reduced. The expected mechanism by which the siRNA is now able to suppress mCherry expression is as follows: siRNA and the endosome release factor load into the pores of MOFs; these MOF complexes are taken up into the cell by endocytosis, and are initially located in endosomes; the MOF begins to unload its contents in the endosome environment, releasing the endosome release factor and siRNA; the endosome release factor breaks down the endosome membrane before the endosome can break down the siRNA, releasing the siRNA into the cytoplasm; and finally, the siRNA can cleave mRNA, silencing expression of mCherry.

The proposed mechanism for the successful gene silencing is confirmed by the SIM image in Figure 4.7d. In contrast to Figure 4.7b, yellow areas show that the siRNA-MOF complex is no longer colocalised with endosomes, showing that the cofactor successfully released it from the endosome into the cytoplasm. Furthermore, siRNA can be seen independently in parts of the cell, as red pixels not colocalised with any other colour.

4.2.3 MOFs deliver DCA to restart the Krebs cycle, with TPP as a cofactor to target mitochondria

As described in Section 4.1, most cancers occur when cells bypass the Krebs cycle, such that mitochondria continuously divide without undergoing apoptosis [55]. This makes cancerous cells difficult to destroy. While chemotherapy techniques, such as radiation therapy, can kill the cancer cells, treatment is often non-discriminatory and destroys all cells in the area, leading to unwelcome side-effects [214–216].

It has been shown that DCA is able to induce apoptosis in cancer cells by restoring mitochondrial function and restarting the Krebs cycle [206]. However clinical trials so far have been unsuccessful; perhaps because of a low half-life in the body, so that a therapeutic concentration does not reach the cells. Loading DCA into a MOF could assist with delivery. In this experiment the MOF UiO-66 was used which, like the NU-1000 used in our previous MOF experiments, has a large pore size and good biocompatibility [208].

Inspired by our previous work loading a cofactor with siRNA, we investigated the effect of adding a mitochondria-targeting cofactor to the MOF alongside the DCA, with the hypothesis that this would destroy more cancer cells with a lower concentration of MOF than previously shown [208]. The compound triphenyl phosphonium (TPP) has been shown to help molecules to target mitochondria, so TPP was loaded into UiO-66 MOFs as a cofactor alongside DCA. To obtain a thorough understanding of how the MOF complex was acting inside the cell, we used the LAG SIM to image HeLa cancer cells under different treatment conditions.

The UiO-66 MOF used for this study was not naturally fluorescent, so was furthermore loaded with calcein to visualise it in the 488 nm excitation channel. Mitochondria were labelled with MitoTrackerTM in the 561 nm channel, and the cell nucleus was visualised with SiR-DNA in the 640 nm channel.

Sample SIM images are shown in Figure 4.9. We confirm, with 3D reconstructions obtained by optical sectioning, that the UiO-66 MOF is successfully taken up into HeLa cells. Furthermore, we can now visually observe the physical effect this treatment has on the mitochondria.

Figure 4.9a shows untreated HeLa cells with inactive mitochondria forming long, stringy projections. After 8 h of incubation with DCA-loaded MOF, some of the mitochondria in Figure 4.9b have become more circular, resembling the textbook shape of healthy mitochondria performing the Krebs cycle [55]. Figure 4.9c shows a similar effect after just 30 min when the mitochondria-targeting cofactor TPP is also loaded into the MOFs. After 8 h of incubation with DCA and TPP both loaded into the MOF, all the mitochondria in Figure 4.9d resemble the healthy textbook shape.

To confirm this visual inspection of mitochondrial shape quantitatively, analysis was performed in CellProfiler [217]. Image stacks of 54 cells were captured and reconstructed in two independent imaging sessions with optical-sectioning LAG SIM, and from these images 4811 mitochondria were extracted using the `IdentifyPrimaryObjects` plugin, as shown in Figure 4.10. The ellipticity of each classified mitochondrion was then calculated with the `MeasureObjectSizeShape` plugin, to produce a long list of mitochondria ellipticity under the different treatment conditions shown of Figure 4.9.

The mitochondria ellipticity statistics were analysed with a one-way ANOVA test in Graphpad Prism [218], and the results are presented in Figure 4.11. Adding DCA-loaded MOF causes a decrease in ellipticity, however there is not a statistically significant difference between the mean ellipticities, with $p = 0.2083$. With TPP also added to the MOF as a cofactor, a $p = 0.0194$ significant decrease in the ellipticity of mitochondria is observed. Furthermore after 8 h of treatment there is a significant decreases between the DCA+TPP MOFs and the DCA-only MOFs. These statistics support the visual inspection of Figure 4.9.

Furthermore, these findings corroborate data from FACS measurements shown in Figure 4.12. The graphs show that MOFs loaded with both DCA and TPP are more effective at inducing cell death than MOFs with DCA alone, measured by the MTS assay [219, 220]. Moreover, the graphs show that this enables a lower concentration of drug to be administered for the same therapeutic effect.

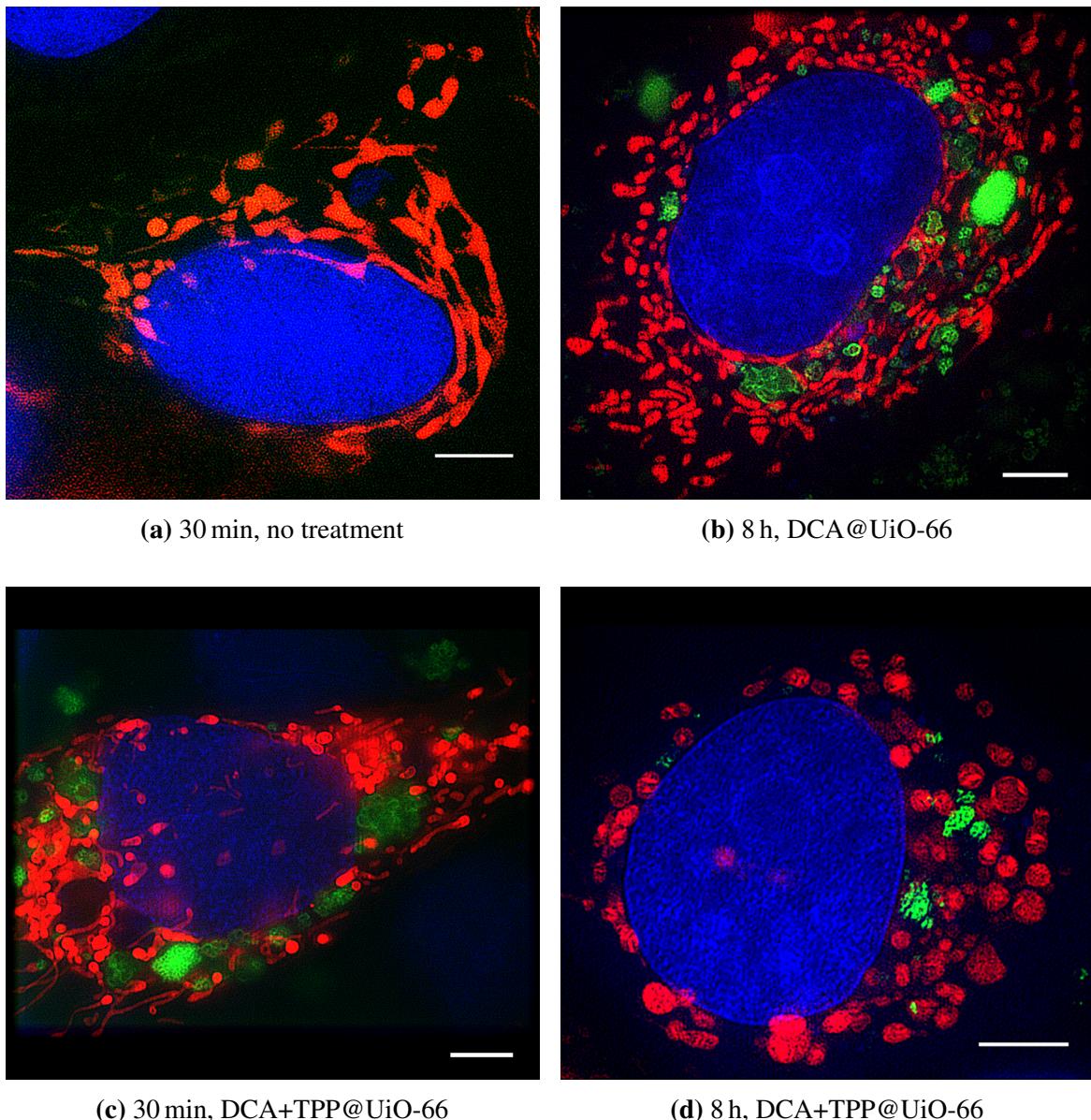


Figure 4.9: Images show mitochondria coloured in red, MOF visible by calcein-loading coloured in green, and the nucleus labelled by SiR-DNA coloured in blue. (a) shows inactive mitochondria in untreated cells, with long stringy projections. Treating the cells with DCA-loaded UiO-66 MOF, shown in (b), causes mitochondria to lose their long, elliptical shape. (c) and (d) show that this process is enhanced by loading TPP as a cofactor, resulting in circular mitochondria after 8 h of treatment. Culture and staining of the HeLa cells was performed by SH. Scalebars are 5 μm .

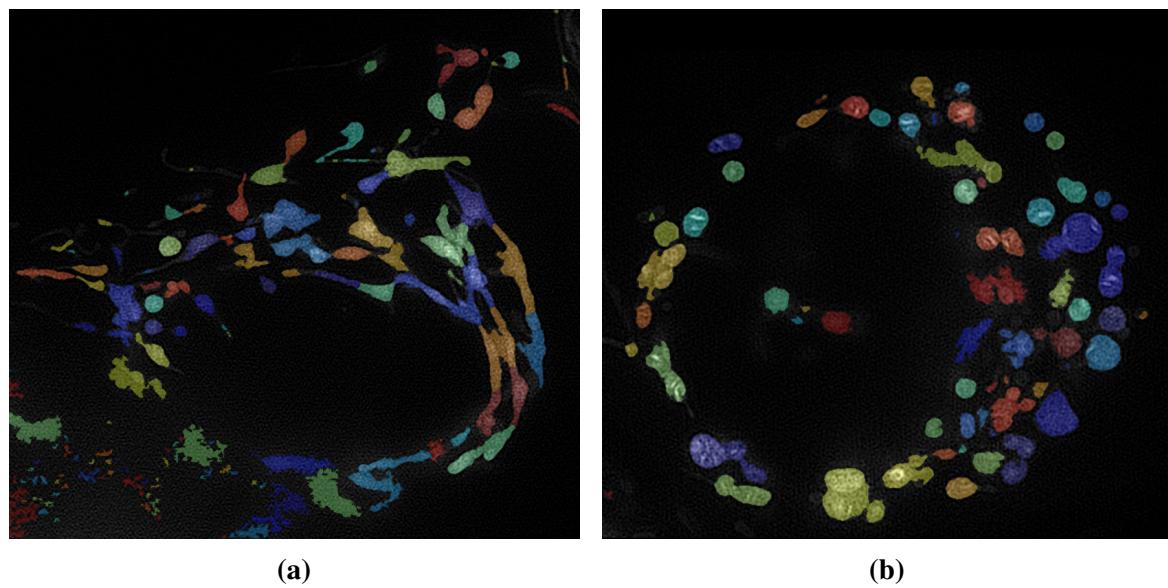


Figure 4.10: The CellProfiler[217] protocol identifies and segments mitochondria in the cell, shown here applied to Figures 4.9a and 4.9d. The ellipticity of segments is calculated to provide statistics about the change in mitochondria shape caused by DCA-loaded UiO-66. Scalebars are 5 μ m.

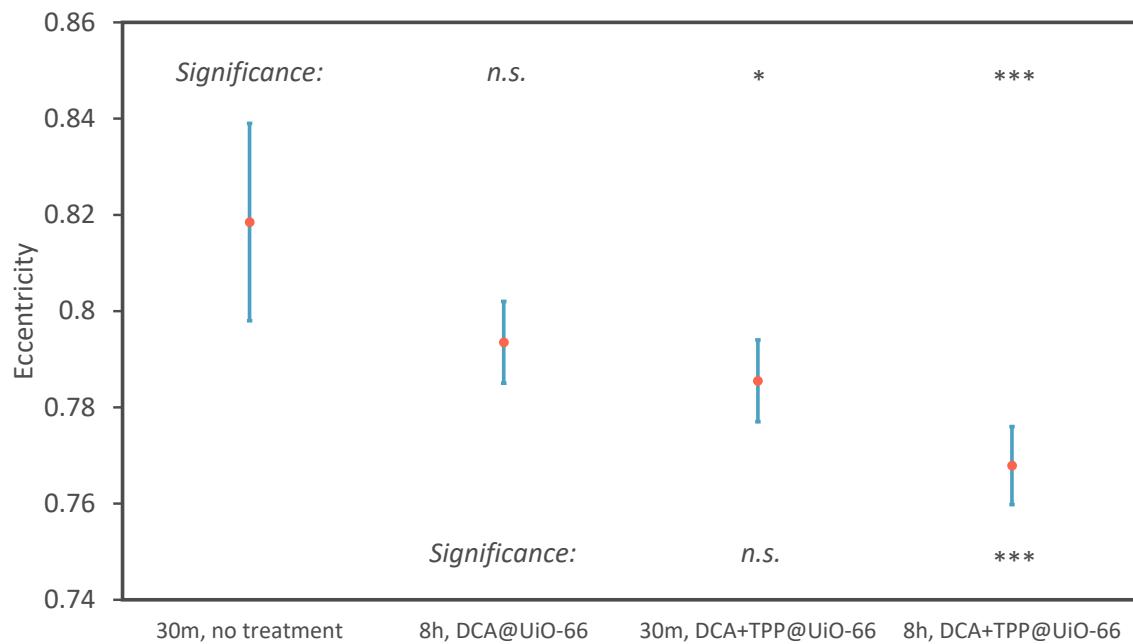


Figure 4.11: Mitochondria are affected by the DCA-UiO-66 treatment, making them more circular in shape. The reduction in eccentricity is not significant without addition of TPP as a mitochondria-targetting cofactor. * : $p < 0.05$, *** : $p < 0.001$

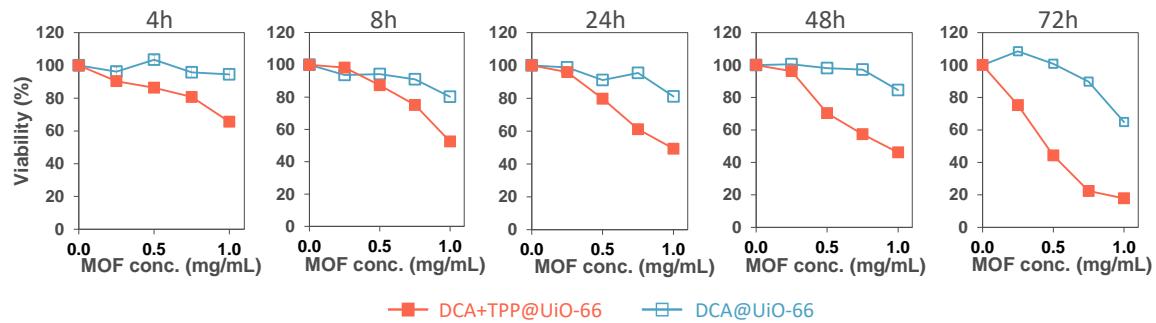


Figure 4.12: A cytotoxicity study corroborates observations of microscopy images that MOFs loaded with TPP as a cofactor are more effective at inducing cell death in cancerous HeLa cells than MOF loaded with DCA alone. This allows treatment with lower concentrations of drug for the same therapeutic effect. Viability measurements were performed with the MTS assay by *SH*.

4.3 Conclusion

The three experiments presented in this chapter utilise MOFs as a drug delivery vehicle for cancer treatment. We have shown through a variety of techniques that MOFs can be successfully loaded with therapeutic molecules which would otherwise be degraded in extra-cellular space, and subsequently can deliver these drugs to the cell.

Furthermore, we have observed a significant decrease in cancer cell viability when MOFs are used as a delivery vehicle. Two techniques, based on delivering siRNA for gene silencing and DCA for mitochondrial degradation which restarts the Krebs cycle, have emerged as successful treatment strategies on cancerous HeLa cells. Concerns regarding the burst release effect, where much of the MOF's therapeutic payload is released before the MOF has entered the cell, have been addressed with a temperature treatment strategy which induces pore collapse to delay release.

The utilisation of optical sectioning provided by the LAG SIM has allowed both visual confirmation of biological hypotheses and large datasets for statistical analysis.

Throughout the chapter, 3D reconstructions have shown that MOF is able to pass the cell membrane and enter cells; an endocytosis study showed that this occurs through the clathrin-mediated pathway. Inside the cells, MOFs are colocalised with endosomes; this can prevent the MOF's payload having a therapeutic effect without addition of an endosome release factor. When Proton-Sponge®, KALA or NH₄Cl were loaded into the MOF as cofactors, a significant decrease in cell viability was measured, and furthermore SIM images confirmed that MOF was no longer colocalised with endosomes.

LAG SIM's fast imaging speed allows a large number of cells to be imaged in a single session. This firstly has the advantage that cells do not spend long on the microscope stage, where non-optimal conditions can cause apoptosis of cells, but rather cells can be left in the incubator for most of a 24-hour time course study. Secondly, meaningful statistics can be gathered when a large number of cells are imaged, allowing visual observations to be rendered as statistically significant results.

Overall this chapter has demonstrated the LAG SIM's capability for facilitating hypothesis-driven research supporting and confirming measurements from other techniques and providing visualisation to develop new treatments in the fight against cancer.

4.4 Future work

Using MOFs as a drug delivery vehicle has been demonstrated as an effective treatment for cultured HEK-293 cells. The next step towards developing a drug that can be used to treat humans is to assess the treatment on a model organism. Work is now underway to apply the siRNA-MOF treatment to mice, to observe the effectiveness of the therapy at organism level and assess any toxicity issues.

This chapter has presented just 3 MOFs, NU-901, NU-1000, and UiO-66, which have desirable properties for use as a drug delivery vehicle. However our previous work has shown that due to the tens-of-thousands of MOF structures possible, selecting the perfect MOF is non-trivial [203]. To address this I have built an online tool for screening a database of MOFs to select the one most-suited to a given application [203]. Though the tool, accessible at <https://aam.ceb.cam.ac.uk/mof-explorer/>, currently displays properties useful for optimising gas storage solutions, adding biologically relevant axes such as toxicity and payload release time could reveal an even better MOF for drug delivery than those already investigated.

Chapter 5

ER: Endoplasmic reticulum network dynamics

5.1 Introduction

5.1.1 ER malfunction is linked to a variety of diseases

The endoplasmic reticulum (ER) is an interconnected network of tubular pipes and reservoirs present in eukaryotic cells [40]. Its function includes the biogenesis of membrane and secretory proteins, and delivery of these functional solutes throughout the cell [53]. Since the ER network extends to the cell periphery, molecules can be transported to distant sites in the cell.

When demand is greater than the capacity of ER to fold proteins, ER stress can occur [221]. ER stress can also be caused by proteins that cannot properly fold inside the ER [222]. A number of signal transduction proteins in the cell monitor proteins secreted by the ER, and if an imbalance is detected the ER stress response begins [221].

As the largest organelle in a cell, recent studies have shown ER forms membrane contacts with all other cellular organelles, with a higher number of contact sites than any other organelle-organelle interaction [223, 224]. Since ER is found in almost all human cells, it is therefore not surprising that a wide range diseases have been associated with ER stress, including neurodegeneration, atherosclerosis, type 2 diabetes, liver disease, metabolic diseases, and cancer [221, 225].

The rate at which luminal ER content is distributed affects the efficiency of ER-mediated intracellular connectivity [226, 227]. Understanding the fundamental processes of ER protein mobility is therefore vital to curing diseases caused by mutations in ER proteins.

5.1.2 Flow in the ER is an active process

ER proteins are delivered around the cell by moving through a network of tubes which make up the ER. It has been shown with fluorescence recovery after photo-bleaching (FRAP) that protein mobility is an ATP-dependent process - that is, proteins move slower when ATP is depleted [228, 229].

More recently, we have shown with single particle tracking that molecules in the ER do not move with diffusive motion [230]. Rather, they have a fast, directed velocity as they flow along ER tubes, but move with slow, diffusive motion at nodes where 3 or more tubes connect, as shown in Figure 5.1a. Appendix B.1 contains further details on the calculation of the flow models.

Figure 5.1b shows that applying data from 80 563 single particle tracks to a mathematical model with both diffusive and flow terms provided a better fit to experimental data than a model with diffusive motion only. This is particularly convincing in the cumulative frequency distribution shown in Figure 5.1c. Further details on the motion analysis modelling are provided in Appendix B.1.

Consistent with previous studies [229], tracking directed flow revealed that depleting ATP causes a $4\times$ decrease in mean flow velocity, shown in Figure 5.2a.

It was suggested that the decrease in protein mobility could be explained by an increase in the resistance to motion due to crowding of luminal ER proteins. To counter this argument,

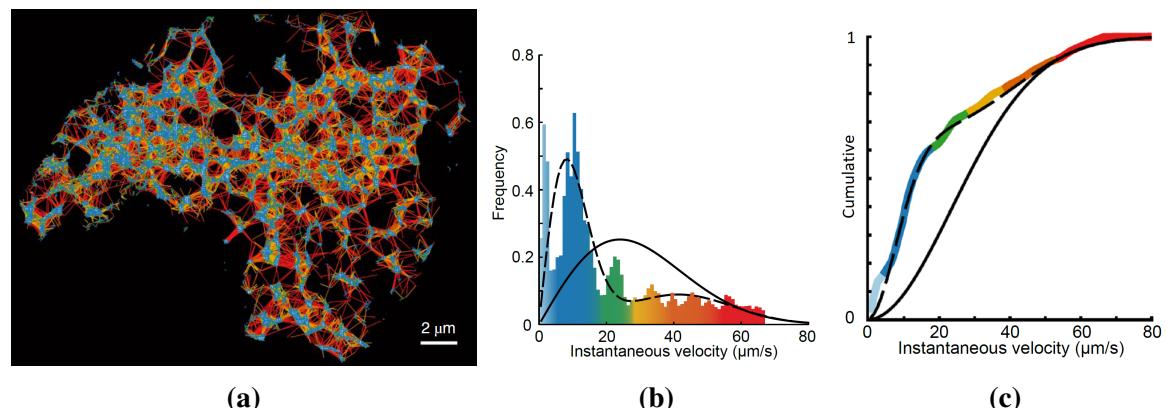


Figure 5.1: (a) shows tracks of single particles in the ER, colour-coded according to the scale in (b) and (c). Fast, directed movement occurs in tubules, and slow movement occurs in nodes. A diffusive model, shown by the solid black line in (b) and (c) does not fit the distribution of velocities; a much better fit is provided using a model which combines directed and diffusive flow, shown by the dashed line. Cell culture and staining was performed by Edward Avezov, and single particle tracking by David Holcman. Graphs were created from $n = 80\,563$ tracks.

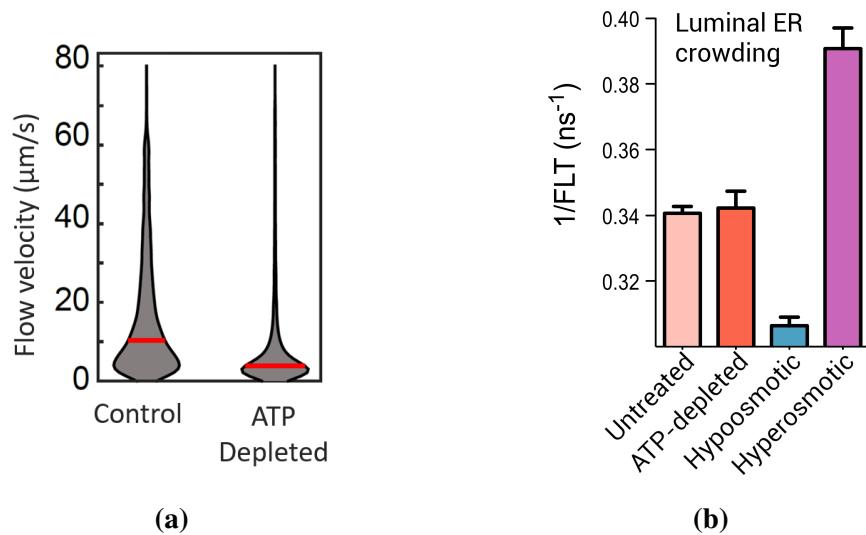


Figure 5.2: (a) shows that mean flow velocity decreased $4\times$ when ATP was depleted from the cell. A FRET probe was used to assess crowding of luminal ER proteins, shown in (b). Although hypoosmotic conditions caused a decrease in crowding and hyperosmotic conditions caused an increase in crowding, due to expansion and contraction of the cell size respectively, depleting ATP did not change crowding of luminal proteins, countering the argument that a decrease in flow velocity is due to crowding of proteins. Measurements were performed by Edward Avezov.

a FRET-based probe was used to measure ER crowdedness. Treating with a hyperosmotic buffer induces cell shrinking, which increases crowding, decreasing the fluorescence lifetime. Conversely, a hypoosmotic buffer induces cell swelling, decreasing crowding, and the fluorescence lifetime of our sensitive FRET probe increases. When depleting ATP, there is no change in lifetime; therefore we dismiss the hypothesis that the change in protein mobility is due to increased crowdedness, but relies on an active, ATP-dependent mechanism.

Further details on the FRET probe implementation are provided in Appendix B.2.

5.1.3 Structure of this chapter

As part of a collaboration with David Holcman and Edward Avezov, the authors who first observed directed ER flow, I investigated potential mechanisms for flow. Utilising the unique capabilities of structured illumination microscopy (SIM) and computational analysis, evidence is presented in this chapter that ER luminal flow is caused by pinching of the ER membrane.

The results and methods section which follows details 3 key contributions which I personally made to the published study:

1. Measuring dynamic network arrangement with SIM imaging and computational analysis
2. Imaging tubule pinching with TIRF-SIM and extracting statistics regarding pinching frequency
3. Computational simulation of tubule pinching to investigate its biological advantage over diffusive transport

Conclusions and future work extending the findings presented here are given at the end of the chapter. Background investigation which I did not personally perform is shown in Appendix B, and further details of the full study can be found in *Nature Cell Biology* [230].

5.2 Results and methods

5.2.1 High-speed SIM and image analysis reveals a dynamic ER network

Utilising the high-speed imaging capability of the LAG SIM, we observed that the ER is a dynamic network, with nodes and tubular endpoints constantly moving to rearrange the network. Imaging was performed at a raw frame rate of 99 Hz, corresponding to a reconstructed frame rate of 11 Hz.

For optimal imaging conditions, COS-7 cells were selected for the investigation. These characteristically flat cells, with a height of $<10\text{ }\mu\text{m}$, meant that the whole ER network could be imaged by LAG SIM in TIRF mode, which physically removes out-of-focus light. The ER network was labelled with ER-TrackerTM Green, a fluorescent dye from ThermoFisher excited at 488 nm. This allowed SIM to achieve its highest resolution imaging of 80 nm.

To verify our observations, and ensure relevance to human pathology, we also used HEK-293 cells labelled with tetramethylrhodamine (TMR) attached to luminal ER proteins with HaloTag® technology. These thicker cells were imaged with optical sectioning LAG SIM, since TIRF only showed small fragments of ER close the coverglass. Images in this chapter are described as COS-7-ERT, COS-7-TMR, and HEK-293-ERT, and HEK-293-TMR to describe COS-7 and HEK-293 cells labelled with ER-TrackerTM Green or TMR respectively.

Figure 5.3 shows a small area of the ER network in COS-7-ERT cells. The network's tubules and nodes are clearly visible, as well as large and small loops in the network and sheet structures. The time series shows that the ER network undergoes fast rearrangements, with a number of tubules growing and shrinking and nodes rearranging at any given time. The speed of this process highlights the need for the fast, high-resolution imaging technique provided by LAG SIM.

To draw statistics describing the network from the images, the following steps were performed:

1. Generate a binary image of the network for each frame
2. Skeletonise the network, making it one pixel wide
3. Classify nodes, endpoints, and tubes with MATLAB's `bwmorph` functions
4. Extract statistics from labelled binary image stacks

A common method for generating a binary image is thresholding, where all pixels with a value above a certain threshold are classified as `true` and all those with values below the threshold are assigned `false`. Whilst this scheme was effective at differentiating ER from background in the thin COS-7 cells, non-uniform background light in HEK-293 cells, visible in Figure 5.4a, meant that a simple binary threshold produced the poor network extraction shown in Figure 5.4b. Furthermore, the binary threshold required for each image set varies depending on exposure time, laser power, and expression level of the fluorescent label in that particular cell. Therefore a single value threshold was not appropriate to generate binary images.

Binary morphology was also considered as a classification technique, where opening and closing operations were performed with line segments and disks respectively. Due to the variety of line segment lengths and shapes, however, a repeatable set of operations could not be devised, and structuring elements required fine tuning for each image set.

Instead a machine learning technique was used to classify ER pixels from background pixels. Using the Trainable Weka Segmentation plugin for ImageJ [231], a set of training data was generated by manually classifying ER pixels and background pixels. The Adaboost

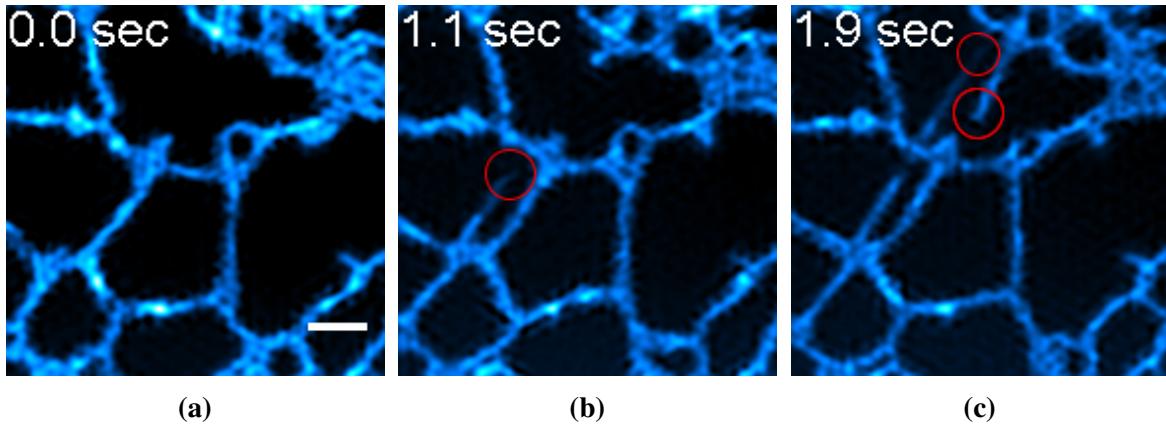


Figure 5.3: The ER network is in a state of dynamic equilibrium, with a small number of nodes being created and destroyed at any given time to cause network rearrangement. Images are of COS-7-ERT; note that these images are a selection from a more detailed timelapse, and these examples are not evenly spaced. Scalebar is 1 μm .

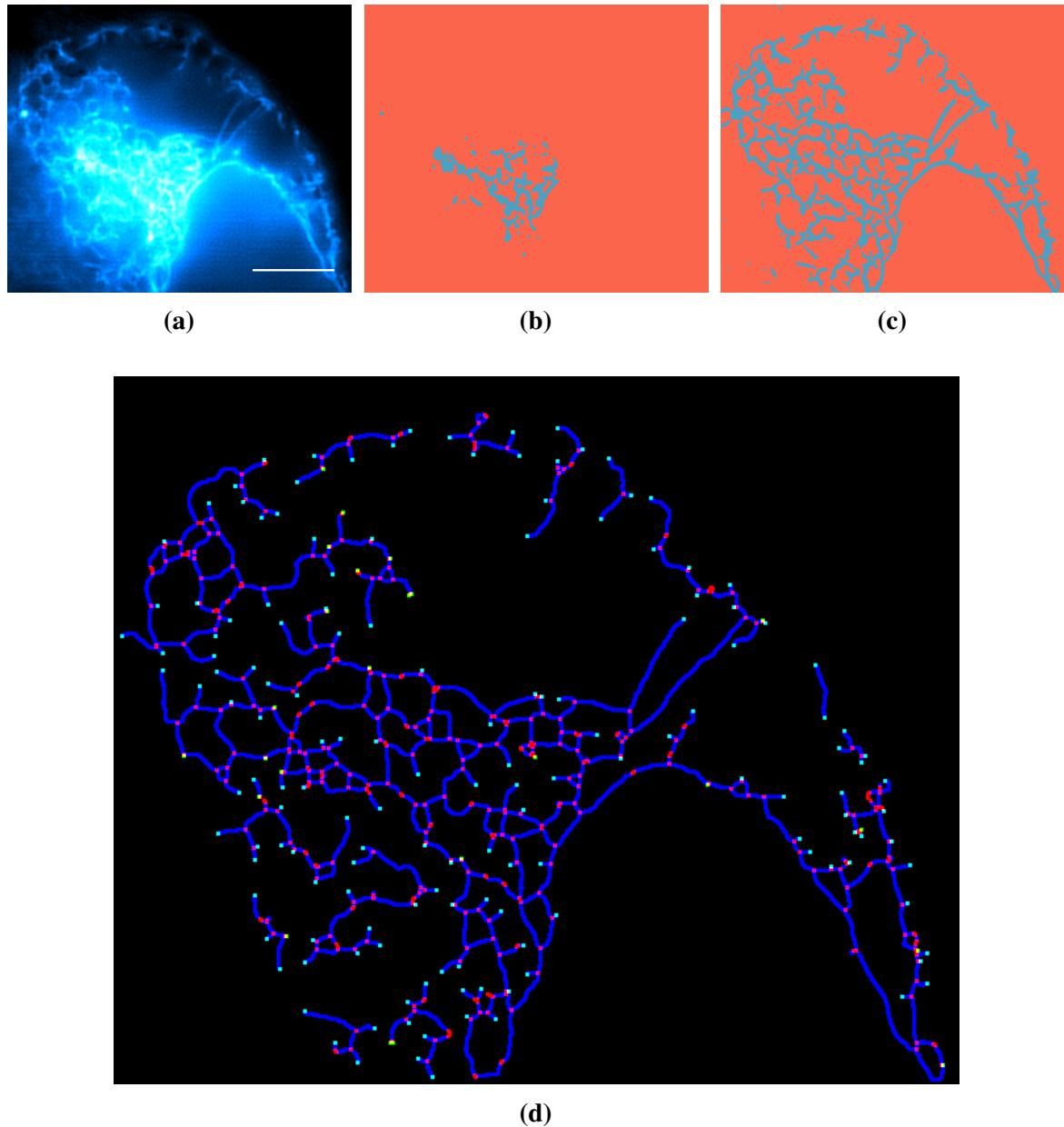


Figure 5.4: (a) shows a SIM reconstruction of ER where the background is not uniform and honeycomb artefacts have appeared, as described in Section 2.1.3. A simple threshold for separating ER from the background, shown in (b), does not work well on HEK-293 cells due to the variations in background intensity. A machine learning technique, implemented with the Weka segmentation plugin for ImageJ, identifies the ER and background more reliably, as shown in (c). A MATLAB script was applied to the binary image to classify ER tubes, nodes, and end-points, labelled in (d) as blue, green, and red respectively. Tracking of nodes and endpoints, and analysis of tubule length distribution, could then be performed on the network. Images are of HEK-293-ERT; scalebar is $5\text{ }\mu\text{m}$.

algorithm was used to quickly refine the classification [232, 233], leading to a reliable and automated adaptive thresholding method. The output of Weka Segmentation in Figure 5.4c successfully extracts the ER network even with the unideal imaging properties of HEK-293 cells.

After binary images were skeletonised, a classified network map, as shown in Figure 5.4d, was generated with MATLAB for every frame of every image set. From this, various statistics could be extracted, including tubule length distribution, tip growth rate, and node movement.

The distribution of tubule lengths, shown at various time points in Figure 5.5, does not change over time. This shows that the ER is in a dynamic equilibrium. Although there are local changes in structure and connectivity, the overall network properties do not change.

Movement of end points and nodes was initially suggested as a hypothesis for the distribution of particle movement inside the ER. However tracking of nodes revealed an average speed of just $(0.39 \pm 0.42) \mu\text{m s}^{-1}$, around $60\times$ slower than the $22.9 \mu\text{m s}^{-1}$ mean speed of particles found in the single particle tracking analysis of Figure 5.1b. Furthermore, we calculated that just $(0.14 \pm 0.04)\%$ of tubules were growing at any given time. We conclude that tubule growth and node movement have a negligible contribution to particles' diffusion and flow velocities.

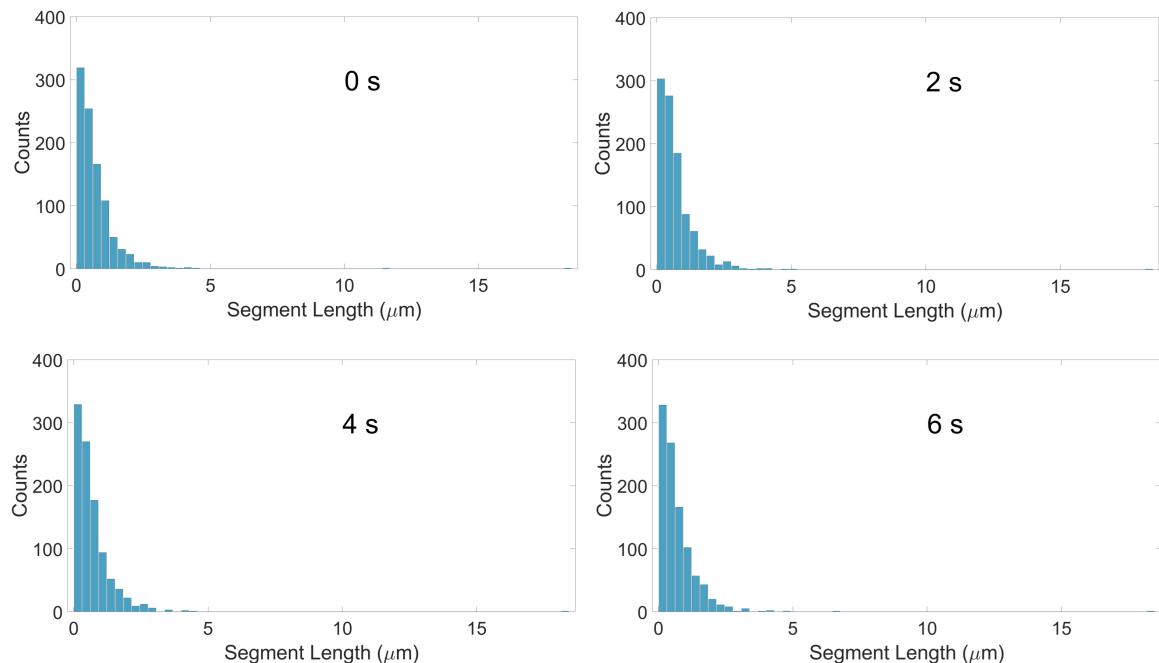


Figure 5.5: Although individual tubules are growing and shrinking at any given time, as shown in Figure 5.3, the distribution of tubule lengths does not vary over time, showing that the ER is in a dynamic equilibrium.

5.2.2 Active tubule pinching correlates with luminal flow velocity

The motion of luminal proteins suggested the possibility of nanoperistalsis-like propulsion [234], attainable by tubule contractions. Using the LAG SIM in TIRF mode, we obtained images at 11 Hz, with a resolution of 80 nm, and were able to observe pinching of the tubule.

ER tubules are 60 nm to 100 nm in width, so observing pinching requires a microscope that can image beyond the diffraction limit [235]. When ER tubules pinch, their width decreases to below the resolution limit of SIM; however this can still be observed as a decrease in intensity, confirmed by the simulation shown in Figures 5.6a-f.

To clearly visualise pinches, the ‘Edges’ colourmap was used in ImageJ, highlighting lines of constant intensity. Note that, whilst this allows pinching events to be accurately counted, there is not enough resolution in SIM to measure the width of the tubule when it is pinched. Nevertheless we believe that SIM is still the only technique capable of capturing these dynamic events, due to its combination of spatial resolution and imaging speed.

Counting the number of pinches in each tube over time gave the mean frequency of pinches as (1.70 ± 0.48) pinches/s in COS-7 cells. Figure 5.6i shows that when ATP was depleted, the frequency decreased $3.9\times$, to (0.44 ± 0.41) pinches/s. This decrease in pinching frequency is very similar to the velocity decrease observed with single particle tracking in Figure 5.2a, suggesting that there is a strong link between tubule pinching and directed motion in the ER.

Further evidence for tubule pinching was gathered when the cell was artificially placed under excessive stress. Whilst acquiring time series videos of the ER on the LAG SIM, we occasionally observed a reversible blebbing behaviour of the ER. This phenomenon, shown in Figure 5.7, was particularly evident when imaging samples labelled with TMR fluorescent dye with the 561 nm laser power set to 95 mW or more, equivalent to 50 W cm^{-2} at the sample. The same behaviour was observed when calcium manipulation [236] was performed on the cell. Notably, the ER recovers its original shape after the stress is removed, demonstrating the ability of ER tubules to undergo reversible pinching.

Furthermore, 3D images of a fixed ER network, obtained using optical sectioning SIM, reveal tubules pinched in 3D. This is available to view interactively with FPBioimage, at <https://fpb.ceb.cam.ac.uk/ER>. Despite tubules’ size being at the resolution limit of SIM, the evidence from the different experiments presented here provide confidence that pinching is not an imaging artefact.

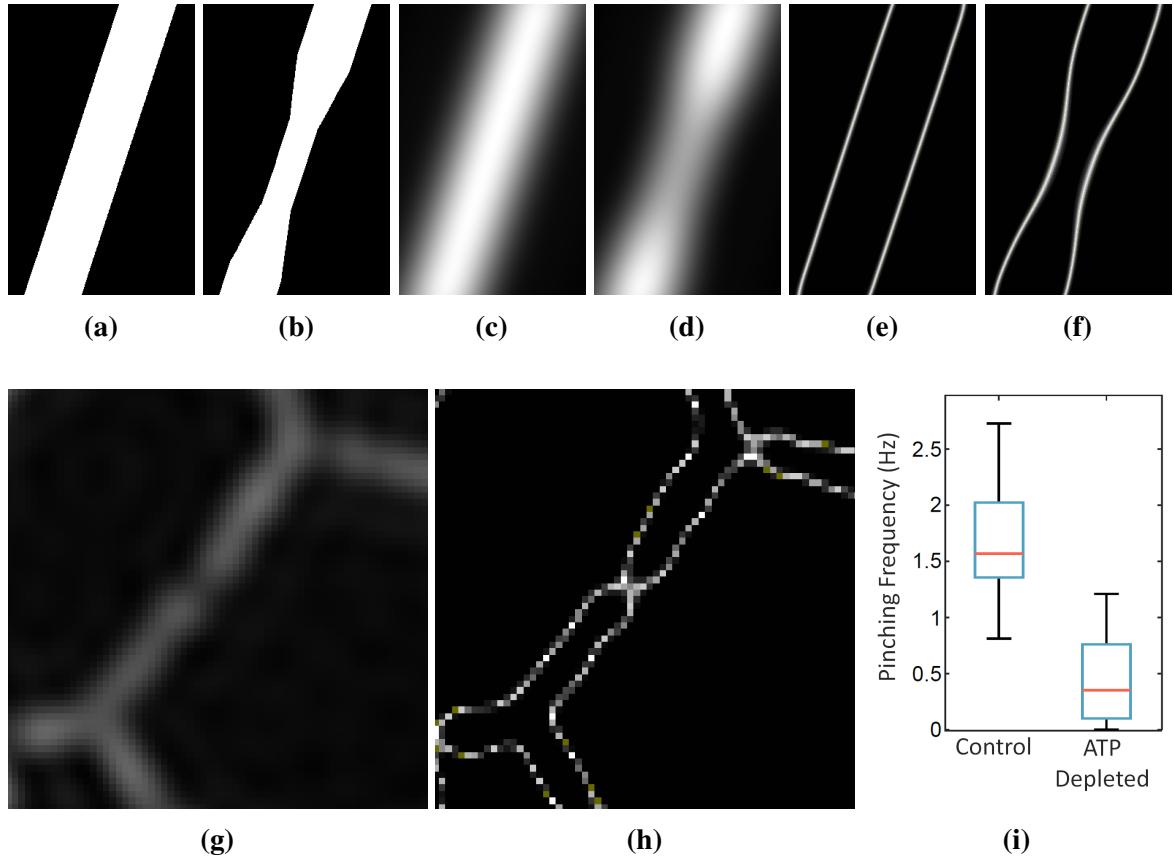


Figure 5.6: A simulation in (a-f) shows that pinching events can be observed even when the pinch width is below the resolution limit of SIM. (a) and (b) show an unpinched and pinched tubule at full simulation resolution, with a width of 80 nm. (c) and (d) show the tubule after convolution with a SIM-resolution point-spread function; (e) and (f) show that using the ‘Edges’ colourmap clearly shows the pinch. (g) shows experimental data of a tubule pinching in COS-7-ERT cells, highlighted in (h) by the ‘Edges’ colourmap. (i) shows that ATP depletion reduces pinching frequency 3.9×, with a similar dependence on ATP as luminal flow velocity seen in Figure 5.2a.

5.2.3 Graphical simulation shows pinching accelerates luminal mixing

In order to visualise the concept of luminal flow, and to understand its purpose, I wrote a 2D Unity program modelling an ER network with labelled proteins.

An ER network was automatically modelled with the following steps:

1. Create binary image of ER network, with method outlined in Section 5.2.1
2. Create a collision boundary from binary image with an edge-tracing algorithm
3. Create a simplified skeleton network from binary image
4. Spawn ‘pinch points’ on the skeleton, aligned with the skeleton direction and the width of the network at that point

The program will accept any binary network image as an input, and will perform these steps automatically using custom C# implementations of the Moore-Neighbourhood tracing algorithm for tracing tubule edges [237], the Zhang-Suen algorithm for skeletonisation [238], and the Ramer–Douglas–Peucker algorithm for simplifying the tubule skeleton and edges [239, 240].

Dynamics are modelled in the network at a particle level. Each particle which is spawned in the network has a physics script attached to it controlling its diffusive motion and its flow velocity when in the vicinity of a pinch.

Diffusive motion was modelled as Brownian motion [241]. After travelling a random distance, sampled from a distribution centred on the mean free path, particles randomly change direction as though collided into by an invisible particle [242]. Diffusion properties

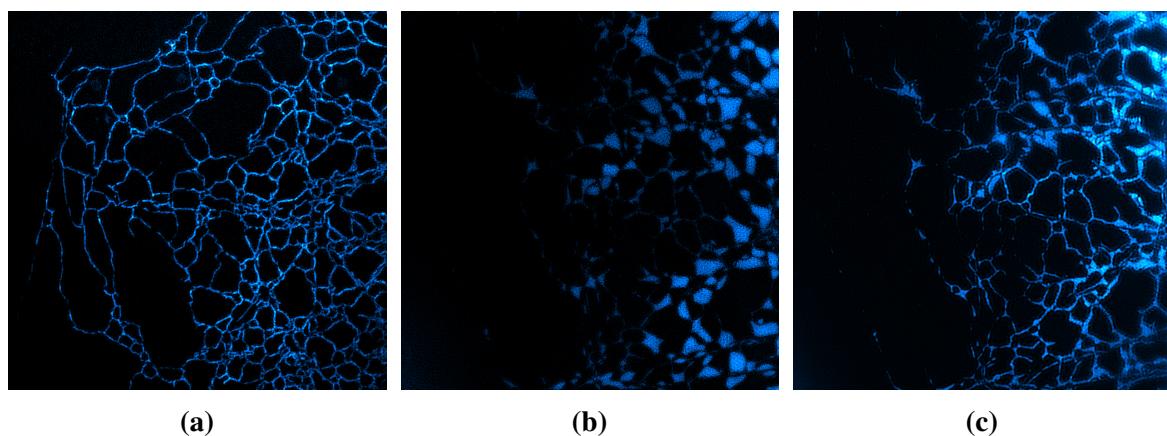


Figure 5.7: When the cell shown in (a) is strongly perturbed from normal conditions, for example by phototoxicity or calcium manipulation, extreme pinching causes blebbing in the ER. (b) shows the cell after imaging for 30 s with 50 W cm^{-2} 561 nm laser light. When the perturbation is removed, the ER recovers its normal shape; (c) shows the same cell after 10 min of recovery time. This experiment confirms the ability of ER to undergo reversible pinching.

are set in the simulation with two of either mean free path δ , time between collisions τ , mean velocity v , or the diffusion coefficient D , related by $D = \frac{\delta^2}{2\tau} = \frac{v\delta}{2}$ [243, ch. 19].

Pinching of the ER tubules simulated a force on particles in the vicinity of the pinch point. As the tubule contracts, this force acts away from the pinch, in a direction along the tubule, and decays exponentially over time. When the pinch point relaxes, the force acts in the opposite direction, sucking particles towards the pinch point. The magnitude of the force is an adjustable parameter, to control overall energy in the system.

Inelastic collisions were modelled with the Nvidia PhysX engine [244]. Coefficient of restitution can be adjusted for each material, such that a collision with the ER membrane is different to a particle-particle collision.

With this physics model, it was possible to simulate different scenarios to understand experimental observations. Firstly, the model was executed without any pinching, simulating purely diffusional motion. In the second scenario, pinching physics was turned on, such that particles in the vicinity of a pinch point experienced a pushing force as the tubule pinched or a pulling force as it relaxed. The parameters were tuned to ensure an equal mean particle speed in both scenarios for a fair comparison.

The two scenarios were compared by measuring the distribution of particles in the network over time, as shown in Figures 5.8 and 5.9. Particles starting at the left hand side of the network were labelled blue; particles in the middle, green; and particles starting on the right pink. By counting the number of particles in different areas of the network, a quantitative description of network mixing can be obtained.

Comparing Figures 5.8 and 5.9 shows that pinching provides the ER a faster way of moving particles stochastically around the network. Under Brownian motion alone, after one minute of simulation pink particles which started on the right-hand-side of the network make up just 6% of particles in the left-hand third of the network. Conversely with ER pinching dynamics, after one minute pink particles make up 24% of particles on the left-hand-side. Pinching dynamics allow particles are able to travel greater distances in the network in less time, even with the same average speed as the diffusional model.

Furthermore, particle motion with ER pinching results in homogeneous mixing in a shorter space of time. After 30 s of simulation with pinching, the middle section of the graph is 84% mixed - where mixing percentage is measured as the sum of the second- and third-most abundant particle colour in an area divided by $2 \times$ the most abundant particle colour. At the same timepoint under pure Brownian motion, however, the middle section is only 36% mixed; and the edge sections even less so. Figure 5.10 shows how mixing in the middle section develops over time; pinching allows mixing to occur $3 \times$ faster than diffusive motion alone. Overall, particles are mixed faster with ER pinching.

Mixing by Brownian motion

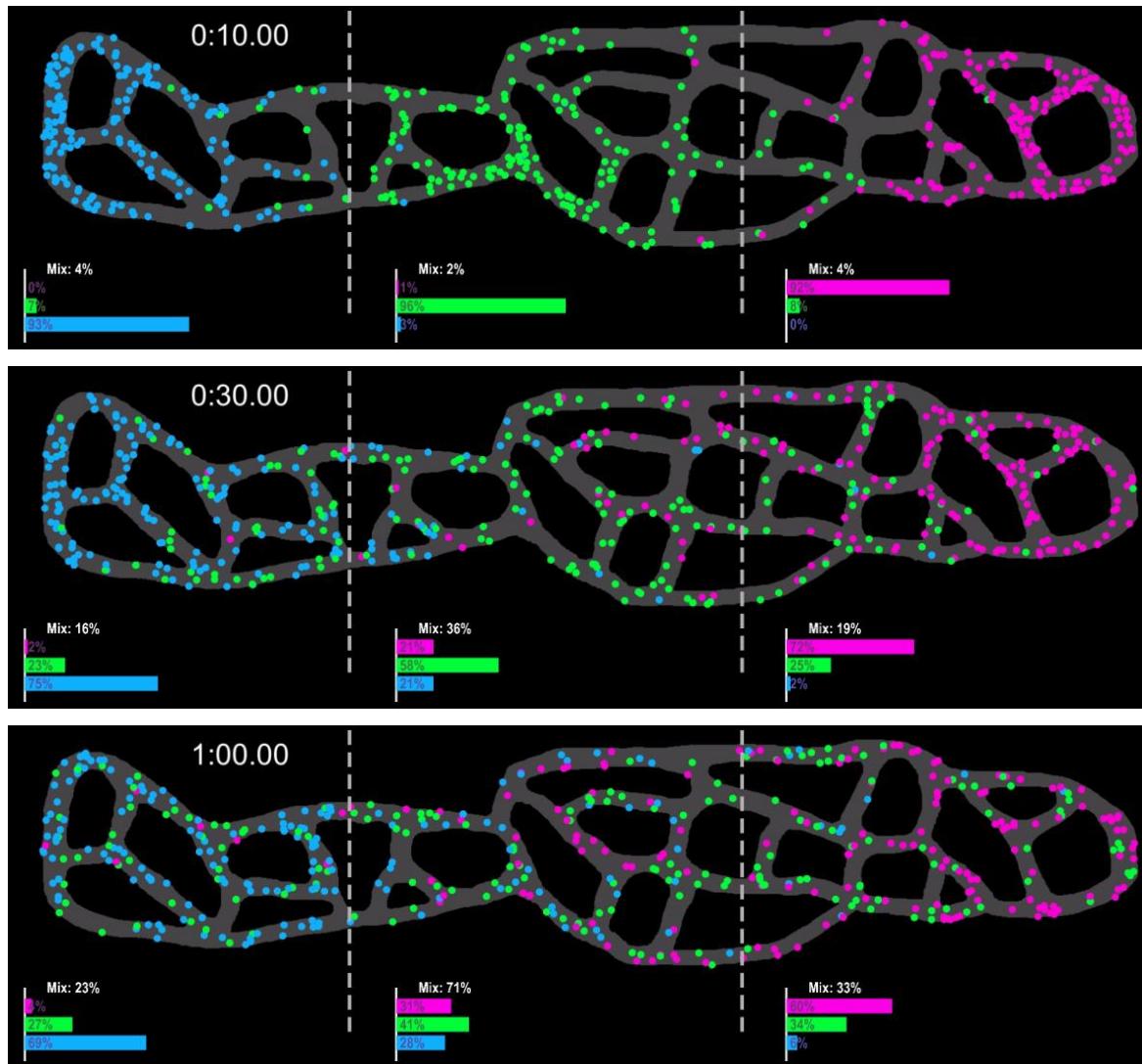


Figure 5.8: After 10 s of mixing by Brownian motion, most particles are still in the third of the network they started in. After 1 min of mixing, the middle section is 71% mixed, although the two edge sections are just 23% and 33% mixed, showing that particles do not transverse the network quickly under Brownian motion alone.

Pinching events are uncoordinated, so that mixing with pinching is stochastic rather directed, which agrees with experimental observations. Particles gain enough speed from a pinch to move away from its influence when the pinch relaxes, so that particles do not stay stuck in one area but are able to freely mix in the network.

To prevent too much energy being added to the network from multiple pinching events, collisions were inelastic and drag was added at network junctions to slow the particles.

Mixing by tubule pinching

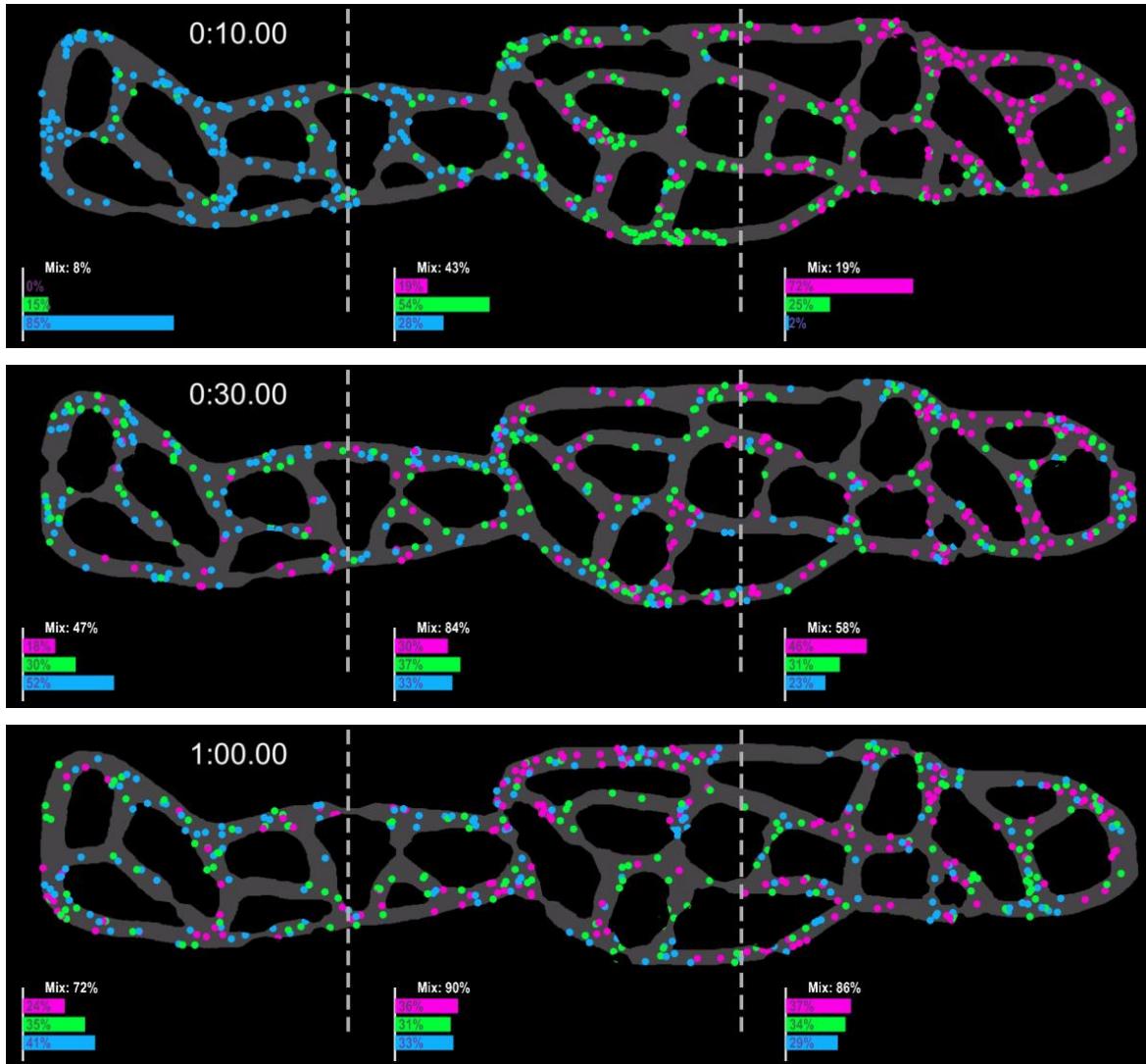


Figure 5.9: When the simulation is executed with tubule pinching, mixing occurs much faster than by Brownian motion despite the same average particle speed.

Parameters were tuned to ensure the same average particle speed in the two mixing scenarios for a fair comparison. A plot of a single particle's trajectory through the network is shown in Figure 5.11, which is consistent with the experimental data in Figure 5.1a, showing fast directed movement along tubules and slower Brownian motion at junctions.

Understanding the reason for ER pinching allows us to associate diseases with ER malfunction. In cells with long projections, such as motor neurons, transporting luminal protein to remote parts of the cell quickly could be vital for healthy cellular function.

5.3 Conclusion

In [230], we present evidence that the movement of proteins in the ER is not diffusive, but also has an ATP-dependent directed-flow component. By utilising the fast high-resolution imaging capabilities of LAG SIM, as well as computational simulation and analysis, this chapter has suggested that contraction of ER tubules is responsible for luminal flow.

Imaging at 11 Hz with 80 nm resolution revealed ER networks in a dynamic equilibrium, constantly growing, shrinking, and rearranging. Network analysis and tracking showed that this is not responsible for flow, however.

The capacity of ER to undergo reversible contractions was revealed when high laser power was used to image ER labelled with TMR, which caused blebbing in the network. Rapid tubule pinching was revealed when re-examining individual segments of ER captured in TIRF-SIM mode. The size and speed of the pinching phenomenon means that high-speed TIRF-SIM is currently the only technique able to visualise these events.

Using the Nvidia PhysX engine to perform simulations of proteins moving in the ER provided insight into the advantages of active flow over diffusion. Tuning the simulation to maintain the same average particle speed in both the pinching and Brownian motion schemes, ensuring a fair comparison, showed that mixing occurs up to $3\times$ faster in a network with

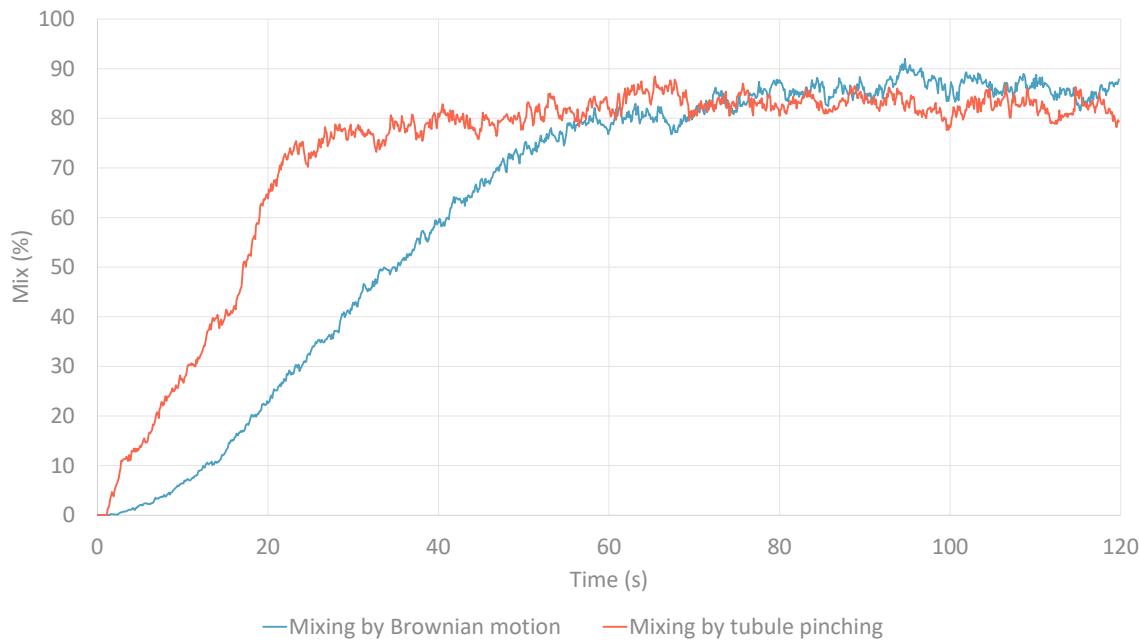


Figure 5.10: Mixing percentage for the middle third of the ER simulation was plotted against time for the two conditions. Mixing occurs about $3\times$ faster with tubule pinching, despite the same average particle speed, allowing particles to reach remote parts of the network faster.

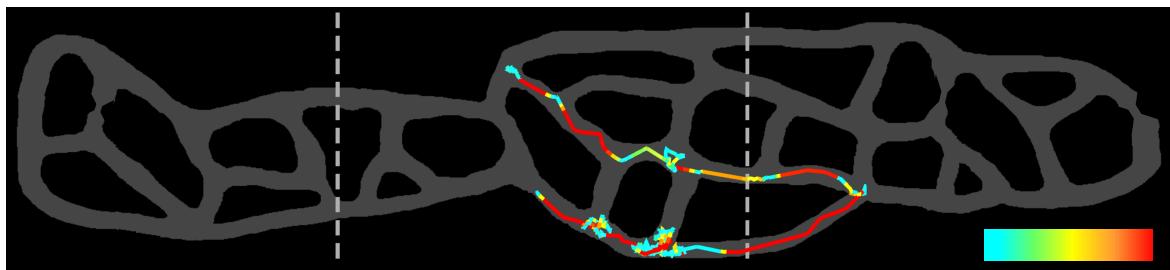


Figure 5.11: A single particle was tracked in the simulation, and its velocity recorded using the same colour scale as Figure 5.1, where red shows fast particle velocity and blue slow velocity. The simulated model is in good agreement with experimental data, with fast directed movement along tubules and slower Brownian motion at junctions. Colour bar from slow (blue) to fast (red).

pinching. Pinching therefore reduces the time taken for proteins to reach remote areas of the network, such as by the cell membrane. The interactive simulation is provided online at <https://laseranalyticsgroup.github.io/er>, where model parameters such as pinch size, particle size, and mean free path can be adjusted, for example to input values measured from electron microscopy data.

Since ER is found in most eukaryotic cells, it is unsurprising that its malfunction is associated with a wide range of diseases. By contributing to the community's fundamental understanding of ER mechanisms, we advance our capability for finding cures.

5.4 Future research

The dynamic rearrangement of the ER network has been observed by several groups [81, 245, 246]. Furthermore is it now known that ER exhibits strong contact with all other organelles [224, 247]. Nevertheless, the biochemical mechanisms for network rearrangement are not well understood.

By utilising high-speed multi-colour SIM we have observed lysosomes localised at the endpoints of growing ER tubules, which appear to pull the network into its new shape. Preliminary data, shown in Figure 5.12, indicates that ER tubules grow faster when assisted by lysosomes, and are also more likely to form new junctions by connecting to another area of the network than ER tubules growing without lysosomes. Further investigation into specific proteins involved in this phenomenon are ongoing, as well as calculations into the dynamic forces required for ER network rearrangement.

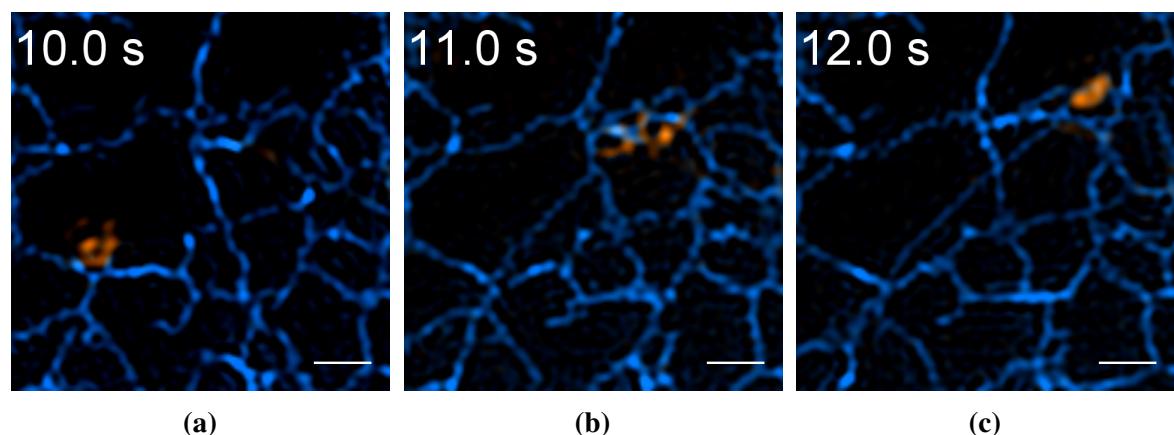


Figure 5.12: Preliminary multi-colour SIM data captured with the Optosplit shows lysosomes, coloured in orange, forming strong contacts with ER tubule endpoints, suggesting that lysosomes assist with rearrangement of the ER network. Scalebars are 1 μm .

Chapter 6

Conclusion

This thesis has detailed the development of two tools, LAG SIM and FPBioimage, and demonstrated their application to answering biological questions about utilising MOFs for drug delivery and discovering a new mechanism for luminal ER transport.

Over the past 350 years, optical microscopy has provided us with knowledge of the biology of life through its unique capability of observing living systems that are invisible to the naked eye. The last two decades have seen a number of techniques developed which surpass the diffraction limit of light, allowing new observations to further develop our understanding. This vital contribution was recognised with a Nobel prize in 2014 awarded to PALM/STORM and STED, two techniques which can resolve features as small as 20 nm.

Of course, circumventing this physical limit is not without compromise, and for these techniques temporal resolution is sacrificed for the gain in spatial resolution. Whilst a diffraction-limited widefield microscope can capture videos at 100 frames per second, a STORM image takes 5 min for each frame to achieve its 10 \times resolution enhancement. This is not useful for observing fast dynamic events taking place within living cells.

Structured illumination microscopy (SIM) produces images with twice the resolution of widefield images, requiring 9 raw images per resolution-enhanced frame. This compromise between ultimate resolution and ultimate speed gives SIM a unique capability to capture dynamic sub-cellular events. Furthermore, SIM can provide optical sectioning, necessary for acquisitions to be reconstructed in 3D.

Chapter 2 described LAG SIM, a versatile and user-friendly structured illumination microscope, capable of imaging at 11 Hz with a resolution of 80 nm. The microscope provides optical sectioning, either physically removing out-of-focus light with TIRF illumination, or computationally by attenuation of the OTF as part of the SIM reconstruction process. Use of an Optosplit before the camera enables simultaneous 3-colour imaging, allowing dynamic

interactions of organelles to be studied without any sacrifice of either spatial or temporal resolution.

Super-resolution microscopes can be difficult to use. There is often a steep learning curve for new users simply to acquire images. Even once the raw data has been captured, reconstructing it into super-resolved images usually requires expert skill and practice.

LAG SIM has been designed to be operated by non-expert users. The LabVIEW interface makes capturing time-lapses, z-stacks, bookmarked cells, and large field-of-view mosaics a simple process, with the complicated synchronisation of hardware completely hidden from the user. Furthermore, the SIM reconstruction process has been simplified with an ImageJ plugin specifically designed for the LAG SIM, including a tool for finding optimal parameters for artefact-free reconstruction, and another for completing batch reconstruction jobs.

LAG SIM's unique advantages over widefield microscopes, combined with its ease-of-use, have lead to a wide variety of biological investigations performed with the instrument. Chapter 2 included a showcase of experiments selected to show the range of applications for which LAG SIM is suited. Suggested parameters for the LAG SIM ImageJ plugin have been provided alongside each reconstruction as a resource to future users seeking to answer their own biological questions.

The optical sectioning capability of LAG SIM allows extraordinary 3D datasets to be constructed, revealing information that cannot be observed in 2D slices. Sharing this data with collaborators around the world, however, is not a trivial tasks. Large file sizes and complicated or expensive software make sharing 3D data a notorious problem in the microscopy community.

Chapter 3 presented my new volumetric rendering program, FPBioimage, with the unique selling point that it runs in a web browser. This allows remote users to explore the data interactively, viewing it from any perspective and utilising a number of advanced rendering features such as transparency and cutting planes. Personal exploration provides an immersive experience, giving users a much clearer understanding of the data than a 2D projection or pre-recorded video can provide.

FPBioimage was designed with user-friendly operation as the first priority, not only for those viewing the data but also for researchers who need to share their own volumetric data. In combination with a suite of plugins and mobile apps built to complement the main tool, FPBioimage has made sharing and publishing 3D volumetric data a one-click process.

FPBioimage is not restricted to the field of microscopy, and enables web-based rendering of any volumetric data. The tool has found use in a diverse range of applications, including medical imaging and computer-generated 3D simulations. With a focus on creating a tool

that is intuitive and easy to use, all researchers around the world can now immediately view their colleagues experimental results, even when separated by thousands of miles.

Chapter 4 detailed an application of these two tools, LAG SIM and FPBioimage, for novel cancer treatments, demonstrating their capacity for facilitating hypothesis-driven research.

A problem of some potentially therapeutic drugs, such as siRNA or DCA, is their breakdown in extra-cellular media. This either means that drugs are not viable for use as a medicine, or that treatments require high doses of the drug which leads to unwelcome side effects. To allow these drugs to arrive inside the cell without degradation, they can be delivered inside carrier molecules. The large pore size of metal organic frameworks (MOFs) makes them an ideal candidate for such a drug delivery system.

Using optical sectioning SIM, Chapter 4 showed that these MOF complexes are successfully endocytosed by the cell, and can therefore be used as effective drug delivery systems, with the potential of treating a wide range of cancers. All volumetric imaging data collected as part of the study was published online with FPBioimage, as an example of how the tool can be used to allow others to examine the data and confirm the findings themselves.

One of LAG SIM's unique capabilities is fast, multi-colour TIRF microscopy in high resolution. This was utilised in Chapter 5 to reveal a new phenomenon in the endoplasmic reticulum (ER) for transporting proteins around the cell.

Collaborators had observed directed laminar flow in the ER. Acquisitions made on LAG SIM at 11 Hz with 80 nm resolution revealed pinching of the ER membrane which correlates with luminal flow speed.

This new discovery was investigated further using a computational model, which showed that pinching reduces the time taken for molecules to travel through the network up to $3\times$ compared to Brownian particle motion. This contribution to the community's fundamental understanding of cell biology will further assist the effort to find cures for the wide range of diseases associated with ER malfunction.

My past three years work, presented in this document as a resource to those who require the tools to answer their own biological questions, has already had a meaningful impact on the community's knowledge and understanding. This is evidenced by publications in several well-respected journals [107, 156, 203, 230, 248]. Ongoing experiments resulting from these investigations give me confidence that this work will continue to provide exciting discoveries.

References

- [1] Russell L De Valois. *Spatial vision*. Oxford psychology series ; 14. Oxford, 1990.
- [2] Antoni van Leeuwenhoek. *The select works of Anthony van Leeuwenhoek: containing his microscopical discoveries in many of the works of nature*, volume 1. Translator: Samuel Hoole, 1800.
- [3] Robert Hooke. *Micrographia; or, Some physiological descriptions of minute bodies made by magnifying glasses*. Jo. Martyn, and Ja. Allestry, London, England, 1665.
- [4] Jan Swammerdam. *The book of nature; or, the history of insects: reduced to distinct classes, confirmed by particular instances, and illustrated with copper-plates*. By John Swammerdam, M.D. With the life of the author, by Herman Boerhaave, M.D. Translated by Thomas Flloyd. Revised and improved by John Hill, M.D. C. G. Seyffert, London, England, 1758.
- [5] John Dollond. Xiv. a letter from Mr. John Dollond to Mr. James Short, F. R. S. concerning an improvement of refracting telescopes. *Philosophical Transactions of the Royal Society*, 48:103–107, 1753.
- [6] M. Daumas. *Scientific Instruments of the Seventeenth and Eighteenth Centuries*. Books that matter. Praeger Publishers, 1972.
- [7] Ernst Abbe. Beiträge zur Theorie des Mikroskops und der mikroskopischen Wahrnehmung. *Archiv für mikroskopische Anatomie*, 9(1):413–418, 1873.
- [8] R Reinhold. Configuration for the enlarged imaging of objects by electron beams. *Patent DE*, 906737:1931–05, 1931.
- [9] Oliver C Wells and David C Joy. The early history and future of the SEM. *Surface and Interface Analysis: An International Journal devoted to the development and application of techniques for the analysis of surfaces, interfaces and thin films*, 38(12–13):1738–1742, 2006.
- [10] L. Reimer. *Transmission Electron Microscopy: Physics of Image Formation and Microanalysis*. Springer Series in Optical Sciences. Springer Berlin Heidelberg, 2013.
- [11] J. Kuo. *Electron Microscopy: Methods and Protocols*. Methods in molecular biology. Humana Press, 2007.
- [12] Niels de Jonge and Diana B Peckys. Live cell electron microscopy is probably impossible. *ACS nano*, 10(10):9061–9063, 2016.

- [13] Richard N Day and Michael W Davidson. *The fluorescent protein revolution*. CRC Press, 2014.
- [14] J.R. Lakowicz. *Principles of Fluorescence Spectroscopy*. Springer US, 2007.
- [15] D. Halliday, J. Walker, and R. Resnick. *Principles of Physics*. Wiley, 2010.
- [16] JS Ploem. The use of a vertical illuminator with interchangeable dichroic mirrors for fluorescence microscopy with incidental light. *Zeitschrift fur wissenschaftliche Mikroskopie und mikroskopische Technik*, 68(3):129–142, 1967.
- [17] Kenneth W Dunn, Małgorzata M Kamocka, and John H McDonald. A practical guide to evaluating colocalization in biological microscopy. *American Journal of Physiology-Cell Physiology*, 300(4):C723–C742, 2011.
- [18] Anda Cornea and P Michael Conn. *Fluorescence Microscopy: Super-Resolution and other Novel Techniques*. Elsevier, 2014.
- [19] Eric Betzig, George H Patterson, Rachid Sougrat, O Wolf Lindwasser, Scott Olenych, Juan S Bonifacino, Michael W Davidson, Jennifer Lippincott-Schwartz, and Harald F Hess. Imaging intracellular fluorescent proteins at nanometer resolution. *Science*, 313(5793):1642–1645, 2006.
- [20] Michael J Rust, Mark Bates, and Xiaowei Zhuang. Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM). *Nature methods*, 3(10):793, 2006.
- [21] Mike Heilemann, Sebastian Van De Linde, Mark Schüttelpe, Robert Kasper, Britta Seefeldt, Anindita Mukherjee, Philip Tinnefeld, and Markus Sauer. Subdiffraction-resolution fluorescence imaging with conventional fluorescent probes. *Angewandte Chemie International Edition*, 47(33):6172–6176, 2008.
- [22] Mats GL Gustafsson. Surpassing the lateral resolution limit by a factor of two using structured illumination microscopy. *Journal of microscopy*, 198(2):82–87, 2000.
- [23] Laurence J Young, Florian Ströhl, and Clemens F Kaminski. A guide to structured illumination TIRF microscopy at high speed with multiple colors. *Journal of visualized experiments: JoVE*, (111), 2016.
- [24] Eldho Markose, B Vikraman, and M Veerabahu. Three dimensional CT reconstruction: a comparison between 2D, 3D CT and original anatomical structures. *Journal of maxillofacial and oral surgery*, 8(1):8–12, 2009.
- [25] Yi Zhang, Yan Zhou, Xinhua Yang, Peng Tang, Quanguang Qiu, Yong Liang, and Jun Jiang. Thin slice three dimensional (3D) reconstruction versus CT 3D reconstruction of human breast cancer. *The Indian journal of medical research*, 137(1):57, 2013.
- [26] V Rangarajan et al. Three dimensional (3D) CT reconstruction in cancer imaging. *Indian J Med Res*, 137(1):10–11, 2013.
- [27] JG White, WB Amos, and M Fordham. An evaluation of confocal versus conventional imaging of biological structures by fluorescence light microscopy. *The Journal of Cell Biology*, 105(1):41–48, 1987.

- [28] Minsky Marvin. Microscopy apparatus, December 19 1961. US Patent 3,013,467.
- [29] Jan Huisken, Jim Swoger, Filippo Del Bene, Joachim Wittbrodt, and Ernst HK Stelzer. Optical sectioning deep inside live embryos by selective plane illumination microscopy. *Science*, 305(5686):1007–1009, 2004.
- [30] James Sharpe, Ulf Ahlgren, Paul Perry, Bill Hill, Allyson Ross, Jacob Hecksher-Sørensen, Richard Baldock, and Duncan Davidson. Optical projection tomography as a tool for 3D microscopy and gene expression studies. *Science*, 296(5567):541–545, 2002.
- [31] Bo Huang, Wenqin Wang, Mark Bates, and Xiaowei Zhuang. Three-dimensional super-resolution imaging by stochastic optical reconstruction microscopy. *Science*, 319(5864):810–813, 2008.
- [32] Sri Rama Prasanna Pavani, Michael A Thompson, Julie S Biteen, Samuel J Lord, Na Liu, Robert J Twieg, Rafael Piestun, and WE Moerner. Three-dimensional, single-molecule fluorescence imaging beyond the diffraction limit by using a double-helix point spread function. *Proceedings of the National Academy of Sciences*, 106(9):2995–2999, 2009.
- [33] Gleb Shtengel, James A Galbraith, Catherine G Galbraith, Jennifer Lippincott-Schwartz, Jennifer M Gillette, Suliana Manley, Rachid Sougrat, Clare M Waterman, Pakorn Kanchanawong, Michael W Davidson, et al. Interferometric fluorescent super-resolution microscopy resolves 3d cellular ultrastructure. *Proceedings of the National Academy of Sciences*, 106(9):3125–3130, 2009.
- [34] Erdal Toprak, Hamza Balci, Benjamin H Blehm, and Paul R Selvin. Three-dimensional particle tracking via bifocal imaging. *Nano letters*, 7(7):2043–2045, 2007.
- [35] Shu Jia, Joshua C Vaughan, and Xiaowei Zhuang. Isotropic three-dimensional super-resolution imaging with a self-bending point spread function. *Nature photonics*, 8(4):302, 2014.
- [36] Philipp J Keller, Annette D Schmidt, Joachim Wittbrodt, and Ernst HK Stelzer. Reconstruction of zebrafish early embryonic development by scanned light sheet microscopy. *Science*, 322(5904):1065–1069, 2008.
- [37] Fang Huang, George Sirinakis, Edward S Allgeyer, Lena K Schroeder, Whitney C Duim, Emil B Kromann, Thomy Phan, Felix E Rivera-Molina, Jordan R Myers, Irnov Irnov, Martin J Booth, Joerg Bewersdorf, et al. Ultra-high resolution 3D imaging of whole cells. *Cell*, 166(4):1028–1040, 2016.
- [38] Kelvinsong. File:Animal Cell.svg - Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Animal_Cell.svg, 2012. Accessed: 2018-10-10.
- [39] Frithjof Hammersen and Johannes Sobotta. *Histology: a color atlas of cytology, histology, and microscopic anatomy*. Urban & Schwarzenberg, 4 edition, 2002.
- [40] Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter. *Molecular Biology of the Cell, Sixth Edition*. Garland Science, 6 edition, 2015.

- [41] Jan Kapuscinski. DAPI: a DNA-specific fluorescent probe. *Biotechnic & Histochemistry*, 70(5):220–233, 1995.
- [42] Gražvydas Lukinavičius, Claudia Blaukopf, Elias Pershagen, Alberto Schena, Luc Reymond, Emmanuel Derivery, Marcos Gonzalez-Gaitan, Elisa D’Este, Stefan W Hell, Daniel Wolfram Gerlich, et al. SiR–Hoechst is a far-red DNA stain for live-cell nanoscopy. *Nature communications*, 6:8497, 2015.
- [43] Karen G Porter and Yvette S Feig. The use of DAPI for identifying and counting aquatic microflora. *Limnology and oceanography*, 25(5):943–948, 1980.
- [44] B. Alberts, D. Bray, K. Hopkin, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Essential Cell Biology, Fourth Edition*. Taylor & Francis Group, 2013.
- [45] Yoshiaki Yano and Katsumi Matsuzaki. Tag–probe labeling methods for live-cell imaging of membrane proteins. *Biochimica et Biophysica Acta (BBA)-Biomembranes*, 1788(10):2124–2131, 2009.
- [46] Jae-Jung Lee, Jyunghyun Son, Hyung-Ho Ha, and Young-Tae Chang. Fluorescent labeling of membrane proteins on the surface of living cells by a self-catalytic glutathione s-transferase omega 1 tag. *Molecular BioSystems*, 7(4):1270–1276, 2011.
- [47] Ingrid Chamma, Olivier Rossier, Grégory Giannone, Olivier Thoumine, and Matthieu Sainlos. Optimized labeling of membrane proteins for applications to super-resolution imaging in confined cellular environments using monomeric streptavidin. *nature protocols*, 12(4):748, 2017.
- [48] M. Marsh. *Endocytosis*. Frontiers in molecular biology. Oxford University Press, 2001.
- [49] M Marsh and HT McMahon. The structural era of endocytosis. *Science*, 285(5425):215–220, 1999.
- [50] Harvey T McMahon and Emmanuel Boucrot. Molecular mechanism and physiological functions of clathrin-mediated endocytosis. *Nature reviews Molecular cell biology*, 12(8):517, 2011.
- [51] Michael J Geisow and W Howard Evans. pH in the endosome: measurements during pinocytosis and receptor-mediated endocytosis. *Experimental cell research*, 150(1):36–46, 1984.
- [52] Jeremy A Pike, Iain B Styles, Joshua Z Rappoport, and John K Heath. Quantifying receptor trafficking and colocalization with confocal microscopy. *Methods*, 115:42–54, 2017.
- [53] R.D. Dyson. *Cell biology: a molecular approach*. Allyn and Bacon, 1978.
- [54] Lindsey Costantini and Erik Snapp. Probing endoplasmic reticulum dynamics using fluorescence imaging and photobleaching techniques. *Current protocols in cell biology*, 60(1):21–7, 2013.

- [55] A.W. Murray and T. Hunt. *The Cell Cycle: An Introduction*. Oxford University Press, 1993.
- [56] O Warburg. Über den Stoffwechsel der Tumoren (London: Constable), 1930.
- [57] David S Goodsell. Inside a living cell. *Trends in biochemical sciences*, 16:203–206, 1991.
- [58] Anna Melkov and Uri Abdu. Regulation of long-distance transport of mitochondria along microtubules. *Cellular and Molecular Life Sciences*, pages 1–14, 2017.
- [59] Marcus Dyba, Stefan Jakobs, and Stefan W Hell. Immunofluorescence stimulated emission depletion microscopy. *Nature biotechnology*, 21(11):1303, 2003.
- [60] Eva Wegel, Antonia Göhler, B Christoffer Lagerholm, Alan Wainman, Stephan Uphoff, Rainer Kaufmann, and Ian M Dobbie. Imaging cellular structures in super-resolution with SIM, STED and Localisation Microscopy: A practical comparison. *Scientific reports*, 6:27290, 2016.
- [61] J. Pawley. *Handbook of Biological Confocal Microscopy*. Springer US, 2012.
- [62] T. Wilson and C. Sheppard. *Theory and Practice of Scanning Optical Microscopy*. Academic Press, 1984.
- [63] A. Periasamy. *Methods in Cellular Imaging*. Methods in Physiology. Springer New York, 2013.
- [64] Daniel Axelrod. Cell-substrate contacts illuminated by total internal reflection fluorescence. *The Journal of Cell Biology*, 89(1):141–145, 1981.
- [65] Makio Tokunaga, Naoko Imamoto, and Kumiko Sakata-Sogawa. Highly inclined thin illumination enables clear single-molecule imaging in cells. *Nature methods*, 5(2):159, 2008.
- [66] Mark AA Neil, Rimas Juškaitis, and Tony Wilson. Method of obtaining optical sectioning by using structured light in a conventional microscope. *Optics letters*, 22(24):1905–1907, 1997.
- [67] Dejan Karadaglić and Tony Wilson. Image formation in structured illumination wide-field fluorescence microscopy. *Micron*, 39(7):808–818, 2008.
- [68] Benjamin Thomas, Michelle Momany, and Peter Kner. Optical sectioning structured illumination microscopy with enhanced sensitivity. *Journal of Optics*, 15(9):094004, 2013.
- [69] Dan Dan, Baoli Yao, and Ming Lei. Structured illumination microscopy for super-resolution and optical sectioning. *Chinese science bulletin*, 59(12):1291–1307, 2014.
- [70] Michael Shaw, Lydia Zajiczek, and Kevin O’Holleran. High speed structured illumination microscopy in optically thick samples. *Methods*, 88:11–19, 2015.
- [71] E. Hecht. *Optics*. Pearson Education, Incorporated, 4 edition, 2002.

- [72] W Lukosz. Optical systems with resolving powers exceeding the classical limit. *JOSA*, 56(11):1463–1471, 1966.
- [73] Rainer Heintzmann and Christoph G Cremer. Laterally modulated excitation microscopy: improvement of resolution by using a diffraction grating. In *Optical Biopsies and Microscopic Techniques III*, volume 3568, pages 185–197. International Society for Optics and Photonics, 1999.
- [74] Kai Wicker, Ondrej Mandula, Gerrit Best, Reto Fiolka, and Rainer Heintzmann. Phase optimisation for structured illumination microscopy. *Optics express*, 21(2):2032–2049, 2013.
- [75] Mats GL Gustafsson. Nonlinear structured-illumination microscopy: wide-field fluorescence imaging with theoretically unlimited resolution. *Proceedings of the National Academy of Sciences*, 102(37):13081–13086, 2005.
- [76] Rainer Heintzmann and Thomas Huser. Super-resolution structured illumination microscopy. *Chemical reviews*, 117(23):13890–13908, 2017.
- [77] Bo-Jui Chang, Li-Jun Chou, Yun-Ching Chang, and Su-Yu Chiang. Isotropic image in structured illumination microscopy patterned with a spatial light modulator. *Optics express*, 17(17):14710–14721, 2009.
- [78] C.S. Williams and O.A. Becklund. *Introduction to the Optical Transfer Function*. SPIE Press monograph. SPIE Press, 2002.
- [79] Mats GL Gustafsson, Lin Shao, Peter M Carlton, CJ Rachel Wang, Inna N Golubovskaya, W Zacheus Cande, David A Agard, and John W Sedat. Three-dimensional resolution doubling in wide-field fluorescence microscopy by structured illumination. *Biophysical journal*, 94(12):4957–4970, 2008.
- [80] Erwin Kreyszig. *Advanced engineering mathematics*. Wiley India Pvt. Limited, 8 edition, 2006.
- [81] Jonathon Nixon-Abell, Christopher J Obara, Aubrey V Weigel, Dong Li, Wesley R Legant, C Shan Xu, H Amalia Pasolli, Kirsten Harvey, Harald F Hess, Eric Betzig, et al. Increased spatiotemporal resolution reveals highly dynamic dense tubular matrices in the peripheral ER. *Science*, 354(6311):aaf3928, 2016.
- [82] Christiaan H Righolt, Johan A Slotman, Ian T Young, Sabine Mai, Lucas J van Vliet, and Sjoerd Stallinga. Image filtering in structured illumination microscopy using the Lukosz bound. *Optics express*, 21(21):24431–24451, 2013.
- [83] Victor Perez, Bo-Jui Chang, and Ernst Hans Karl Stelzer. Optimal 2D-SIM reconstruction by two filtering steps with Richardson–Lucy deconvolution. *Scientific reports*, 6:37149, 2016.
- [84] Nadya Chakrova, Bernd Rieger, and Sjoerd Stallinga. Deconvolution methods for structured illumination microscopy. *JOSA A*, 33(7):B12–B20, 2016.

- [85] Vanessa Blake. Official opening - Department of Chemical Engineering and Biotechnology. <https://www.ceb.cam.ac.uk/news/news-list/official-opening>, 2018. Accessed: 2018-06-25.
- [86] Shen Y.R. *The principles of nonlinear optics*. Wiley, New York, 1984.
- [87] Kevin O'Holleran and Michael Shaw. Polarization effects on contrast in structured illumination microscopy. *Optics letters*, 37(22):4603–4605, 2012.
- [88] Rajaram Nityananda. The interference of polarised light. *Resonance*, 18(4):309–322, 2013.
- [89] Masud Mansuripur. Effects of high-numerical-aperture focusing on the state of polarization in optical and magneto-optic data storage systems. *Applied Optics*, 30(22):3154–3162, 1991.
- [90] Jean M. Bennett. *Handbook of Optics*, volume 2. McGraw-Hill Inc., 2 edition, 1995.
- [91] Firas Mubaid and Claire M Brown. Less is more: longer exposure times with low light intensity is less photo-toxic. *Microscopy Today*, 25(6):26–35, 2017.
- [92] Labstep Ltd. Labstep | The ultimate tool for scientists - Research, discover and collaborate. <https://www.labstep.com/>, 2018. Accessed: 2018-10-22.
- [93] Lin Shao, Peter Kner, E Hesper Rego, and Mats GL Gustafsson. Super-resolution 3D microscopy of live whole cells using structured illumination. *Nature methods*, 8(12):1044, 2011.
- [94] Jordan R Beach, Lin Shao, Kirsten Remmert, Dong Li, Eric Betzig, and John A Hammer III. Nonmuscle myosin II isoforms coassemble in living cells. *Current Biology*, 24(10):1160–1166, 2014.
- [95] J. Sanders and E. Kandrot. *CUDA by Example: An Introduction to General-Purpose GPU Programming, Portable Documents*. Pearson Education, 2010.
- [96] Kevin O'Holleran and Michael Shaw. Optimized approaches for optical sectioning and resolution enhancement in 2D structured illumination microscopy. *Biomedical optics express*, 5(8):2580–2590, 2014.
- [97] Marcel Müller, Viola Mönkemöller, Simon Hennig, Wolfgang Hübner, and Thomas Huser. Open-source image reconstruction of super-resolution structured illumination microscopy data in ImageJ. *Nature communications*, 7:10980, 2016.
- [98] Johannes Schindelin, Ignacio Arganda-Carreras, Erwin Frise, Verena Kaynig, Mark Longair, Tobias Pietzsch, Stephan Preibisch, Curtis Rueden, Stephan Saalfeld, Benjamin Schmid, et al. Fiji: an open-source platform for biological-image analysis. *Nature methods*, 9(7):676, 2012.
- [99] Marcel Müller. fairSIM/fairSIM: Free analysis and interactive reconstruction for structured illumination microscopy, 2017. Accessed: 2018-06-26.
- [100] Marcus Fantham. Index of /LagSIM. <http://sites.imagej.net/LagSIM/>, 2018. Accessed: 2018-06-26.

- [101] Siân Culley, David Albrecht, Caron Jacobs, Pedro Matos Pereira, Christophe Leterrier, Jason Mercer, and Ricardo Henriques. Quantitative mapping and minimization of super-resolution optical imaging artifacts. *Nature methods*, 15(4):263, 2018.
- [102] R.G. Brown and P.Y.C. Hwang. *Introduction to Random Signals and Applied Kalman Filtering with MATLAB Exercises, 4th Edition*. Wiley, 2012.
- [103] S Eichstädt, F Schmähling, G Wübbeler, K Anhalt, L Bünger, U Krüger, and C Elster. Comparison of the Richardson–Lucy method and a classical approach for spectrometer bandpass correction. *Metrologia*, 50(2):107, 2013.
- [104] William Hadley Richardson. Bayesian-based iterative method of image restoration. *JOSA*, 62(1):55–59, 1972.
- [105] Leon B Lucy. An iterative technique for the rectification of observed distributions. *The astronomical journal*, 79:745, 1974.
- [106] CJR Sheppard and Min Gu. The significance of 3-D transfer functions in confocal scanning microscopy. *Journal of Microscopy*, 165(3):377–390, 1992.
- [107] Michelle H Teplensky, Marcus Fantham, Peng Li, Timothy C Wang, Joshua P Mehta, Laurence J Young, Peyman Z Moghadam, Joseph T Hupp, Omar K Farha, Clemens F Kaminski, and David Fairen-Jimenez. Temperature treatment of highly porous zirconium-containing metal–organic frameworks extends drug delivery release. *Journal of the American Chemical Society*, 139(22):7522–7532, 2017.
- [108] Elizabeth A Specht, Esther Braselmann, and Amy E Palmer. A critical and comparative review of fluorescent tools for live-cell imaging. *Annual review of physiology*, 79:93–117, 2017.
- [109] Juan Wang, Yun Wei, Shasha Zhao, Ying Zhou, Wei He, Yang Zhang, and Wensheng Deng. The analysis of viability for mammalian cells treated at different temperatures and its application in cell shipment. *PLoS one*, 12(4):e0176120, 2017.
- [110] Claudia Orellana-Tavra, Emma F Baxter, Tian Tian, Thomas D Bennett, Nigel KH Slater, Anthony K Cheetham, and David Fairen-Jimenez. Amorphous metal–organic frameworks for drug delivery. *Chemical Communications*, 51(73):13878–13881, 2015.
- [111] James Jonkman and Claire M Brown. Any way you slice it—a comparison of confocal microscopy techniques. *Journal of biomolecular techniques: JBT*, 26(2):54, 2015.
- [112] Gunther Zimmermann, Björn Papke, Shehab Ismail, Nachiket Vartak, Anchal Chandra, Maike Hoffmann, Stephan A Hahn, Gemma Triola, Alfred Wittinghofer, Philippe IH Bastiaens, et al. Small molecule inhibition of the KRAS–PDE δ interaction impairs oncogenic KRAS signalling. *Nature*, 497(7451):638, 2013.
- [113] Julian Downward. Targeting ras signalling pathways in cancer therapy. *Nature Reviews Cancer*, 3(1):11, 2003.
- [114] Onno Kranenburg. The KRAS oncogene: past, present, and future. *Biochimica et biophysica acta*, 1756(2):81–82, 2005.

- [115] Z.W. Hall. *An Introduction to Molecular Neurobiology*. Sinauer Associates, 1992.
- [116] Ichiji Tasaki. The electro-saltatory transmission of the nerve impulse and the effect of narcosis upon the nerve fiber. *American Journal of Physiology-Legacy Content*, 127(2):211–227, 1939.
- [117] Kinuko Suzuki, Diane Armao, Jeffrey A Stone, and Suresh K Mukherji. Demyelinating diseases, leukodystrophies, and other myelin disorders. *Neuroimaging Clinics of North America*, 11(1):vii–15, 2001.
- [118] Andrew c. File:Neuron with oligodendrocyte and myelin sheath.svg - Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Neuron_with_oligodendrocyte_and_myelin_sheath.svg, 2010. Accessed: 2018-10-10.
- [119] C. Hammond. *Cellular and Molecular Neurobiology*. Elsevier Science, 2012.
- [120] Steven A Goldman and Nicholas J Kuypers. How to make an oligodendrocyte. *Development*, 142(23):3983–3995, 2015.
- [121] Nicoletta Kessaris, Matthew Fogarty, Palma Iannarelli, Matthew Grist, Michael Wegner, and William D Richardson. Competing waves of oligodendrocytes in the forebrain and postnatal elimination of an embryonic lineage. *Nature neuroscience*, 9(2):173, 2006.
- [122] Verena C Wimmer and Andreas Möller. High-resolution confocal imaging in tissue. In *Histology Protocols*, pages 183–191. Springer, 2010.
- [123] Steven L Jacques. Optical properties of biological tissues: a review. *Physics in Medicine & Biology*, 58(11):R37, 2013.
- [124] Fabrice De Chaumont, Stéphane Dallongeville, Nicolas Chenouard, Nicolas Hervé, Sorin Pop, Thomas Provoost, Vannary Meas-Yedid, Praveen Pankajakshan, Timothée Lecomte, Yoann Le Montagner, et al. Icy: an open bioimage informatics platform for extended reproducible research. *Nature methods*, 9(7):690, 2012.
- [125] Springer Nature. Availability of data, materials, code and protocols : authors & referees @ npg. <https://www.nature.com/authors/policies/availability.html>, 2018. Accessed: 2018-10-17.
- [126] American Association for the Advancement of Science. Science Journals: editorial policies | Science | AAAS. <http://www.sciencemag.org/authors/science-journals-editorial-policies>, 2018. Accessed: 2018-10-17.
- [127] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific data*, 3, 2016.
- [128] Wesley R Legant, Lin Shao, Jonathan B Grimm, Timothy A Brown, Daniel E Milkie, Brian B Avants, Luke D Lavis, and Eric Betzig. High-density three-dimensional localization microscopy across large volumes. *Nature methods*, 13(4):359, 2016.

- [129] Qinyi Fu, Benjamin L Martin, David Q Matus, and Liang Gao. Imaging multicellular specimens with real-time optimized tiling light-sheet selective plane illumination microscopy. *Nature communications*, 7:11088, 2016.
- [130] Raghav K Chhetri, Fernando Amat, Yinan Wan, Burkhard Höckendorf, William C Lemon, and Philipp J Keller. Whole-animal functional and developmental imaging with isotropic spatial resolution. *Nature methods*, 12(12):1171, 2015.
- [131] Eleanor Williams, Josh Moore, Simon W Li, Gabriella Rustici, Aleksandra Tarkowska, Anatole Chessel, Simone Leo, Bálint Antal, Richard K Ferguson, Ugis Sarkans, et al. Image data resource: a bioimage data integration and publication platform. *Nature methods*, 14(8):775, 2017.
- [132] J. Broeke, J.M.M. Perez, and J. Pascau. *Image Processing with ImageJ*. Packt Publishing, 2015.
- [133] Alvy Ray Smith. Volume graphics and volume visualization: A tutorial. *Technical Memo*, 176, 1987.
- [134] Pixar. The Pixar Story. Documentary film, 2007.
- [135] Disney-Pixar. Pixar Animation Studios: Our story. <https://www.pixar.com/our-story-1/#our-story-main>, 2018. Accessed: 2018-06-26.
- [136] Siggraph '88: Acm siggraph 88 panel proceedings. New York, NY, USA, 1988. ACM.
- [137] S. Buss and S.R. Buss. *3D Computer Graphics: A Mathematical Introduction with OpenGL*. 3-D Computer Graphics: A Mathematical Introduction with OpenGL. Cambridge University Press, 2003.
- [138] Heang K Tuy and Lee Tan Tuy. Direct 2-D display of 3-D objects. *IEEE Computer Graphics and Applications*, 4(10):29–34, 1984.
- [139] Marc Levoy. Display of surfaces from volume data. *IEEE Computer graphics and Applications*, (3):29–30, 1988.
- [140] Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. In *ACM Siggraph Computer Graphics*, volume 22, pages 65–74. ACM, 1988.
- [141] Frank Dachille, Kevin Keeger, Baoquan Chen, Ingmar Bitter, and Arie Kaufman. High-quality volume rendering using texture mapping hardware. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware*, pages 69–ff. ACM, 1998.
- [142] Harvey Ray and Deborah Silver. Resample and composite engine for real-time volume rendering. *US Patent Application*, US20020113787A1, 2000.
- [143] NVIDIA Corporation. GeForce GTX 1080 | Specifications | GeForce. <https://www.geforce.co.uk/hardware/desktop-gpus/geforce-gtx-1080/specifications>, 2016. Accessed: 2018-06-26.
- [144] Bitplane and Oxford Instruments. Bitplane: Imaris - Microscopy Image Analysis Software. <http://www.bitplane.com/>, 1992. Accessed: 2018-06-26.

- [145] Paul A Yushkevich, Joseph Piven, Heather Cody Hazlett, Rachel Gimpel Smith, Sean Ho, James C Gee, and Guido Gerig. User-guided 3D active contour segmentation of anatomical structures: significantly improved efficiency and reliability. *Neuroimage*, 31(3):1116–1128, 2006.
- [146] Utkarsh Ayachit. The paraview guide: a parallel visualization application. 2015.
- [147] CIBC ImageVis3D. An interactive visualization software system for large-scale volume data. <http://www.sci.utah.edu/cibc-software/imagevis3D.html>, 2015. Accessed: 2018-06-26.
- [148] Christopher R Madan. Creating 3D visualizations of mri data: A brief guide. *F1000Research*, 4, 2015.
- [149] Barry M Leiner, Vinton G Cerf, David D Clark, Robert E Kahn, Leonard Kleinrock, Daniel C Lynch, Jon Postel, Larry G Roberts, and Stephen Wolff. A brief history of the internet. *ACM SIGCOMM Computer Communication Review*, 39(5):22–31, 2009.
- [150] Khronos Group. Khronos details webgl initiative to bring hardware-accelerated 3D graphics to the internet. <https://www.khronos.org/news/press/khronos-webgl-initiative-hardware-accelerated-3D-graphics-internet>, 2009. Accessed: 2018-06-26.
- [151] @Fyrd and @Lensco. Can I Use... Support tables for HTML5, CSS3, etc. <https://caniuse.com/#feat=webgl>, 2013. Accessed: 2018-10-10.
- [152] Dean Jackson, Jeff Gilbert, and Khronos Group. Webgl 2.0 specification: Editor’s draft Fri Aug 31 2018. <https://www.khronos.org/registry/webgl/specs/latest/2.0/>, 2018. Accessed: 2018-09-30.
- [153] Unity Technologies. Unity - Multiplatform - Publish your game to over 25 platforms. <https://unity3D.com/unity/features/multiplatform>, 2016. Accessed: 2018-06-26.
- [154] NVIDIA Corporation. OpenGL SDK Guide. Whitepaper at http://developer.download.nvidia.com/SDK/10/opengl/OpenGL_SDK_Guide.pdf, 2007. Accessed: 2018-06-26.
- [155] Abhishek Kumar, Yicong Wu, Ryan Christensen, Panagiotis Chandris, William Gandler, Evan McCreedy, Alexandra Bokinsky, Daniel A Colón-Ramos, Zhirong Bao, Matthew McAuliffe, G Rondeau, and Hari Schroff. Dual-view plane illumination microscopy for rapid and spatially isotropic imaging. *nature protocols*, 9(11):2555, 2014.
- [156] Marcus Fantham and Clemens F Kaminski. A new online tool for visualization of volumetric data. *Nature Photonics*, 11(2):69, 2017.
- [157] M Levoy. Volume rendering—visible volume rendering. *Computer Graphics and Applications*, 8:29–37, 1988.
- [158] Jürgen Mayer, Alexandre Robert-Moreno, Renzo Danuser, Jens V Stein, James Sharpe, and Jim Swoger. OPTiSPIM: integrating optical projection tomography in light sheet microscopy extends specimen characterization to nonfluorescent contrasts. *Optics letters*, 39(4):1053–1056, 2014.

- [159] Ann Torrence. Martin newell’s original teapot. In *ACM SIGGRAPH 2006*, page 29. ACM, 2006.
- [160] Harvey Ray and Deborah Silver. The RACE II Engine for Real-Time Volume Rendering. In I. Buck, G. Humphreys, and P. Hanrahan, editors, *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware*. The Eurographics Association, 2000.
- [161] Erik Lindholm, John Nickolls, Stuart Oberman, and John Montrym. NVIDIA Tesla: A unified graphics and computing architecture. *IEEE micro*, 28(2), 2008.
- [162] Statista. Most popular internet browser versions 2018 | Statistic. <https://www.statista.com/statistics/268299/most-popular-internet-browsers/>, 2018. Accessed: 2018-09-30.
- [163] University of Dundee and Open Microscopy Environment. OMERO Features | Open Microscopy Environment (OME). <https://www.openmicroscopy.org/omero/features/view/>, 2018. Accessed: 2018-09-30.
- [164] Melissa Linkert, Curtis T Rueden, Chris Allan, Jean-Marie Burel, Will Moore, Andrew Patterson, Brian Loranger, Josh Moore, Carlos Neves, Donald MacDonald, et al. Metadata matters: access to image data in the real world. *The Journal of Cell Biology*, 189(5):777–782, 2010.
- [165] Marcus Fantham. FPBioimage Host. <https://fpbhost.firebaseio.com/>, 2018. Accessed: 2019-04-05.
- [166] S Jayaraman, S Esakkirajan, and T Veerakumar. *Digital Image Processing*. Tata McGraw Hill Education, 2011.
- [167] Josh Moore, Melissa Linkert, Colin Blackburn, Mark Carroll, Richard K Ferguson, Helen Flynn, Kenneth Gillen, Roger Leigh, Simon Li, Dominik Lindner, et al. OMERO and Bio-Formats 5: flexible access to large bioimaging datasets at scale. In *Medical Imaging 2015: Image Processing*, volume 9413, page 941307. International Society for Optics and Photonics, 2015.
- [168] Niko Ehrenfeuchter. 3D Rendering on WebViewer NG Priorities List | Trello. <https://trello.com/c/QKtSahbg/10-3D-rendering#comment-5914230dbde8ecde73c94104>, 2017. Accessed: 2018-10-21.
- [169] fleur - FPBioimage Viewer. <https://s3.amazonaws.com/fpbhost/fleur/index.html>, 2018. Accessed: 2018-10-21.
- [170] Google. Google Cardboard - Google VR. <https://vr.google.com/cardboard/>, 2018. Accessed: 2018-10-10.
- [171] Google. Get Cardboard - Google VR. <https://vr.google.com/cardboard/get-cardboard/>, 2018. Accessed: 2018-10-10.
- [172] Fantham, Marcus. Marcus Fantham on Twitter: “Due to popular demand at #FOM2018, I’ll soon release FPBioimage as a volumetric viewer for @FijiSc /ImageJ....”. https://twitter.com/fantm_science/status/978846458572140545, 2018. Accessed: 2018-10-10.

- [173] Poll Everywhere. LiveSlides - Embed any website in a PowerPoint or Keynote presentation. <https://www.liveslides.com/>, 2016. Accessed: 2018-10-10.
- [174] AS Ahmad, N Ormiston-Smith, and PD Sasieni. Trends in the lifetime risk of developing cancer in great britain: comparison of risk for those born from 1930 to 1960. *British journal of cancer*, 112(5):943, 2015.
- [175] R. Hesketh. *Betrayed by Nature: The War on Cancer*. MacSci. St. Martin's Press, 2012.
- [176] J Craig Venter, Mark D Adams, Eugene W Myers, Peter W Li, Richard J Mural, Granger G Sutton, Hamilton O Smith, Mark Yandell, Cheryl A Evans, Robert A Holt, et al. The sequence of the human genome. *Science*, 291(5507):1304–1351, 2001.
- [177] David R Bentley, Shankar Balasubramanian, Harold P Swerdlow, Geoffrey P Smith, John Milton, Clive G Brown, Kevin P Hall, Dirk J Evers, Colin L Barnes, Helen R Bignell, et al. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature*, 456(7218):53, 2008.
- [178] Andrew Fire, SiQun Xu, Mary K Montgomery, Steven A Kostas, Samuel E Driver, and Craig C Mello. Potent and specific genetic interference by double-stranded rna in *caenorhabditis elegans*. *Nature*, 391(6669):806, 1998.
- [179] Lisa Timmons and Andrew Fire. Specific interference by ingested dsrna. *Nature*, 395(6705):854, 1998.
- [180] Gregory J Hannon. Rna interference. *Nature*, 418(6894):244, 2002.
- [181] Andrew J Hamilton and David C Baulcombe. A species of small antisense rna in posttranscriptional gene silencing in plants. *Science*, 286(5441):950–952, 1999.
- [182] Sayda M Elbashir, Jens Harborth, Winfried Lendeckel, Abdullah Yalcin, Klaus Weber, and Thomas Tuschl. Duplexes of 21-nucleotide rnas mediate rna interference in cultured mammalian cells. *Nature*, 411(6836):494, 2001.
- [183] M. Knowles and P. Selby. *Introduction to the Cellular and Molecular Biology of Cancer*. OUP Oxford, 2005.
- [184] Yair Dorsett and Thomas Tuschl. siRNAs: applications in functional genomics and potential as therapeutics. *Nature Reviews Drug Discovery*, 3(4):318, 2004.
- [185] H Chial. Tumor suppressor (ts) genes and the two-hit hypothesis. *Nature Education*, 1(1):177, 2008.
- [186] H.F. Lodish. *Molecular Cell Biology*. W H Freeman & Company, 1999.
- [187] Chiranjib Chakraborty, Ashish Ranjan Sharma, Garima Sharma, C George Priya Doss, and Sang-Soo Lee. Therapeutic miRNA and siRNA: moving from bench to clinic as next generation medicine. *Molecular Therapy-Nucleic Acids*, 8:132–143, 2017.
- [188] European Patent Office. Espacenet - results view. <https://worldwide.espacenet.com/searchResults?query=sirna+and+cancer>, 2018. Accessed: 2018-10-15.

- [189] Gaurav Tiwari, Ruchi Tiwari, Birendra Sriwastawa, L Bhati, S Pandey, P Pandey, and Saurabh K Bannerjee. Drug delivery systems: An updated review. *International journal of pharmaceutical investigation*, 2(1):2, 2012.
- [190] ThermoFisher Scientific. . <https://www.thermofisher.com/uk/en/home/brands/product-brand/lipofectamine.html>, 2018. Accessed: 2018-10-15.
- [191] Che Fu Liu, Raymond Chen, Joseph A Frezzo, Priya Katyal, Lindsay K Hill, Liming Yin, Nikita Srivastava, Haresh T More, P Douglas Renfrew, Richard Bonneau, et al. Efficient dual siRNA and drug delivery using engineered lipoproteoplexes. *Biomacromolecules*, 18(9):2688–2698, 2017.
- [192] Cong-fei Xu and Jun Wang. Delivery systems for siRNA drug development in cancer therapy. *asian journal of pharmaceutical sciences*, 10(1):1–12, 2015.
- [193] Sanjeeb K Sahoo and Vinod Labhsetwar. Nanotech approaches to drug delivery and imaging. *Drug discovery today*, 8(24):1112–1120, 2003.
- [194] Xu Wang, Yiqing Wang, Zhuo Georgia Chen, and Dong M Shin. Advances of cancer therapy by nanotechnology. *Cancer research and treatment: official journal of Korean Cancer Association*, 41(1):1, 2009.
- [195] Jürgen Soutschek, Akin Akinc, Birgit Bramlage, Klaus Charisse, Rainer Constien, Mary Donoghue, Sayda Elbashir, Anke Geick, Philipp Hadwiger, Jens Harborth, et al. Therapeutic silencing of an endogenous gene by systemic administration of modified siRNAs. *Nature*, 432(7014):173, 2004.
- [196] Ya-Lin Chiu, Akbar Ali, Chia-ying Chu, Hong Cao, and Tariq M Rana. Visualizing a correlation between siRNA localization, cellular uptake, and RNAi in living cells. *Chemistry & biology*, 11(8):1165–1175, 2004.
- [197] Liqing Ma, Carter Abney, and Wenbin Lin. Enantioselective catalysis with homochiral metal–organic frameworks. *Chemical Society Reviews*, 38(5):1248–1256, 2009.
- [198] D. Farrusseng. *Metal-Organic Frameworks: Applications from Catalysis to Gas Storage*. EngineeringPro collection. Wiley, 2011.
- [199] Radu Custelcean and Bruce A Moyer. Anion separation with metal–organic frameworks. *European Journal of Inorganic Chemistry*, 2007(10):1321–1340, 2007.
- [200] Honghan Fei, David L Rogow, and Scott RJ Oliver. Reversible anion exchange and catalytic properties of two cationic metal–organic frameworks based on Cu (I) and Ag (I). *Journal of the American Chemical Society*, 132(20):7202–7209, 2010.
- [201] Lauren E Kreno, Kirsty Leong, Omar K Farha, Mark Allendorf, Richard P Van Duyne, and Joseph T Hupp. Metal–organic framework materials as chemical sensors. *Chemical reviews*, 112(2):1105–1125, 2011.
- [202] Sophie E Miller, Michelle H Teplensky, Peyman Z Moghadam, and David Fairen-Jimenez. Metal-organic frameworks as biosensors for luminescence-based detection and imaging. *Interface Focus*, 6(4):20160027, 2016.

- [203] Peyman Z Moghadam, Timur Islamoglu, Subhadip Goswami, Jason Exley, Marcus Fantham, Clemens F Kaminski, Randall Q Snurr, Omar K Farha, and David Fairen-Jimenez. Computer-aided discovery of a metal–organic framework with superior oxygen uptake. *Nature communications*, 9(1):1378, 2018.
- [204] ED Michelakis, L Webster, and JR Mackey. Dichloroacetate (DCA) as a potential metabolic-targeting therapy for cancer. *British journal of cancer*, 99(7):989, 2008.
- [205] Tomohiro Matsuhashi, Takako Hishiki, Heping Zhou, Tomohiko Ono, Ruri Kaneda, Tatsuya Iso, Aiko Yamaguchi, Jin Endo, Yoshinori Katsumata, Anzai Atsushi, et al. Activation of pyruvate dehydrogenase by dichloroacetate has the potential to induce epigenetic remodeling in the heart. *Journal of molecular and cellular cardiology*, 82:116–124, 2015.
- [206] Sébastien Bonnet, Stephen L Archer, Joan Allalunis-Turner, Alois Haromy, Christian Beaulieu, Richard Thompson, Christopher T Lee, Gary D Lopaschuk, Lakshmi Puttagunta, Sandra Bonnet, et al. A mitochondria-k⁺ channel axis is suppressed in cancer and its normalization promotes apoptosis and inhibits cancer growth. *Cancer cell*, 11(1):37–51, 2007.
- [207] ED Michelakis, G Sutendra, P Dromparis, L Webster, A Haromy, E Niven, C Maguire, T-L Gammer, JR Mackey, D Fulton, et al. Metabolic modulation of glioblastoma with dichloroacetate. *Science translational medicine*, 2(31):31ra34–31ra34, 2010.
- [208] Isabel Abánades Lázaro, Salame Haddad, José M Rodrigo-Muñoz, Claudia Orellana-Tavra, Victoria del Pozo, David Fairen-Jimenez, and Ross S Forgan. Mechanistic investigation into the selective anticancer cytotoxicity and immune system response of surface-functionalized, dichloroacetate-loaded, UiO-66 nanoparticles. *ACS applied materials & interfaces*, 10(6):5255–5268, 2018.
- [209] Xiao Huang and Christopher S Brazel. On the importance and mechanisms of burst release in matrix-controlled drug delivery systems. *Journal of controlled release*, 73(2-3):121–136, 2001.
- [210] Yao Fu and Weiyuan John Kao. Drug release kinetics and transport mechanisms of non-degradable and degradable polymeric delivery systems. *Expert opinion on drug delivery*, 7(4):429–444, 2010.
- [211] John M Chirgwin, Alan E Przybyla, Raymond J MacDonald, and William J Rutter. Isolation of biologically active ribonucleic acid from sources enriched in ribonuclease. *Biochemistry*, 18(24):5294–5299, 1979.
- [212] Saim Özkar and Richard G Finke. Nanocluster formation and stabilization fundamental studies. 2. proton sponge as an effective H⁺ scavenger and expansion of the anion stabilization ability series. *Langmuir*, 18(20):7653–7662, 2002.
- [213] Lisa D Cervia, Chun-Chi Chang, Liangli Wang, and Fan Yuan. Distinct effects of endosomal escape and inhibition of endosomal trafficking on gene delivery via electrotransfection. *PloS one*, 12(2):e0171699, 2017.

- [214] Alan Coates, Suzanne Abraham, S Betai Kaye, Timothy Sowerbutts, Cheryl Frewin, RM Fox, and MHN Tattersall. On the receiving end—patient perception of the side-effects of cancer chemotherapy. *European Journal of Cancer and Clinical Oncology*, 19(2):203–208, 1983.
- [215] M de Boer-Dennert, R De Wit, PIM Schmitz, J Djontono, V v Beurden, G Stoter, and J Verweij. Patient perceptions of the side-effects of chemotherapy: the influence of 5HT3 antagonists. *British journal of cancer*, 76(8):1055, 1997.
- [216] Manabu Minami, Shigemi Matsumoto, and Hisanori Horiuchi. Cardiovascular side-effects of modern cancer therapy. *Circulation Journal*, 74(9):1779–1786, 2010.
- [217] Anne E Carpenter, Thouis R Jones, Michael R Lamprecht, Colin Clarke, In Han Kang, Ola Friman, David A Guertin, Joo Han Chang, Robert A Lindquist, Jason Moffat, et al. Cellprofiler: image analysis software for identifying and quantifying cell phenotypes. *Genome biology*, 7(10):R100, 2006.
- [218] GraphPad Software. Graphpad Prism version 7.04 for Windows. <https://www.graphpad.com/>, 1995. Accessed: 2018-10-10.
- [219] Tim Mosmann. Rapid colorimetric assay for cellular growth and survival: application to proliferation and cytotoxicity assays. *Journal of immunological methods*, 65(1-2):55–63, 1983.
- [220] Abcam plc. MTS Assay Kit (Cell Proliferation) (Colorimetric). <https://www.abcam.com/mts-assay-kit-cell-proliferation-colorimetric-ab197010.html>, 2018. Accessed: 2018-10-17.
- [221] Scott A Oakes and Feroz R Papa. The role of endoplasmic reticulum stress in human pathology. *Annual Review of Pathology: Mechanisms of Disease*, 10:173–194, 2015.
- [222] Rammohan V Rao and Dale E Bredesen. Misfolded proteins, endoplasmic reticulum stress and neurodegeneration. *Current opinion in cell biology*, 16(6):653–662, 2004.
- [223] Melissa J Phillips and Gia K Voeltz. Structure and function of ER membrane contact sites with other organelles. *Nature reviews Molecular cell biology*, 17(2):69, 2016.
- [224] Alex M Valm, Sarah Cohen, Wesley R Legant, Justin Melunis, Uri Hershberg, Eric Wait, Andrew R Cohen, Michael W Davidson, Eric Betzig, and Jennifer Lippincott-Schwartz. Applying systems-level spectral imaging and analysis to reveal the organelle interactome. *Nature*, 546(7656):162, 2017.
- [225] Lale Ozcan and Ira Tabas. Role of endoplasmic reticulum stress in metabolic disease and other disorders. *Annual review of medicine*, 63:317–328, 2012.
- [226] Christian A Hübner and Ingo Kurth. Membrane-shaping disorders: a common pathway in axon degeneration. *Brain*, 137(12):3109–3121, 2014.
- [227] Craig Blackstone, Cahir J O’kane, and Evan Reid. Hereditary spastic paraplegias: membrane traffic and the motor pathway. *Nature Reviews Neuroscience*, 12(1):31, 2011.

- [228] Mark J Dayel, Erik FY Hom, and AS Verkman. Diffusion of green fluorescent protein in the aqueous-phase lumen of endoplasmic reticulum. *Biophysical journal*, 76(5):2843–2851, 1999.
- [229] Sarah Nehls, Erik L Snapp, Nelson B Cole, Kristien JM Zaal, Anne K Kenworthy, Theresa H Roberts, Jan Ellenberg, John F Presley, Eric Siggia, and Jennifer Lippincott-Schwartz. Dynamics and retention of misfolded proteins in native ER membranes. *Nature cell biology*, 2(5):288, 2000.
- [230] David Holcman, Pierre Parutto, Joseph E Chambers, Marcus Fantham, Laurence J Young, Stefan J Marciniak, Clemens F Kaminski, David Ron, and Edward Avezov. Single particle trajectories reveal active endoplasmic reticulum luminal flow. *Nature cell biology*, 20:1118–1125, 2018.
- [231] Ignacio Arganda-Carreras, Verena Kaynig, Curtis Rueden, Kevin W Eliceiri, Johannes Schindelin, Albert Cardona, and H Sebastian Seung. Trainable Weka Segmentation: a machine learning tool for microscopy pixel classification. *Bioinformatics*, 33(15):2424–2426, 2017.
- [232] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.
- [233] Robert E Schapire. A brief introduction to boosting. In *Ijcai*, volume 99, pages 1401–1406, 1999.
- [234] S Nadeem and EN Maraj. The mathematical analysis for peristaltic flow of nano fluid in a curved channel with compliant walls. *Applied Nanoscience*, 4(1):85–92, 2014.
- [235] Jennifer Lippincott-Schwartz, Lydia C Yuan, Juan S Bonifacino, and Richard D Klausner. Rapid redistribution of Golgi proteins into the ER in cells treated with brefeldin A: evidence for membrane cycling from Golgi to ER. *Cell*, 56(5):801–813, 1989.
- [236] Kala Subramanian and Tobias Meyer. Calcium-induced restructuring of nuclear envelope and endoplasmic reticulum calcium stores. *Cell*, 89(6):963–971, 1997.
- [237] Abeer George Ghuneim. Contour Tracing. http://www.imageprocessingplace.com/downloads_V3/root_downloads/tutorials/contour_tracing_Abeer_George_Ghuneim/moore.html, 2000. Accessed: 2018-04-13.
- [238] TY Zhang and Ching Y. Suen. A fast parallel algorithm for thinning digital patterns. *Communications of the ACM*, 27(3):236–239, 1984.
- [239] Urs Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer graphics and image processing*, 1(3):244–256, 1972.
- [240] David H Douglas and Thomas K Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2):112–122, 1973.

- [241] Albert Einstein. Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen. *Annalen der physik*, 322(8):549–560, 1905.
- [242] Titus Lucretius Carus. *Titus Lvcretivs Carvs: De rerum natura*, 66 BC. Apud Ioann: Ianssonium, Amstelodami, 1631.
- [243] P.M. Fishbane, S. Gasiorowicz, and S.T. Thornton. *Physics for Scientists and Engineers: Extended*. Pearson College Division, 1998.
- [244] NVIDIA Corporation. PhysX | Technology | GeForce. <https://www.geforce.com/hardware/technology/physx/technology>, 2008. Accessed: 2018-10-18.
- [245] Diana Pendin, James A McNew, and Andrea Daga. Balancing ER dynamics: shaping, bending, severing, and mending membranes. *Current opinion in cell biology*, 23(4):435–442, 2011.
- [246] Tomoyuki Yamanaka and Nobuyuki Nukina. ER dynamics and derangement in neurological diseases. *Frontiers in neuroscience*, 12:91, 2018.
- [247] Yuting Guo, Di Li, Siwei Zhang, Yanrui Yang, Jia-Jia Liu, Xinyu Wang, Chong Liu, Daniel E Milkie, Regan P Moore, U Serdar Tulu, et al. Visualizing intracellular organelle and cytoskeletal interactions at nanoscale resolution on millisecond timescales. *Cell*, 175(5):1430–1442, 2018.
- [248] Janin Lautenschläger, Amberley D Stephens, Giuliana Fusco, Florian Ströhl, Nathan Curry, Maria Zacharopoulou, Claire H Michel, Romain Laine, Nadezhda Nesporvitaya, Marcus Fantham, et al. C-terminal calcium binding of α -synuclein modulates synaptic vesicle interaction. *Nature communications*, 9(1):712, 2018.

Appendix A

LAG SIM Software Implementations

A.1 LabVIEW control software

The LAG SIM control software is written in LabVIEW, a graphical programming language which links variables through block diagrams rather than textual functions. This makes running tasks simultaneously easier in LabVIEW than other languages, making it ideally suited for instrument control where some events must happen in parallel, and others in a strict order. A further advantage of LabVIEW is that the user interface is built simultaneously with the program control logic, making it suitable for building applications designed to be run by non-expert end users.

The user facing controls of the LAG SIM software are detailed in Section 2.3. This section of the appendix includes some of the block diagram logic that powers the microscope control software, equivalent to presenting source code for other programming languages.

LabVIEW supports abstraction with inputs and outputs, like functions in other languages, so repeatable pieces of logic can be wrapped up in their own subVI blocks. The LAG SIM software is made up of 55 such files, and presenting them all here would be impractical; however, the entire project is available to download from <https://github.com/laseranalyticsgroup/sim-labview-control>. Here I will present the implementation of LAG SIM's experiment logging feature, since it is a reusable component that could be useful in many other LabVIEW control programs.

Labstep is an online laboratory management system, designed to make research work compliant, collaborative, and audited. Experiments can be logged in an electronic laboratory notebook and, because they are stored online, can be accessed from anywhere with an internet connection. An example of the Labstep timeline, along with details of an experiment, is shown in Figure A.1. Of particular interest to me was access to the system through an API, allowing requests to be made from any programming language that supports HTTP.

The screenshot shows the Labstep interface with two main panels. The left panel displays a list of experiments under the heading 'My Experiments'. The right panel shows a detailed view of 'Meng's SIM experiment from 02/05/2019 at 12:41'.

Left Panel: My Experiments

- Showing 111-120 of 3955
- Filter: Status: Finished
- Experiments (111)
- Protocols (0)
- Resources (0)
- Orders (0)
- Files (0)

Name	Status	Started on	Finished on	Results
Ioanna's SIM experiment from 08/05/2019 at 11:11	Finished	08/05/2019 at 11:11	08/05/2019 at 18:20	View results/notes (43)
Ioanna's SIM experiment from 08/05/2019 at 11:04	Finished	08/05/2019 at 11:04	08/05/2019 at 12:10	View results/notes (1)
Meng's SIM experiment from 03/05/2019 at 14:00	Finished	03/05/2019 at 14:00	03/05/2019 at 16:41	View results/notes (102)
Ioanna's SIM experiment from 03/05/2019 at 11:11	Finished	03/05/2019 at 11:11	03/06/2019 at 07:51	View results/notes (85)
Meng's SIM experiment from 02/05/2019 at 12:41	Finished	02/05/2019 at 12:41	02/05/2019 at 16:27	View results/notes (2)
Ioanna's SIM experiment from 02/05/2019 at 11:11	Finished	02/05/2019 at 10:11	02/05/2019 at 13:41	View results/notes (43)
Ioanna's SIM experiment from 02/05/2019 at 09:57	Finished	02/05/2019 at 09:57	02/05/2019 at 11:08	View results/notes (3)
Lisp's SIM experiment from 27/04/2019 at 16:00	Finished	27/04/2019 at 16:00	27/04/2019 at 17:40	View results/notes (6)
Lisp's SIM experiment from 27/04/2019 at 15:56	Finished	27/04/2019 at 15:56	27/04/2019 at 17:00	View results/notes (1)
Lisa's SIM experiment from 27/04/2019 at 16:49	Finished	27/04/2019 at 16:49	13/06/2019 at 07:52	View results/notes (2)

Right Panel: Meng's SIM experiment from 02/05/2019 at 12:41

Created on 02/05/2019 at 12:41 by [LAG SIM](#)

Detail Workflow (0) Results/Notes (2)

[+ Add your results/notes](#)

LAG SIM 02/05/2019 at 15:18 Meng just made a 18-frame acquisition with the following parameters: {channels: 2, t-frames: 1, z-planes: 1, bookmark: 0, channel_info: {channel: [wavelength: 483, exposure: 80, tift: false]}, c: channel1: [wavelength: 581, exposure: 60, tift: false]}, optospilt: false, widefield: false} [Edit](#) [Delete](#)

LAG SIM 02/05/2019 at 15:17 Meng just made a 18-frame acquisition with the following parameters: {channels: 2, t-frames: 1, z-planes: 1, bookmark: 1, channel_info: {channel: [wavelength: 483, exposure: 80, tift: false]}, c: channel2: [wavelength: 581, exposure: 60, tift: false]}, optospilt: false, widefield: false} [Edit](#) [Delete](#)

LAG SIM 02/05/2019 at 15:17 Meng just made a 18-frame acquisition with the following parameters: {channels: 2, t-frames: 1, z-planes: 1, bookmark: 0, channel_info: {channel: [wavelength: 483, exposure: 80, tift: false]}, c: channel1: [wavelength: 581, exposure: 60, tift: false]}, optospilt: false, widefield: false} [Edit](#) [Delete](#)

Show 111-120 of 3955

Figure A.1: User activity is automatically logged to an online logbook hosted on Labstep to identify and fix any problems quickly. The left-hand screenshot shows the users which have used the LAG SIM between 27-04-2019 and 08-05-2019; opening one of these experiments (right) shows each of the acquisitions with the instrument parameters recorded in JSON format.

LabVIEW has built-in functions for making HTTP requests to a remote server. To interact with the Labstep API, the functions GET, POST, and PUT are required, each of which has their own block in LabVIEW. A full API list of API endpoints and the HTTP requests that they respond to can be found at Labstep's website, <https://github.com/Labstep/labstep-api-examples>.

For LAG-SIM, the required functionality was: record which user has started an experiment; post a comment to the experiment every time they capture an image which records the microscope settings for that acquisition; log any software errors that occurred, which could happen if a piece of hardware is switched off for example; and finally, end the experiment when the user closes the software.

The block-diagram logic for starting an experiment is shown in Figure A.2a. First, an HTTP request handler is set up, and the API key is added as a header. In parallel to this, a JSON string is created which will become the body of the HTTP request, containing a title ('name') for the experiment, and a description of the experiment which contains the time and date formatted as a string. The body is POSTed to the appropriate endpoint, <https://api.labstep.com/api/generic/experiment-workflow>. Finally, the response of the request is parsed to extract the Thread ID and Experiment ID, which will be required to add comments to the experiment and to finish the experiment respectively. If the subVI cannot extract these IDs, it generates an error; this could occur if, for example, the API key is incorrect or there is no internet connection.

The subVI for starting an experiment is shown as part of a larger block diagram in Figure A.3, where the logic has been abstracted into a block with 2 inputs and 2 outputs. The API key is stored in the program, and the user's name is collected from the user when they open the software; the output wires Experiment ID and Thread ID continue into the program to be used again later.

Figure A.2b shows the subVI for posting a comment to an experiment. The inputs are a Thread ID, from the initialisation subVI, and a string, which can be any list of words or characters. Every time an acquisition is made, this subVI is called with an input string which describes the microscope settings for that particular acquisition. This is set up as a JSON string, as shown in Figure A.1, which leaves potential for further processing on the metadata at a later time.

When the user has finished their session and closes the software, the snippet shown in Figure A.3b is executed. If an error occurred during the software execution, this is logged just before the block to end the experiment is executed using the comment-positing subVI detailed in Figure A.2b. Finally, the 'END' subVI is called to close the experiment on Labstep; the contents of this block are shown in Figure A.2c.

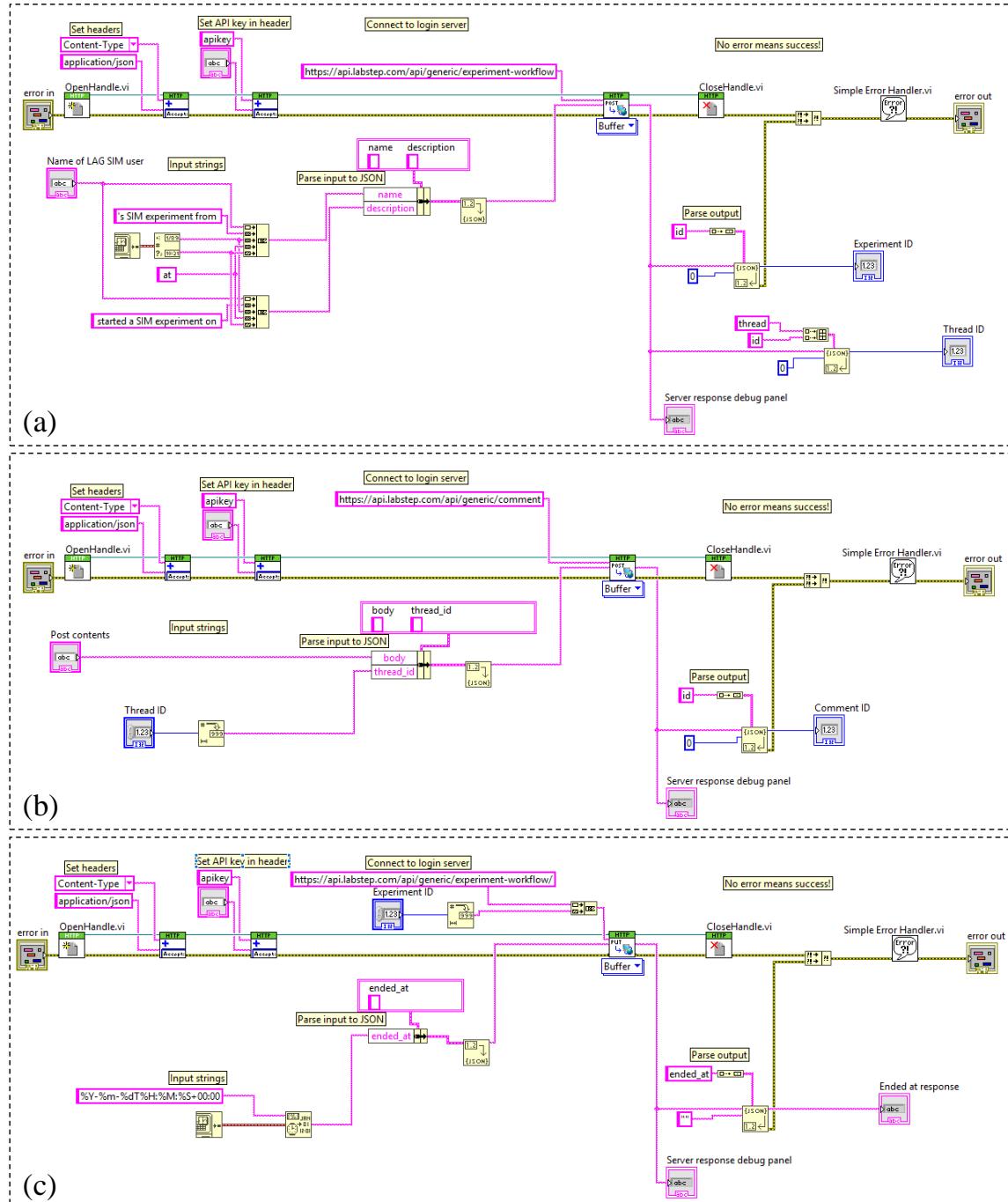


Figure A.2: (a) shows the block diagram logic for creating a new experiment on Labstep with an HTTP request. (b) shows the subVI for adding a comment to the experiment, which can be used to log information about each acquisition. (c) shows the logic for closing the Labstep experiment, which is executed when the LAG SIM control software is closed.

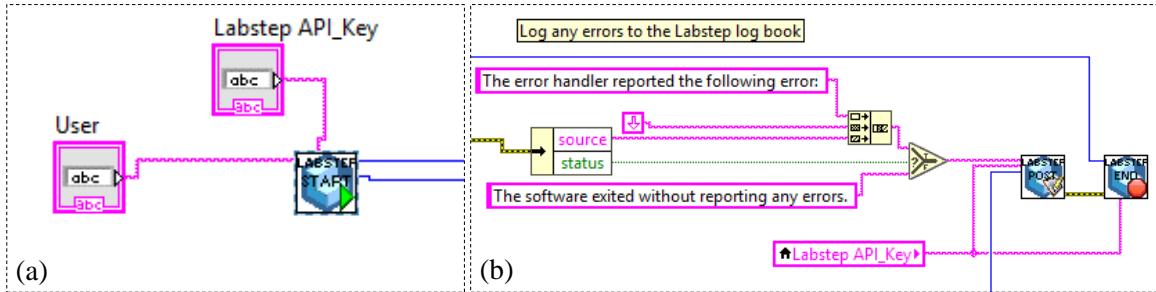


Figure A.3: (a) shows the block diagram from Figure A.2a abstracted to a single block with 2 inputs and 2 outputs. (b) shows the logic that runs when the user exits the LAG SIM control software: any errors are logged to the Labstep timeline for the experiment, using the logic of Figure A.2b, and finally the experiment is ended with the logic of Figure A.2c.

The logging software described in this sectioning is available on the LabVIEW community file exchange, at <https://forums.ni.com/t5/Example-Program-Drafts/Logging-with-Labstep/t-p/3822118>. The blocks can be reused in any other LabVIEW program, providing the computer is connected to the internet. The full LAG SIM control program, which includes the the Labstep integration, is available on Github at <https://github.com/laseranalyticsgroup/sim-labview-control>.

A.2 SROS-SIM reconstruction on GPU

When I arrived in the Laser Analytics Group, SIM data was reconstructed using either Lin Shao's algorithm written in CUDA, as described in Section 2.4.1, or, when super-resolution with optical sectioning was required, a MATLAB script called SROS-SIM written by Florian Ströhl was used.

The main disadvantage of SROS-SIM was its speed. A single 1024×1024 frame of SIM data took 22.7 s to reconstruct on the CPU, assuming that illumination parameters such as spatial frequency and phase shift had been calculated beforehand. Furthermore, calculating the illumination parameters took 105.2 s although these could be computed once per data set and reused for the set.

These speeds were impractically slow for high-throughput experiments, where hundreds or, for timelapse data, thousands of SIM frames require reconstruction. Furthermore, during an experiment it is often useful to view reconstructed data to ensure imaging conditions are optimal; waiting over two minutes to see a single reconstructed frame is not practical when live cells are surviving in unfavourable conditions on the microscope. There was a clear need to increase the speed of the program.

One reason for the particularly slow parameter estimation was an autocorrelation operation, implemented as a convolution. A direct convolution operation has an algorithmic complexity of $O(n^2)$, where n is the number of elements in the array to be convolved. Since the images to be convolved are large 2D arrays, convolution is a particularly expensive operation which becomes exponentially slower the larger the image size.

A mathematically equivalent operation is shown by Equation A.1, where A and B are example arrays and \mathcal{F}^{-1} is the inverse Fourier transform. This requires Fourier transforming the arrays and a point-wise multiplication of the resulting Fourier-space arrays. However, several Fast Fourier Transform (FFT) algorithms have been invented which recursively reduce the Fourier transform into smaller arrays, ultimately Fourier transforming with a complexity of $O(n \log n)$, which is faster than $O(n^2)$. Implementing the auto-correlation as a Fourier transform gave a $2.25\times$ improvement in parameter calculation time, shown in Table A.1.

$$A \otimes B \equiv \mathcal{F}^{-1} \{ \hat{A} \hat{B} \} \quad (\text{A.1})$$

The recursive reduction property of FFT algorithms lend themselves well to calculation on a GPU, as each reduced array can be operated on independently of the other arrays. GPU architectures, which are designed to run many simple calculations in parallel, are well optimised for this type of problem. By moving the arrays onto the graphics memory and computing the FFT on the GPU a further $1.8\times$ speed improvement was obtained compared to computing the FFT on the CPU.

More remarkable was the reduction in time required to compute a frame of SIM reconstruction, which also requires many Fourier transforms and simple, pixel-wise calculations which can be computed in parallel. By computing the reconstruction on the GPU, a speed increase of $17\times$ was achieved, as shown in Table A.1.

A final performance optimisation was realised by reducing the precision to which pixel values were calculated. In the original SROS-SIM program, values were calculated to double-precision floating-point format, requiring 64 bits per pixel. While it is good practice to perform calculations to more precision than required to avoid computational rounding errors, the sCMOS camera only records images to 10-bit precision, so 64 bits is unnecessarily excessive. Furthermore, graphics cards are usually optimised for single-precision calculations - that is, 32 bits per pixel. When arrays were stored and calculated in the single-precision floating point format, Table A.1 shows that speed of calculation was almost doubled compared to computation at double precision.

Overall, an $8\times$ improvement in speed was achieved for parameter estimation, and a $32\times$ improvement for per-frame reconstruction time, reducing the time to reconstruct a SIM frame

to 0.7 s. These improvements made it possible to reconstruct data during acquisition, to see results almost immediately without endangering the longevity of live-cell experiments.

Table A.1: The table shows the performance increase by switching from CPU to GPU computation, as well as other optimisations. This test was performed on a consumer desktop computer, with the following specification: 2-core i3-2120 @3.30 GHz, 8 GB RAM, NVIDIA GeForce GTX 750 Ti with 2 GB graphics RAM.

Condition	Parameter calculation (s)	SIM reconstruction (s)
CPU	105.2	22.7
CPU with FFT-convolution	46.7	22.7
GPU	25.5	1.3
GPU at single precision	13.0	0.7

MALTAB arrays are moved from CPU memory to GPU memory with the `gpuArray()` function. After an array has become a `gpuArray`, any computations involving the array will be performed by the GPU. For computations which can be highly parallelised, for example fast Fourier transforms, this can lead to significant reductions in processing time. Once the GPU computations are complete, the array can be moved back to CPU memory with the `gather()` command, for performing CPU operations such as saving to disk.

To ensure backwards compatibility on machines without GPUs, the revised code checks if the machine has a GPU and sets a boolean flag `gpuPresent`, as shown in Snippet A.1. If this flag is set, the array is sent to GPU with `gpuArray()`; else, it will be left MATLAB matrix which will be computed on the CPU.

Snippet A.1: A check must be performed at the start of the main program to detect if a suitable graphics card is installed for GPU computation

```

1 gpuPresent = false;
2 if (gpuDeviceCount > 0)
3     try
4         g = gpuDevice();
5         fprintf(['GPU device detected... \n' ...
6                 '           Name: ' g.Name '\n', ...
7                 '   ComputeCapability: ' g.ComputeCapability ' \n']);
8         % Tested for 3.0 and better
9         if(g.ComputeCapability > 3.0)
10             disp('Device supported, executing on GPU... ');
11             gpuPresent = true;
12         else
13             disp(['GPU functionality requires ComputeCapability' ...
14                   ' of 3.0 or higher. Executing on CPU... ']);
15         end
16     catch
17         disp('GPU device not compatible. Executing on CPU..');
18     end
19 end

```

To save writing a conditional check around every single array, a simple wrapper function shown in Snippet A.2 called `gpuThis()` was created, which takes as input arguments the array to potentially send to the GPU, and the `gpuPresent` variable calculated in Snippet A.1.

Snippet A.2: A simple helper function, `gpuThis()`, sends an array to graphics memory if a GPU is available, but leaves the array in CPU memory if not

```

1 function output = gpuThis(input, GPU_present);
2
3 if GPU_present
4     output = gpuArray(single(input));
5 else
6     output = single(input);
7 end
8
9 end

```

To make the program GPU-optimised, we now simply put `gpuThis()` around each array that would benefit from GPU computation during its allocation. Snippet A.3 shows a small extract from the original SROS-SIM code in lines 1-6, responsible for loading the OTF for each channel into a 3-dimensional array. The changes highlighted in lines 8-13 show that only 2 lines need modification for GPU optimisation.

Snippet A.3: The function `gpuThis()` should be wrapped around new arrays when they are preallocated

```

1 otf = zeros(HEIGHT,WIDTH,SETUP.colours);
2 for c = 1:SETUP.colours
3     otf(:,:,c) = double(imread([DATA.OTF_path{c}, DATA.OTF_name{c}, DATA.OTF_suffix{c}]));
4     % Normalise
5     otf(:,:,c) = otf/max(max(otf(:,:,c)));
6 end
7
8 otf = gpuThis(zeros(HEIGHT,WIDTH,SETUP.colours), gpuPresent);
9 for c = 1:SETUP.colours
10    otf(:,:,c) = gpuThis(imread([DATA.OTF_path{c}, DATA.OTF_name{c}, DATA.OTF_suffix{c}]), gpuPresent);
11    % Normalise
12    otf(:,:,c) = otf/max(max(otf(:,:,c)));
13 end

```

The small wrapper function `gpuThis` could be used in any other MATLAB code to investigate whether it benefits from GPU computation. The biggest speed increases will be achieved when large matrices require Fourier transforming, or other operations which can be computed in parallel.

The full SROS-SIM software, with the new GPU optimisations, is hosted at <https://github.com/laseranalyticsgroup/jRL-SIM>. Access to the repository is available upon request.

A.3 ImageJ/Fiji plugin

The LAG SIM plugin for ImageJ makes reconstructing artefact-free SIM data straightforward for users who are not experts in SIM reconstruction algorithms. The user interface is detailed in Section 2.5. This appendix contains extracts from the source code explaining the technical implementation of the plugin.

The plugin is implemented in Beanshell, a scripting language which closely resembles Java but allows execution without the need for compilation. The plugin is built upon functions provided by the fairSIM project from Marcel Müller and the Squirrel project from Siân Culley. These projects include efficient and accurate algorithms, but have generic user interfaces which requires unnecessarily repetitive clicking by LAG SIM users. The LAG SIM plugin combines the two projects into one user interface, and automatically inputs the appropriate parameters for our instrument.

The plugin includes a function for batch SIM reconstruction, and a parameter finder for choosing optimal reconstruction parameters. Extracts from the parameter finder source code are presented in Snippet A.4; the full source code is available from the ImageJ update site for LAG SIM at <https://sites.imagej.net/LagSIM>.

Lines 1-6 begin with `//@`, which is an ImageJ-specific syntax for automatically creating a dialog box with these inputs when the function is executed. Furthermore, specifying the input parameters in this syntax allows the function to be called with custom parameters from an ImageJ macro, which could be useful for future users who want to implement the parameter finder as part of a larger program.

Lines 8-13 show a sample of the modules which need to be imported for the Beanshell script to function properly. Most modules are included with a standard ImageJ installation; however, fairSIM and NanoJ-Squirrel are notable dependencies which must be installed separately. Both projects are distributed as ImageJ update sites, available in the ImageJ update menu.

The first task for the parameter finder is set the instrument parameters based on the lens and wavelength used on LAG SIM, shown in lines 22-49. In the native fairSIM plugin, these would need to be entered every time a SIM reconstruction is required; abstracting the values into the LAG SIM plugin saves users time and reduces errors, since users do not have to remember lots of arbitrary numbers. Furthermore, in batch mode the LAG SIM plugin is able to extract wavelength and lens information directly from the filename, removing the need for any user input.

Line 59 calls the fairSIM function `estimateParameters`, which extracts the spatial frequency and phase shift of the illumination pattern to sub-pixel accuracy. Lines 62-78 check the modulation depth of the illumination pattern, and warns the user if it looks low.

This may mean there is too much out-of-focus light in the image for a reliable reconstruction, or the lens and wavelength combination may be incorrect.

Lines 90-101 set up and execute the fairSIM reconstruction algorithm for the test image. This runs in a `while` loop, so that the user can continue to test parameters until they press the Finish button. Each additional test image is added to `outputStack`, so that the user can compare visually how different parameters affect the reconstruction.

If the `Calculate Squirrel metrics` checkbox is ticked, lines 105-107 generate a wide-field image to use as the Squirrel reference image. Line 109 then calls Squirrel's "Calculate Error-Map, RSE and RSP" function as an ImageJ macro, automatically entering the appropriate inputs. The RSP and RSE values are extracted from Squirrel's results window in lines 112-116, and added to LAG SIM's results window in lines 118-123, along with the test number and reconstruction parameters used.

Finally, in lines 129-133 the final SIM parameters tested are saved to the ImageJ preferences file. If the parameter tester or batch reconstruction plugins are run without inputs, these values will be loaded in as the default parameters, so that the user does not have to write them down and re-enter them.

Snippet A.4: Extracts from the Beanshell script used for the LAG SIM parameter tester

```

1 // @String(label="Filter type", choices={"Wiener out", "RL out", "RL in, Wiener out", "RL in
2 , RL out"}) filter
3 // @Double(label="WienerParameter", value=0.05) wienerParameter
4 // @Double(label="Apodisation cutoff", value=1.9) apoCutoff
5 // @Double(label="Apodisation strength", value=0.8) apoBend
6 // @Integer(label="Richardson-Lucy steps", value=5) rlSteps
7 /* etc. */
8
9 import ij.IJ;
10 import ij.ImagePlus;
11 import ij.ImageStack;
12 import org.fairsim.sim_algorithm.SimAlgorithm;
13 import nanoj.squirrel.java.gui.ErrorMap_;
14 /* etc. */
15
16 // Check open image only has 9 frames: we want this to be quick!
17 ImagePlus iPl = ij.WindowManager.getCurrentImage();
18 if (iPl.getStackSize() != 9){
19     MessageDialog errorDlg = new MessageDialog(null, "Raw data stack size error!", "Please
20         use a single-slice, single-channel image for parameter finding.");
21     return;
22 }
23
24 // Find illumination parameters with fairSIM.
25 int nrAng    = 3;      // #angles
26 int nrPha    = 3;      // #phases
27
28 switch(lens){
29     case "60X Water":
30         otfNA = 1.2;
31         pxlSize = 0.0833;
32         break;
33     case "100X Oil":
34         otfNA = 1.49;
35         pxlSize = 0.05;
36         break;
37 }
38
39 switch(wavelength){
40     case "488": // 488nm
41         emWavelen = 525;
42         px1 = -0.056;
43         py1 = -114.567;
44         px2 = 99.744;
45         py2 = 57.189;
46         px3 = -99.811;
47         py3 = 57.389;
48         break;
49     case "561": // 561nm
50         /* etc. */
51 }
52 OtfProvider otf    = OtfProvider.fromEstimate( otfNA, emWavelen, otfCorr );
53 SimParam simParam = SimParam.create( nrBands, nrAng, nrPha, imgSize, pxlSize, otf );

```

```

53 // Set initial guesses for parameters
54 simParam.dir(0).setPxPy(px1, py1);
55 simParam.dir(1).setPxPy(px2, py2);
56 simParam.dir(2).setPxPy(px3, py3);
57
58
59 SimAlgorithm.estimateParameters( simParam, rawImages, fitBand, -1, idf, visualFeedback,
60                                 null);
61
62 // Check modulation depth
63 double mod0 = simParam.dir(0).getRawModulations()[1];
64 double mod1 = simParam.dir(1).getRawModulations()[1];
65 double mod2 = simParam.dir(2).getRawModulations()[1];
66
67 if (mod0 < 0.5 || mod1 < 0.5 || mod2 < 0.5){
68     modStr = "Modulation depth looks quite low. Continue? \n";
69     modStr += "\n Modulation dir1: " + String.format("%.2f", mod0);
70     modStr += "\n Modulation dir2: " + String.format("%.2f", mod1);
71     modStr += "\n Modulation dir3: " + String.format("%.2f", mod2);
72
73     GenericDialog moddlg = new GenericDialog("Modulation information");
74     moddlg.addMessage(modStr);
75     moddlg.showDialog();
76
77     if(moddlg.wasCanceled())
78         return;
79 }
80
81 // Set up test reconstructions, and loop until user finishes
82 ImageStack outputStack = new ImageStack(2*imgSize, 2*imgSize);
83 boolean finished = false;
84
85 while (!finished){
86     if(simDlg.wasCanceled()){
87         finished = true;
88         break;
89     } else {
90         // Set up reconstruction
91         simParam.setWienerFilter( wienerParameter );
92         simParam.setRLiterations( (int)rlSteps );
93         simParam.setFilterStyle( filterStyle );
94         /* etc. */
95
96         // Run reconstruction!
97         Vec2d.Real result = SimAlgorithm.runReconstruction( simParam, rawImages, idf,
98             visualFeedback, otfBeforeShift, clipscale, null);
99
100        // Add reconstruction to test stack
101        String testParams = filter + ", w: " + wienerParameter + ", RL: " + (int)rlSteps + ", "
102          + apoC: " + apoCutoff + ", apoS: " + apoBend + ", at?: " + (otfAttenuation ? "true"
103          : "false") + ", attFW: " + attenFWHM + ", atS: " + attenStrength;
104        FloatProcessor fp = new FloatProcessor(2*imgSize, 2*imgSize, res);
105        outputStack.addSlice(testParams, fp);
106
107        if (metrics){
108
109            if (mod0 < 0.5 || mod1 < 0.5 || mod2 < 0.5){
110                modStr = "Modulation depth looks quite low. Continue? \n";
111                modStr += "\n Modulation dir1: " + String.format("%.2f", mod0);
112                modStr += "\n Modulation dir2: " + String.format("%.2f", mod1);
113                modStr += "\n Modulation dir3: " + String.format("%.2f", mod2);
114
115                GenericDialog moddlg = new GenericDialog("Modulation information");
116                moddlg.addMessage(modStr);
117                moddlg.showDialog();
118
119                if(moddlg.wasCanceled())
120                    return;
121            }
122
123            metrics = false;
124        }
125    }
126 }

```

```
104 // Create widefield projection as Squirrel reference image
105 ZProjector projector = new ZProjector().setImage(iPl);
106 ImagePlus widepl = projector.getProjection();
107 widepl.setTitle("squirrel-wf");
108
109 IJ.run("Calculate Error-Map, RSE and RSP", "reference=[squirrel-wf] super-resolution
110 =[squirrel-sr] rsf=[-- RSF unknown, estimate via optimisation --] max.=5 crop
111 enable maximum=0 show show_rsf-convolved");
112
113 // Extract metrics from Squirrel table, and add to LAG SIM test table
114 Window rw = WindowManager.getWindow("RSP and RSE values");
115 ResultsTable rt = rw.getResultsTable();
116 double rsp = rt.getValue("RSP (Resolution Scaled Pearson-Correlation)", 0);
117 double rse = rt.getValue("RSE (Resolution Scaled Error)", 0);
118 rw.close();
119
120 testResults.incrementCounter();
121 testResults.addValue("Test #", testNumber);
122 testResults.addValue("Squirrel RSP", rsp);
123 testResults.addValue("Squirrel RSE", rse);
124 testResults.addValue("Parameters", testParams);
125 testResults.show("Test results");
126 }
127 }
128 // Save parameters to ImageJ preferences.
129 Prefs.set("lagsim.filter", filter);
130 Prefs.set("lagsim.wienerParameter", String.valueOf(wienerParameter));
131 Prefs.set("lagsim.rlSteps", String.valueOf(rlSteps));
132 Prefs.set("lagsim.apoCutoff", String.valueOf(apoCutoff));
133 /* etc. */
```


Appendix B

Directed flow in the Endoplasmic Reticulum

Chapter 5 presented evidence for active flow in the endoplasmic reticulum (ER). The chapter focussed on work that I had personally contributed to the project; this appendix provides background investigations completed by my collaborators which provide extra context for the work. Further details still can be found in *Nature Cell Biology*.

B.1 Flow calculations

Proteins localised within the ER network, known as luminal proteins, were initially imaged over time using a photoswitchable protein to track their movement. Figure B.1 shows a COS-7 cell expressing Dendra2-ER, a green-to-red photoswitchable fluorescent protein which usually has an emission peak at 507 nm. After a brief illumination with a 405 nm laser, however, the fluorescent protein is photoconverted and its emission peak shifts to 573 nm. This property was exploited to track packets of luminal protein by illuminating a small area of the ER network, then tracking the 573 nm signal, as shown in Figure B.1.

From this experiment, it was noted that luminal proteins did not appear to move with exclusively diffusive motion, but rather with directed motion along ER tubes and with Brownian motion at junctions. However, the photoswitching experiment can only track one particle group at a time, which is impractical for gathering large numbers of tracks for statistical confidence.

To gather a large number of luminal protein tracks, Halo-tagged calreticulin (Ctr) was labelled with chloroalkane–tetramethylrhodamine (TMR). The brightness and photostability of TMR enabled single particle tracking over long trajectories, with multiple tracks per cell.

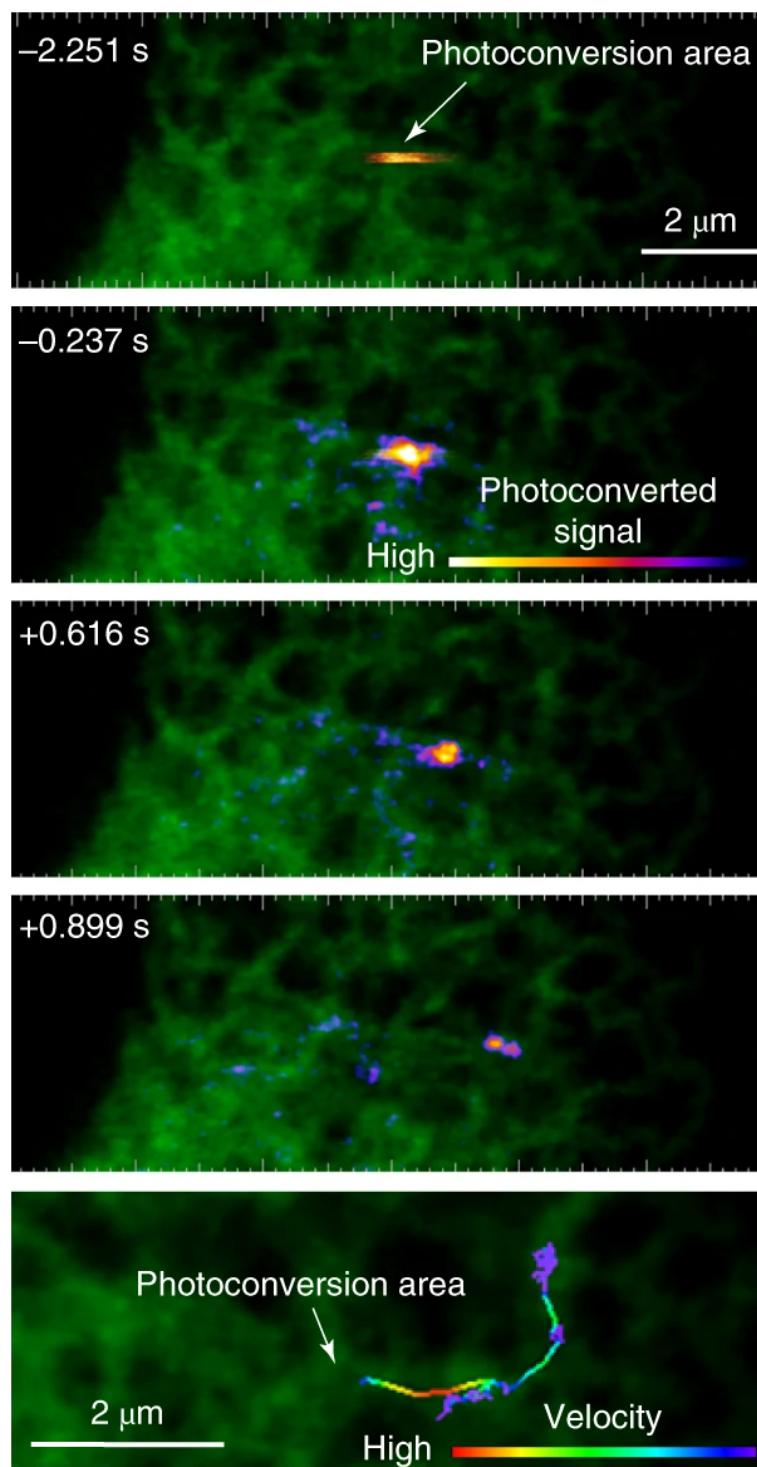


Figure B.1: The figure shows a timelapse of COS-7 labelled with Dendra2-ER, a photo-switchable fluorescent protein. In the first frame, a small area of protein is photoconverted with a 405 nm laser. The converted packet is then tracked, showing fast directed motion as it travels along tubules, but slow Brownian motion at junctions.

The velocity distributions from the single particle tracking were presented in Figure 5.1a, and show the same velocity distribution as Figure B.1, with fast motion along tubes and slow, diffusional motion at junctions.

Two models were created to compare the fit of the data: a purely diffusional model, and a diffusion plus flow model. The diffusion model is given by Equation B.1, where t is a time step, D is the diffusion coefficient, and η is a white noise term. The track data $\mathbf{X}(t)$ and, at a later time point, $\mathbf{X}(t + \Delta t)$ is extracted from the single particle images. From this equation the diffusion coefficient is estimated, and the equation is used to produce the solid black line shown in Figure 5.1b.

$$\mathbf{X}(t + \Delta t) = \mathbf{X}(t) + \sqrt{2D\Delta t}\eta \quad (\text{B.1})$$

The diffusion-only model does not provide a good fit to the data. Instead, a model with forced movement as well as a Brownian drift term was used, as shown in Equation B.2, where \mathbf{F} is a field of force, γ is viscosity, and $\dot{\omega}$ is a noise term derived from thermal agitation.

$$\dot{\mathbf{X}} = \frac{\mathbf{F}(\mathbf{X}(t))}{\gamma} + \sqrt{2D}\dot{\omega}(t) \quad (\text{B.2})$$

Fitting this equation to the single particle tracks produces the dashed line shown in Figure 5.1b. This provides a much closer fit to the experimental data, which is particularly noticeable in the cumulative velocity distribution shown in Figure 5.1c. In total, 80563 single particle tracks were used to calculate model parameters.

B.2 FRET measurements

When ATP was depleted from cells, a decrease in particle velocity was noticed, shown in Figure 5.2a. This could either come from a decrease in active transport due to energy starvation, or an increase in the resistance to motion. To test the latter hypothesis, a method of measuring ER crowdedness was devised by measuring the fluorescence lifetime (FLT) of a sensitive fluorescence resonance energy transfer (FRET)-based probe.

The ER-crowdedness FRET probe was based on an expression vector encoding a cytosol-localized crowding probe, which was modified to encode an ER-localized probe by amino-terminal addition of mouse pro

Under normal conditions, the probe shown an average fluorescent lifetime of 2932 ps. To test the sensitivity of the probe, cells were placed under hyperosmotic or hypo-osmotic conditions with Hanks balanced salt solution (2.5 mM KCl, 1.2 mM CaCl₂, 0.5 mM MgCl₂, 5 mM glucose, 10 mM HEPES pH 7.5) supplemented by 500 mM NaCl or 0 mM NaCl respectively. Under hyperosmotic conditions, the cell shrinks and we expected luminal crowdedness in the ER to increase; this was measured as a decrease in the fluorescent lifetime, to 2546 ps. Conversely, under hypo-osmotic conditions we expect the cell to expand and crowdedness to decrease, which increased the fluorescent lifetime of the FRET probe to 3302 ps.

When ATP was depleted from the cell, no significant change in fluorescent lifetime was observed compared to the untreated condition. This result contradicted the hypothesis that a decrease in particle velocity upon ATP depletion is caused by an increase in resistance from additional crowding, and led us into further investigation of the active process causing ER luminal flow, as detailed in Chapter 5.

The fluorescent lifetime results measured are presented as an ER-crowdedness measure in Figure 5.2b, where error bars showing standard deviation were calculated from 22 independently sampled cells.

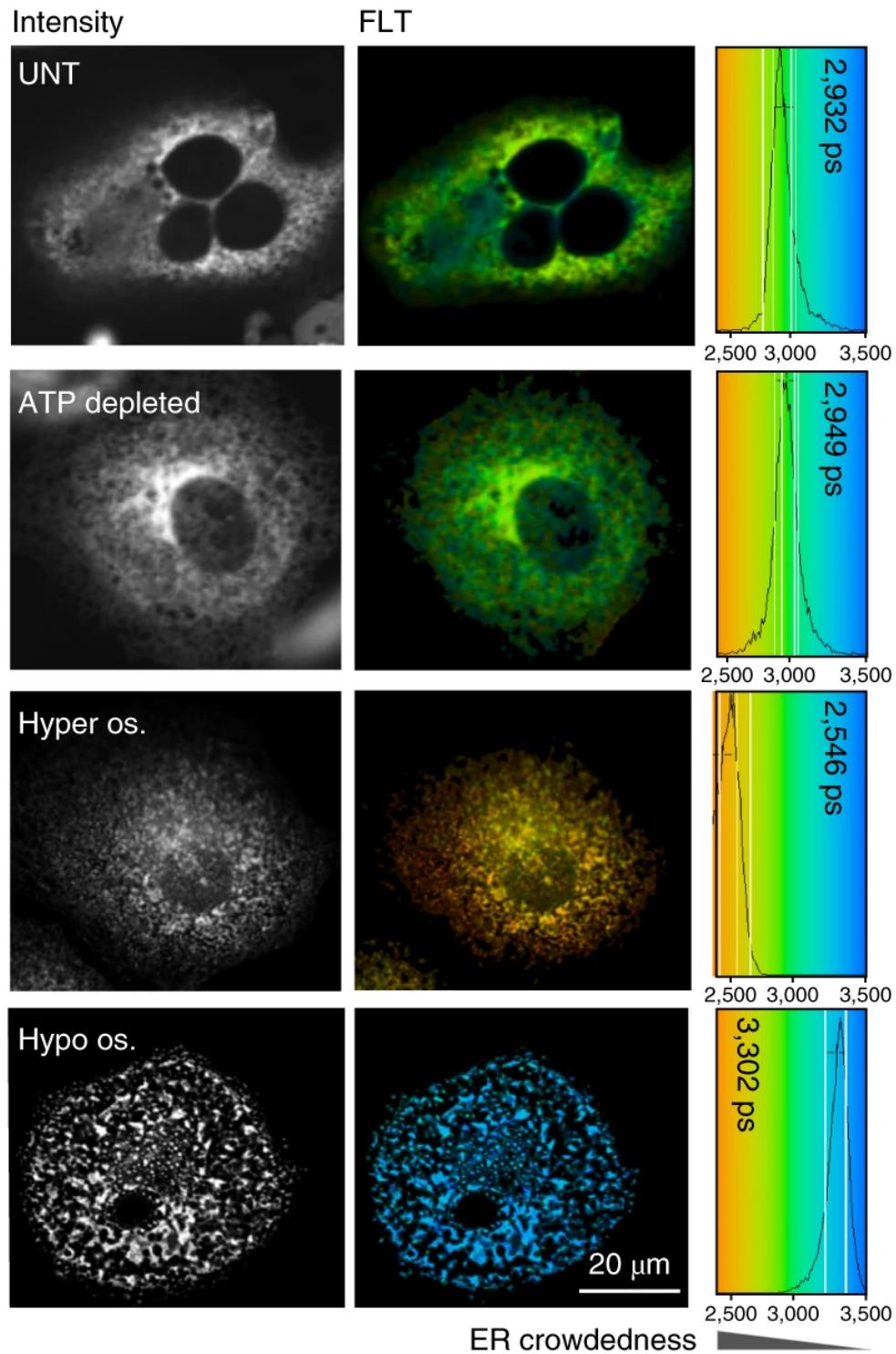


Figure B.2: Fluorescent lifetime (FLT) was measured in SHSY-5Y cells transfected with a FRET-based probe to assess ER crowdedness. There is no significant difference in FLT between the untreated (UNT) case and the ATP depleted condition. However inducing hyperosmotic or hypo-osmotic conditions with salt solutions causes crowdedness to increase or decrease respectively, leading to measurable changes in the FLT.

