

ColocZStats: A 3D Slicer Extension for Assessing Colocalization in Confocal Microscopy

by

© Xiang Chen

A thesis submitted to the
School of Graduate Studies
in partial fulfillment of the
requirements for the degree of
Master of Science

Supervisor: Dr. Oscar Meruvia-Pastor

Dr. Touati Benoukraf

Department of Computer Science

Memorial University of Newfoundland

June 2024

St. John's

Newfoundland

Abstract

Confocal microscopy has evolved as a widely adopted imaging technique in molecular biology and is frequently utilized to achieve accurate subcellular localization of proteins. Applying colocalization analysis on image z-stacks obtained from confocal fluorescence microscopes is a dependable method to reveal the association between different molecules. In addition, despite the established advantages and growing adoption of virtual reality (VR) technology in various microscopy research domains, there has been a scarcity of systems supporting colocalization analysis within VR space.

In this context, several broadly employed biological image visualization platforms were meticulously explored in this study to comprehend the current landscape. It has been observed that while these applications can generate three-dimensional (3D) reconstructions for the z-stacks and transfer them into an immersive VR scene, there is still a common necessity for them to optimize the capability for executing quantitative colocalization analysis on such images.

To constructively improve the above circumstances, an extension called ColocZStats has been developed for 3D Slicer. With a user-friendly interface, ColocZStats allows investigators to conduct intensity thresholding and region-of-interest (ROI) selection on imported 3D image stacks. It can deliver several essential colocalization metrics for structures of interest in the form of diagrams and spreadsheets. While currently serving as a desktop tool, ColocZStats has been continuously enhanced and will be systematically extended into VR in the next phase.

Acknowledgments

First and foremost, I would like to express my sincere gratitude to my supervisors, Dr. Oscar Meruvia-Pastor and Dr. Touati Benoukraf, for their continuous support and patient guidance throughout my master's studies. Their profound knowledge, tremendous enthusiasm, and heartfelt encouragement have significantly impacted my research and the process of writing this thesis. Their positive attitude and innovative ideas, demonstrated consistently in research and daily life, have set an excellent example for me.

Second, I thank Memorial University of Newfoundland's School of Graduate Studies, the Memorial University of Newfoundland's Department of Computer Science, and the Memorial University of Newfoundland's Faculty of Medicine for their financial and equipment support. I also appreciate Teena Thakur and Dr. Anand D. Jeyasekharan (the Cancer Science Institute of Singapore, National University of Singapore) for their help in acquiring the confocal microscopy data for this research.

Furthermore, I would like to take this opportunity to express my appreciation to all the good friends and classmates I met during my master's studies. We have mutually inspired each other to learn professional knowledge, and we have also explored Newfoundland's beautiful scenery together. All these experiences have become my precious memories.

Last but not least, I want to convey my immense thanks to my parents, Mr. Qingdong Chen and Mrs. Yufang Dong, and my fiancée, Miss Hemeihui Zhao. Their

selfless love and unwavering support have enabled me to advance toward pursuing my dreams and achieving my goals continuously. I dedicate my achievements to them.

Contents

Abstract	i
Acknowledgments	ii
List of Figures	xi
List of Tables	xii
1 Introduction	1
1.1 Background and Related Work	1
1.1.1 Colocalization Analysis on Confocal Z-stacks	2
1.1.2 Common Methods for Visual Assessment of Colocalization	4
1.1.3 Unlocking VR's Potential for Colocalization Analysis	7
1.1.4 Pertinent Visualization Tools	12
1.2 Motivation	14
1.3 Overview of the Developed Tool's Features	15
2 Methods	16

2.1	Description of Sample Data Source	16
2.2	ColocZStats Development	18
2.2.1	Selection Criteria for 3D Slicer	18
2.2.1.1	Comparison of 3D reconstructions	18
2.2.1.2	High Extensibility of 3D Slicer	21
2.2.2	Implementation Specifics	22
2.3	Statistical Analysis of Colocalization	24
2.3.1	Pearson’s Correlation Coefficient	24
2.3.2	Intersection Coefficients	26
3	Results	30
3.1	Availability of ColocZStats	30
3.2	Input Image File	32
3.3	User Interface	32
3.4	The Workflow of ColocZStats	34
3.5	Produced Graphical Representations	38
3.5.1	Venn Diagrams	38
3.5.2	2D Histograms	41
3.6	Results Spreadsheet	44
3.7	Case Study	46
3.8	Comprehensive Comparison of Visualization Tools	49

4	Conclusions	52
5	Future Work	55
	Bibliography	58

List of Figures

1.1	Confocal microscope structure. Reprinted from ‘Confocal microscopes,’ in Britannica, 2024, February 23, https://www.britannica.com/technology/microscope/Confocal-microscopes , Copyright 2012 by Encyclopædia Britannica, Inc.	3
1.2	An illustration demonstrating the storage of a confocal image z-stack as a 3D texture in GPU memory, with each block within the texture representing one voxel. Adapted from ‘Virtual Reality Assisted Fluorescence Microscopy Data Visualisation and Analysis for Improved Understanding of Molecular Structures Implicated in Neurodegenerative Diseases,’ by Theart, R. P., 2020, Stellenbosch University, http://hdl.handle.net/10019.1/107804 , p.15, Copyright 2020 by Stellenbosch University.	5
1.3	Optical properties of the hit voxels, such as color and opacity, are integrated to achieve the final visualization effect.	6

1.4	A scene demonstrating the ‘EyePhone’ system in the late 1980s, where two people were able to view and manipulate objects in a computer-generated environment using special goggles and a DataGlove. Reprinted from ‘Dawn of the new everything: Encounters with reality and virtual reality,’ by J. Lanier, 2017, New York, New York: Henry Holt and Company Press. Copyright 2017 by Jaron Lanier.	8
1.5	HTC Vive Cosmos Elite VR System. Reprinted from ‘Cosmos Elite Overview,’ in HTC Vive product page, 2024, February 23, https://www.vive.com/eu/product/vive-cosmos-elite/overview/ , Copyright 2024 by HTC Corporation.	9
1.6	A scenario where contextual information appears when using controllers to touch an organelle of a breast cancer cell in VR space. Reprinted from ‘Journey to the centre of the cell: Virtual reality immersion into scientific data,’ by Johnston, A. P., Rae, J., Ariotti, N., Bailey, B., Lilja, A., Webb, R., ... & Parton, R. G., 2018, Traffic, 19(2), 105-110, Copyright 2017 by John Wiley & Sons A/S.	11

2.1	A series of screenshots displaying the visualization of the same specimen’s channels from an identical perspective across all the mentioned programs. For each sub-figure, from left to right, two separate channels of the sample stack, their superposition, and a magnification of a specific region in the superposition are shown. (A) The volumetric rendering of the two channels in ChimeraX’s desktop version. (B) The same series of scenes as (A). They were captured from a VR environment created by ChimeraX VR. The first rows in (C), (D), (E), and (F), respectively, show the scenes without external lighting in the ExMicroVR, ConfocalVR, SlicerVR, and the desktop viewport of 3D Slicer. The second row of (C) to (F) shows the scenes with external lighting turned on.	20
3.1	A screenshot showing the ‘Slicer Extensions Catalog’ via the ‘Extensions Manager’ in 3D Slicer. ColocZStats is offered within the ‘Quantification’ category.	31
3.2	The Graphical User Interface (GUI) of ColocZStats.	34
3.3	(A) Typical workflow of ColocZStats. (B) Example operation scenarios in ColocZStats that correspond to the steps in the workflow. (C) Visual abstractions that correspond to the steps in the workflow.	35

3.4	(A) and (B) present the thresholded volume renderings when two or three channels of the entire sample image stack were selected. (C) and (D) depict the respective resulting Venn diagrams. (E) and (F) illustrate all the defined thresholds and the colocalization metrics results retained to four decimal places.	40
3.5	A visual explanation of a 2D histogram’s generation process. Adapted from ‘Scientific Volume Imaging-2D histograms,’ 2024, Scientific Volume Imaging B.V., https://svi.nl/TwoChannelHistogram , Copyright 1995-2024 by Scientific Volume Imaging B.V. (A) The volume rendering of channel 1 and an example intensity at a specific position. (B) The volume rendering of channel 2 and an example intensity at the same position. (C) The coordinates in the 2D histogram are generated by the combination of those intensities.	42
3.6	An illustration depicting the relationship between the example 2D histogram and its corresponding Venn diagram. Coordinate of point (a): (1500,25470); Coordinate of point (b): (0,2000); Coordinate of point (c): (1500,2000); Coordinate of point (d): (1500,0); Coordinate of point (e): (9006,2000).	43

3.7	All of the sub-sheets in each spreadsheet result from each single computation. (A) The selected channels and their respective threshold ranges. (B) The timestamp of this computation. (C) The custom annotation for the specified image stack. (D) The related information of the defined ROI for this computation. (E) The PCCs for all possible pairwise combinations of the selected channels. (F) All the resulting intersection coefficients. (G) All the resulting 2D histograms, with each histogram illustrating a pair of channels. (H) The resulting illustration embeds the Venn diagram and all coefficient results.	45
3.8	(A) Volume rendering of all channels within the same ROI box. (B) Venn diagrams corresponding to all the cases. (C) All the associated thresholds and colocalization metrics' results. (D) Additional histograms illustrating the proportions of overlaps. They were created based on the data derived directly from the Venn diagrams. 'Overlap 1': The overlap between Channel 1 and Channel 2. 'Overlap 2': The overlap between Channel 1 and Channel 3. 'Overlap 3': The overlap between Channel 2 and Channel 3. 'Overlap 4': The overlap of all three channels.	48

List of Tables

3.1 Programs Features Overview. 51

Chapter 1

Introduction

A software tool called ColocZStats has been proposed in this thesis. ColocZStats can help clearly visualize confocal 3D datasets and produce objective colocalization metrics.

To better contextualize this contribution, this chapter begins by elucidating the background and relevant work closely associated with the development of the tool. Subsequently, the motivation behind ColocZStats is clarified through an investigation into the current status of similar functionalities possessed by a series of related platforms. Finally, the features of the developed application are outlined.

1.1 Background and Related Work

This section discusses the advantages of confocal microscopy, the concept and the importance of colocalization analysis, the principle of visually assessing colocalization

and prevalent visual modeling techniques for confocal z-stacks. It also explores the evolving trends of VR applications in microscopic visualization and VR's significant potential for advancing colocalization analysis. Additionally, it provides a synopsis of several pertinent visualization software programs.

1.1.1 Colocalization Analysis on Confocal Z-stacks

Compared to conventional fluorescence microscopes, the most significant advantage of confocal microscopes is that they can exclude the out-of-focus light from either above or below the current focal plane [1]. This capability facilitates precisely detecting the specific organelle in which the target molecule is present. In addition, the confocal microscope's features of sharpening fluorescence images and reducing haze contribute to enhancing image clarity [2]. Many of these two-dimensional (2D) image slices can be continuously collected from focal planes at different depths along the z-dimension and eventually assembled to produce a z-stack comprising the specimen's entire 3D data [2, 3]. Figure 1.1 depicts a typical configuration of a confocal microscope [4].

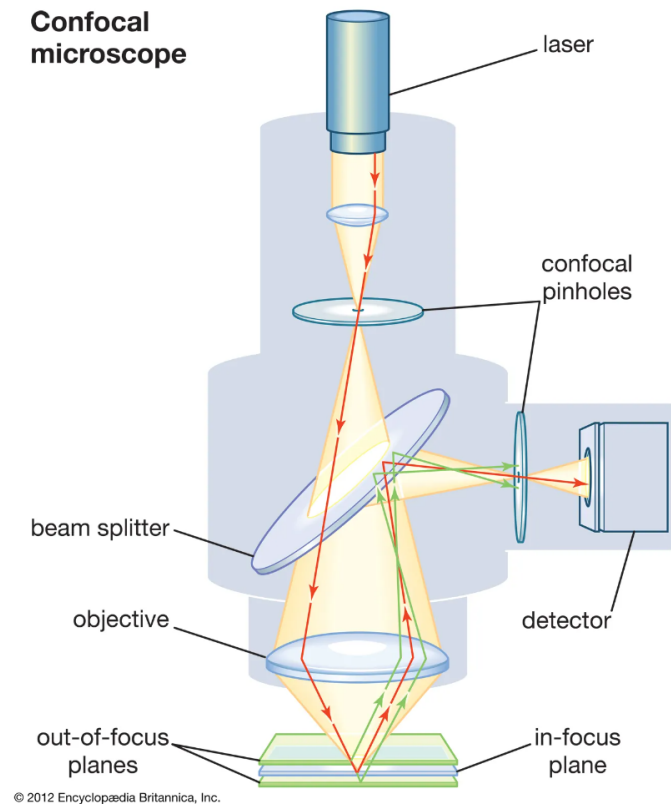


Figure 1.1: Confocal microscope structure. Reprinted from ‘Confocal microscopes,’ in Britannica, 2024, February 23, <https://www.britannica.com/technology/microscope/Confocal-microscopes>, Copyright 2012 by Encyclopædia Britannica, Inc.

Examining interactions between different proteins or molecular structures holds significant importance in biological sciences. Biologists often perform colocalization analysis on confocal image z-stacks to better understand the roles and interactions of proteins [5, 6]. Colocalization detects the spatial overlap between distinct fluorescent labels with different emission wavelengths to determine whether the fluorophores

are close or within the same region [7, 8]. Colocalization involves two aspects: co-occurrence and correlation. Co-occurrence refers to the simple spatial intersection of different fluorophores. Correlation refers to distinct fluorophores codistributed proportionally with a more apparent statistical relationship [8, 9]. Typical application examples of colocalization analysis include confirming whether a specific protein associates with microtubules [10, 11] or mitochondria [12], or verifying whether different proteins associate with identical plasma membrane domains [9, 13].

1.1.2 Common Methods for Visual Assessment of Colocalization

Visualizing superimposed fluorescence micrographs is the most common method for assessing colocalization. When the images of each fluorescence label are merged, their combined contribution can be indicated by the color of the microstructure appearance. For instance, because of the combined effects of green and red fluorescence, the colocalization of fluorescein and rhodamine can be recognized in yellow structures [9].

When analyzing 3D datasets, a popular visualization method involves performing 3D reconstruction to enhance the user's perception when observing samples. Volume rendering has been extensively utilized to achieve this objective [3, 14, 15]. Volume rendering includes diverse techniques for projecting a 3D scalar dataset onto a 2D viewing plane from any angle, creating the illusion of a solid 3D structure [16, 17]. A standard 3D volumetric dataset comprises a sequence of evenly-spaced 2D slice

images, each with a consistent number of pixels arranged in a regular grid pattern. In the context of 3D reconstruction, such a dataset is typically regarded as composed of volume elements, known as ‘voxels,’ arranged in a discretely sampled regular volumetric grid [16]. Figure 1.2 shows the correspondence between a z-stack and its 3D texture.

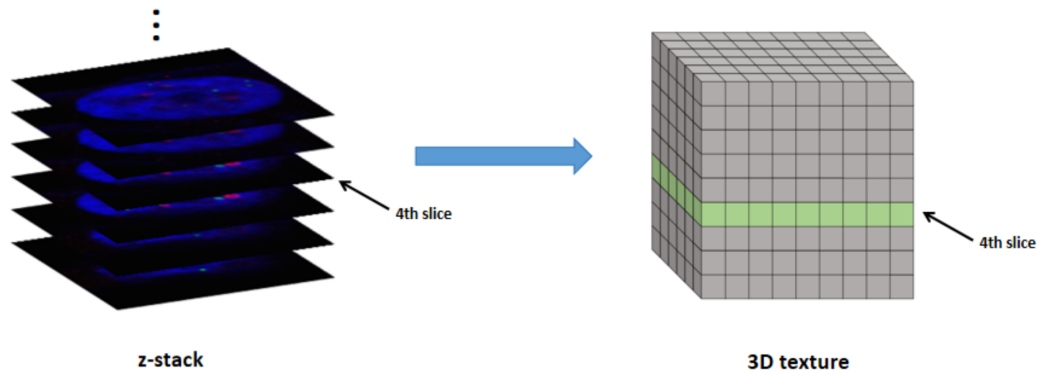


Figure 1.2: An illustration demonstrating the storage of a confocal image z-stack as a 3D texture in GPU memory, with each block within the texture representing one voxel. Adapted from ‘Virtual Reality Assisted Fluorescence Microscopy Data Visualisation and Analysis for Improved Understanding of Molecular Structures Implicated in Neurodegenerative Diseases,’ by Theart, R. P., 2020, Stellenbosch University, <http://hdl.handle.net/10019.1/107804>, p.15, Copyright 2020 by Stellenbosch University.

Volume rendering techniques are divided into two categories: Direct Volume Rendering (DVR) and Indirect Volume Rendering (IVR). Given that fluorescence mi-

croscopy data analysis usually focuses on the overlap of different molecular structures, DVR techniques are more suitable for visualizing such data than IVR techniques, which require extracting surface geometry before rendering [16]. This preference arises from the feature of DVR techniques that can model each voxel as a light-emitting medium with specific emission and absorption properties, enabling the creation of a 3D semi-translucent appearance for the data [18], thereby facilitating a clearer exposition of the internal structures. ‘Ray Casting’ is an effective technique that is classified as DVR. By sampling data values along the ray’s path passing through the data’s volume, this technique can integrate all the hit voxels’ optical properties for rendering [19, 20]. An illustration of the ‘Ray Casting’ scheme is presented in Figure 1.3.

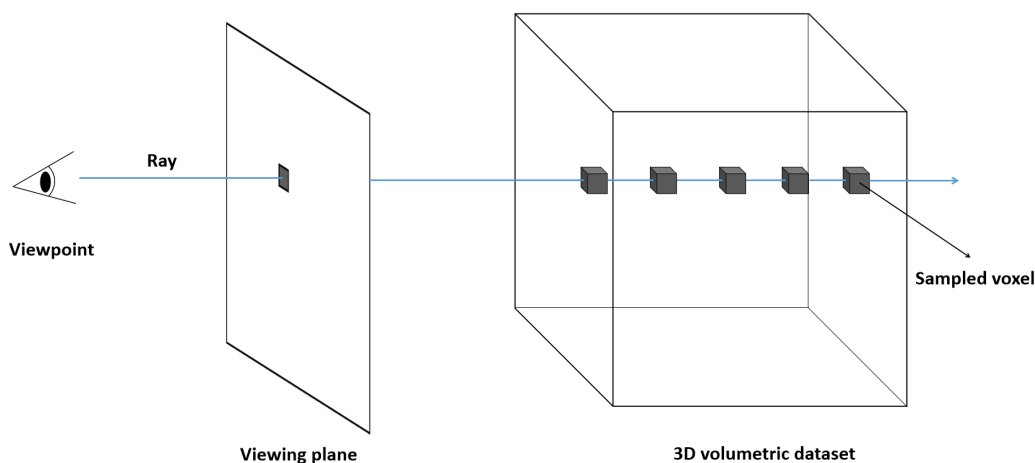


Figure 1.3: Optical properties of the hit voxels, such as color and opacity, are integrated to achieve the final visualization effect.

1.1.3 Unlocking VR's Potential for Colocalization Analysis

Another cutting-edge visualization technology is VR. VR is a digitally created immersive 3D simulated environment that closely resembles reality. While being fully immersed in this environment, users can navigate through and interact with virtual objects. Ivan Sutherland, widely regarded as the creator of the first VR Head-Mounted Display (HMD) system, emphasized the significant potential of immersive digital platforms in providing an intuitive grasp of scientific and mathematical fields in an insightful article as early as 1965 [21, 22]. Since the development of the first batch of commercial electronic gadgets, such as the 'EyePhone' [23] and other disruptive devices supporting 3D visualization and interaction by innovator Jaron Lanier in the 1980s, VR technology has continuously evolved and become increasingly popular.

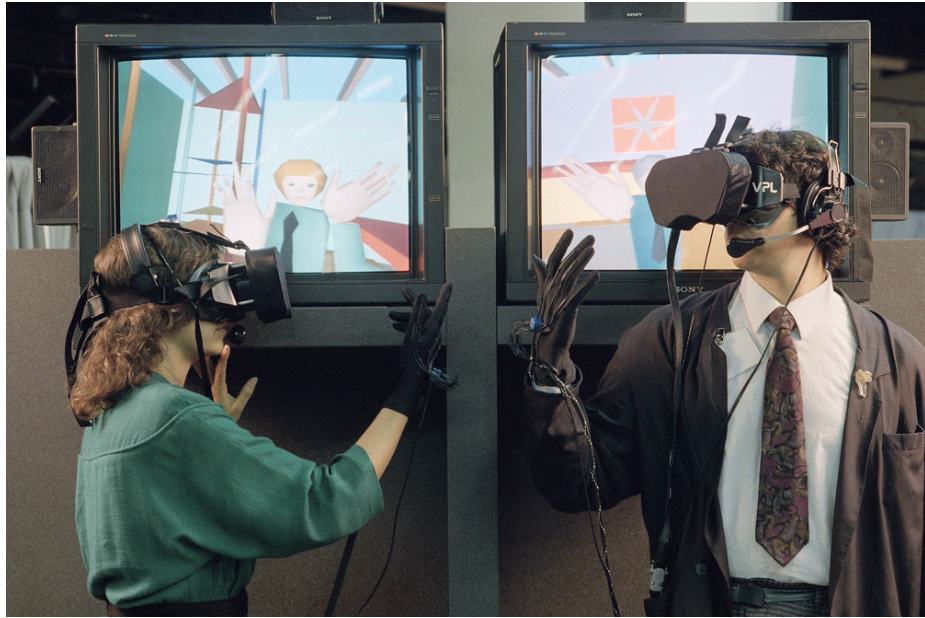


Figure 1.4: A scene demonstrating the ‘EyePhone’ system in the late 1980s, where two people were able to view and manipulate objects in a computer-generated environment using special goggles and a DataGlove. Reprinted from ‘Dawn of the new everything: Encounters with reality and virtual reality,’ by J. Lanier, 2017, New York, New York: Henry Holt and Company Press. Copyright 2017 by Jaron Lanier.

Recent advancements in VR technology have given rise to HMDs that offer low latency head tracking, high resolution, and expansive Field-of-View (FoV) [24]. As the mainstream display device for modern VR, HMD delivers distinct images to each eye, facilitating a 3D sense of the surroundings. The HTC Vive is one of the representatives of advanced VR systems. Its HMD provides an approximate FoV of around 110° through two displays refreshed at a rate of 90 Hz. The Vive HMD’s tracked orientation and position are mainly updated by inertial measurement units within

the headset using dead reckoning, achieving higher update rates [25]. In addition to an HMD, the HTC Vive includes two controllers and two infrared laser emitter units called ‘lighthouses’. The HTC Vive has found applications in various realms, such as medicine, engineering, and robotics [26]. Figure 1.5 below illustrates one of the HTC Vive series products, the HTC VIVE Cosmos Elite [27].



Figure 1.5: HTC Vive Cosmos Elite VR System. Reprinted from ‘Cosmos Elite Overview,’ in HTC Vive product page, 2024, February 23, <https://www.vive.com/eu/product/vive-cosmos-elite/overview/>, Copyright 2024 by HTC Corporation.

Research related to bioinformatics involves significant amounts of highly diverse data, including data generated from proteomics, metabolomics, and transcriptomics

[28]. From the modeling perspective, the objects studied in bioinformatics are usually at the nanoscale level. Effective models are vital to promoting related research in such domains invisible to the naked eye [29]. In the past, the low rendering speed has been a significant limiting factor for the effectiveness of microscopy visualization. Benefiting from recent enhancements in consumer graphics performance, the required frame rates for VR with acceptable rendering quality have become feasible [16].

The progress in VR-related technologies mentioned above, along with VR's enticing immersive features, bring considerable promise to the display of microscopy models. In the past few years, VR has been increasingly utilized for visualizing cellular and molecular structures [30]. Typical examples encompass a VR program designed by Angus Johnston et al. [31] for educational purposes, offering a virtual journey inside breast cancer cells obtained by serial block-face electron microscopy, a web tool called 'RealityConvert' that can create VR scenarios with protein structures [32], and some applications that use VR to promote drug design, such as 'molecular rift' [33] and '3D-Lab' [30, 34]. In a VR scene, users can intuitively manipulate complex biological data, obtaining a more realistic 3D awareness in scientific exploration. For instance, they can grasp the image to rotate and scale, swiftly adjust its viewing parameters, and examine 3D structures at different levels. Figure 1.6 illustrates a scenario exploring breast cancer cells in VR space using Angus Johnston's program.

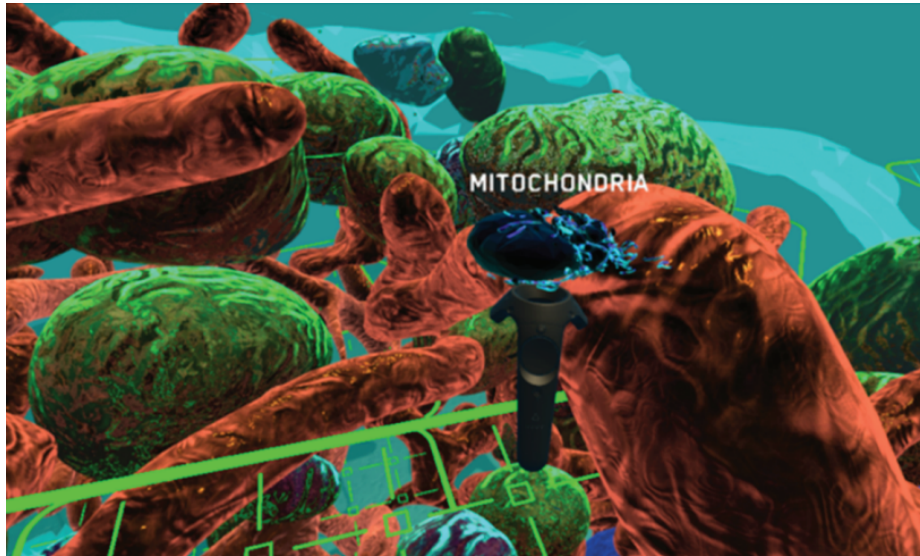


Figure 1.6: A scenario where contextual information appears when using controllers to touch an organelle of a breast cancer cell in VR space. Reprinted from ‘Journey to the centre of the cell: Virtual reality immersion into scientific data,’ by Johnston, A. P., Rae, J., Ariotti, N., Bailey, B., Lilja, A., Webb, R., ... & Parton, R. G., 2018, *Traffic*, 19(2), 105-110, Copyright 2017 by John Wiley & Sons A/S.

Although the application of VR in various subfields of bioinformatics has been steadily increasing, a study published in 2017 by Theart, R. P., Loos, B., and Niesler, T. R. [3] revealed that, before that time, no applications had provided the capability for colocalization analysis in VR. Nevertheless, this study, along with its subsequent research [35], collectively demonstrated the substantial potential and advantages of performing colocalization analysis for z-stacks based on the generated volume-rendered images in an immersive VR environment, namely that the efficiency

of conducting such analysis and the precision of inspecting and assessing biological samples can be significantly improved.

1.1.4 Pertinent Visualization Tools

In consideration of the above background, several well-known applications, with a focus on 3D graphics systems with VR functionalities, were investigated during the study period to gain an overview of the current status of these commonly accepted bioimaging visualization and analysis platforms for visualizing confocal microscopy data and evaluating its degree of colocalization.

ExMicroVR [36] is a VR tool created for the immersive visualization and manipulation of multi-channel confocal image stacks. Its VR environment and easy-to-use user interface allow it to considerably expand microscopic samples so biologists can view and explore the molecules' structures in greater detail [37]. Another software, ConfocalVR [38], in its current form, is an upgraded version of ExMicroVR. Not only does it have more added interactive features, but it also provides a range of relatively advanced image analysis capabilities, such as adding markers in 3D scenes, counting the number of interesting objects, and measuring the distance between them, allowing researchers to further investigate the complexity of cellular structures [39]. ChimeraX [40] is an interactive platform for visualizing diverse types of data, including atomic structures, sequences, and 3D multi-channel microscopy data [41, 42]. It supplies around 100 different analysis functionalities [30, 43]. Moreover, it integrates

a fast macromolecular crystallographic information file (mmCIF) reader to write mmCIF files. Data from the Protein DataBank (PDB) [44], Research Collaboratory for Structure Bioinformatics (RCSB) [45], and Electron Microscopy Data Bank (EMDB) [46], can be accessed from local files or downloaded directly from online web services. ChimeraX VR is the VR extension of ChimeraX [47], enabling users to interact with cellular protein structures with stereo depth perception. The 3D image data generated by light microscopy, electron microscopy, and x-ray crystallography can be displayed as colored translucent voxels, surfaces, and meshes in ChimeraX VR [43]. When activated, a floating panel, precisely the same as the desktop interface, is available in the VR scene to help users use controllers to perform all necessary manipulations on the images. 3D Slicer [48] is a broadly recognized, accessible, and open-source platform that provides multifarious biomedical image processing and visualization features [49, 50]. Similarly, SlicerVR is the VR extension within 3D Slicer. Thanks to the capabilities of the 3D Slicer ecosystem, SlicerVR offers seamless VR integration in this popular image computing application [51] so that observers can quickly transfer images displayed in desktop mode to the VR scenario. Additionally, it supports multi-user collaboration, allowing images within the exact scene to be manipulated synchronously [52].

1.2 Motivation

Through this investigation, it was found that all of the platforms above are capable of reading datasets generated by fluorescence confocal microscopy and converting them into 3D semi-translucent voxels that can be delivered to VR scenes for visualization. For instance, ConfocalVR offers a ‘blend function’ option that allows the extraction of voxels representing the spatially intersecting regions between two channels. The 3D rendering of these voxels can aid researchers in visually assessing the level of colocalization between channels. However, for more comprehensive colocalization analysis, relying solely on subjective identification of the relative distribution of different molecules from a visual perspective is insufficient. Objective and quantitative analysis is crucial. Despite the commendable image visualization and diverse analytical capabilities inherent in these platforms, there is still room for them to improve in obtaining colocalization statistics for confocal stacks. As indicated by a published work of Stefani et al. [53], developing a dedicated tool to gain objective colocalization measurements remains one of the goals and challenges for ConfocalVR.

To address the aforementioned limitation and take advantage of the comparatively superior high extensibility of 3D Slicer, which supports the creation of interactive and batch-processing tools for various purposes [54], a free, open-source extension called ColocZStats has been developed for 3D Slicer.

1.3 Overview of the Developed Tool's Features

ColocZStats is currently designed as a desktop application, which enables users to visually observe the spatial relationship between different biological microstructures while performing thresholding and ROI selection on the channels' 3D volumetric representations via an easy-to-use graphical user interface (GUI) and then acquiring critical colocalization metrics with one-mouse click. The proposal of this tool contributes to supplementing the capabilities of 3D Slicer in visualizing multi-channel confocal image stacks and quantifying colocalization, thereby broadening its scope as an integrated image analysis platform. Notably, the SlicerVR is constantly being enhanced. For instance, one of its features that is in the plans for future versions is the capability to use interactive floating widgets in the immersive environment [55], providing considerable feasibility for expanding ColocZStats into VR.

Chapter 2

Methods

This chapter begins by providing information on the acquisition process of the sample image data used in this thesis, which encompasses cell culture, staining, and imaging, serving as a reference for the materials applied. Following that, the reasons for choosing 3D Slicer as the development framework are explained. Next, the discussion shifts to the specifics of the implementation. In the last section, the discussion delves into the principles and formulas of the colocalization coefficients that ColocZStats can generate.

2.1 Description of Sample Data Source

To showcase the capabilities of ColocZStats discussed in this thesis, the confocal z-stack data collected during a study on the colocalization of DSS1 nuclear bodies with other nuclear body types was utilized. DSS1, also known as SEM1, is a gene that

encodes a protein crucial for various cellular processes, most notably the function of the 26S proteasome complex in protein degradation [56].

More specifically, a human ovarian clear cell carcinoma cell line (RMG-I) was seeded at 100,000 cells per well onto a coverslip in a 6-well plate and allowed to grow till 70% confluency. On the day of staining, the cells were fixed with 4% ice-cold paraformaldehyde for 15 mins at room temperature (RT) and blocked with 2% Bovine Serum Albumin (BSA) in 0.1% Phosphate Buffer Saline containing 0.1% Triton-X (PBSTx) for 30 mins. Following fixation and blocking, the cells were incubated with a primary antibody cocktail containing anti-DSS1 (Catalogue# NB100-1334, Novus Biologicals) and anti-PML (Catalogue# sc-966, SCBT) for 1h at RT. After that, the cells were washed 3 times with 0.1% PBSTx for 5 mins each and incubated with a secondary antibody cocktail containing anti-goat-647 (for DSS1), anti-mouse-488 (for PML) and Hoechst 33342 (Catalogue# H3570, Invitrogen) for 1h at RT. Following this incubation, the cells were subjected to 3 washes, each lasting 5 mins, with 0.1% PBSTx to ensure thorough cleansing. Subsequently, z-stack imaging was performed using a Zeiss LSM800 confocal microscope with Airyscan. The above process utilized three distinct dyes to specifically label DSS1 nuclear bodies (Red), promyelocytic leukemia (PML) nuclear bodies (Green), and the nucleus (Blue). The data file was named 'Sample Image Stack.tif'.

2.2 ColocZStats Development

The criteria for selecting the underlying framework, as well as an outline of the implementation details, are illustrated in this section.

2.2.1 Selection Criteria for 3D Slicer

The decision to select 3D Slicer as the development framework primarily stems from two aspects: an assessment of the visualization effects in these investigated visualization platforms and an assessment of their extensibility.

2.2.1.1 Comparison of 3D reconstructions

High-quality 3D reconstruction or rendering is an essential prerequisite for accurately performing colocalization analysis on confocal z-stacks, as it can reveal more detail in structures, enhance users' recognition of colocalization areas, and elevate colocalization analysis sensitivity. Figure 2.1 shows two channels of the same sample dataset and their combination displayed as colored semi-transparent voxels in ChimeraX, ChimeraX VR, ExMicroVR, ConfocalVR, SlicerVR, and 3D Slicer separately. All screenshots of the VR environments were captured using an HTC Vive Cosmos Elite VR System provided by Dr. Oscar Meruvia-Pastor from the Department of Computer Science, Memorial University of Newfoundland. Although 3D Slicer supports volume rendering, before the development of ColocZStats, it could not automatically split multi-channel confocal z-stack channels and color them, nor did it provide specialized

GUI widgets for separate manipulation of each channel or navigation between them. Therefore, an image processing package, FIJI [57], was initially utilized to save the channels as separate multi-page z-stack files. Subsequently, these files were selected and imported as scalar volumes into the ‘Volume Rendering’ module of 3D Slicer for rendering. Their colors were then manually configured to ensure consistency with the others.

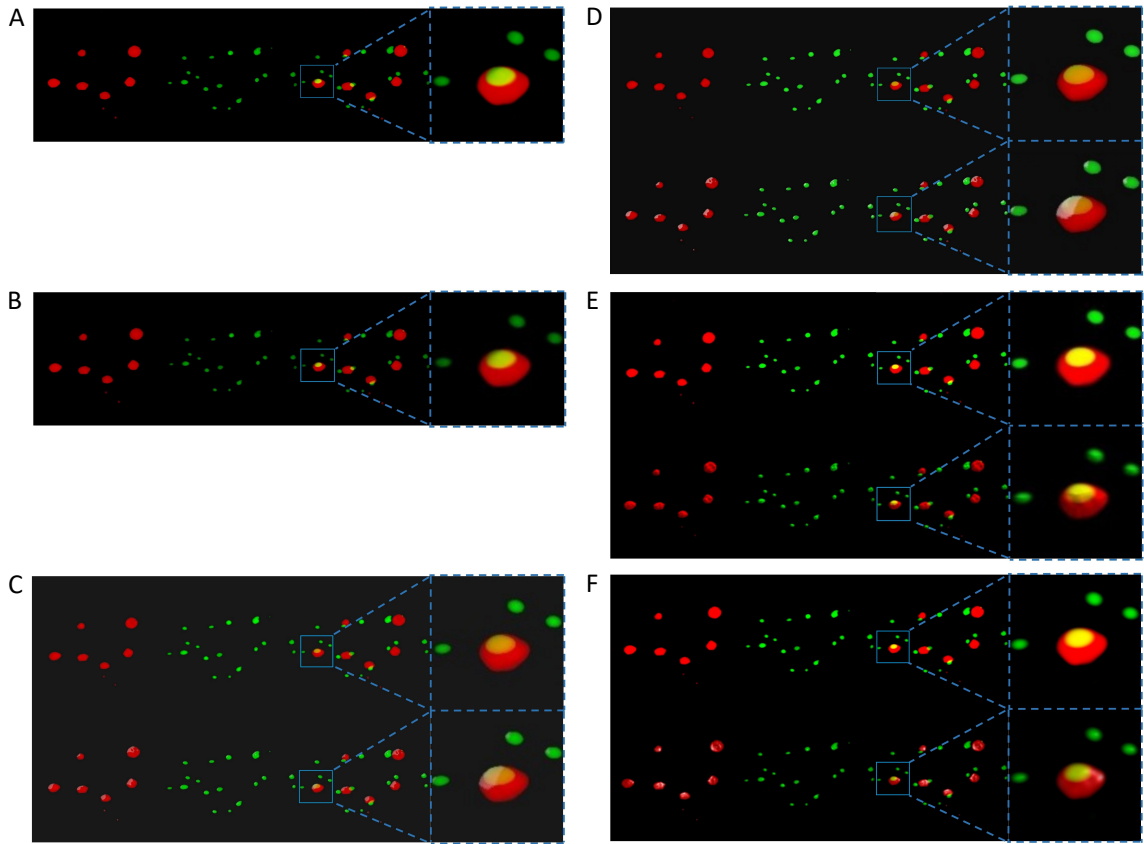


Figure 2.1: A series of screenshots displaying the visualization of the same specimen's channels from an identical perspective across all the mentioned programs. For each sub-figure, from left to right, two separate channels of the sample stack, their superposition, and a magnification of a specific region in the superposition are shown. (A) The volumetric rendering of the two channels in ChimeraX's desktop version. (B) The same series of scenes as (A). They were captured from a VR environment created by ChimeraX VR. The first rows in (C), (D), (E), and (F), respectively, show the scenes without external lighting in the ExMicroVR, ConfocalVR, SlicerVR, and the desktop viewport of 3D Slicer. The second row of (C) to (F) shows the scenes with external lighting turned on.

Currently, ChimeraX and ChimeraX VR do not support applying lighting to volumetric renderings, but only to surfaces and meshes [58, 59]. However, adjustable highlights or shadows are necessary for showing 3D semi-translucent voxels, as this can help create refined volumetric representations and allow viewers to distinguish intricate topologies and distance relationships more easily [60, 61]. Unlike them, the remaining programs in Figure 2.1 possess the capability of adding lighting and shadows to such renderings. For example, the ‘Volume Rendering’ module of 3D Slicer provides many options for calculating shading effects [62], which enables finer adjustments to the rendered volume’s appearance. Regarding the rendering effects, it was observed that the quality produced by 3D Slicer or SlicerVR is not inferior to that of other visualization software programs.

2.2.1.2 High Extensibility of 3D Slicer

More importantly, it has also been learned that researchers can efficiently develop, assess new methods, and add more capabilities through custom modules by benefiting from the highly extendable features of 3D Slicer. That is to say, in 3D Slicer, developers do not need extra time to redevelop primary data import/export, visualization, or interaction functionalities. Instead, they can easily call or integrate these characteristics and focus on developing new required features [48, 63]. Although ChimeraX is also extensible, many of its functional Application Programming Interfaces (APIs) were not documented during the tool development, and many existing APIs were

experimental. In contrast, 3D Slicer has provided relatively extensive API documentation and developer tutorials. Moreover, the 3D Slicer community [64] has offered a large and active forum for developers, where considerable issues related to the tool development can be found, discussed, or raised for timely feedback. The other two, ConfocalVR and ExMicroVR, have not provided publicly available APIs. All the above factors have constituted the rationale for developing the tool for 3D Slicer.

2.2.2 Implementation Specifics

ColocZStats is a scripted extension that utilizes the 3D Slicer APIs [65] and is implemented using Python. The code was written in PyCharm [66], an integrated development environment (IDE) designed explicitly for Python programming. The fundamental organizational structures of ColocZStats are ‘classes,’ which represent independent code blocks containing a group of functions and methods.

Once a multi-channel confocal z-stack is loaded into ColocZStats, in the program’s background, the stack’s metadata will be parsed to extract individual Numpy arrays of each channel. Simultaneously, several methods from a base module, ‘util’ [67], and a core class, ‘vtkMRMLDisplayNode’ [68], of 3D Slicer will be iteratively invoked to set up separate scalar volume nodes for each channel’s Numpy array and to assign individual pseudo colors to them. Also, the rendering method from 3D Slicer’s ‘Volume Rendering’ module [69, 70], will be applied to all individual channels separately, eventually presenting their merged visual appearance. Similar to most open-source

medical imaging systems, such as MITK [71, 72], itkSNAP [73, 74], and CustusX [75], the volume rendering back-end of 3D Slicer is based on the Visualization Toolkit (VTK) [76]. VTK is a powerful cross-platform library that supports a variety of visualization and image-processing techniques, making it widely adopted in open-source and commercial visualization software [77]. The ‘Volume Rendering’ module provides three volume rendering methods: (i) VTK CPU Ray Casting, (ii) VTK GPU Ray Casting, and (iii) VTK Multi-Volume. The ‘VTK GPU Ray Casting’ is the default method for rendering because graphics hardware can significantly accelerate rendering [69].

Moreover, at the moment a stack is loaded, a set of separate GUI elements will be created for each channel to achieve purposes such as adjusting channels’ threshold values or revealing any number of channels in the scene by controlling their visibility. The GUIs in ColocZStats are provided by the Qt toolkit [78]. Numerous modules in 3D Slicer offer a suite of reusable, modifiable GUI widgets, allowing developers to integrate them seamlessly into custom user interfaces via Qt Designer [79]. Qt Designer is a tool for crafting and constructing GUIs with Qt Widgets. For instance, in ColocZStats, the widget integrated for controlling each channel’s threshold range is the ‘qMRMLVolumeThresholdWidget’ [80], which is also a widget within 3D Slicer’s ‘Volumes’ module.

By integrating and leveraging several existing classes and methods in 3D Slicer, the initial need for a series of manual operations to create appropriate volumetric

representations for channels has evolved into the current state where all these steps can be automatically completed, with each channel equipped with individually controllable GUI widgets. The above process is also an illustrative example of how the high extensibility of 3D Slicer can be exploited to fulfill certain specific requirements of the tool.

2.3 Statistical Analysis of Colocalization

As previously mentioned in Chapter 1, ColocZStats not only helps researchers identify the relative distribution of different cellular molecules but can also generate effective metrics for quantifying colocalization to help investigate the presence of spatial associations between different channels. In the following subsections, these metrics are described in detail.

2.3.1 Pearson’s Correlation Coefficient

In the field of colocalization analysis, Intensity Correlation Coefficient-Based (ICCB) analysis methods constitute one of the primary categories of methods for assessing colocalization events. They depend on the image’s channel intensity information, which provides a powerful way to quantify the degree of spatial overlapping of two channels [81]. A large number of colocalization analysis tools employing ICCB methods have been widely integrated into various image analysis applications [6]. Pearson’s correlation coefficient (PCC) is one of the most commonly employed ICCB methods

[9]. It originated in the 19th century and has been extensively used to evaluate the linear correlation between two data sets [82]. When analyzing colocalization, PCC is employed to quantify the linear relationship between the signal intensities in one channel and the related values in another [83]. The PCC can be considered as a normalized assessment of two channels' covariance [83]. The formula for PCC applied in ColocZStats is defined below, with the signal intensities of two channels at each voxel included in the calculation:

$$PCC = \frac{\sum_{i=1}^n (ch1_i - \overline{ch1})(ch2_i - \overline{ch2})}{\sqrt{\sum_{i=1}^n (ch1_i - \overline{ch1})^2} \sqrt{\sum_{i=1}^n (ch2_i - \overline{ch2})^2}} \quad (2.1)$$

For any pair of selected channels in the tool, the $ch1_i$ and $ch2_i$ represent the intensity values of each channel at voxel i , respectively, and the $\overline{ch1}$ and $\overline{ch2}$ represent the mean intensities of each channel, respectively.

The range of PCC is between +1 and -1. +1 indicates that the two channels are entirely linearly correlated, and -1 means that the two channels are perfectly but inversely correlated. The value of zero means that the distributions of the two channels are uncorrelated [9]. Although PCC is theoretically not influenced by thresholds, they are incorporated into the tool's computation to handle specified threshold settings uniformly across all coefficients computed by ColocZStats. The specific approach of setting thresholds for calculating PCC is similar to that of the 'Colocalization Analyzer' in the Huygens Essential software [84, 85]. The Huygens Essential software [86]

is a widely acknowledged desktop software for the visualization and analysis of microscopic images [87, 88, 89]. The distinction between the two is that, in ColocZStats, not only the value of the lower threshold but also that of the upper threshold can be specified for each channel. The purpose of allowing the setting for channels' upper threshold values is analogous to the design of ConfocalVR's previous 'Cut Range Max' widget or its latest 'Max Threshold' widget, which is used to filter out over-saturated voxels that may be occasionally observed by observers in some particular circumstances [53]. During the calculation process, if a particular channel is set with lower and upper thresholds, voxel intensities greater than the upper threshold will be set to zero. Based on that, the lower threshold will be subtracted from the remaining voxel intensities. If any negative voxel values occur after the subtraction, they will be set to zero.

2.3.2 Intersection Coefficients

In distinction from the PCC, which is calculated based on actual voxel intensities, more straightforward coefficients can be calculated based only on the presence or absence of signals in a voxel, regardless of its actual intensity value. All intersection coefficients calculated by ColocZStats are examples of such coefficients, and the computation methods were borrowed from the Huygens Colocalization Analyzer [85, 90]. Similarly, the difference is that the upper threshold value for each selected channel is included in ColocZStats's calculation. To be more specific, once a voxel's intensity

value is within a specific intensity range bounded by upper and lower thresholds, it can be regarded as having some meaningful signal. If so, its value could be considered as 1, regardless of its actual intensity, and 0 otherwise. This indicates that a binary image $ch1_{weight}$ with intensity $ch1_{weight,i}$ at voxel i can be created based on the voxel's actual intensity $ch1_i$ and the channel's intensity range. This explanation is illustrated by taking the first channel, $ch1$, of all selected channels as an example (the same applies to all the other channels):

$$ch1_{weight,i} = \begin{cases} 0 & \text{if } ch1_i \leq ch1_{lower} \text{ or } ch1_i > ch1_{upper} \\ 1 & \text{if } ch1_{lower} < ch1_i \leq ch1_{upper} \end{cases} \quad (2.2)$$

Based on the definition of $ch1_{weight,i}$, another metric called the global intersection coefficient (I) can be calculated. It is defined as the ratio of the total volume of voxels where all channels intersect to the total volume of all channels. In other words, it calculates the proportion of voxels having valid intensity values in all thresholded channels. Multiplying this figure by 100 is interpreted as the intersection's volume percentage. For any two specified channels in ColocZStats, the formula of I is defined as follows:

$$I = \frac{\sum_{i=1}^n (ch1_{weight,i} ch2_{weight,i})}{\sum_{i=1}^n ch1_{weight,i} + \sum_{i=1}^n ch2_{weight,i} - \sum_{i=1}^n (ch1_{weight,i} ch2_{weight,i})} \quad (2.3)$$

The numerator refers to the intersecting voxels' total volume. For each voxel, the overlapping contribution is defined as the product of $ch1_{weight,i}$ and $ch2_{weight,i}$. The denominator represents the total volume of the two channels, which is defined as the sum of the volume of the first channel and the volume of the second channel minus their intersection's total volume (to prevent double counting). Another two individual intersection coefficients are derived from I , which describes what proportion of the first and second channels are intersecting:

$$i_1 = \frac{\sum_{i=1}^n (ch1_{weight,i} ch2_{weight,i})}{\sum_{i=1}^n ch1_{weight,i}} \quad (2.4)$$

$$i_2 = \frac{\sum_{i=1}^n (ch1_{weight,i} ch2_{weight,i})}{\sum_{i=1}^n ch2_{weight,i}} \quad (2.5)$$

Another contribution of ColocZStats is that it allows researchers to choose up to three channels in a z-stack for statistical analysis. Extending from the above-mentioned formulas, the intersection coefficients' formulas tailored to this scenario have also been proposed. In this case, the formula for I is given as:

$$I = \frac{\sum_{i=1}^n (ch1_{weight,i} ch2_{weight,i} ch3_{weight,i})}{\sum_{i=1}^n ch1_{weight,i} + \sum_{i=1}^n ch2_{weight,i} + \sum_{i=1}^n ch3_{weight,i} - \sum_{i=1}^n (ch1_{weight,i} ch2_{weight,i}) - \sum_{i=1}^n (ch1_{weight,i} ch3_{weight,i}) - \sum_{i=1}^n (ch2_{weight,i} ch3_{weight,i}) + \sum_{i=1}^n (ch1_{weight,i} ch2_{weight,i} ch3_{weight,i})} \quad (2.6)$$

Likewise, the three channels' intersecting volume of a given voxel i is defined as the product of $ch1_{weight,i}$, $ch2_{weight,i}$, $ch3_{weight,i}$, while the sum acts as the numerator. The

denominator is the total volume of the three channels, determined by the inclusion-exclusion principle for three sets [91]. Also, three respective intersection coefficients can be obtained to exhibit the proportion of the intersection in each channel:

$$i_1 = \frac{\sum_{i=1}^n (ch1_{weight,i} ch2_{weight,i} ch3_{weight,i})}{\sum_{i=1}^n ch1_{weight,i}} \quad (2.7)$$

$$i_2 = \frac{\sum_{i=1}^n (ch1_{weight,i} ch2_{weight,i} ch3_{weight,i})}{\sum_{i=1}^n ch2_{weight,i}} \quad (2.8)$$

$$i_3 = \frac{\sum_{i=1}^n (ch1_{weight,i} ch2_{weight,i} ch3_{weight,i})}{\sum_{i=1}^n ch3_{weight,i}} \quad (2.9)$$

Chapter 3

Results

This chapter starts with a description of how ColocZStats is made available to the public and the requirements for input images. Subsequently, the user interface and workflow of ColocZStats are illustrated. Following that, the graphical representations and results spreadsheet generated by ColocZStats are demonstrated. Furthermore, the usage of ColocZStats for the colocalization analysis process is exemplified through a sample case study. Finally, a comprehensive comparison of the features of ColocZStats and the other discussed visualization applications is conducted.

3.1 Availability of ColocZStats

ColocZStats is freely available under an MIT license [92] and requires the stable version of 3D Slicer [93] for compatibility. The Slicer community maintains a website called the ‘Slicer Extensions Catalog’ for finding and downloading extensions, and

ColocZStats has been distributed on it [94]. The ‘Extensions Manager’ in 3D Slicer provides direct access to the website, facilitating easy installation, updating, or uninstallation of extensions with a few clicks in the application [95]. As shown in Figure 3.1, ColocZStats can be found within the ‘Quantification’ category in the catalog. After installing the ColocZStats, it will be presented to users as a built-in extension. More information about installing and using ColocZStats can be found on the homepage of its repository on GitHub [96].

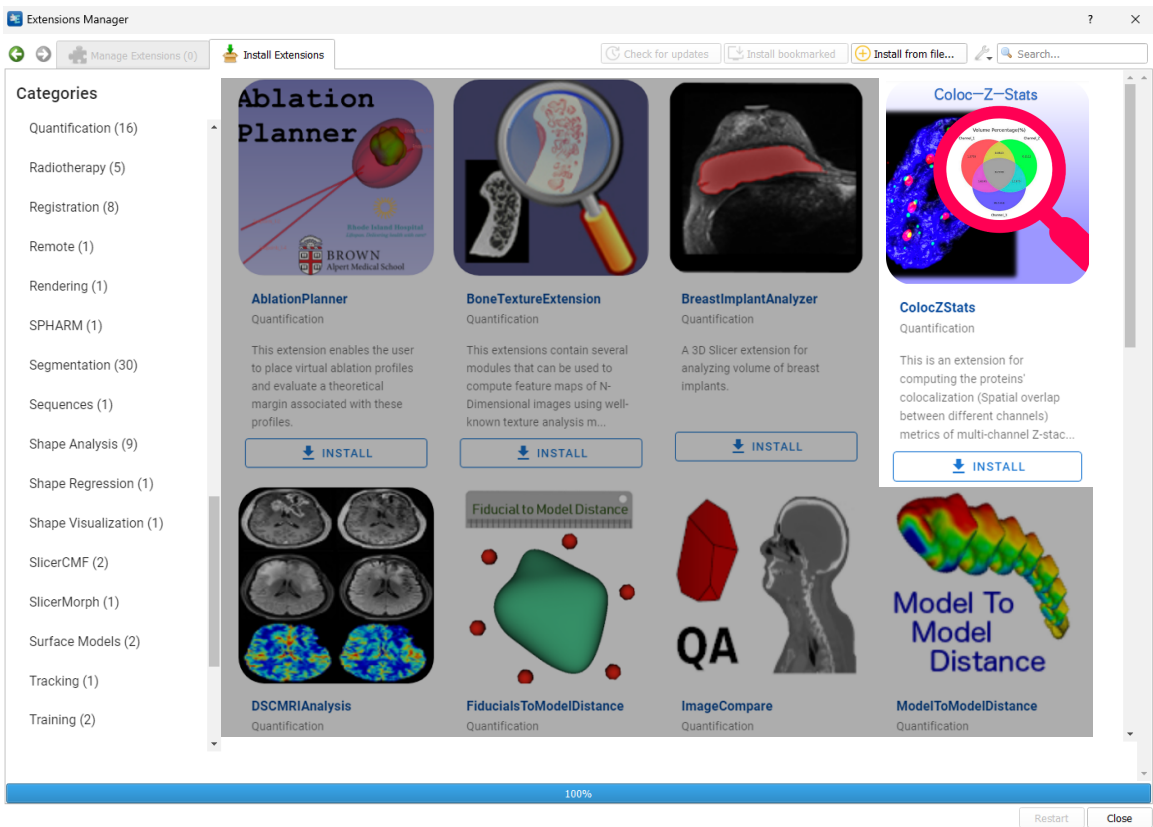


Figure 3.1: A screenshot showing the ‘Slicer Extensions Catalog’ via the ‘Extensions Manager’ in 3D Slicer. ColocZStats is offered within the ‘Quantification’ category.

3.2 Input Image File

For compatibility with ColocZStats, each input image file must be a 3D multi-channel confocal image z-stack in TIFF format that maintains the original intensity values, with each channel in grayscale. Each channel must possess the same dimensions, image order, and magnification. Even though ColocZStats supports loading z-stacks containing up to fifteen channels, only up to three channels can be specified for each colocalization computation because the statistical analysis and the possible interactions between channels become more complex with each additional channel.

3.3 User Interface

An example of ColocZStats's user interface is shown in Figure 3.2, which consists of two separate areas: a control panel on the left side that provides a series of interactive widgets, and a 3D view on the right side that displays the volumetric rendering of the loaded sample image stack. Users can adjust these volume-rendered images and calculate the corresponding colocalization metrics by manipulating the widgets on the control panel. The details of the widgets shown in Figure 3.2 are depicted below: (a) A clickable eye icon for managing the visibility of the entire image stack's rendering. (b) A clickable eye icon for managing the visibility of the adjustable box for selecting ROI regions. (c) A combo box for switching between multiple loaded image stacks, displaying the stack's filename without extensions by default. (d) A

button for repositioning the rendering within the ROI box to the 3D view's center. (e) A button for triggering a pop-up text box for customizing the displayed name on the combo box. (f) A button for deleting the current stack from the scene, along with its associated annotation and GUI widgets. (g) A button for performing colocalization analysis. (h) A text field for adding a customized annotation for the current stack. (i) A checkbox for managing the visibility of each channel. The indices in these default channel name labels start from 1. (j) A button for triggering a pop-up text box for customizing the corresponding channel's name label. (k) A drop-down list for selecting the channel threshold control mode. It comprises three options: 'Auto,' 'Manual,' and 'Off'. (l) Adjustable sliders for setting lower and upper thresholds. Both values can also be specified in the two input fields. (m) A drop-down box for displaying the initial threshold boundaries of the associated channel. The following section describes a typical workflow for ColocZStats.

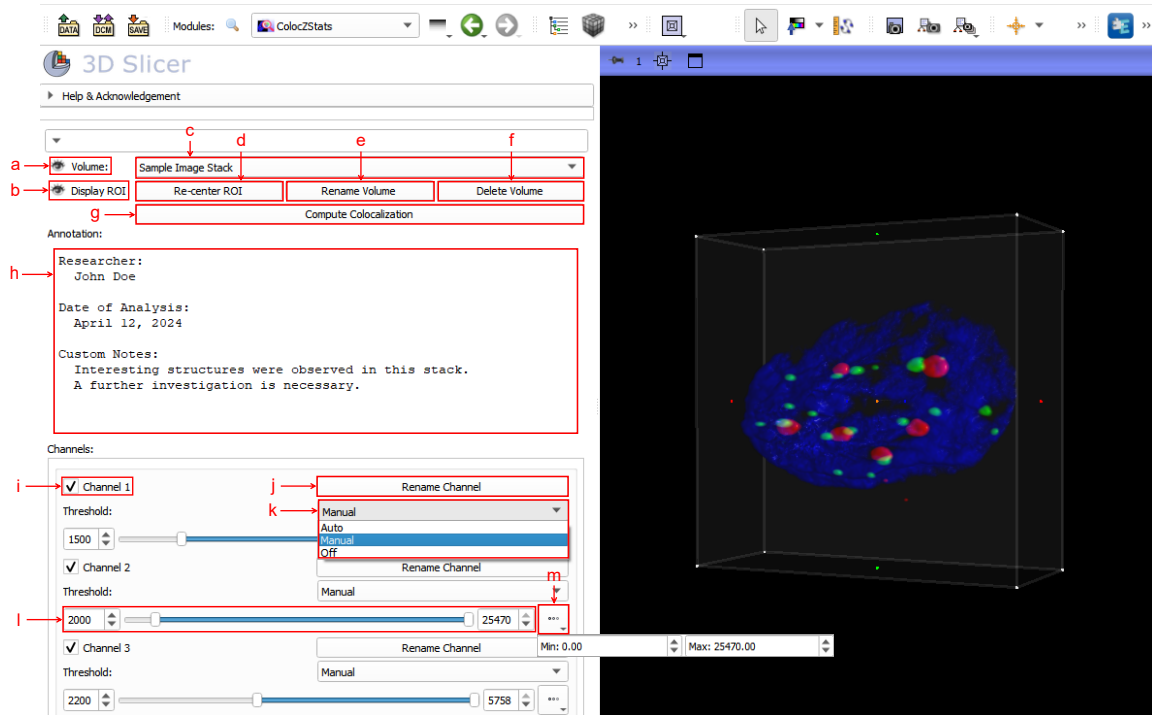


Figure 3.2: The Graphical User Interface (GUI) of ColocZStats.

3.4 The Workflow of ColocZStats

A typical workflow of ColocZStats is shown in Figure 3.3(A), which is divided into five steps: (i) ‘Inputs’; (ii) ‘Visualization’; (iii) ‘Channel Selection & Thresholding’; (iv) ‘ROI Selection’; and (v) ‘Analysis.’ The sequence of certain steps in this workflow can be slightly adjusted based on particular conditions or personal preferences.

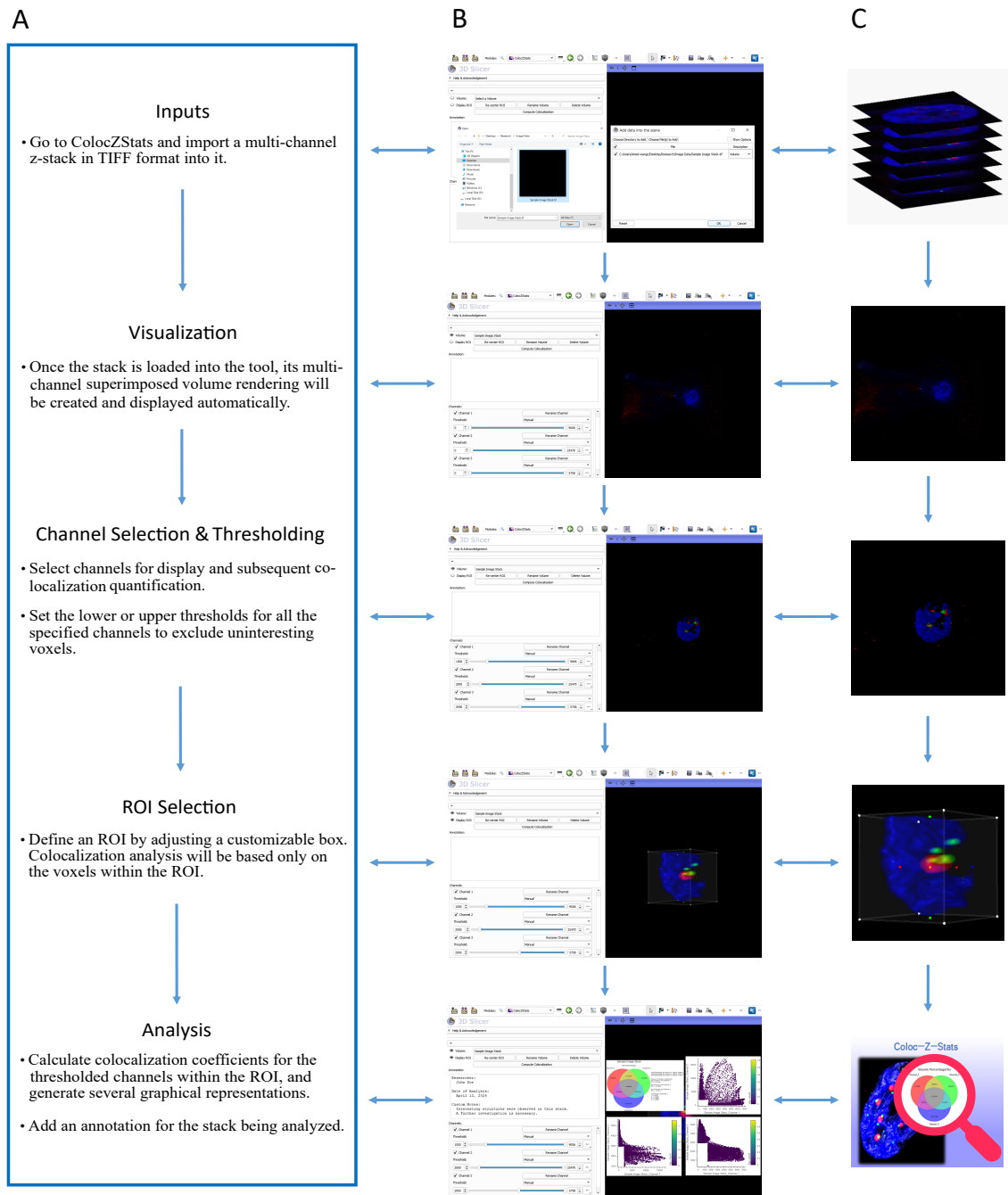


Figure 3.3: (A) Typical workflow of ColocZStats. (B) Example operation scenarios in ColocZStats that correspond to the steps in the workflow. (C) Visual abstractions that correspond to the steps in the workflow.

In 3D Slicer, a file browser can be opened by clicking the ‘DATA’ button at the upper-left corner to load a multi-channel confocal z-stack file into the scene. All channels’ colored volume-rendered images will immediately appear in the 3D view by default when the stack is loaded into ColocZStats, and they can be moved and rotated arbitrarily with mouse clicks and movements.

The widgets in the ‘Channels’ sub-panel will also be displayed concurrently upon loading the input file. The checkbox in front of each channel’s name label is not only used for managing the visibility of the channel’s rendered volume but also for determining whether the channel will be included in the colocalization analysis. Figure 3.3(B) is an example where all three channels of the sample data are selected. After the channel selection, threshold segmentation is often necessary because it is a helpful approach for extracting interesting voxels from the image. For individual channels, as any slider for controlling the thresholds is adjusted, the display range of the channel’s volumetric representation will be changed synchronously, facilitating users’ observation and the decision-making for the following analysis. By default, when a confocal z-stack is imported, the checkboxes of all its channels are turned on, and the threshold ranges for all channels are displayed as their original ranges, respectively.

The ROI box integrated into ColocZStats facilitates researchers’ analysis of any parts of the loaded stack. By default, there is no ROI box for the imported stack in the scene. Upon the first click on the eye icon beside the ‘Display ROI’, an ROI box

will be created, and after that, its visibility can be customized to be toggled on or off. While the ROI box is displayed, depth peeling will automatically be applied to the volumetric rendering to produce a better translucent appearance. The handle points on the ROI box can be dragged to crop the rendered visualization with six planes. As any handle point is dragged, the volumetric rendering outside the box will disappear synchronously. The final analysis will only include the voxels inside the ROI box. When there is a need to analyze the entire stack, the ROI box should be adjusted to enclose it completely. The functionality to extract the voxels inside the ROI is enabled by the ‘Crop Volume’ module [97], and the related methods will be called in the backend when a calculation is executed. The ‘Crop Volume’ module is a built-in loadable module of 3D Slicer that allows the extraction of a rectangular sub-volume from a scalar volume. With the above control options, users can conveniently configure voxel intensity thresholds and ROI for all channels to be analyzed while intuitively focusing on the critical structures to obtain the final colocalization metrics.

The ‘Analysis’ step is performed when clicking the ‘Compute Colocalization’ button; all the thresholded channels and overlapping areas inside the ROI will be identified in the program’s background process. Subsequently, the colocalization coefficients described previously will be computed. Meanwhile, a series of graphical representations will pop up on the screen, such as a Venn diagram illustrating volume percentages and 2D histograms showing the combinations of intensities for all possible pairs of selected channels. In addition, users can add a custom annotation for the associated

image stack at any step after it is loaded, to record any necessary information.

3.5 Produced Graphical Representations

The following subsections provide detailed descriptions of the graphical representations obtained from the ‘Analysis’ step in the workflow.

3.5.1 Venn Diagrams

ColocZStats allows two or three channels to be selected simultaneously to perform colocalization measurements. The Venn diagrams generated from the above two scenarios allow researchers to quickly recognize the volume percentages of overlapping regions along with that of the remaining parts. A popular Python package, ‘matplotlib-venn’ [98], was utilized to implement this functionality while leveraging its ‘venn2_unweighted’ and ‘venn3_unweighted’ functions for the creation of the two kinds of Venn diagrams consisting of two or three circles without area-weighting, respectively.

In the Venn diagrams generated, the colors in these circles match the colors in the volume-rendered images, and the area where all circles intersect signifies the part where all specified channels intersect, with the displayed percentage corresponding to the result obtained by multiplying the global intersection coefficient by 100 and retaining four decimal places. Based on this, the percentages of the remaining parts of these channels can be derived and displayed in the other areas of the Venn diagrams.

Besides, the Venn diagram's title matches the specified name on the GUI's combo box, and the channel names shown correspond to the channel name labels defined on the interface.

The examples shown in Figure 3.4 demonstrate a typical scenario of setting lower thresholds for each channel and configuring their upper thresholds to their maximum values. Figures 3.4(E) and (F) are also components of the result illustrations composed together with the two Venn diagrams, respectively, and for clarity, they were extracted as two separate figures.

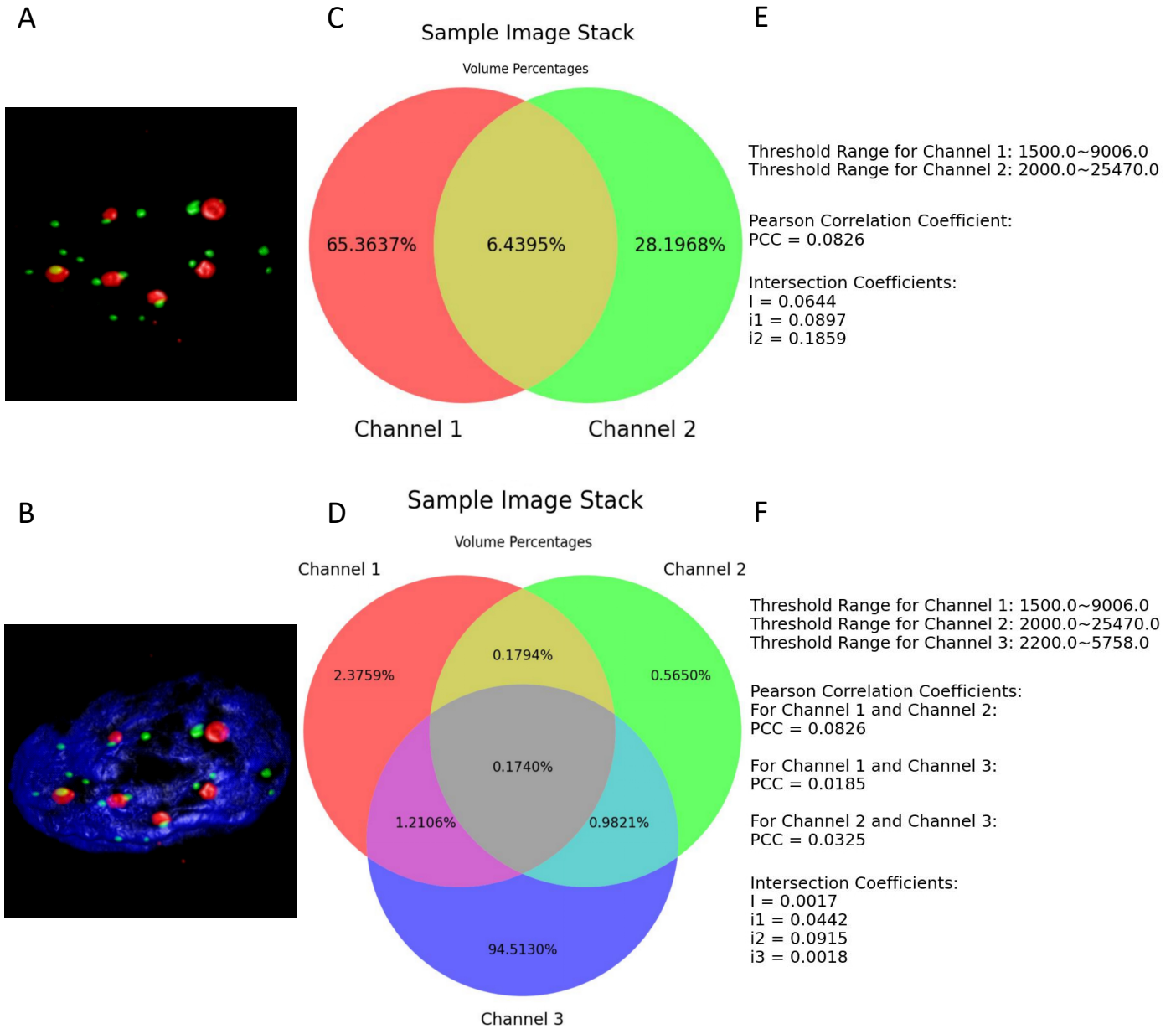


Figure 3.4: (A) and (B) present the thresholded volume renderings when two or three channels of the entire sample image stack were selected. (C) and (D) depict the respective resulting Venn diagrams. (E) and (F) illustrate all the defined thresholds and the colocalization metrics results retained to four decimal places.

3.5.2 2D Histograms

The 2D histograms produced by ColocZStas serve as a supplementary tool for visually assessing colocalization, providing a qualitative indication. The feature of generating 2D histograms has become a basic functionality of most colocalization analysis software. One specific application of 2D histograms is that they can be employed to identify populations within different compartments [99, 100]. For any two channels, a 2D histogram illustrates the connection of intensities between them, where the x-axis corresponds to the intensities of the first channel, and the y-axis corresponds to the intensities of the second channel [101]. The histogram's points can be observed closely gathering along a straight line if the two channels are highly correlated. The line's slope indicates the two channels' fluorescence ratio [9]. Following the method introduced in Caltech's 'Introduction to Data Analysis in the Biological Sciences' course in 2019 [102], in ColocZStats, a Python library called 'Holoviews' [103], was applied to plot the 2D histograms. With the scenario corresponding to Figure 3.4(A), Figure 3.5 demonstrates the generation of a 2D histogram.

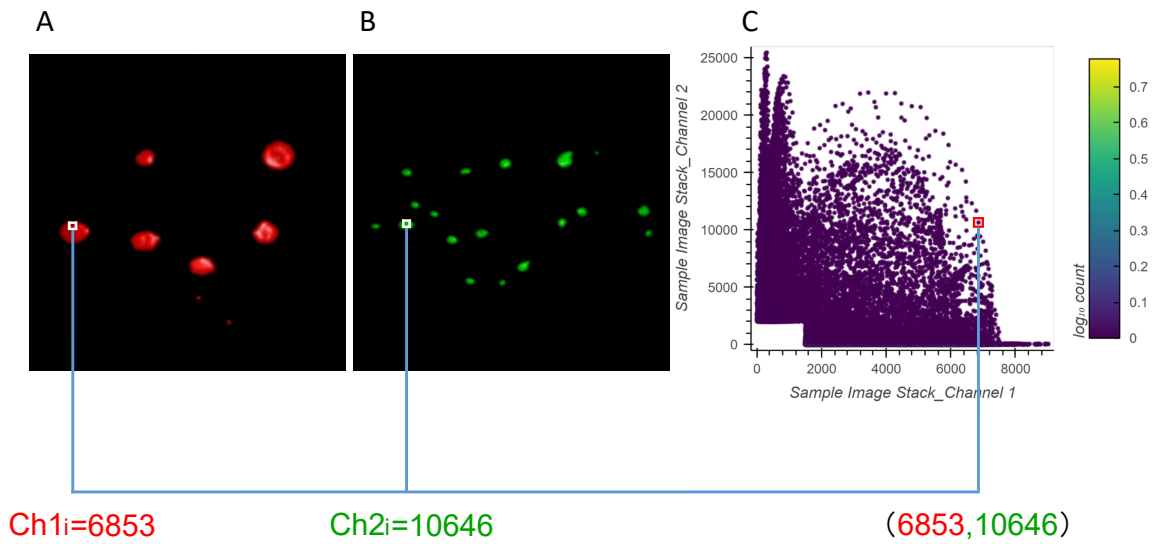


Figure 3.5: A visual explanation of a 2D histogram’s generation process. Adapted from ‘Scientific Volume Imaging-2D histograms,’ 2024, Scientific Volume Imaging B.V., <https://svi.nl/TwoChannelHistogram>, Copyright 1995-2024 by Scientific Volume Imaging B.V. (A) The volume rendering of channel 1 and an example intensity at a specific position. (B) The volume rendering of channel 2 and an example intensity at the same position. (C) The coordinates in the 2D histogram are generated by the combination of those intensities.

For any precise position within the channels, the related intensities of both channels are combined to define a coordinate in the 2D histogram. Simultaneously, the count of points at this coordinate is incremented by one [104]. As shown in Figure 3.5(C), the blank area in the lower-left corner represents the background. For any background’s voxel, the intensity values of its two channels are both outside the re-

spective valid channel threshold ranges, so such voxels will not be plotted as points in the 2D histogram. The definition of the valid threshold ranges mentioned in this context is consistent with those applied when calculating the intersection coefficients. The color bar on the histogram's right side indicates the number of points with the same intensity combinations. The 2D histograms generated by ColocZStats will be saved as static images and interactive HTML files that can be viewed in more detail.

The correspondence between the example 2D histogram and its related Venn diagram is depicted in Figure 3.6.

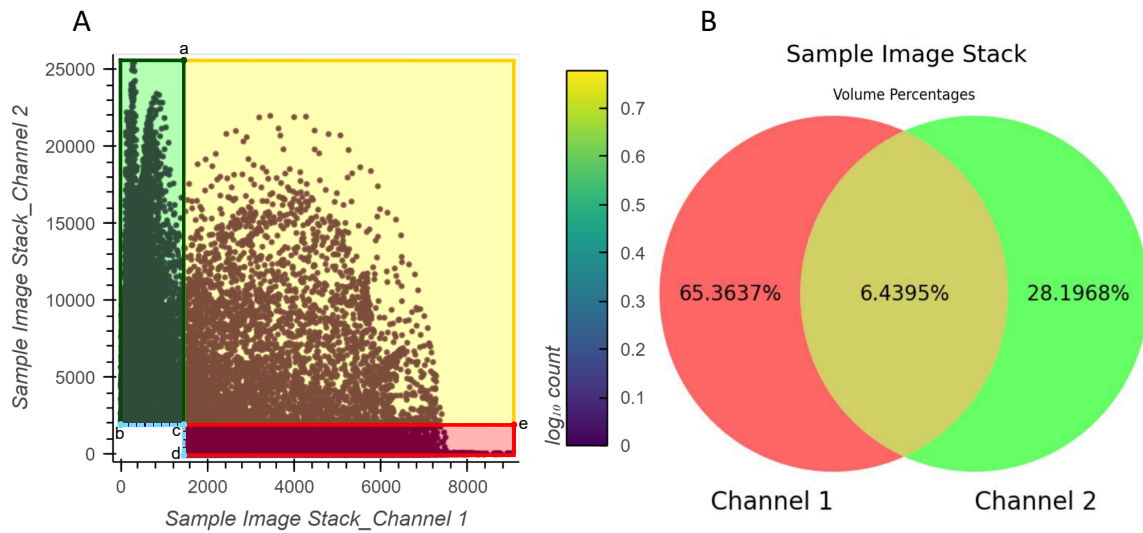


Figure 3.6: An illustration depicting the relationship between the example 2D histogram and its corresponding Venn diagram. Coordinate of point (a): (1500,25470); Coordinate of point (b): (0,2000); Coordinate of point (c): (1500,2000); Coordinate of point (d): (1500,0); Coordinate of point (e): (9006,2000).

The defined threshold value boundaries for the channels delineate four regions in Figure 3.6(A). In this 2D histogram, three colored regions, and their outlines aligned with the boundaries, are used to highlight all possible distribution positions of points.

All points within the yellow region and along its yellow outlines represent all the voxels that contain valid signals in both channels, and the percentage of all these points aligns with the percentage displayed in the yellow area of Figure 3.6(B). Likewise, the points distributed in the red or green parts represent the voxels with exclusively valid signals in channel 1 or channel 2, respectively, and the percentage of points in the two regions are consistent with the values shown in the Venn diagram's red and green areas, respectively. Also, no points are plotted on coordinates corresponding to 'b,' 'c,' and 'd' or the boundaries formed between them, as they lie outside the defined threshold ranges.

3.6 Results Spreadsheet

After clicking the 'Compute Colocalization' button, a comprehensive results spreadsheet will be automatically saved for researchers' further reference or sharing. In Figure 3.7, an example is presented, showcasing all sub-sheets of the spreadsheet.



Figure 3.7: All of the sub-sheets in each spreadsheet result from each single computation. (A) The selected channels and their respective threshold ranges. (B) The timestamp of this computation. (C) The custom annotation for the specified image stack. (D) The related information of the defined ROI for this computation. (E) The PCCs for all possible pairwise combinations of the selected channels. (F) All the resulting intersection coefficients. (G) All the resulting 2D histograms, with each histogram illustrating a pair of channels. (H) The resulting illustration embeds the Venn diagram and all coefficient results. 45

Among them, Figure 3.7(D) displays some crucial information related to the defined ROI. The information is automatically and selectively extracted from a JSON file containing all details of the ROI node, and the JSON file is for reloading the ROI box [105]. By invoking the 3D Slicer’s built-in exporting method for such JSON files [106] in ColocZStats’ background process, the JSON file will also be automatically saved after each calculation. Notably, the JSON file can also be saved as one of the batch files produced by clicking the ‘SAVE’ button of 3D Slicer. All these files are referred to by an MRML file created synchronously to reload the operation scenario. Various elements for each imported stack, including volume-rendered image, ROI box, annotation, and all the other widgets, can be reloaded from this MRML file [107].

3.7 Case Study

This section provides an example demonstrating a specific scenario of applying ColocZStats to perform a colocalization analysis task. It primarily focuses on how changes in the threshold range of an individual channel influence the objective quantitative colocalization indicators and the variations in colocalization degree that can be revealed during this process. The ROI box remains unchanged throughout the four cases shown in Figure 3.8, and the red channel is individually assigned four distinct lower threshold values. In contrast, the threshold ranges for the other channels remain constant. By observing the 3D rendering appearance of the data provided by this tool, a continuous change can be found; that is, as the red channel’s lower

threshold gradually increases, the volume of its overlap with other channels shows an evident decreasing trend. Combining a series of objective quantifications generated by ColocZStats for each case aids in validating this subjective visual impression and obtaining a more reliable assessment.

Through Figure 3.8(C), a noticeable phenomenon regarding the variation of colocalization coefficients is exposed: as the lower threshold value of channel 1 increases, the PCCs between channel 1 and the other two channels gradually approach 0, indicating a diminishing linear correlation between channel 1 and the other two channels. Meanwhile, as demonstrated by the additional histograms created based on the Venn diagrams, with the reduction in the threshold range of channel 1, the proportions of overlapping regions between channel 1 and the other two channels also decrease. Integrating the above observations effectively validates a biological fact: the reduction of channel 1 leads to a decline in its colocalization with the other two channels. In addition, the values of the global intersection coefficient and its three derived coefficients also gradually decrease, implying a decline in the overall colocalization degree of the three channels. This example also exhibits how the visual representations and coefficients produced by this tool can provide multiple perspectives for assisting colocalization analysis.

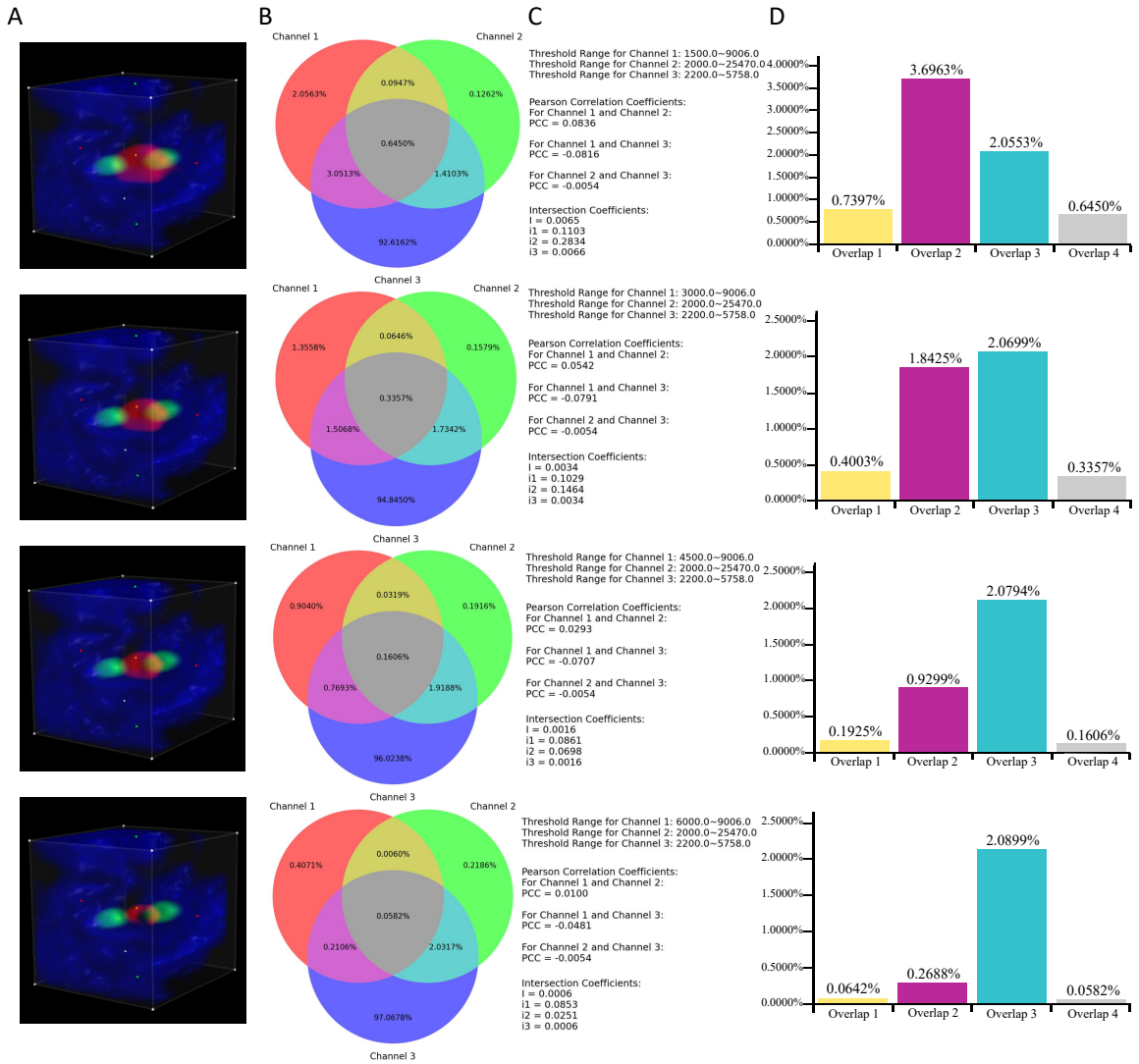


Figure 3.8: (A) Volume rendering of all channels within the same ROI box. (B) Venn diagrams corresponding to all the cases. (C) All the associated thresholds and colocalization metrics' results. (D) Additional histograms illustrating the proportions of overlaps. They were created based on the data derived directly from the Venn diagrams. 'Overlap 1': The overlap between Channel 1 and Channel 2. 'Overlap 2': The overlap between Channel 1 and Channel 3. 'Overlap 3': The overlap between Channel 2 and Channel 3. 'Overlap 4': The overlap of all three channels.

3.8 Comprehensive Comparison of Visualization Tools

The related features of the programs mentioned in the ‘Pertinent Visualization Tools’ subsection, including ColocZStats, for visualizing confocal z-stacks and measuring their colocalization are summarized in Table 3.1. This table divides these features into four categories, encompassing the most meaningful comparable features for the execution of colocalization analysis. Consequently, although certain programs may have many other functionalities, they are not included in this table. Regarding the choice of programs, ChimeraX VR and SlicerVR are VR extensions of ChimeraX and 3D Slicer correspondingly, and their image processing and analysis capabilities almost entirely depend on the specifics of each platform. As a result, only ChimeraX and 3D Slicer are listed in this table for comparison. For each feature in the table, if the programs themselves or any plugins or modules they incorporate offer matching functionality, the corresponding cell is marked with a check. For instance, in the case of 3D Slicer, its ‘Volume Rendering’ module supports loading multi-page Z-stack files; thus, the corresponding cell is checked. Another module in 3D Slicer, ‘ImageStacks,’ allows loading the image sequence of z-stack slices, resulting in the respective cell being checked as well. It is worth noting that because the ‘ImageStacks’ module is designed specifically for working with image stacks, all the listed image formats in the ‘Supported image format’ for 3D Slicer are consistent with those supported by the ‘ImageStacks’ module.

The category ‘Confocal Image Z-stack Observation’ summarizes the general prac-

tical features for meticulously observing the channels in confocal z-stacks. Most of them are common to several tools in this comparison. Among these, as for ‘ExMicroVR’ or ‘ConfocalVR’, if one intends to manipulate individual channels separately, a necessary step is to extract each channel of the original z-stack as a single multi-page z-stack file using FIJI and save all of them into a designated directory for loading. Therefore, while both tools support users in manipulating channels individually, they do not have the functionality to automatically separate all channels of the z-stack to enable this operation. Before the completion of ColocZStats development, as indicated by the information from the 3D Slicer community [108], 3D Slicer still needed specialized tools for properly visualizing multi-channel confocal z-stacks and manipulating channels separately. Therefore, in the table, no features related to operating z-stack channels corresponding to 3D Slicer have been checked. On a side note, in the context of the third and seventh items within this category, ‘Delete’ refers to removing channels’ rendering and matching GUI widgets from the scene rather than deleting any associated files from the file system.

From the information presented in Table 3.1, it is indicated that in contrast to other platforms, ColocZStats not only retains several necessary control options for channels of confocal z-stacks, but can also objectively analyze the colocalization in such stacks. Significantly, this comparison also reveals some limitations in the current functionalities of ColocZStats, such as the need for improvements to make it compatible with immersive VR environments. The current prominent limitations of

ColocZStats and potential directions for future enhancements are discussed in detail in the ‘Future Work’ Chapter.

Features	Programs	ExMicroVR	ConfocalVR	ChimeraX	3D Slicer	ColocZStats
Confocal Image Z-stack Loading						
Load multi-page (multi-image) z-stack files		✓	✓	✓	✓	✓
Load image sequence of z-stack slices				✓	✓	
Supported image format		NIFTI	NIFTI	TIFF/PNG/PGM	TIFF/PNG/JPG/BMP	TIFF
Confocal Image Z-stack Observation						
Automatically split all channels of each z-stack for independent manipulation				✓		✓
Switch among multiple multi-channel z-stacks				✓	✓	✓
Delete all channels of any z-stack simultaneously				✓		✓
Turn on/off individual channel’s visibility		✓	✓	✓		✓
Adjust lower threshold for each channel		✓	✓	✓		✓
Adjust upper threshold for each channel		✓	✓	✓		✓
Delete each channel individually			✓	✓		
Define ROI		✓	✓	✓	✓	✓
Center ROI in the visual field		✓	✓	✓		✓
Extract overlapping voxels of any 2 channels			✓			
Image Measurement and Analysis						
Quantify colocalization of confocal z-stacks						✓
Save quantified image analysis results			✓	✓	✓	✓
Create a 2D histogram for each channel pair						✓
Edit and save annotations				✓	✓	✓
Operating Environment						
Desktop				✓	✓	✓
VR (HMD)		✓	✓	✓	✓	
Multi-user VR session		✓	✓	✓	✓	

Table 3.1: Programs Features Overview.

Chapter 4

Conclusions

Through a comprehensive examination of several widely recognized biological image visualization applications, all with VR functionalities and some of which already own extensive analytical features, it has been identified that these tools lack dedicated built-in options to perform the quantitative colocalization analysis for multi-channel z-stacks generated by confocal microscopes, hindering the integration of this crucial analytical method with VR technology. The development of an open-source 3D Slicer extension named ColocZStats has contributed to expanding potential solutions for addressing this challenge. Certain functionalities from multiple modules within 3D Slicer were reasonably utilized and integrated by ColocZStats, enabling the effective presentation of confocal microscopy images' merged multi-channel volumetric appearances. This endeavor aims to support researchers in quickly discerning spatial relationships between molecular structures within organisms. On top of that,

ColocZStats can generate colocalization metrics for thresholded channels within ROIs of samples, which has positive significance for biologists to gain detailed and objective insights into biological processes.

More specifically, ColocZStats allows users to select up to three channels of each Z-stack concurrently for analysis. It permits customized control over channels' lower or upper threshold limits according to researchers' specific needs and obtaining metrics such as PCCs for all possible channel pairwise combinations and intersection coefficients. The above distinctive characteristics distinguish ColocZStats from most tools, which typically restrict users from performing colocalization analysis between two channels at a time and only allow setting the lower threshold for each channel. Via the Venn diagram it generates, the ratios of all parts, including the part where all channels intersect, can be clearly displayed. Moreover, users can further enhance their understanding of the intensity relationship between different channels through the generated 2D histograms. For each calculation, all the results and diagrams are conveyed to researchers through a supplementary spreadsheet, making it easy for them to share or compare data. In summary, with the ultimate goal of merging with SlicerVR, ColocZStats is presently functioning as a desktop extension for 3D Slicer, incorporating an intuitive GUI that allows users to customize ROIs and define the threshold ranges for all stack channels while supporting the one-click generation and saving of colocalization analysis results. More importantly, the development of the ColocZStats extension has further enhanced the comprehensiveness of 3D Slicer,

which means that the extensive audience of 3D Slicer can now seamlessly perform colocalization analysis for confocal stacks without frequently switching between different tools.

Chapter 5

Future Work

At present, as a purpose-built tool for colocalization analysis, ColocZStats requires prioritized improvements in the following aspects. Firstly, there is a need to expand the variety of colocalization metrics. In addition to the existing PCC, other coefficients belonging to the ICCB methodology that are also extensively employed, such as Manders' coefficients, Spearman's coefficient, and the overlap coefficient, could be integrated into the extension to further enhance its analytical capabilities. As more colocalization coefficients become incorporated into ColocZStats in the future, enhancing the capability of image pre-processing becomes increasingly essential. The need for this enhancement arises from the potential impact of excessive noise in microscopic images, which affects the correlation between distinct signals and leads to an underestimation of colocalization analysis results [109]. Deconvolution is a well-established method for image filtering and restoration [110]. Integrating this

technique into the tool could significantly help eliminate image noise, improve image quality [111], and consequently improve the accuracy of subsequent analysis.

Meanwhile, to further increase the coverage of ColocZStats for a wider range of input images and the efficiency of analyzing multi-channel image stacks, it is anticipated that features allowing compatibility with stacks containing more channels and permitting the simultaneous selection of more channels in a single computation will be implemented.

Moreover, as elucidated in the aforementioned ‘User Interface’ section, ColocZStats supports changing channels’ thresholds by manually adjusting the sliders or entering values in the input fields to help biologists customize structures of interest. Nonetheless, relying on visual inspection of images to estimate suitable thresholds can be challenging and may result in inconsistent outcomes. Figure 3.2(k), as one of the components of the ‘qMRMLVolumeThresholdWidget’ provided by 3D Slicer, although it includes an ‘Auto’ option that can be used to assign a threshold range for any channel automatically, incorporating more reliable, robust, and objective automated threshold methods will undoubtedly further enhance the functionality of ColocZStats. The method developed by Costes et al. [112] to determine the appropriate threshold value for background identification is expected to be appropriately integrated into ColocZStats. The Costes method is founded on the linear fitting of channels’ 2D histograms [112, 113] and has been proven to be a robust and reproducible approach that can be readily automated [9].

Enabling users to utilize ColocZStats within an immersive VR environment is a primary objective for future work. At the time of writing, integrating arbitrary interactive Qt widgets into the VR environment is an ongoing development effort by the SlicerVR development team. Once this feature is fully implemented, this would imply that any functionality in 3D Slicer and its extensions could be easily accessed in VR through these virtual widgets [52, 114]. This work illuminates the significant potential for extending ColocZStats into the immersive scene offered by Slicer VR, which will provide valuable assets and support for this forthcoming endeavor. Next, ColocZStats will be continually enhanced in various aspects, aiming to make it a more efficient and flexible software tool.

Bibliography

- [1] Confocal Techniques. <https://imb.uq.edu.au/research/facilities/microscopy/training-manuals/microscopy-online-resources/image-capture/confocal-techniques>. Accessed: 15-April-2024.
- [2] Andres Collazo, Olivier Bricaud, and Kalpana Desai. Use of confocal microscopy in comparative studies of vertebrate morphology. In *Methods in enzymology*, volume 395, pages 521-543. Elsevier, 2005.
- [3] Rensu P Theart, Ben Loos, and Thomas R Niesler. Virtual reality assisted microscopy data visualization and colocalization analysis. *BMC bioinformatics*, 18(2): 1-16, 2017.
- [4] Confocal microscopes. <https://www.britannica.com/technology/microscope/Confocal-microscopes>. Accessed: 23-February-2024.
- [5] Shanica N Pompey, Peter Michaely, and Katherine Luby-Phelps. Quantitative fluorescence co-localization to study protein-receptor complexes. In *Protein-Ligand Interactions: Methods and Applications*, Springer, pages 439-453, 2013.

- [6] Susanne Bolte and Fabrice P Cordelières. A guided tour into subcellular colocalization analysis in light microscopy. *Journal of microscopy*, 224(3): 213-232, 2006.
- [7] Thilo D Lacoste, Xavier Michalet, Fabien Pinaud, Daniel S Chemla, A Paul Alivisatos, and Shimon Weiss. Ultrahigh-resolution multicolor colocalization of single fluorescent probes. *Proceedings of the National Academy of Sciences*, 97(17): 9461–9466, 2000.
- [8] Jeremy Adler and Ingela Parmryd. Colocalization analysis in fluorescence microscopy. *Cell Imaging Techniques: Methods and Protocols*, 97–109, 2013.
- [9] Kenneth W Dunn, Malgorzata 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.
- [10] Gary J Bassell, Honglai Zhang, Anne L Byrd, Andrea M Femino, Robert H Singer, Krishan L Taneja, Lawrence M Lifshitz, Ira M Herman, and Kenneth S Kosik. Sorting of β -actin mrna and protein to neurites and growth cones in culture. *Journal of Neuroscience*, 18(1): 251–265, 1998.
- [11] Céline S Nicolas, Kyu-Ho Park, Aziza El Harchi, Jacques Camonis, Robert S Kass, Denis Escande, Jean Mérot, Gildas Loussouarn, Françoise Le Bouffant, and Isabelle Baró. I ks response to protein kinase a-dependent kcnq1 phospho-

- rylation requires direct interaction with microtubules. *Cardiovascular research*, 79(3): 427–435, 2008.
- [12] RONALD M Lynch, WALTER Carrington, KEVIN E Fogarty, and FREDRIC S Fay. Metabolic modulation of hexokinase association with mitochondria in living smooth muscle cells. *American Journal of Physiology-Cell Physiology*, 270(2): C488–C499, 1996.
- [13] E Lachmanovich, DE Shvartsman, Y Malka, C Botvin, YI Henis, and AM Weiss. Co-localization analysis of complex formation among membrane proteins by computerized fluorescence microscopy: application to immunofluorescence copatching studies. *Journal of microscopy*, 212(2): 122-131, 2003.
- [14] L Lucas, N Gilbert, D Ploton, and N Bonnet. Visualization of volume data in confocal microscopy: comparison and improvements of volume rendering methods. *Journal of microscopy*, 181(3): 238–252, 1996.
- [15] Yee-Chien Liu and Ann-Shyn Chiang. High-resolution confocal imaging and three-dimensional rendering. *Methods*, 30(1): 86–93, 2003.
- [16] Rensu Theart. *Virtual reality assisted fluorescence microscopy data visualisation and analysis for improved understanding of molecular structures implicated in neurodegenerative diseases*. PhD thesis, Stellenbosch: Stellenbosch University, 2020.

- [17] Volume Rendering Definition. <https://www.heavy.ai/technical-glossary/volume-rendering>. Accessed: 10-April-2024.
- [18] Direct Volume Rendering. https://www.kth.se/social/files/565e35dff27654457fb84363/08_VolumeRendering.pdf. Accessed: 23-February-2024.
- [19] Ricardo Marques, Luís Paulo Santos, Peter Leskovsky, and Céline Paloc. GPU ray casting. Grupo Português de Computação Gráfica, 2009.
- [20] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. *ACM SIGGRAPH computer graphics*, 18(3): 165–174, 1984.
- [21] Ivan E Sutherland. The ultimate display. *Multimedia: From Wagner to virtual reality*, 1, 1965.
- [22] Helen M Deeks. *Biomolecular Applications of Interactive Molecular Dynamics in Virtual Reality*. PhD thesis, Bristol: University of Bristol, 2021.
- [23] Jaron Lanier. *Dawn of the new everything: Encounters with reality and virtual reality*. Henry Holt and Company, 2017.
- [24] CW Sensen and B Hallgrimsson. Advanced imaging in biology and medicine: technology, software environments, applications. Springer, 2009.

- [25] Diederick C Niehorster, Li Li, and Markus Lappe. The accuracy and precision of position and orientation tracking in the htc vive virtual reality system for scientific research. *i-Perception*, 8(3): 2041669517708205, 2017.
- [26] Miguel Borges, Andrew Symington, Brian Coltin, Trey Smith, and Rodrigo Ventura. Htc vive: Analysis and accuracy improvement. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2610–2615. IEEE, 2018.
- [27] HTC VIVE Cosmos Elite Overview. <https://www.vive.com/eu/product/vive-cosmos-elite/overview/>. Accessed: 23-February-2024.
- [28] Björn Sommer, Marc Baaden, Michael Krone, and Andrew Woods. From virtual reality to immersive analytics in bioinformatics. *Journal of integrative bioinformatics*, 15(2), 2018.
- [29] Gaurav Sharma, Constantinos Mavroidis, and Antoine Ferreira. Virtual reality and haptics in nano-and bionanotechnology. *Handbook of theoretical and computational nanotechnology*, 10: 1–33, 2005.
- [30] Thomas D Goddard, Alan A Brilliant, Thomas L Skillman, Steven Vergenz, James Tyrwhitt-Drake, Elaine C Meng, and Thomas E Ferrin. Molecular visualization on the holodeck. *Journal of molecular biology*, 430(21): 3982–3996, 2018.

- [31] Angus PR Johnston, James Rae, Nicholas Ariotti, Benjamin Bailey, Andrew Lilja, Robyn Webb, Charles Ferguson, Sheryl Maher, Thomas P Davis, Richard I Webb, et al. Journey to the centre of the cell: Virtual reality immersion into scientific data. *Traffic*, 19(2): 105–110, 2018.
- [32] Alexandre Borrel and Denis Fourches. Realityconvert: a tool for preparing 3d models of biochemical structures for augmented and virtual reality. *Bioinformatics*, 33(23): 3816–3818, 2017.
- [33] Magnus Norrby, Christoph Grebner, Joakim Eriksson, and Jonas Bostrom. Molecular rift: virtual reality for drug designers. *Journal of chemical information and modeling*, 55(11): 2475–2484, 2015.
- [34] Christoph Grebner, Magnus Norrby, Jonatan Enström, Ingemar Nilsson, Anders Hogner, Jonas Henriksson, Johan Westin, Farzad Faramarzi, Philip Werner, and Jonas Boström. 3d-lab: a collaborative web-based platform for molecular modeling. *Future medicinal chemistry*, 8(14): 1739–1752, 2016.
- [35] Rensu P Theart, Ben Loos, Yigael SL Powrie, and Thomas R Niesler. Improved region of interest selection and colocalization analysis in three-dimensional fluorescence microscopy samples using virtual reality. *PloS one*, 13(8): e0201965, 2018.
- [36] ExMicroVR. <https://www.immsci.com/home/exmicrovr>. Accessed: 10-April-2024.

- [37] ExMicroVR-Download. <https://www.immsci.com/exmicrovr-download>. Accessed: 10-April-2024.
- [38] ConfocalVR. <https://www.immsci.com/home/confocalvr>. Accessed: 10-April-2024.
- [39] ConfocalVR-User-Guide. https://www.immsci.com/confocalvr_users_guide. Accessed: 10-April-2024.
- [40] ChimeraX. <https://www.cgl.ucsf.edu/chimerax>. Accessed: 10-April-2024.
- [41] ChimeraX-Commands. <https://www.rbvi.ucsf.edu/chimerax/docs/user/commands/open.html>. Accessed: 10-April-2024.
- [42] ChimeraX-Volume-Viewer. <https://www.rbvi.ucsf.edu/chimerax/docs/user/tools/volumeviewer.html#multichannel>. Accessed: 10-April-2024.
- [43] Eric F Pettersen, Thomas D Goddard, Conrad C Huang, Elaine C Meng, Gregory S Couch, Tristan I Croll, John H Morris, and Thomas E Ferrin. Ucsf chimeraX: Structure visualization for researchers, educators, and developers. *Protein Science*, 30(1): 70–82, 2021.
- [44] Helen M Berman, Tammy Battistuz, Talapady N Bhat, Wolfgang F Bluhm, Philip E Bourne, Kyle Burkhardt, Zukang Feng, Gary L Gilliland, Lisa Iype, Shri Jain, et al. The protein data bank. *Acta Crystallographica Section D: Biological Crystallography*, 58(6): 899–907, 2002.

- [45] Peter W Rose, Bojan Beran, Chunxiao Bi, Wolfgang F Bluhm, Dimitris Dimitropoulos, David S Goodsell, Andreas Prlić, Martha Quesada, Gregory B Quinn, John D Westbrook, et al. The rcsb protein data bank: redesigned web site and web services. *Nucleic acids research*, 39(suppl 1): D392–D401, 2010.
- [46] Sanja Abbott, Andrii Iudin, Paul K Korir, Sriram Somasundharam, and Ardan Patwardhan. The rcsb protein data bank: redesigned website and web services. *Current protocols in bioinformatics*, 61(1): 5–10, 2018.
- [47] ChimeraX-Virtual-Reality. <https://www.cgl.ucsf.edu/chimerax/docs/user/vr.html>. Accessed: 10-April-2024.
- [48] Andriy Fedorov, Reinhard Beichel, Jayashree Kalpathy-Cramer, Julien Finet, Jean-Christophe Fillion-Robin, Sonia Pujol, Christian Bauer, Dominique Jennings, Fiona Fennessy, Milan Sonka, et al. 3d slicer as an image computing platform for the quantitative imaging network. *Magnetic resonance imaging*, 30(9): 1323–1341, 2012.
- [49] Ron Kikinis, Steve D Pieper, and Kirby G Vosburgh. 3d slicer: a platform for subject-specific image analysis, visualization, and clinical support. In *Intraoperative imaging and image-guided therapy*, pages 277–289. Springer, 2013.
- [50] 3D Slicer image computing platform. <https://www.slicer.org/>. Accessed: 10-April-2024.

- [51] Andras Lasso, Hannah H Nam, Patrick V Dinh, Csaba Pinter, Jean-Christophe Fillion-Robin, Steve Pieper, Sankhesh Jhaveri, Jean-Baptiste Vimort, Ken Martin, Mark Asselin, et al. Interaction with volume-rendered three-dimensional echocardiographic images in virtual reality.. *Journal of the American Society of Echocardiography*, 31(10): 1158–1160, 2018.
- [52] Csaba Pinter, Andras Lasso, Saleh Choueib, Mark Asselin, Jean-Christophe Fillion-Robin, Jean-Baptiste Vimort, Ken Martin, Matthew A Jolley, and Gabor Fichtinger. Slicervr for medical intervention training and planning in immersive virtual reality. *IEEE transactions on medical robotics and bionics*, 2(2): 108–117, 2020.
- [53] Caroline Stefani, Adam Lacy-Hulbert, and Thomas Skillman. Confocalvr: immersive visualization for confocal microscopy. *Journal of molecular biology*, 430(21): 4028–4035, 2018.
- [54] Tina Kapur, Steve Pieper, Andriy Fedorov, Jean-Christophe Fillion-Robin, Michael Halle, Lauren O’Donnell, Andras Lasso, Tamas Ungi, Csaba Pinter, Julien Finet, et al. Increasing the impact of medical image computing using community-based open-access hackathons: The NA-MIC and 3D Slicer experience. *Medical Image Analysis*, 33: 176-180, 2016.
- [55] Add-interactive-Qt-panel-to-VR-scene. <https://github.com/KitwareMedical/SlicerVirtualReality/issues/43>. Accessed: 10-April-

2024.

- [56] Konstantinos Paraskevopoulos, Franziska Kriegenburg, Michael H Tatham, Heike I Rösner, Bethan Medina, Ida B Larsen, Rikke Brandstrup, Kevin G Hardwick, Ronald T Hay, Birthe B Kragelund, et al. Dss1 is a 26S proteasome ubiquitin receptor. *Molecular cell*, 56(3): 453–461, 2014.
- [57] Fiji-An image processing package of ImageJ2. <https://imagej.net/software/fiji/>. Accessed: 20-February-2024.
- [58] Visualizing 4D Light Microscopy with ChimeraX. <https://www.rbvi.ucsf.edu/chimera/data/light-apr2017/light4d.html>. Accessed: 10-April-2024.
- [59] ChimeraX material Command. <https://www.cgl.ucsf.edu/chimerax/docs/user/commands/material.html>. Accessed: 10-April-2024.
- [60] AGAVE 3D pathtrace image viewer. <https://www.allencell.org/pathtrace-rendering.html>. Accessed: 10-April-2024.
- [61] Chapter 39. Volume Rendering Techniques. NVIDIA Developer. <https://developer.nvidia.com/gpugems/gpugems/part-vi-beyond-triangles/chapter-39-volume-rendering-techniques>. Accessed: 10-April-2024.
- [62] Volume rendering—3D Slicer documentation. https://slicer.readthedocs.io/en/latest/user_guide/modules/volumerendering.html. Accessed: 10-April-2024.

- [63] Francesca Vipiana and Lorenzo Crocco. *Electromagnetic Imaging for a Novel Generation of Medical Devices: Fundamental Issues, Methodological Challenges and Practical Implementation*. Springer Nature, 2023.
- [64] 3D Slicer Community. <https://discourse.slicer.org>. Accessed: 10-April-2024.
- [65] Slicer API — 3D Slicer documentation. https://slicer.readthedocs.io/en/latest/developer_guide/api.html. Accessed: 10-April-2024.
- [66] PyCharm: the Python IDE for Professional Developers by JetBrains. <https://www.jetbrains.com/pycharm>. Accessed: 10-April-2024.
- [67] Slicer/Base/Python/slicer/util.py at main · Slicer/Slicer. GitHub. <https://github.com/Slicer/Slicer/blob/main/Base/Python/slicer/util.py>. Accessed: 10-April-2024.
- [68] Slicer: vtkMRMLDisplayNode Class Reference. <https://apidocs.slicer.org/master/classvtkMRMLDisplayNode.html>. Accessed: 10-April-2024.
- [69] Volume rendering—3D Slicer documentation. https://slicer.readthedocs.io/en/latest/user_guide/modules/volumerendering.html. Accessed: 10-April-2024.

- [70] Slicer: vtkSlicerVolumeRenderingLogic Class Reference. <https://apidocs.slicer.org/master/classvtkSlicerVolumeRenderingLogic.html>. Accessed: 10-April-2024.
- [71] Alfred M Franz, Daniel Maleike, Markus Fangerau, Matthias Baumhauer, Lena Maier-Hein, Klaus H Maier-Hein, Hans-Peter Meinzer, and Ivo Wolf. The medical imaging interaction toolkit: challenges and advances. *International journal of computer assisted radiology and surgery*, 8(10): 607–620, 2013.
- [72] The Medical Imaging Interaction Toolkit (MITK). <https://www.mitk.org>. Accessed: 10-April-2024.
- [73] 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.
- [74] ITK-SNAP Home. <http://www.itksnap.org>. Accessed: 10-April-2024.
- [75] Christian Askeland, Ole Vegard Solberg, Janne Beate Lervik Bakeng, Ingerid Reinertsen, Geir Arne Tangen, Erlend Fagertun Hofstad, Daniel Høyer Iversen, Cecilie Våpenstad, Tormod Selbekk, Thomas Langø, et al. Custusx: an open-source research platform for image-guided therapy. *International journal of computer assisted radiology and surgery*, 11: 505–519, 2016.

- [76] Simon Drouin and D Louis Collins. Prism: An open source framework for the interactive design of gpu volume rendering shaders. *PLoS One*, 13(3): e0193636, 2018.
- [77] Mohammadmehdi Bozorgi and Frank Lindseth. Gpu-based multi-volume ray casting within vtk for medical applications. *International journal of computer assisted radiology and surgery*, 10: 293–300, 2015.
- [78] Qt Wiki. <https://wiki.qt.io/Main>. Accessed: 10-April-2024.
- [79] Qt Designer Manual. <https://doc.qt.io/qt-6/qtdesigner-manual.html>. Accessed: 10-April-2024.
- [80] Slicer: qMRMLVolumeThresholdWidget Class Reference. <https://apidocs.slicer.org/master/classqMRMLVolumeThresholdWidget.html>. Accessed: 10-April-2024.
- [81] Mariya Georgieva, Diego I Cattoni, Jean-Bernard Fiche, Thibaut Mutin, Delphine Chamousset, and Marcelo Nollmann. Nanometer resolved single-molecule colocalization of nuclear factors by two-color super resolution microscopy imaging. *Methods*, 105: 44–55, 2016.
- [82] Jeremy Adler and Ingela Parmryd. Quantifying colocalization by correlation: the pearson correlation coefficient is superior to the mander’s overlap coefficient. *Cytometry Part A*, 77(8): 733–742, 2010.

- [83] Jesse S Aaron, Aaron B Taylor, and Teng-Leong Chew. Image colocalization–co-occurrence versus correlation. *Journal of cell science*, 131(3): jcs211847, 2018.
- [84] Huygens Colocalization Analyzer. Scientific Volume Imaging. <https://svi.nl/Huygens-Colocalization-Analyzer>. Accessed: 10-April-2024.
- [85] ColocalizationTheory. Scientific Volume Imaging. <https://svi.nl/ColocalizationTheory>. Accessed: 10-April-2024.
- [86] Huygens Essential. Scientific Volume Imaging. <https://svi.nl/Huygens-Essential>. Accessed: 10-April-2024.
- [87] Katie L Pennington, Colten M McEwan, James Woods, Colin M Muir, AG Pramoda Sahankumari, Riley Eastmond, Eranga R Balasooriya, Christina M Egbert, Sandeep Kaur, Tyler Heaton, et al. SGK2, 14-3-3, and HUWE1 Cooperate to Control the Localization, Stability, and Function of the Oncoprotein PTOV1. *Molecular Cancer Research*, 20(2): 231–243, 2022.
- [88] Jack Davis, Thorsten Meyer, Martin Smolnig, Daniel GJ Smethurst, Lisa Neuhaus, Jonas Heyden, Filomena Broeskamp, Elizabeth SM Edrich, Oskar Knittelfelder, Dagmar Kolb, et al. A dynamic actin cytoskeleton is required to prevent constitutive vdac-dependent mapk signalling and aberrant lipid homeostasis. *Iscience*, 26(9), 2023.

- [89] LB Volk, KL Cooper, T Jiang, ML Paffett, and LG Hudson. Impacts of arsenic on rad18 and translesion synthesis. *Toxicology and applied pharmacology*, 454: 116230, 2022.
- [90] Colocalization Coefficients. Scientific Volume Imaging. <https://svi.nl/ColocalizationCoefficients>. Accessed: 10-April-2024.
- [91] Shin-Guang Chen. Reduced recursive inclusion-exclusion principle for the probability of union events. In *2014 IEEE international conference on industrial engineering and engineering management*, pages 11–13. IEEE, 2014.
- [92] SlicerColoc-Z-Stats/LICENSE at main · benoukraflab/SlicerColoc-Z-Stats. GitHub. <https://github.com/benoukraflab/SlicerColoc-Z-Stats/blob/main/LICENSE>. Accessed: 10-April-2024.
- [93] Download. 3D Slicer. <https://slicer.org/download.html>. Accessed: 10-April-2024.
- [94] slicer-extensions-webapp. <https://extensions.slicer.org/catalog/All/32448/win>. Accessed: 10-April-2024.
- [95] Extensions manager in 3D Slicer. https://slicer.readthedocs.io/en/latest/user_guide/extensions_manager.html. Accessed: 23-February-2024.

- [96] GitHub - ChenXiang96/SlicerColoc-Z-Stats: This is a 3D Slicer extension for computing proteins' colocalization (spatial overlap between different channels) of Z-stack images. GitHub. <https://github.com/ChenXiang96/SlicerColoc-Z-Stats>. Accessed: 10-April-2024.
- [97] Crop Volume — 3D Slicer documentation. https://slicer.readthedocs.io/en/latest/user_guide/modules/cropvolume.html. Accessed: 10-April-2024.
- [98] matplotlib-venn. PyPI. <https://pypi.org/project/matplotlib-venn>. Accessed: 10-April-2024.
- [99] Paul S Brown, Exing Wang, Benjamin Aroeti, Steven J Chapin, Keith E Mostov, and Kenneth W Dunn. Definition of distinct compartments in polarized madin-darby canine kidney (mdck) cells for membrane-volume sorting, polarized sorting and apical recycling. *Traffic*, 1(2): 124–140, 2000.
- [100] Exing Wang, Janice G Pennington, James R Goldenring, Walter Hunziker, and Kenneth W Dunn. Brefeldin a rapidly disrupts plasma membrane polarity by blocking polar sorting in common endosomes of mdck cells. *Journal of cell science*, 114(18): 3309–3321, 2001.
- [101] Huygens Essential-User Guide for version 17.10. <https://cumming.ucalgary.ca/sites/default/files/teams/45/EssentialUserGuide1710.pdf>. Accessed: 10-April-2024.

- [102] 7. Colocalization — BE/Bi 103 a documentation. <http://bebi103.caltech.edu.s3-website-us-east-1.amazonaws.com/2019a/content/recitations/colocalization.html>. Accessed: 10-April-2024.
- [103] Jean-Luc R Stevens, Philipp Rudiger, and James A Bednar. Holoviews: Building complex visualizations easily for reproducible science. In *Proceedings of the 14th Python in Science Conference*, pages 61–69. SciPy, 2015.
- [104] Scientific Volume Imaging-2D histograms. <https://svi.nl/TwoChannelHistogram>. Accessed: 10-April-2024.
- [105] Markups—3D Slicer documentation. https://slicer.readthedocs.io/en/latest/developer_guide/modules/markups.html#markups-json-file-format-mrk-json. Accessed: 10-April-2024.
- [106] Script repository—3D Slicer documentation. https://slicer.readthedocs.io/en/latest/developer_guide/script_repository.html#write-markup-roi-to-json-fil. Accessed: 10-April-2024.
- [107] MRML Overview—3D Slicer documentation. https://slicer.readthedocs.io/en/5.6/developer_guide/mrml_overview.html. Accessed: 10-April-2024.
- [108] Best data import practices for microscopy. 3DSlicer Community. <https://discourse.slicer.org/t/>

best-data-import-practices-for-microscopy/22056. Accessed: 10-April-2024.

- [109] Adler, SN Pagakis, and I Parmryd. eplicate-based noise corrected correlation for accurate measurements of colocalization. *Journal of microscopy*, 230(1): 121–133, 2008.
- [110] Lukas Landmann and Permsin Marbet. Colocalization analysis yields superior results after image restoration. *Microscopy research and technique*, 64(2): 103–112, 2004.
- [111] Yong Wu, Vadim Zinchuk, Olga Grossenbacher-Zinchuk, and Enrico Stefani. Critical evaluation of quantitative colocalization analysis in confocal fluorescence microscopy. *Interdisciplinary Sciences: Computational Life Sciences*, 4: 27–37, 2012.
- [112] Sylvain V Costes, Dirk Daelemans, Edward H Cho, Zachary Dobbin, George Pavlakis, and Stephen Lockett. Automatic and quantitative measurement of protein-protein colocalization in live cells. *Biophysical journal*, 86(6): 3993–4003, 2004.
- [113] 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

[114] NA-MIC Project Weeks. SlicerVR: interactive UI panel in the VR environment. https://projectweek.na-mic.org/PW35_2021_Virtual/Projects/SlicerVR. Accessed: 10-April-2024.