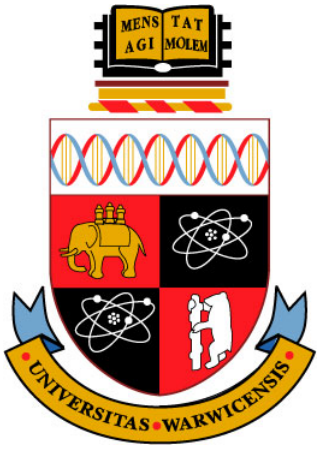


Analysing Images of Protein Distributions using Fluorescent Microscopy

CS350 Data Science Project

Kim Ta

Supervisor: Dr. Julia Brettschneider

Department of Statistics

University of Warwick

2021-2022

Abstract

This paper explores methods of image segmentation on microscopic images of proteins to analyse their distribution. We look at automating the identification of the region of interest in these microscopic images and investigate its importance in statistical calculations.

Proteins are the building blocks of life, performing essential processes that are key to understanding diseases and improving healthcare. The structure of a protein gives insight into how proteins are affected, controlled and modified, as its function is highly dependent on its shape [34].

Research showed that analysis done on microscopic images is affected by their background noise, where singling out the protein structure from the rest of the image provides more reliable results. We found the seeded growing region method to be one of the most effective methods for identifying the protein structure and implemented this into a web application.

Furthermore, we applied our application and research to the biomedical industry, where further research opened possibilities for faster diagnoses and other improvements.

Our GitHub repository containing the finished application and code files can be found using this link: `https://github.com/kimieta/CS350-Project`

Keywords: *Colocalization, Region of Interest, Fluorescent Microscopy, Protein Structure, Seeded Growing Region, Background noise.*

Acknowledgements

I would like to thank Dr Julia Brettschneider, the supervisor of this project, for her support and guidance throughout. Her feedback was invaluable and made this project what it is today. I am furthermore grateful for her mental support and advice during a tough time.

I'd also like to thank Professor Dmitry Chistikov, the second assessor, for his time and feedback on my presentation. His questions helped me identify errors in an assumption I previously made and put me back on the right path.

Abbreviations

Region of Interest	ROI
Pearson's Correlation Coefficient	PCC
Seeded Growing Region	SGR

Contents

List of Figures

1 Introduction

The purpose of this project is to primarily investigate methods of image segmentation in the biomedical industry. We will apply this to microscopic images of proteins by structuring our research around image segmentation in cell biology and its usefulness in industry. The focus is on fluorescent microscopic images as they capture high-resolution images of proteins that we can examine. It is noted that there are other microscopy techniques to obtain these types of images, however fluorescent microscopy is most useful for our purpose.

A protein's function and distribution can be studied by exploring its structure, as its shape determines how it interacts with other molecules [36]. Thus, we centre our attention on applying image segmentation to identify the shape of the protein. In biology, when we single out a specific structure or cell, we refer to this as identifying the region of interest. We use additional resources to display the usefulness of understanding protein structure in medical diagnosis and treatments.

Our research also covers another approach to studying the distribution of a protein, where we look at colocalization. The shape of a cell should be accurately identified in order to perform statistical analysis or to make conclusions about certain features in a sample. Take an experiment with results where cells in the image have unusual shapes, we would need to quantify how different the shapes are from normal to conclude any statistical irregularities. This means that manually identifying the cell shape (by hand-drawing an outline) wouldn't provide repeatable results and thus, not valid. Instead, automating the region of interest identification ensures reliability in results.

We automate the region of interest selection by creating a web application where users can upload images and choose certain inputs, then an algorithm will run to produce the image with an outline showing the region of interest. The implementation will depend on our chosen method (algorithm) and our target audience. We use automation with

the goal of increasing time efficiency and reliability in biological-image analysis, where our aimed audience is researchers and doctors in the biomedical field. This means the application must be easy to use from a non-technical point of view.

This project requires heavy research of biomedical terminology and any current work surrounding this topic.

We acknowledge that there are many other ways and methods to analyse a protein to give insight into their distributions, however, this project specifically focuses on the region of interest identification. With significantly more time, we can expand to exhaust more facets of protein analysis and apply it to other parts of the medical industry.

1.1 Terminology

Region of Interest

Image selection is the process of partitioning a digital image into multiple segments or regions. The term "region of interest" refers to a specific type of image selection in which only sections of the image where cells are present, or where the protein's spatial distributions overlap. This helps to distinguish parts of the image which are important and separate them for analysis.

Detecting the region of interest of an image can be done manually by tracing an outline or by describing its shape, the disadvantage here is that the exact outline will not be obtained if we were to retrace the cell in the same image. Whereas by automating this process, we benefit by speeding up the process of detecting the region of interest and eliminating user bias. It is useful when having a large dataset of images and creates more reliable, and reproducible results.

Colocalization

In biology, colocalization refers to the spatial overlap between two

or more cells or cellular components [19]. The colocalization between two cells can be measured both quantitatively and quantitatively.

Qualitatively, we would look at the microscopic image of a cell with two interacting proteins and describe the extent to which the regions overlap. Quantitatively, we use numerical operations to quantify the degree to which the two proteins are interacting.

This project uses a quantitative approach because it produces robust and reproducible results, it has the advantage of obtaining the value quicker, being easier to analyse and taking up less storage space compared to qualitatively.

We can link colocalization and the region of interest by discussing their relationships with the background noise of a digital image, this is explored later on.

1.2 Motivation and Key Questions

In 2019, a highly contagious virus took over the world. This virus was known as COVID-19, an illness that had devastating effects on the human body, including respiratory problems [37].

When observing the illness's progression through studying proteins such as thrombin, it was suspected that COVID-19 affected the body's ability to break down blood clots, resulting in symptoms like shortness of breath and feeling tired or exhausted [41].

The study of proteins can tell us a lot about how diseases work and we aim to design proteins to combat diseases [39]. Proteins perform essential processes and their structure and shape tell us how they are affected, controlled and modified.

When applying data science to real-life problems, we are interested in the efficiency of methods to analyse images of proteins and what

affects their accuracy. To do this, we need to understand how they are implemented, which involves the challenge of devising algorithms that have appropriate time complexities. The second obstacle would be suiting our implementation to the target audience, as the requirements are situational to the type of cells or features we are trying to identify. This means we require a lot of background research specific to the biomedical industry.

It is impactful on the biomedical industry as we can learn about how to improve diagnostic accuracy and efficiency, especially in cases like during the COVID-19 pandemic where hospitals were overfilled and short of staff. As this is an ongoing problem, there are still limited data resources to specialise our algorithms for the COVID-19 and blood clotting issues.

Furthermore, this project provides the personal opportunity to learn new skills and find solutions to an ongoing problem in the biomedical industry. We capture the core aspects of data science through analysing complex datasets, implementing algorithms and spotting flaws in existing solutions.

2 Objectives

The end goal of this project is to create an application that can find the region of interest of an image and is suited to users in the biomedical industry. We lead up to this through work done in our project aims below.

2.1 Project Aims

We approached this project with heavy research to identify the essential topics and use these to both set aims and guide the timeline of this project as part of project management.

The project aims are as follows:

1. Investigate how the colocalization value of an image is affected by the background noise of that image. (**MUST**)
2. Use mathematical reasoning to justify conclusions surrounding the effects of background noise on colocalization. (**COULD**)
3. Explore the different types of image segmentation on microscopic images. (**MUST**)
4. Implement seeded growing region as a method of automating the region of interest selection. (**MUST**)
5. Create an application from aim four. (**SHOULD**)
6. Given fluorescent microscopic images from blood samples, understand how blood clots can be detected using the region of interest algorithm. (**COULD**)

The requirements stated above were prioritised using the "MoSCoW Requirements" technique [3].

2.2 System Requirement Analysis

Project aims four and five involve the construction and execution of a prototype, so we implement system requirements to ensure our software is fulfilling the target users' needs. It also provides a basis for tests, validation and verification.

The functional aims are as follows:

1. The system **should** be easy to use from a non-technical user's point of view (i.e. easy to navigate around).
2. The system **must** output an outline on the inputted image.
3. The system **could** show the process of outlining the image.
4. Pressing the submit button **should** start the algorithm.
5. The system **should** indicate the application is running.

The non-functional aims are as follows:

1. The system **should** accept standard image files types.
2. The system **must** provide error messages for invalid inputs.

3 Data

The dataset used in this project consists of fluorescent microscopic images of platelets in blood samples [5]. We use these images to test our prototype and learn more about its application in the biomedical. The data handled by the investigation can be located in the folder available under the "Results" section of the article linked in the above reference.

Since the dataset consists of images, the most well-grounded method for data cleaning would be manually cleaning the dataset. As this research topic is still relatively new, there is not enough data to decide which images in the study present anomalies. Instead, we can assume reliability from the controls implemented during the experiment.

3.1 Data Background

This data is from the research study “Prevalence of readily detected amyloid blood clots in ‘unclotted’ Type 2 Diabetes Mellitus and COVID-19 plasma”, Pretorius, E., Venter, C., Laubscher, G.J. et al [5].

This study contains fluorescent microscopic images from platelet-poor plasma (PPP) [1] samples of 40 patients:

- 20 COVID-19 positive patients,
- 10 age-matched type II diabetic patients (T2DM), and
- 10 healthy non-smokers patients with CRP [2] levels in healthy ranges.

SARS-CoV-2 (more commonly known as COVID-19) is an illness of "short" duration which can be rapidly progressive and in need of urgent care [6]. This study investigates blood samples of patients who have or have had COVID-19, to measure their body's ability to remove blood clots. This is important as COVID-19 has been linked to excessive blood clotting [7], resulting in clots travelling to vital organs and causing serious consequences.

The results of the study showed the plasma in the proteins in both COVID-19 and Long COVID patients [3] to be greatly resistant to breaking down clots in the presence of trypsin, which is an enzyme that helps digest proteins and break up blood clots [12]. This suggested a significant failure in fibrinolytic processes in patients with COVID-19 or Long COVID.

3.2 Legal, Ethical and Professional Issues

The dataset used is open access and licensed under a Creative Commons Attribution 4.0 International License [43]. Sharing, adaptation, distribution and reproduction are permitted given appropriate accreditation to the original author(s) and sources.

[1] PPP is blood plasma with a very low number of platelets, it is used to measure how quickly the blood can form a clot by timing and combining it with thromboplastin and calcium. [8]

[2] CRP is a protein sent to the bloodstream in response to a bacterial infection, it is an inflammatory response, which initiates blood clotting. [9][10]

[3] A COVID-19 phenotype (also known as PASC: Post-Acute Sequelae SARS-CoV-2 infection) observed in recovering patients [11].

The data was collected with ethical clearance from the Health Research Ethics Committee (HREC) of Stellenbosch University. Reference: N19/03/043, project ID: 9521. Volunteers had the objectives of the experiment and the risks explained, and informed consent was obtained before blood collection.

To provide both reliability to the results and confidentiality to the subjects of the experiments, the data was anonymised.

4 Literature Review

Colocalization is one of the most popular methods to measure protein distribution, it's an accessible method used to understand a protein's molecular network organisation [48]. This is followed by using Pearson's Correlation Coefficient (PCC) to measure colocalization between cellular components. These methods motivate the initiation of our project.

Most studies on colocalization and protein distributions use fluorescent microscopic imaging [47] as this technique is easy to use and analyse. Studies on colocalization tend to note its sensitivity to image noise however never usually analysed or explore these ideas and effects further. Instead, they give techniques to reduce these effects. Therefore, one of our topics of interest in this project is the effects of image noise on the colocalization value, and we aim to visualise these effects with mathematical reasoning.

In the analysis of protein distribution, the running theme involves analysing spatial distributions [50]. When looking at more complex analyses of spatial distributions, some have approached this problem by using fluorescence lifetime imaging microscopy (FILM) and fluorescence resonance energy transfer (FRET) to study protein interaction [49]. This is a different approach to measuring protein distribution in biomedical imaging, as it models spatial and temporal resolution to detect dynamic interactions of proteins inside living cells. Furthermore, the study concentrates more on the biological side and interpretation of biological systems and processes, whereas our project concentrates on applying data science to biomedical imaging.

Despite the popularity of mapping spatial distributions, there exist countless packages that already perform this statistical analysis. Spatstats [46] is an open-source toolbox used to analyse Spatial Point Patterns, it is a package in R and is one of the easiest packages to access and use.

Since our project looks at gaps and improvements in analysing protein distributions, we pay the most attention to topics that are reviewed less but are still important in the biomedical industry.

5 Background

5.1 Fluorescent Microscopy

Fluorescent microscopy [4] is a form of microscopic imaging which we use to look at specific cells or cellular structures in a sample. [14]

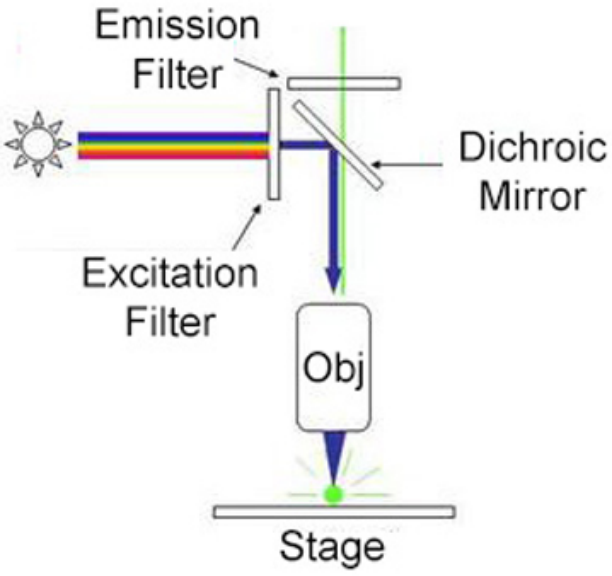


Figure 5.1: Fluorescent microscope [15]

In this technique, we first add a fluorescent dye to the sample. This dye is referred to as a fluorophore and attaches to (tags) a specific protein, i.e. the protein we are interested in and want to image. After adding the fluorophore to the sample, a light of a relatively short wavelength (usually blue or ultraviolet) is reflected onto the sample, causing the fluorophore to be excited and give off light with a lower energy of a longer wavelength. This is what produces our magnified images.

Fluorescent microscopy is widely used because of its high sensitivity and ability to specifically label structures of interest [16]. It is also

[4] Microscopy is the use of microscopes to view objects that cannot be seen with the human eye.

seen to be simpler to use than other imaging techniques and can image living cells or organisms, along with having the ability to provide 3D images of the sample.

In conventional light microscopy, visible light is reflected onto the sample to produce a magnified image [42]. Compared to fluorescent microscopy, the light used is of a longer wavelength and is applied to the sample differently. Light microscopy does not magnify at the same level as fluorescent microscopy and produces lower quality imaging of living cells [17].

Each microscopic technique has different qualities, and we choose the type depending on the purpose of our images, take this image below. This image consists of the same two samples using two different microscopy techniques.

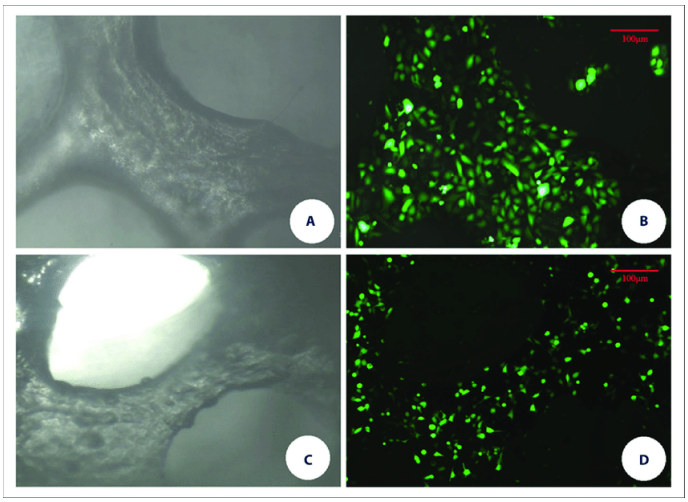


Figure 5.2: Demineralized bone matrix scaffolds under dynamic perfusion culture (A,B) and static culture (C,D) [18]

The two techniques used are light microscopy (A and C) and fluorescent microscopy (B and D). Light microscopy presents the overall structure of the cells in the sample, whereas fluorescent microscopy shows detailed shapes of the proteins inside the sample, and this is what our research is more interested in.

5.2 Pearson's Correlation Coefficient

Pearson's correlation coefficient (PCC) is a statistical measure that measures the linear interdependence between two variables or sets of data [21], used to measure colocalization [20].

The equation for Pearson's Correlation Coefficient is:

$$r_p = \frac{\sum(R_i - \overline{R})(G_i - \overline{G})}{\sqrt{\sum(R_i - \overline{R})^2 \sum(G_i - \overline{G})^2}}$$

Where R_i and G_i are the R and G colour intensities at pixel i of the image.

PCC gives a value between a negative one and a positive one. A PCC value close to a positive one indicates a stronger correlation between the two cells and hence more colocalization compared to a value closer to a negative one, indicating the opposite.

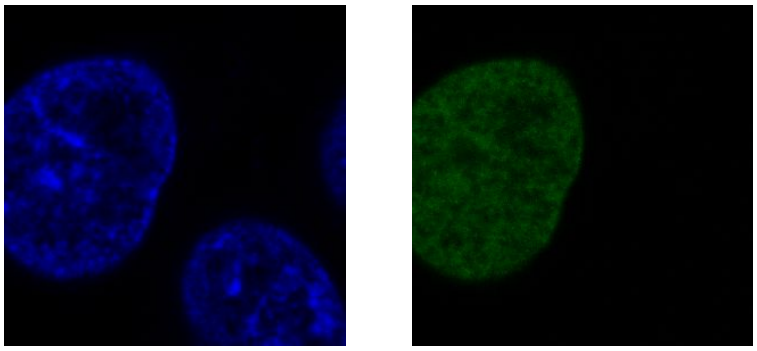
5.3 The Background Noise of a Fluorescent Microscopic Image

In this project, we are interested in ways to ensure reliability in the analysis of microscopic images of proteins. So this section discusses how an image's background noise affects the colocalization value by linking it to the region of interest.

Take the fluorescent microscopic image of histones below.

This sample contains two fluorophores, highlighted by the green and blue dye, each corresponding to a different protein. The sample has been split into two images to easily view the distribution of the two proteins.

Fluorescent microscopic image of proteins



(a) Blue Channel (b) Green channel

When calculating the colocalization value of these two proteins using the entire image, we calculate over three histones with the protein labelled by the blue dye but only one histone with the protein labelled by the green dye.

The analysis of the sample may not accurately represent the degree to which the proteins are interacting, as both images range over a different number of histones, meaning results may be less reliable. It also affects the reproducibility of the experiments, as the following samples would need to measure over the same number of histones. Instead, we should only analyse one histone at a time.

We use this sample to investigate the effects of background noise on colocalization by:

- Removing the space in the image where cells are not present, and
- Cropping the image so that lower levels of colour intensities are not included.

And we will do this by:

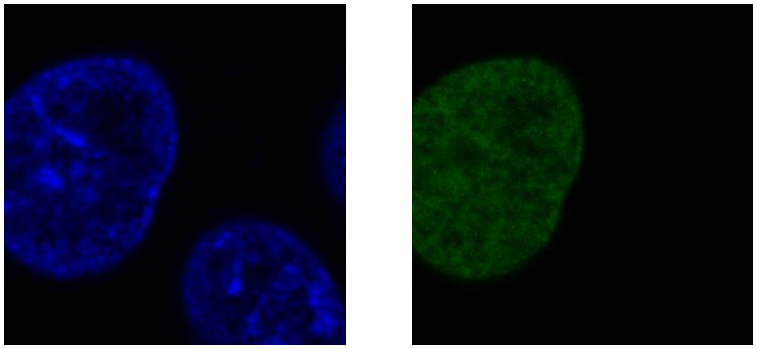
- Analysing the mathematical equation, and
- Running simulations to measure the:

1. PCC of the original image,
2. PCC of simulation 1 image with extra background space,
3. PCC of simulation 1 image with a lot of extra background space,
4. PCC of simulation 1 image cropped to one cell, and
5. PCC of a random section of cell.

5.3.1 Simulations

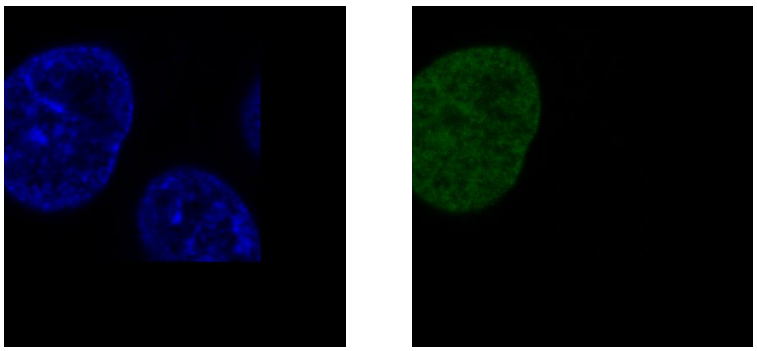
The images of the simulations are as follows.

Simulation 1: Original image (300 x 300 pixels)



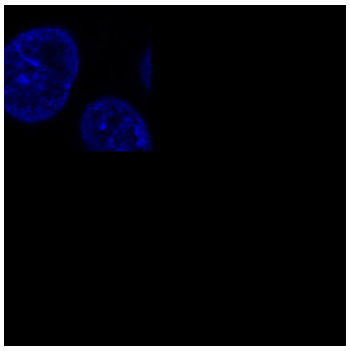
(a) Blue Channel (b) Green channel

Simulation 2: Original with extra blank space (400 x 400 pixels)

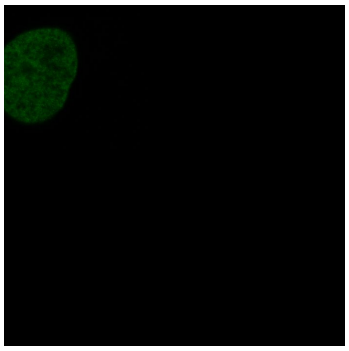


(a) Blue Channel (b) Green channel

Simulation 3: Original image with a lot of extra blank space (700 x 700 pixels)

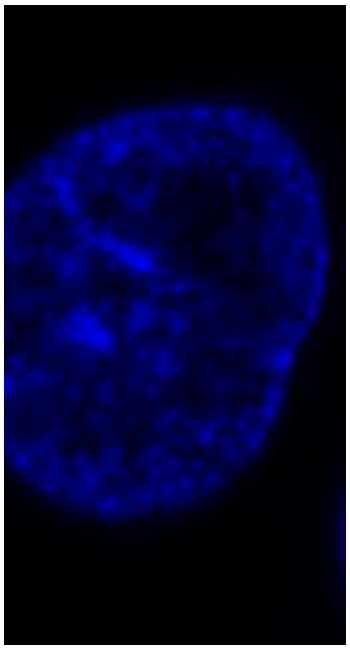


(a) Blue Channel

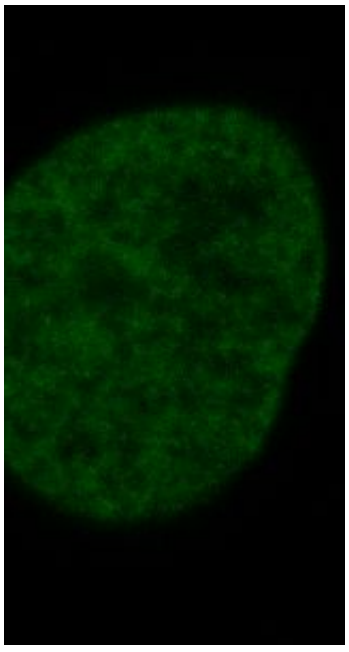


(b) Green channel

Simulation 4: Cropped image (160 x 260 pixels)

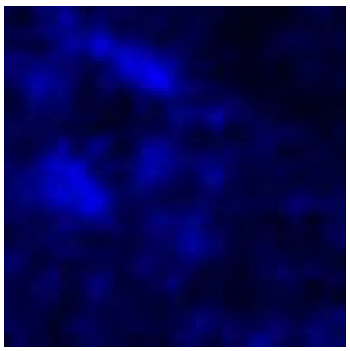


(a) Blue Channel

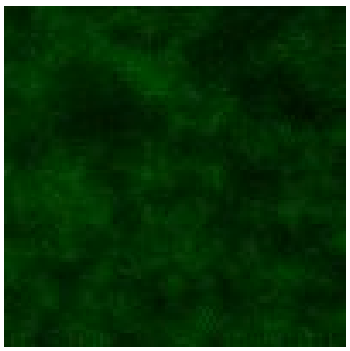


(b) Green channel

Simulation 5: random section of the cell (100 x 100 pixels)



(a) Blue Channel



(b) Green channl

5.3.2 Results

We calculated the PCC values for the simulations above and recorded them in the table below.

Note that the column "Normalisation Factor" refers to the PCC value in proportion to simulation 1.

Simulation	**Modification**	**Dimensions (pixels)**	**PCC (4 d.p.)**	**Normalisation factor (2 d.p.)**
1	None	300 x 300	-0.1271	1
2	Extended	400 x 400	-0.0676	0.53
3	Extended	700 x 700	-0.0212	0.16
4	Cropped	160 x 260	-0.1971	1.55
5	Cropped	100 x 100	-0.4089	3.22

Simulations 1 - 3: *"Remove the space in the image where cells are not present."*

These simulations look at the same image but with different amounts of background space added. The original image gave a Pearson's correlation value (PCC) of -0.1271 (simulation 1). For the image with extra black space added, the PCC value was -0.0676 (simulation 2). To emphasise this further, we added more blank space and obtained a PCC value of -0.0212 (simulation 3).

For these images, we can conclude that increasing the amount of black space in an image decreases the strength of correlation in the Pearson's correlation value.

In these examples, we also wanted to see if there existed a proportional difference or pattern. Increasing the blank space by 100 pixels (33% increase) resulted in the PCC value being half its original value. Increasing the blank space by 400 pixels (133% increase) resulted in the PCC value being 16% of its original value. Therefore the extent to which increasing the background affected the colocalization value was unclear.

Simulations 4 - 5: *"Crop the image so that lower levels of colour intensities are not included."*

This simulation looks at the effects of removing background noise of an image, i.e. removing parts of the images that still have traces of proteins.

In our original image, our protein tagged by the blue dye was present in three histones, but our green tagged protein was only present in one. When we measured the colocalisation value of the entire image, we got a PCC value of -0.1271 (simulation 1). Now, compared to a version of the image that was cropped to a single histone, we got the PCC value of -0.1971 (simulation 4). We further investigated its effects by creating a simulation that only contains a section of a histone, with no blank background space and reduced background noise. In this case, we got a PCC value of -0.4089 (simulation 5).

For these images, we can conclude that as we start choosing more specific regions of the image without no information (the blank areas), the strength of correlation in Pearson's correlation value increases. This is expected as simulations 4 and 5 are essentially the reverse of simulations 1,2 and 3.

In all cases, despite having the same cell, the correlation values (which represent colocalisation between the two proteins in the cell) are different. This highlights the importance of identifying the region of interest in these types of images before performing statistical analysis.

Note this is only for the given example, numbers would differ depending on the cell and image and results may be different.

5.3.3 Reproducibility and Reliability

All code is provided in the appendix and the GitHub project repository, under the folder name "blankspacesimulations" including the original and edited images. In the case that readers would like to edit the original image for reproducibility, the process of obtaining each image is also included.

Simulation 1:

- original images (not edited).

Simulation 2:

- add 100 pixels width to the right,
- and 100 pixels length to the bottom of the image.
- Fill white space with black (hex code 000000).

Simulation 3:

- add 400 pixels width to the right,
- and 400 pixels length to the bottom of the image.
- Fill white space with black (hex code 000000).

Simulation 4:

- Crop image width to 160 pixels (from the left).

Simulation 5:

- Crop image width to 180 pixels,
- and length to 200 pixels from the top left corner,
- then rotate the image by 180 degrees,
- then crop the image to 100 pixels width,
- and 100 pixels length from the top right corner,
- and rotate by 180 degrees.

We edited these images with Microsoft Paint, but any graphics editor can be used.

The images used for the simulations came in pairs, as they are images of the same histone with two different proteins tagged. So the mock images were created following precise and strict instructions to ensure reliability and that images align. No changes were done to distort the original image by stretching or changing the resolution.

5.3.4 Hypothesis

We hypothesise that decreasing the amount of background space where cells are not present in an image would increase the strength of Pearson's Correlation value (both ways negative or positive) because the background area affects the PCC value, indicating a stronger colocalization value than it should be. I.e. the background noise of an image makes the PCC value closer to 0 than it should be [22].

5.3.5 Justification

In the attempt to mathematically justify our hypothesis, we would have to first consider the mathematical equation for Pearson's Correlation Coefficient.

DISCLAIMER: This is only a draft proof.

Therefore large numerical expansions (labelled as [expansion {number}]) will be located in the appendix.

Goal: to compare PCC values of a digital image and the same identical image but with the areas removed where no cells are present (what we call the background space).

The equation for Pearson's Correlation Coefficient is:

$$r_p = \frac{\sum(R_i - \overline{R})(G_i - \overline{G})}{\sqrt{\sum(R_i - \overline{R})^2 \sum(G_i - \overline{G})^2}} \tag{5.1}$$

Where R_i and G_i are the R and G colour intensities at pixel i of the image.

To do this, we first initialise our sample space I, $i \in I$ where $I = (R, G)$.

Next, we define the sample space of our image without the pixels where there is no R and G colour intensity present, i.e. $I_0 = (0, 0)$, and
$I_{/0} = (R, G)/(0, 0)$
$I = I_0 \cap I$

Let r represents the PCC value of the whole image, and $\widetilde{r}$ the PCC value of the image without pixels of zero R and G colour intensities. Then the goal is to prove:

$$|r| \leq |\widetilde{r}| \iff |\frac{r_{nom}}{r_{denom}}| \leq |\frac{\widetilde{r}_{nom}}{\widetilde{r}_{denom}}| \iff \frac{|r_{nom}|}{r_{denom}} \leq \frac{|\widetilde{r}_{nom}|}{\widetilde{r}_{denom}} \iff$$

$$|r_{nom}|\widetilde{r}_{denom} \leq |\widetilde{r}_{nom}|r_{denom}$$

We can do this by proving the following:

1. $|r_{nom}| \leq |\widetilde{r}_{nom}|$ and

2. $\widetilde{r}_{denom} \leq r_{denom}$

First breaking down the nominator of the PCC equation, for the numerator of the PCC equation, we get that:

$$\widetilde{r}_{nom} = \sum_{i \in I_{/0}}^{n} R_i G_i - \overline{R}_0 \sum_{i \in I_{/0}}^{n} G_i - \overline{G}_0 \sum_{i \in I_{/0}}^{n} R_i + \sum_{i \in I_{/0}}^{n} \overline{R}_0 \overline{G}_0 \tag{5.2}$$

$$r_{nom} = \sum_{i \in I}^{n+1} R_i G_i - \overline{R} \sum_{i \in I}^{n+1} G_i - \overline{G} \sum_{i \in I}^{n+1} R_i + \sum_{i \in I}^{n+1} \overline{RG} \tag{5.3}$$

We look at if we want to remove just one pixel where R and G equal 0. So space set I contains $n+1$ pixels and set $I_{/0}$ contains n pixels. If we can prove it for one pixel, then for removing any larger amount of pixels, it will also follow the same case.

We can say that these two expressions are equal as the sum of R_i and G_i points will be the same, since the only difference between the two ranges is adding a zero

$$\sum_{i \in I_{/0}}^{n} R_i G_i = \sum_{i \in I}^{n+1} R_i G_i \tag{5.4}$$

Likewise for the below, also simplifying the expression for ease of reason

$$\sum_{i \in I_{/0}}^{n} R_i = \sum_{i \in I}^{n+1} R_i = x \tag{5.5}$$

And

$$\sum_{i \in I_{/0}}^{n} G_i = \sum_{i \in I}^{n+1} G_i = y \tag{5.6}$$

The standard mean formula:

$$\overline{R_0} = \frac{1}{n}\sum_{i \in I_{/0}}^{n} R_i \tag{5.7}$$

Therefore,

$\overline{R} = \frac{1}{n+1}\sum_{i \in I}^{n+1} R_i = \frac{1}{n+1}\sum_{i \in I_{/0}}^{n} R_i$ (5) $= \frac{1}{n+1}(n\overline{R_0})$ (7) $= \frac{n}{n+1}\overline{R}_0$

$$\overline{R} = \frac{n}{n+1}\overline{R}_0 \tag{5.8}$$

This is the same for G.

Case 1

Expanding the numerator of r, we get that

$$r_{nom} = \frac{\sum_{i \in I}^{n+1} R_i G_i}{n+1} + \frac{n}{n+1}\widetilde{r}_{nom} \tag{5.9}$$

[expansion 1]

Next, we proved that $r_{nom} \leq \widetilde{r}_{nom}$

[expansion 2]

Therefore, $r_{nom} \leq \widetilde{r}_{nom} \implies |r_{nom}| \leq |\widetilde{r}_{nom}|$.

Case 2

Next we have to look at the denominator, we can start with: $\sum(R_i - \overline{R})^2$

$$\sum(R_i - \overline{R})^2 = \frac{\sum_{i \in I} R_i^2}{n+1} + \frac{n}{n+1}\sum_{i \in I_{/0}}(R_i - \overline{R})^2 \tag{5.10}$$

[expansion 3]

Next, we want to prove that $r_{denom} > \widetilde{r}_{denom}$

Expanding the denominator of r^2, we get that

$$(r_{denom})^2 = (\widetilde{r}_{denom})^2 + \frac{1}{(n+1)^2}(cn + \sum_{i \in I} R_i^2 \sum_{i \in I} G_i^2 - 1) \qquad (5.11)$$

Where $c = \overline{R}^2 \sum_{i \in I} G_i^2 (n - 2\sum_{i \in I} R_i^2) + \overline{G}^2 \sum_{i \in I} R_i^2 (n - 2\sum_{i \in I} G_i^2)$

[expansion 4]

And we get that, $(r_{denom})^2 \geq (\widetilde{r}_{denom})^2$, therefore $r_{denom} \geq \widetilde{r}_{denom}$.

By case 1, $|r_{nom}| \leq |\widetilde{r}_{nom}|$ and by case 2, $r_{denom} \geq \widetilde{r}_{denom}$. Therefore $|r_{nom}|\widetilde{r}_{denom} \leq |\widetilde{r}_{nom}|r_{denom}$ which means that $|r| \leq |\widetilde{r}|$. As this is true for removing one pixel where R and G colour intensities as zero, this would be true for any number of pixels removed greater than one.

5.4 Mander's Overlap Coefficient

Our simulations used Pearson's Correlation Coefficient to measure our degree of colocalization. We considered another method known as the Manders Overlap coefficient. Mander's Overlap Coefficient (MOC) is a quantitative measure to quantify the degree of colocalization between fluorophores, it compares the co-occurrence of fluorescence among pixels [23].

The equation for Mander's Overlap Coefficient is:

$$MOC = \frac{\sum_i (R_i \times G_i)}{\sqrt{\sum_i (R_i)^2 \sum_i (G_i)^2}}$$

Where R_i and G_i are the R and G average grey levels with the range

-1 to +1, and MOC gives a range of 0 to 1, where the larger the MOC, the stronger evidence for colocalisation. [24].

5.4.1 PCC vs MOC

We chose our method of measuring colocalization by researching their disadvantages and seeing which one was most suitable for our type of images.

PCC is sensitive to differences in mean signal intensities, as shown in the "background of a fluorescent microscopic image" section, the background noise of an image affects its value.

MOC is not sensitive to differences in mean signal intensities (it is not contained in the equation), however, its ability to differentiate between different patterns of colocalisation is limited [25]. MOC is affected by both changes in co-occurrence and correlation, where different distributions of intensity[5] dramatically alter the correlation [26].

Therefore in this project, we concentrate on using PCC.

5.5 Methods of Detecting the Region of Interest

In our application, we want to obtain the region of interest using the most appropriate methods. So this section investigates some popular methods to understand their entailments and decide which one to implement. We will be looking at:

- Automated Threshold Detection,
- Fuzzy Set Theory, and
- Seeded Growing Region.

[5] Gaussian, gamma, uniform, exponential

Note that the results from our research lead us to choose seeded growing region as the method to implement, and automated threshold detection is the method used in an existing package to which we compare our implementation to.

5.5.1 Automated Threshold Detection

Automated Threshold Detection is a method to obtain the region of interest of an image. This is a threshold-based technique that was developed by Costes et al., and returns a threshold value that can be used to identify the background of an image based on the range of pixel values which returns a positive PCC value [27].

The threshold of the image is found by:

- Measuring the PCC for all pixels in the image, then
- Measuring the PCC for the next lower colour intensities of the image, and
- Repeating the second step with lower colour intensities until the PCC value drops to or below zero.

This method is advantageous as it is both reproducible and robust, effective for images with high signal-to-background ratios and struggles with images that have very high labelling density or large differences in the number of structures labelled.

5.5.2 Fuzzy Set Theory

Another threshold-based image segmentation technique is through using the fuzzy set theory. The threshold is found by averaging the grey level of a pixel by comparing it to the pixel's neighbours [28].

The measure of fuzziness usually indicates the degree of fuzziness. We can measure this via entropy:

$$E(A) = \frac{1}{nln2} \sum_{i} S(\mu_A(x_i)), \quad i = 1, 2, ...n.$$

Where $\mu_A(x_i)$ denotes the grade of possessing some brightness property μ_A, which is the distance between the grey tone image and its nearest two-tone version.

This is based on Shannon's function:

$$S(\mu_A(x_i)) = -\mu_A(x_i)ln[\mu_A(x_i)] - [1 - \mu_A(x_i)]ln[1 - \mu_A(x_i)].$$

To measure the entropy of an image set X, we can extend these equations to a two-dimensional plane:

$$E(X) = \frac{1}{MNln2} \sum_m \sum_n S(\mu_x(x_{mn}))$$

with $m = 0, 1, ..., M - 1$ and $n = 0, 1, ..., N - 1$

When we talk about the index of fuzziness, we refer to measuring the distance between the grey-level image and its crisp version (average amount of fuzziness). As for nonfuzziness, this takes the absolute difference between the crisp image and its complement (average amount of nonfuzziness)

There also exists an algorithm using fuzzy geometric properties, developed by Sankar K. Pal and Azeriel Rosenfeld. To choose the appropriate nonfuzzy threshold, compactness of fuzziness is minimised to obtain the fuzzy and nonfuzzy version of an ill-defined image [29].

5.6 Seeded Growing Region

Region growing is a region-based image selection method. The method starts with an initial set of small areas in a digital image [6], which are referred to as "seeds", and then the seeds grow by successively adding in neighbouring pixels. The process stops when every pixel in the image is assigned to one (and only one) seed group [30].

5.7 Research Methods

Our data was obtained from a case study that used primary research to provide its results. The study was reliable as it provided exact procedures to acquire the data and gave access to all raw files. We did consider collecting our own data, however, because of current COVID restrictions, we were unable to gain access to the Warwick Medical School resources.

[6] coordinates or pixels on the image

We used this case study to apply our research and project to the industry. Secondary data analysis was our main source of information. This was collectively from academic journals and articles. We looked at qualitative approaches to our problem and it was helpful as we needed lots of information to create our algorithms and come to our conclusions.

Quantitative resources were less helpful for the research proportion of this project as our goals and aim mainly consisted of high detailed research where we build upon it through creating an application and making hypotheses.

Instead, we used experiments and simulations after we created our application to test its feasibility and efficiency. This method was only suitable once the primary research stages had been completed as it was required to create the application.

6 Design

As part of this project, we aimed to create an application that found the region of interest of a digital image. From our research, we concluded that seeded growing region would be the most appropriate method to implement for this project.

6.1 Audience

This project involved applying data science to the biomedical industry, so our target audience is users in this field with an interest in analysing microscopic images of cells. Therefore, this application needs to be suitable for non-technical users and easy to load and run. We do this by considering the application's features and user interface (UI).

6.2 User Interface and Properties

To create an application that would be suitable for non-technical users, we had to construct a design that was both simple and well-organised.

We followed the basic principles of user interface design to build our UI and make it user friendly and easily understandable. Having a simple and consistent design for each page makes it easy to navigate around, where we included tabs for each section of the page so that it is straightforwards for the user to go back to other tabs, meaning all actions in the application are reversible. The input section on the left-hand side of the web page contains a submit button so that the user can identify when they start running the application.

Our first page in the application consists of information. This page describes the purpose of the application, the inputs, the languages used to code it and a link to the GitHub repository in which all background information surrounding this project is included, along with the original code source.

Following Neilson's Usability Principles, we use consistent language on the information page to avoid confusion between different words used for the same meaning and include inputs to allow flexibility and increase interactivity. We avoided using any jargon that could not easily be explained.

The second page applies the inputted constraints to the uploaded image to output the image modified to a different resolution and show where the initial seed point is located on the image. This allows the user to check their inputted constraints before they run the algorithm, to ensure this is what they are trying to analyse.

Our third page provides the image with the region of interest identified by an outline, this is found by our implementation using the SGR algorithm. The last page provides an animation of the SGR outlining process.

6.3 Inputs and Constraints

In this application, the user uploads their digital image and has to choose some inputs. These constraints include the coordinates of the initial seed point (pixel within the cell or structure we want to be identifying), the threshold, and the colour of the outline. Having these inputs allow for flexibility in the application and increases interactivity.

We allow users to pick the colour of the outline to make the application more accessible. This is because colourblindness is very common, approximately 1 in 12 men and 1 in 200 women are colourblind, with red-green colourblindness being most common [31]. So by allowing the choice to choose outline colour, we can reach a wider audience.

The threshold is the amount in which we distinguish the difference between the pixel belonging to the seed group or the non-seed group. Using a lower threshold means we will be disregarding parts of the image noise where the protein is less present, so the region of interest will be more rigid and specific. Higher thresholds will include more of the background around the cell. Therefore depending on what the user wants will depend on the chosen threshold value. For our case of looking at the colocalization of a cell, we want an accurate outline, so we would choose a medium threshold that includes as much of the image as possible, without too much background noise or space.

The resolution of an image determines its quality. An image of lower resolution has fewer pixels than that of a higher resolution, this algorithm will run faster on an image of lower resolution. This is the quality versus computational speed tradeoff. Consequently, despite the resolution of the images, the algorithm still produces a rough accurate

outline. So in cases of large datasets, we opt for using lower resolution images to achieve a faster computational speed.

7 Implementation

Our application runs an algorithm based on the seeded growing region method to find our region of interest. This section discusses the process we used to implement this technique. These are the steps in the algorithm:

1. Select Image seed and resolution,
2. Add seed to seed group,
3. Convert image to greyscale,
4. Recursively search each pixel around the seed group,
    - If the pixel meets the manually set threshold, add the pixel to the seed group,
    - Else is belongs to the not seed group.
5. Finish when all pixels have been added to a group (seed or not seed)

.

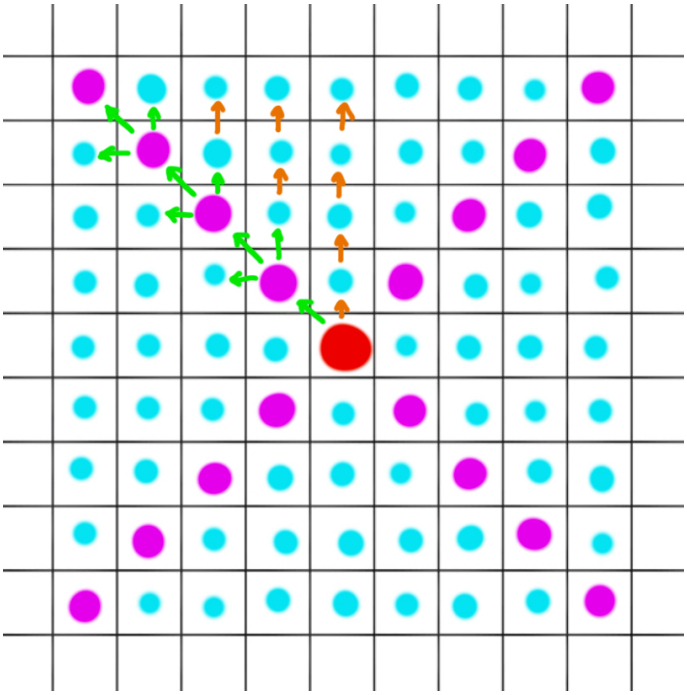


Figure 7.1: SGR algorithm: pointer movements

We start by choosing a seed, manually done. On the image above, it is referenced as a red dot. Around the first pixel, there will be four edge and four corner pointers. These will be initiated to start the algorithm.

In step four, we recursively search each pixel around the seed group by using two functions that relate to whether the pointer is an edge pointer or a corner pointer.

Edge pointers are indicated by the blue dots on the diagram above, and will only move on to search one pixel at a time. They lead on to only creating one new pointer at a time, where their direction of movement is shown by the orange arrow.

Function EdgePointer (direction, x-cord, y-cord)

```
0: if direction is up then
0:    Pointer moves up one unit
0: else if direction is down then
0:    Pointer moves down one unit
0: else if direction is right then
0:    Pointer moves right one unit
0: else if direction is left then
0:    Pointer moves left one unit
0: end if
0: return Pointer =0
```

Each edge pointer will move in the same direction it was originally set, e.g. an edge pointer going up will only have subsequent edge pointers going up.

Corner pointers are indicated by the pink dots on the diagram above, they move in three directions (hence creating three new pointers, one corner pointer and two edge pointers), where their movement is shown by the green arrows.

CornerPointer (quadrant, x-cord, y-cord)

```
0: if quadrant is 1 then
0:    Edge pointer one moves up one unit
0:    Edge pointer two moves left one unit
0:    Corner Pointer moves up and left one unit
0: else if quadrant is 2 then
0:    Edge pointer one moves up one unit
0:    Edge pointer two moves right one unit
0:    Corner Pointer moves up and right one unit
0: else if quadrant is 3 then
0:    Edge pointer one moves down one unit
0:    Edge pointer two moves right one unit
0:    Corner Pointer moves down and right one unit
0: else if quadrant is 4 then
0:    Edge pointer one moves down one unit
0:    Edge pointer two moves left one unit
0:    Corner Pointer moves down and left one unit
0: end if
0: return Edge pointer 1, 2 and corner pointer =0
```

The direction of movement of subsequent pointers created by the corner pointer will depend on the quadrant it exists in.

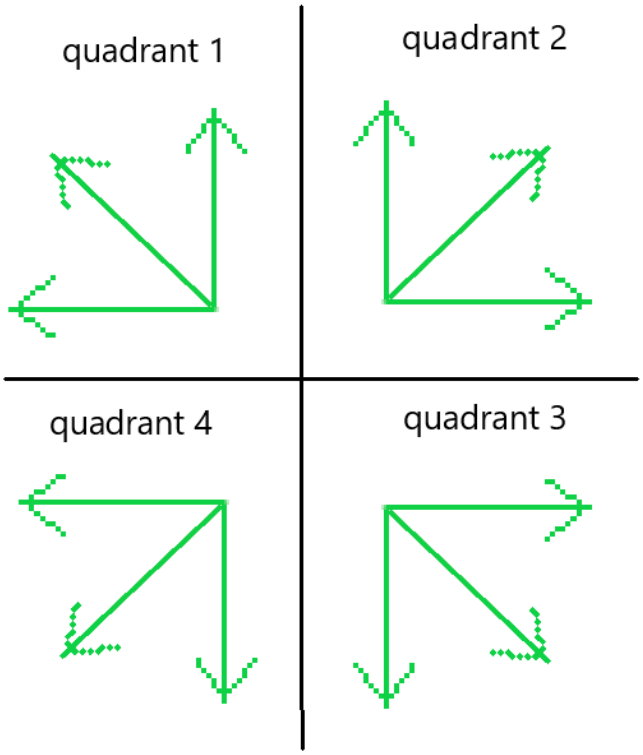


Figure 7.2: SGR algorithm: corner pointer quadrants and directions

The algorithm is a recursive call to create new pointers each time a new pixel is crossed in order to cover the whole image. At each pointer, the algorithm will add the pixel to the seed group if the greyscale value of that pixel meets the set threshold. Otherwise, it will be assigned to the non-seed group, hence we have two group assignments.

If a pixel meets the threshold AND subsequent pixels do not, then we know we are at the edge of the shape we want to outline, thus we colour the pixel. And this continues around the seed until all pixels are assigned a group.

7.1 Converting to Grayscale

In our implementation, step three talks about converting the image to greyscale.

Pixels in a digital image are made up of three values, for red, green and blue colours (RGB value). To convert an image to greyscale, we get the average of the pixel's RGB value. Greyscale is where pixels are represented by one value, displaying only black, white and grey colours (multiple levels). This makes it easier to compare to a threshold value, as we only need to work with a single figure.

The equation for working out the greyscale value is:

$$Greyscalevalue = \frac{R + G + B}{3} \tag{7.1}$$

7.2 Error Checking

The choice of threshold and seed coordinates is influenced by the image. Users would choose these inputs based on what is in their image and how rigid they want the outline to be. So the second page of the application helps ensure the inputs are valid before the user runs the application.

We made it easy for users to choose their seed coordinate by including an x and y-axis on the image, this means they can pinpoint an exact location. In the event that the user chooses an invalid seed value, i.e. the point is outside the image, then the application will return an error message providing the valid range of values the user can choose the seed coordinate to be in. In general, the seed coordinate should be in an area of contrast to the background.

As for the threshold, if the constraint is invalid, an error message will be provided with advice on how to choose a more suitable threshold value, as it is harder to determine what the acceptable range would be.

7.3 Flaws and Advancements

This application currently can only identify one region of interest. However, we have provided the research with the capability to develop this application into identifying multiple structures of interest.

The succession of our seeded growing region highly depends on our chosen seed(s). With more time, we would like to make the application more efficient by automating the seed selection.

To mathematically choose which pixels we want to use as seeds, we can look at the **gradient-based homogeneity criteria** [32]. In this method, a cost function is set as a criterion to consider features of the image around the seed, meaning it helps choose the seed with respect to the background of the image, the strength of the region boundary, size, shape and colour.

The cost function is defined as:

$$G_m = G_{max} - \frac{G(x,y)}{G_{max}} - G_{min}$$

Such that $0 < G_m < 1$, where $G(x,y)$ is the gradient magnitude at pixel under consideration, and G_{min}, G_{max} is the minimum and maximum gradient present in the image.

Secondly, part of this project included investigating images of proteins using colocalization. We mathematically discussed this in the "background noise of a fluorescent microscopic image" section.

This is another opening for broadening the features of our application. With additional time, we could use our current ROI results on the images and use PCC to calculate the colocalization value of the image. We would require the sample to contain two different proteins tagged

and have to decide whether the application takes in one (combined) image or two separate images (for each protein tagged).

8 Control and Integration

The choice of programming language was R as it contained a wide variety of statistics-related libraries that were integrated into the application, along with being a tool popularly used for quantitative analysis, hence suited to the aims of the project.

Packages in R (instructions of importing included in GitHub repository) included packages for:

- image importing and processing - ***imager***
- working with data frames - ***dplyr***
- complex image processing - ***magick***
- creating graphics - ***ggplot2***
- creating animations- ***animation***
- building an interactive web application - ***shiny*** version 1.7.1.
- adding animations into a Rshiny application - ***shinycssloaders***
- styling the Rshiny application - ***bslib***

RShiny was a particular package in R that provided a web framework to build the application on, this was chosen because of the ability to create interactivity and visualisations along with its shareability.

We used Git as a form of version control, this was important when keeping track of changes and identifying where key choices were made in the creation of the application. It allowed for reverting to previous editions of code in the cases of unknown problems or bugs. We also used GitHub as a secondary form of storage, it acted as a backup in case of a hardware failure and final editions of all code have been uploaded there.

9 Project Management

This dissertation is part of an individual project that ran over the course of an academic year, with the aim to showcase the ability to apply data science in a wider context.

Previous elements in this project consisted of a specification document and an oral presentation. The specification document was made in the first few weeks of the project, giving detailed aims to guide the development of the project and ensure its feasibility. This was essential to organising the project and making sure it could be completed within the time frame. Note that this document can be found on the linked GitHub.

9.1 Time Organisation

The start of the project focused on heavy research to help identify and define the focus, methods, and goals to outline the purpose. Proceeding on, we created Gantt charts to schedule and organise the components of the project.

Our Gantt charts ran over the course of two terms (two ten-week periods) and consisted of weekly tasks, milestones and deadlines. These Gantt charts can be found in appendix B.

Our timetables only included tasks during the academic terms due to the uncertainty of schedules during the non-academic term times. This was planned to allow time for document writing which is not specified in the charts, and any additional fixes or to solve any problems encountered.

The weekly tasks outlined the research focus, we used this to organise the workload of this project into definable tasks that aline with the coding aspects of this project. We chose to have tasks to be completed weekly rather than setting dates to allow for flexibility and time adjust-

ments. Its positives included having maximum productivity without exceeding the deadline.

Milestones were implemented to monitor our progress. Weekly meetings with the project supervisor were scheduled to keep the project on track, the milestones helped set a basis to show the development of the project and the problems encountered.

9.2 Risk Management

In the early stages of our project, we performed a risk assessment to identify possible problems we could encounter and recorded them in the table below. We then used these to structure our Gantt chart.

Risk	Impact	Solution	Severity	Likeli - hood	Total Risk
Illnesses (like covid) and personal circumstances	Inability to progress with tasks on the project	Include leeway time and prioritise tasks	6	6	36
Libraries fail	Libraries cannot be used	Revert the library version to the one that worked before	4	2	8
ROI implementation is non-functional	Application cannot run or perform to what is needed	Research existing packages to identify the ROI as a backup	8	2	16
Proof cannot be made for the image noise issue	There is no evidence to back-up our hypothesis	Run simulations and find evidence from research	3	7	21
Hardware failure	Lose all the work	Use GitHub to store work online and constantly back up work	10	3	30
External readers cannot compile or run the code	Application cannot be peer-reviewed	Use the most up to date version of languages and programs used and state them for reference.	5	5	25

The classification of our risks was done using the risk assessment matrix [44].

- Severity: 1 (not severe) - 10 (very severe).
- Likelihood: 1 (not likely) - 10 (highly likely).
- Risk: the multiplication of severity and likelihood.

We use the "Total Risk" column to identify which risks to be most cautious of when undertaking this project.

In response to the high chances of unforeseen problems, we included leeway time between each term to allow for any delays. We recognize that these problems may not occur, in this case, the extra time would be used to improve document writing and additional research similar to the topic.

External deadlines were included to be mindful of the work capabilities, this helped create a realistic timetable that could be followed efficiently. The project requirements were prioritized using the MoSCoW method to ensure a working prototype was produced at the end of the project that was both functional and met the requirements.

9.3 Development Methodology

We approached this project's management with a plan-driven model methodology. We wanted to use careful planning and sequential phases to design the project so that one task follows another, making it structured and easy to trace.

The waterfall methodology relies on careful planning and detailed documentation. Thus, we used a waterfall methodology and integrated some agile components, this is because the project was initialised with a project specification to define the needs, objectives and constraints. So we could already confirm what tools were needed and the features required in our project and application.

Despite this methodology's difficulty to adjust to major changes, it provided to be more useful for this project compared to full-on agile software development. Agile development requires an initial heavy understanding of the topic area, which is difficult due to unfamiliarities with technology in the biomedical sector and likewise, the target audience. Our project did not require any SCRUM stages because our coding phases were a follow up of sequential research and lead to more investigations. Secondly, it was not as suitable because agile development usually depends on consistent collaboration between the targeted audience and developer [33].

Overall, a plan-driven methodology was more appropriate since it provided a clear structure that helped design, test and delivers a product that meets the goals we defined earlier, with little change to the original plan. This method is very organised and advantageous for time management [45].

9.4 System Architecture

We used a self-monitoring architecture to provide logic to our application and designed it to take action if a problem was detected. Our computations were implemented on separate channels and the outputs were compared. If failures occurred, they were judged and error messages were outputted.

9.5 Testing Methodology

Our testing methodology ensured the application was functional and met the user needs and product specifications. We demonstrated this through validation and verification.

The testing methods consisted of two different (dynamic) software methods known as white-box and black-box testing to inspect the code's implementation and functionality respectively.

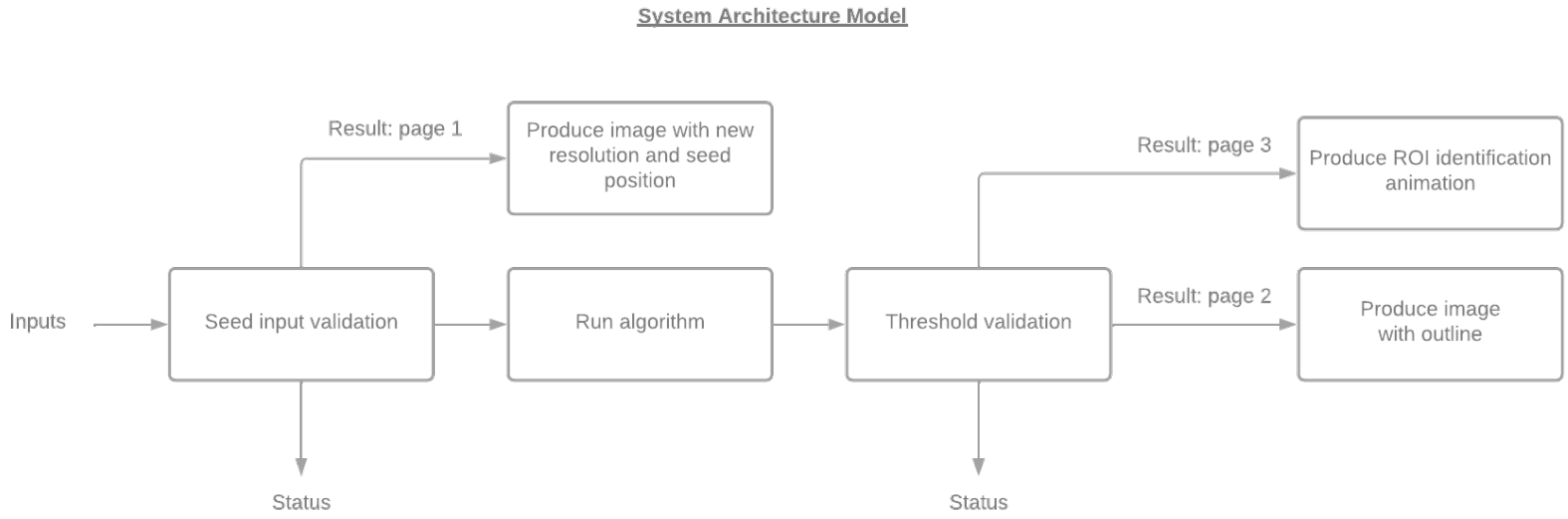


Figure 9.1: System Architecture Diagram

White-box testing aimed to verify the inner workings of the application and ensure it was usable. We did this by:

- Component-wise testing - *testing each part of the code before integrating it together*, and
- Running simulations on test and mock data - *testing the edge cases and reducing possible errors.*

Code walkthroughs and error checking were our forms of static testing, which involved testing without execution.

Black-box testing aimed to measure the readability and usefulness of the outputted application results, along with its accessibility. We did this by:

- Presenting the project and giving an application demonstration,
- Comparing it to existing applications, and
- Using feedback from peer assessments.

9.6 Results

Note: We provided a folder on our GitHub containing a small subset of images with pre-tested inputs for users to trial the application accompanied by expected results. This can be found in the "Testing" folder with easy to follow instructions and the raw code.

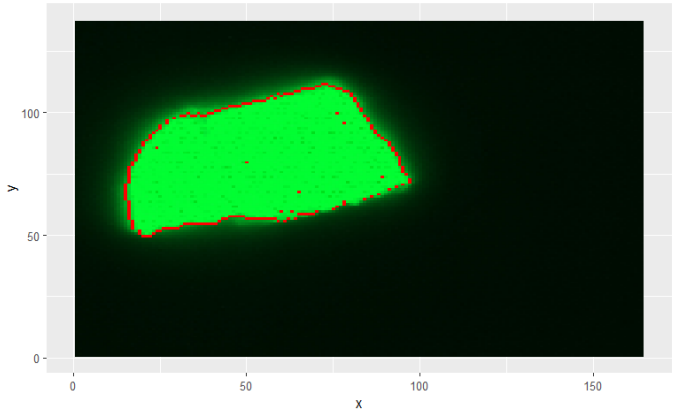
White-box testing

In our previous subsection *"Inputs and Constraints"* under the section *"Design"*, we mentioned the effects of changing resolution or threshold on the accuracy of the region of interest identified. To further explore and show these effects, we ran some tests below.

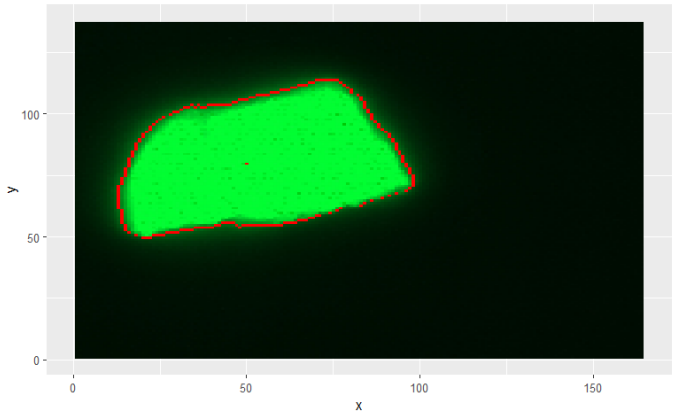
Statement for changing the threshold: “Running the algorithm with a higher threshold will include more of the background around the structure we are trying to identify”.

Figure 9.2: Effect of changing the threshold

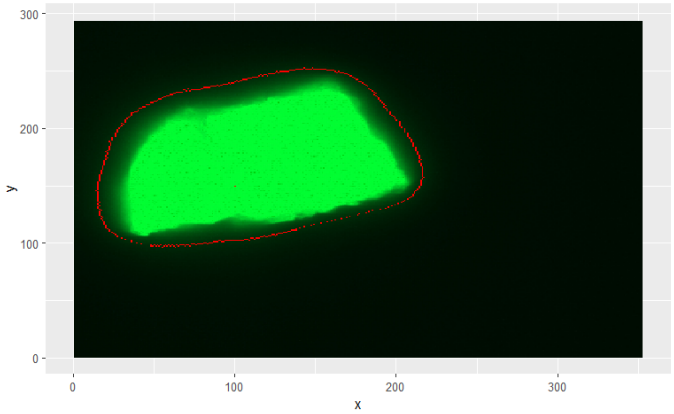
(a) Threshold: 0.1,
Resolution: 7% decrease



(b) Threshold: 0.3,
Resolution: 7% decrease



(c) Threshold: 0.5,
Resolution: 7% decrease

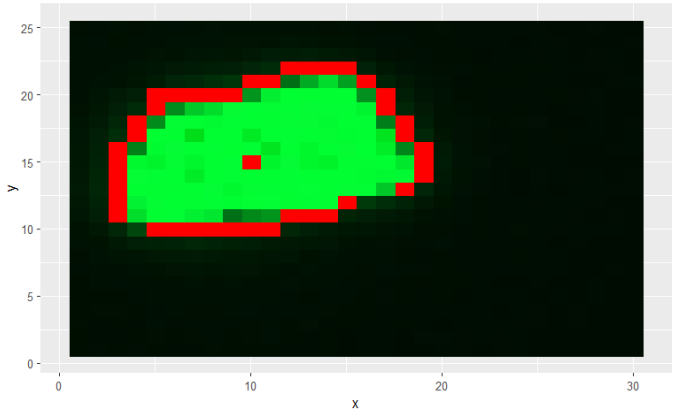


As we can see from figure 8.2 a), with a lower threshold, the outline is tighter and excludes part of the image with low levels of colour intensity. Compared to a higher threshold, shown in figure 8.2 c), we can see the whole cell is contained in the outline in addition to some background noise.

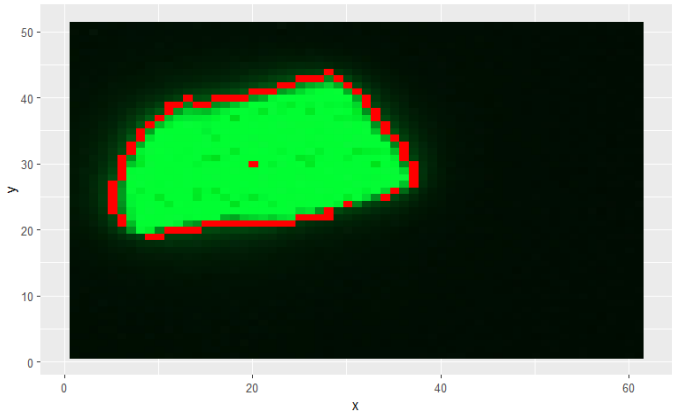
Statement for changing the resolution: “An image of lower resolution has worse quality image, however, runs faster and still provides an accurate outline.

Figure 9.3: Effect of changing the resolution

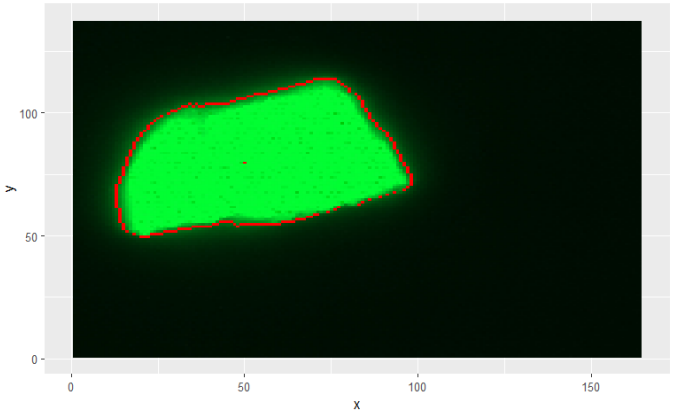
(a) Threshold: 0.3,
Resolution: 80% decrease



(b) Threshold: 0.3,
Resolution: 40% decrease



(c) Threshold: 0.3,
Resolution: 7% decrease



As we can see here, despite a lower quality image, the image still presents an accurate outline, the main issue being the image is of worse quality. We found this to be a good tradeoff as the algorithm ran much faster.

The testing of the code and software itself was successful, the application can be used by anyone to upload images and run the algorithm. It provides error messages for cases where a region of interest cannot be found due to the constraints chosen. Simulations also helped to identify any edge cases we may have been missing.

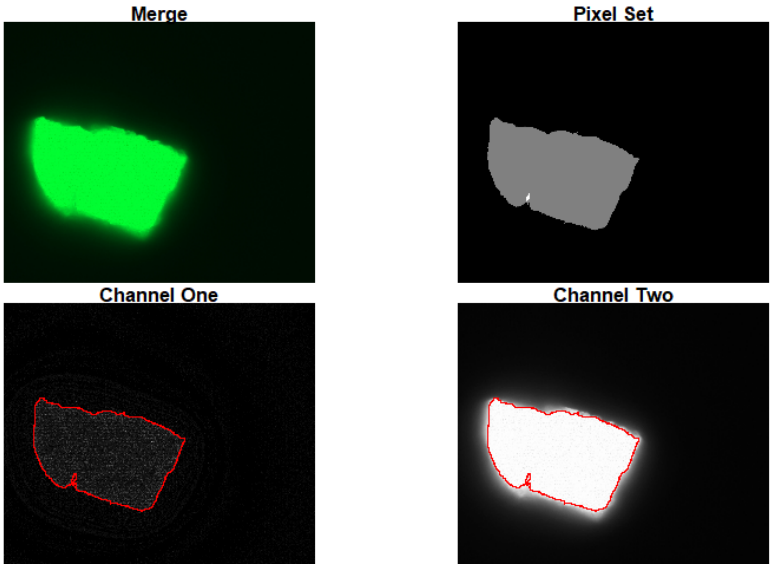
Black-box testing

The design of the application was clean and simple. From peer reviews, the application was easy to understand and use, where our only real problem or difficulty was installing R, as the application required it to run. Therefore, we created a detailed GitHub page explaining the application further in the case of uncertainties, and this included all the test files mentioned above and the program details.

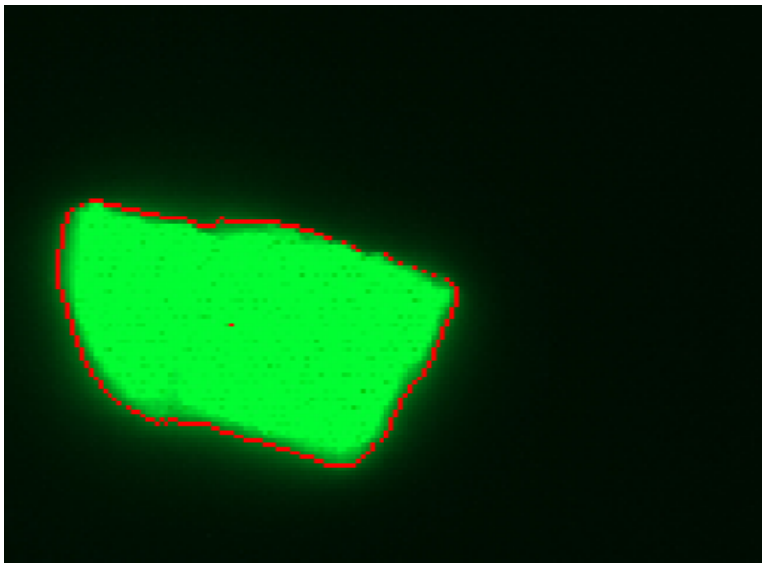
The second part of our testing was comparing it to an existing package that performed the same function. *Colocr* is a package in R that measures the region of interest using the Costes method (section “automated threshold detection").

Figure 9.4: Application Comparison

(a) Automated Threshold Detection



(b) Seeded Growing Region



Testing on the same image, both applications identified the region of interest well. In the *Colocr* package, the region of interest was highlighted by converting the image to grey and white so that the outline showed up more clearly.

The main difference between our application and *Colocr* was that the *Colocr* package used a threshold-based method whereas ours used a region-based method. Threshold-based methods use a threshold to distinguish what belongs in the region of interest, so it can exclude major parts of the shape and make it look irregular compared to a normal cell shape.

Take figure 8.4 a), the bottom left had a divot in the cell where protein the protein was still present but in lower concentrations. This is not always beneficial as it is missing an important part of the cell that our algorithm includes. Therefore, our application using a region-based method is more beneficial for our project purpose because it is more likely to identify non-rigid smooth shapes like cells.

9.7 Flaws and Improvements

In our second Gantt chart, the tasks in the research section were less specific as we could not predict the technical issues we could get during the implementation phase and what extra research we would need. This could have been improved by cutting down that chunk into smaller sections, where we consider specific portions of the report writing.

The proposal timelines did not include the validation of the R shiny application, just the creating and testing of the baseline code, however, extra time accounted for in risk management allowed for this.

When we tested and reviewed our application, it was noted that the outline looks slightly rough. With more time, we can improve this by slightly changing how we manage the graphics to smooth out the outline. Secondly, we saw that inputs needed to be valid for the algorithm to run. Currently, we cannot provide a valid range for the threshold, only for seed coordinates. We compromised by creating an application where actions are reversible so that users could retry as many times as they want to, along with providing pre-tested images and constraints.

9.8 Successes

The flexibility we included in the Gantt charts was proven to be beneficial in the end. During the project's oral presentation, feedback highlighted anility in some areas which we were able to improve due to the grace periods we gave ourselves, this would not have been possible if we scheduled more tasks in these time periods, as we would have either not met our aims or have had incorrect statements.

Because of this, we were able to complete the project on time with no extensions needed for any submissions. Our risk management also meant we had extra time to implement more features than expected due to being ahead of schedule.

We found that our development methodology suited the project well as we were able to evenly distribute the workload throughout the year due to good organisation. Our chosen system architecture allowed for a feasible model that helped create a working prototype with all the required features.

10 Evaluation

We aimed to analyse the distribution of proteins in fluorescent microscopic images, and this was achieved through careful planning and evenly distributing the workload throughout the project period. We performed detailed research on the background noise and methods to analyse these images, this lead us to look at the region of interest in images and measure colocalisation.

We found strengths in having a detailed literature report, it allowed us to more easily create an application that suited the target audience, and helped to understand the potential to expand the project with more time. The Gantt charts tracked progress and ensured the completion of the application with sufficient testing and validation.

The only flaw in the project management was that planning did not account for report writing. However, the process and additional notes were constantly documented throughout the project, this made it easier to refer back to when report writing. Overall, the timeline was realistic and all milestones were achieved.

Below we summarise our objectives and how they were achieved.

Project Aim	**Prioritisation**	**Completion and results**
Investigate how the colocalization value of an image is affected by the background noise of that image.	MUST	Completed a literature review on previous work and extensive background research.
Use mathematical reasoning to justify conclusions surrounding the effects of background noise on colocalization	COULD	Hypothesised the effects and drafted a mathematical proof to justify the hypothesis.
Explore the different types of image segmentation on microscopic images	MUST	Research and explored algorithms in these segmentation techniques.
Implement seeded growing region as a method of automating the region of interest selection.	MUST	Used R to implement SGR.
Create an application from aim four.	SHOULD	Created a fully working application which can be found on the linked GitHub.
Given fluorescent microscopic images from blood samples, understand how blood clots can be detected using the region of interest algorithm.	COULD	Used scholarly research to investigate its possibilities.

We followed a waterfall integrated methodology, which was useful in the long run as we were able to successfully follow a step by step timeline. The only minor shift in focus was after research revealed existing packages to calculate the region of interest. We used this as motivation to create an application that was more easily accessible and performed better given the structures of interest.

Our biggest obstacle faced was understanding the effects of background noise on the colocalization value of an image, this was highlighted in a presentation we gave on this project prior to the submission

of this dissertation. With further research, we found our initial hypothesis "we assume that increasing the blank space in images increases the amount of space both colour channels are not in, therefore increasing the colocalization value" to be incorrect. More research was needed to find that this statement was only partially correct, the background noise of an image actually made the colocalization value close to zero than it should be. This was confirmed with research and simulations, we also reasoned this through a drafted mathematic proof, this is only a draft as other assumptions made are not in the proof.

It was challenging to work on applications for a different industry. Thorough research was required to understand new terminologies and the goals of our target audience were different to our usual audience. Moreover, there was a limitless opportunity to add new features to our application, however, we wanted to make it user friendly and ensure a working prototype was finished and tested. Essentially, this leaves the application open to enhancements by adding new features we mentioned in parts of our research. For example, calculating the colocalization of the ROI or other variations depending on its use.

10.1 Future Work and Application to Industry

Region of interest identification is commonly applied to the medical industry by identifying the volume or size of specific components in an image [13]. In this particular case, it could be used to examine a patient's blood sample to identify the size of blood clots and the severity. In the future, automation of this could improve the diagnostic procedure for identifying the severity of a patient's blood clotting or other medical conditions, especially important when hospitals are understaffed or very busy.

This would be from the time saved visually assessing countless images and giving quantitative reassurance for a diagnosis and administering medication. This is currently not possible due to the time taken to obtain fluorescent microscopic images and the high costs associated.

To further analyse proteins, we could also explore other methods of analysing protein distributions, for example tracking and mapping the changes in proteins under certain conditions to understand their function and predict their structure after certain chemical changes.

Our project focused on the region of interest in microscopic images, identifying the structure of a protein. However, the dataset was based on research seeing the effects of COVID-19 on blood clots in humans. Thus, with a larger dataset of images, we could quantitatively determine the association between blood clots and COVID-19 by comparing the difference in ROI of platelet cells of samples between the patients.

10.2 Author's Assessment of the Project

This project is considered an achievement because all goals were exceeded and the technical contribution involved an easy to use application that identified the region of interest which is suited to microscopic images of proteins, where we provided a detailed GitHub for peer review and collaboration.

The application is relevant to one's degree as it makes use of the core aspects of data science, the implementation involved algorithmic solving and statistical research to select our method. It is important in the biomedical industry as it can help in the diagnosis of medical conditions and understanding diseases in more detail.

11 Conclusion

The first steps of our project involved heavy research to understand the processes used to obtain the microscopic images, this gave insight into the types of shapes and structures we were interested in. After investigating the different methods used for image segmentation in the biomedical industry, we chose the seeded growing region as our most suitable method to find the region of interest.

When exploring these different methods, research showed region-based segmentation was more suitable than threshold-based segmentation for protein images as it was less sensitive to small non-similar objects [40]. Therefore, region-based segmentation would still identify the full shape of the cell whereas threshold-based segmentation was more likely to exclude parts of the cell where less protein was present.

Secondly, we were able to apply the region of interest to colocalization and find that background noise makes the colocalization value of an image closer to zero than it should be. This illustrates the unreliability in analysing these types of images without the structure being singled out from the background, resulting in non-valid results.

Moving on, we implemented seeded growing region into a web application that is user friendly and contains an appropriate number of features to still be both interactive and meet the user requirements. We included an informative GitHub repository to make the application easier to use and it also opens the opportunity for improvements or a basis to build on.

With more time, the application could be specialised to diagnose certain conditions using microscopic image samples. For example, by combing ROI with statistical analysis, there is the possibility of automating the diagnosis of severe blood clotting by comparing the ROI coverage to what it should normally be. For this to be implemented, more research and data are required. The purpose of mentioning this

was to emphasise the link from our dataset background to our application. As an extension, it can furthermore be useful in the medical industry for situations when hospitals are understaffed or doctors are unavailable and medical attention is desperately needed, e.g. the start of the COVID-19 pandemic.

There are still many limitations to the above. There still exists the time and monetary costs to obtain these types of images before we can efficiently apply them. Despite this, we have learned methods to efficiently analyse protein distributions and which were more favourable depending on the types of microscopic images, it taught us about the importance of the region of interest selection in which we were able to implement one specific method and apply it to the biomedical industry.

References

[1] Alberts, B. Johnson, A. Lewis, J. et al. Molecular Biology of the Cell, 4th edition, New York: Garland Science, Analyzing Protein Structure and Function. Published: 2002. Available from: https://www.ncbi.nlm.nih.gov/books/NBK26820/. Accessed: 30/11/2021.

[2] BSCB. Why cell biology is so important?. Published: N.d. Available from: https://bscb.org/learning-resources/softcell-e-learning/why-cell-biology-is-so-important/. Accessed: 30/11/2021.

[3] Agile Business Consortium. Moscow Prioritisation, Ch 10. Published: N.d. Available from: https://www.agilebusiness.org/page/ProjectFramework_10_MoSCoWPrioritisation#:~:text=MoSCoW%20. Accessed: 20/03/2022.

[4] its.ny.gov. System Requirements Analysis, PP 31. Published: N.d. Available from: https://its.ny.gov/sites/default/files/documents/systemreq.pdf. Accessed: 20/03/2022.

[5] Pretorius, E. Venter, C. Laubscher, G.J. et al. Prevalence of readily detected amyloid blood clots in 'unclotted' Type 2 Diabetes Mellitus and COVID-19 plasma: a preliminary report. Cardiovasc Diabetol. Published: 2020. Available from: https://doi.org/10.1186/s12933-020-01165-7. Accessed: 30/11/2021.

[6] Johns Hopkins Medicine. What Is Coronavirus?. Published: 2022. Available from: https://www.hopkinsmedicine.org/health/conditions-and-diseases/coronavirus. Accessed: 30/11/2021.

[7] Merschel, M. COVID-19 linked to risk of dangerous blood clots in stroke patients. Published: 2022. Available from: https://www.heart.org/en/news/2022/02/03/covid-19-linked-to-risk-of-dangerous-blood-clots-in-stroke-patients#:~:text=COVID%2D19%20linked%20to%20risk%20of%20dangerous%20blood%20clots%20in%20stroke%20patients,-By%20Michael%20Merschel&text=Older%20stroke%20patients%20who%20had,Disease%20Control%20and%20Prevention%20shows. Accessed: 30/11/2021.

[8] Michael, A. Richard, A. Manning and Lewis Practical Haematology (Twelfth Edition), Investigation of a Thrombotic Tendency. Published: 2017. Available from: https://www.sciencedirect.com/topics/immunology-and-

microbiology/platelet-poor-plasma. Accessed: 30/11/2021.

[9] MedlinePlus. C-Reactive Protein (CRP) Test. Published: N.d. Available from: https://medlineplus.gov/lab-tests/c-reactive-protein-crp-test/#:~:text=CRP%20is%20a%20protein%20made,the%20injured%20or%20affected%20area. Accessed: 10/01/2022.

[10] University Of California. UC Davis Study Identifies C-reactive Protein As Cause Of Blood Clot Formation. Published: 2003. Available from: www.sciencedaily.com/releases/2003/01/030113072324.htm. Accessed: 10/01/2022.

[11] Mahase, E. BMJ. Covid-19: What do we know about "long covid"?. Published: 2020. Available from: https://www.bmj.com/content/370/bmj.m2815.long. Accessed: 30/11/2021.

[12] Domingues, M. Carvalho, F. Santos, N. Nanomechanics of Blood Clot and Thrombus Formation. Published: 2022. Available from: https://www.annualreviews.org/doi/abs/10.1146/annurev-biophys-111821-072110. Accessed: 10/01/2022.

[13] Rudin, S. Bednarek, DR. Kezerashvili, M. Granger, WE. Serghany, JE. Guterman, LR. Hopkins, LN. Szymanski, B. Loftus, RJ. Clinical application of region-of-interest techniques to radiologic imaging, Radiographics. Published: 1996. Available from: https://pubmed.ncbi.nlm.nih.gov/8835978/. Accessed: 20/01/2022.

[14] Chang, JB. Gao, R. Chen, F. Light-Sheet Fluorescence Microscopy for Multiscale Biological Imaging. Published: 2021. Available from: https://www.sciencedirect.com/topics/medicine-and-dentistry/fluorescence-microscopy. Accessed: 20/01/2022.

[15] Sanderson, J. Smith, I. Parker, I, Bootman, M.Fluorescence Microscopy. Published: 2014. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4711767/. Accessed: 20/01/2022.

[16] Koenig, F. Fluorescence Microscopy vs. Light Microscopy. Published: 2020. Available from: https://microscopeinternational.com/fluorescence-vs-light-microscopy/#:~:text=Because%20traditional%20light%20microscopy%20uses,more%20detailed%20and%20reliable%20image. Accessed: 17/11/2021.

[17] Sanderson, MJ. Smith, I. Parker, I. Bootman, MD. Fluorescence microscopy, Cold Spring Harb Protoc. Published: 2014. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4711767/. Accessed: 17/11/2021.

[18] Cai, D.X. et al., Dynamic Perfusion Culture of Human Outgrowth Endothelial Progenitor Cells on Demineralized Bone Matrix In Vitro. Published: 2016. Available from: https://www.researchgate.net/publication/309543341_Dynamic_Perfusion_Culture_of_Human_Outgrowth_Endothelial_Progenitor_Cells_on_Demineralized_Bone_Matrix_In_Vitro. Accessed: 26/11/2021.

[19] Spring, K R. Colocalization of Fluorophores in Confocal Microscopy. Published: N.d. Available from: https://www.olympus-lifescience.com/en/microscope-resource/primer/techniques/confocal/applications/colocalization/. Accessed: 26/11/2021.

[20] Adler, J. Parmryd, I. Quantifying colocalization by correlation: the Pearson correlation coefficient is superior to the Mander's overlap coefficient. Published: N.d. Available from: https://pubmed.ncbi.nlm.nih.gov/20653013/#:~:text=Abstract,perceived%20problems%20with%20the%20PCC. Accessed: 17/11/2021.

[21] Glen, S. Correlation Coefficient: Simple Definition, Formula, Easy Steps. Published: N.d. Available from: https://www.statisticshowto.com/probability-and-statistics/correlation-coefficient-formula/. Accessed: 17/11/2021.

[22] ImageJ, et.al. What is colocalization?. Published: N.d. Available from: https://imagej.net/imaging/colocalization-analysis. Accessed: 17/11/2021.

[23] Adler, J. Parmryd, I. Quantifying colocalization by correlation: the Pearson correlation coefficient is superior to the Mander's overlap coefficient. Published: 2010. Available from: ttps://pubmed.ncbi.nlm.nih.gov/20653013/#:~:text=Abstract,perceived%20problems%20with%20the%20PCC. Accessed: 19/11/2021.

[24] Manders, EMM. Verbeek, F.J. Aten, J.A. Measurement of co-localization of objects in dual-colour confocal images. Published: 1993. Available from: https://www.premium.fm.usp.br/download/Manders_Eventos_de_Colocalizacao.

pdf. Accessed: 19/11/2021.

[25] Adler, J. Parmryd, I. Quantifying colocalization: The case for discarding the Manders overlap coefficient. Published: 2021. Available from: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cyto.a.24336. Accessed: 21/11/2021.

[26] Gavrilovic, M. Wahlby, C. Quantification and Localization of Colocalization. Published: N.d. Available from: https://www.diva-portal.org/smash/get/diva2:40219/FULLTEXT01.pdf. Accessed: 21/11/2021.

[27] Dunn, K. Kamocka, M. McDonald, J. A practical guide to evaluating colocalization in biological microscopy. Published: 2011. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3074624/. Accessed: 21/11/2021.

[28] Huang, L.K. Wang, M.J.J. Image thresholding by minimizing the measures of fuzziness. Pattern recognition, 28(1):41–51. Published: 1995. Accessed: 21/11/2021.

[29] Rosenfeld, A. Pal, S.K. Image enhancement and thresholding by optimization of fuzzy compactness. Published: 1987. Available from: https://www.cb.uu.se/~joakim/course/fuzzy/vt10/CE1/PalRosenfeld.pdf. Accessed: 21/11/2021.

[30] Fan, Minjie, Lee, Thomas. Variants of Seeded Region Growing. IET Image Processing. Published: 2015. Available from: https://www.researchgate.net/publication/269338276_Variants_of_Seeded_Region_Growing#:~:text=Seeded%20region%20growing%20(SRG)%20is,adding%20neighboring%20pixels%20to%20them. Accessed: 22/11/2021.

[31] About Colour Blindness. Published: N.d., Available from: https://www.colourblindawareness.org/colour-blindness/#:~:text=Colour%20Blindness,-What%20is%20colour&text=Colour%20(color)%20blindness%20. Accessed: 21/04/2022.

[32] Rai, G.N. Gradient-Based Seeded Region Grow method for CT Angiographic Image Segmentation. Published: 2010, Available from: https://arxiv.org/pdf/1001.3735.pdf. Accessed: 10/03/2022.

[33] Wrike. What Is Agile Methodology in Project Management?. Pub-

lished: N.d. Available from: https://www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management/. Accessed: 21/04/2022.

[34] Structural Biology. Why Structure Protein Prediction Matters. Published: 2020. Available from: https://www.dnastar.com/blog/structural-biology/why-structure-prediction-matters/#:~:text=Having%20a%20protein%20structure%20provides,the%20intent%20of%20changing%20function. Accessed: 20/01/2022.

[35] MRC. Medical Imaging. Published: N.d. Available from: https://www.ukri.org/what-we-offer/browse-our-areas-of-investment-and-support/medical-imaging/#:~:text=Medical%20imaging%20is%20used%20in,magnetic%20resonance%20imaging%20(MRI). Accessed: 20/01/2022.

[36] Smith, Y. Pharm, B. Protein Structure and Function. Published: 2018. Available from: https://www.news-medical.net/life-sciences/Protein-Structure-and-Function.aspx#:~:text=The%20structure%20of%20protein%20sets,%2C%20therefore%2C%20determines%20its%20function. Accessed: 19/11/2021.

[38] Weebly, P. How does shape affect protein structure and function?. Published: 2015. Available from: https://socratic.org/questions/how-does-shape-affect-protein-structure-and-function. Accessed: 19/11/2021.

[37] Seladi-Schulman, J. What to Know About COVID-19 and Blood Clots. Published: 2020. Available from: https://www.healthline.com/health/coronavirus-and-blood-clots. Accessed: 19/11/2021.

[39] Davis, JP. Shettigar, V. Tikunova, SB. Little, SC. Liu, B. Siddiqui, JK. Janssen, PML. Ziolo, MT. Walton, SD. Designing proteins to combat disease: Cardiac troponin C as an example, Archives of Biochemistry and Biophysics. Published: 2016. Available from: https://www.sciencedirect.com/science/article/pii/S0003986116300261. Accessed: 19/11/2021.

[40] Mukherjee, S. Acton, ST. Region-Based Segmentation in Presence of Intensity Inhomogeneity Using Legendre Polynomials, IEEE Signal Processing Letters, vol. 22, no. 3, pp. 298-302. Published: 2015, Available from: https://ieeexplore.ieee.org/abstract/document/6874542. Accessed: 21/04/2022.

[41] Katsoularis, I. Fonseca-RodrÃguez, O. Farrington, P. Jerndal, H. Lundevaller, EH, Sund, M. et al. Risks of deep vein thrombosis, pulmonary embolism, and bleeding after covid-19: nationwide self-controlled cases series and matched cohort study. Published: 2022. Available from: https://www.bmj.com/content/377/bmj-2021-069590. Accessed: 20/01/2022.

[42] Schadler, K. An Introduction to the Light Microscope, Light Microscopy Techniques and Applications. Published: 2021. Available from: https://www.technologynetworks.com/analysis/articles/an-introduction-to-the-light-microscope-light-microscopy-techniques-and-applications-351924. Accessed: 20/11/2021.

[43] Creative Commons. Attribution 4.0 International. Published: N.d. Available from: https://creativecommons.org/licenses/by/4.0/. Accessed: 17/01/2022.

[44] Markovic, I. How to use the risk assessment matrix to organize your project better. Published: 2019. Available from: https://tms-outsource.com/blog/posts/risk-assessment-matrix/. Accessed: 21/04/2022.

[45] Lucid Chart. The Pros and Cons of Waterfall Methodology. Published: N.d. Available from: https://www.lucidchart.com/blog/pros-and-cons-of-waterfall-methodology. Accessed: 21/04/2022.

[46] Baddelet, A. Turner, R. Rubak, E. spatstat: Spatial Point Pattern Analysis, Model-Fitting, Simulation, Tests. Published: 2022. Available from: https://cran.r-project.org/web/packages/spatstat/index.html. Accessed: 15/04/2022.

[47] Lachmanovich, E. Shvartsman, DE. Malka, Y. Botvin, C. Henis, I. Weiss, AM. Co-localization analysis of complex formation among membrane proteins by computerized fluorescence microscopy: application to immunofluorescence co-patching studies. Published:2003. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1046/j.1365-2818.2003.01239.x. Accessed: 20/11/2021.

[48] Lagache, T. Grassart, A. Dallongeville, S. et al. Mapping molecular assemblies with fluorescence microscopy and object-based spatial statistics. Published: 2018. Available from: https://doi.org/10.1038/s41467-018-03053-x. Accessed: 20/11/2021.

[49] Elangovan, M. Day, RN. Periasamy, A. Nanosecond fluorescence resonance energy transfer-fluorescence lifetime imaging microscopy to localize the protein interactions in a single living cell. Published: 2002. Available from: https://doi.org/10.1046/j.0022-2720.2001.00984.x. Accessed: 20/22/2021.

[50] DePristo, MA. Lynne Chang, L. Vale, RD. Khan, SM. Lipkow, K. Introducing simulated cellular architecture to the quantitative analysis of fluorescent microscopy, Progress in Biophysics and Molecular Biology, Volume 100. Published: 2009. Available from: https://doi.org/10.1016/j.pbiomolbio.2009.07.002. Accessed: 20/22/2021.

A Mathematical Workings

[expansion 1]

r_{nom} (2)

$= \sum_{i\in I}^{n+1} R_i G_i - \overline{R}y - \overline{G}x + \sum_{i\in I}^{n+1} \overline{RG}$ (5), (6)

$= \sum_{i\in I}^{n+1} R_i G_i - \overline{R}y - \overline{G}x + (n+1)\overline{RG}$ (summing over constant)

$= \sum_{i\in I}^{n+1} R_i G_i - \frac{n}{n+1}\overline{R}_0 y - \frac{n}{n+1}\overline{G}_0 x + (n+1)\frac{n}{n+1}\overline{R}_0\frac{n}{n+1}\overline{G}_0$ (8)
$= \sum_{i\in I}^{n+1} R_i G_i + \frac{n}{n+1}(-\overline{R}_0 y - \overline{G}_0 x + n\overline{R}_0\overline{G}_0)$

$= \sum_{i\in I}^{n+1} R_i G_i + \frac{n}{n+1}(\widetilde{r}_{nom} - \sum_{i\in I_{/0}}^{n} R_i G_i)$ (2)

$= \frac{\sum_{i\in I}^{n+1} R_i G_i}{n+1} + \frac{n}{n+1}\widetilde{r}_{nom}$ (5)

[expansion 2]

$r_{nom} \leq \widetilde{r}_{nom} \iff$

$\frac{\sum_{i\in I}^{n+1} R_i G_i}{n+1} + \frac{n}{n+1}\widetilde{r}_{nom} \leq \widetilde{r}_{nom} \iff$, (9)

$\frac{\sum_{i\in I}^{n+1} R_i G_i}{n+1} \leq \frac{\widetilde{r}_{nom}}{n+1} \iff$

$\sum_{i\in I}^{n+1} R_i G_i \leq \widetilde{r}_{nom} \iff$, (2)

$\sum_{i\in I}^{n+1} R_i G_i \leq \sum_{i\in I_{/0}}^{n} R_i G_i - \overline{R}_0 \sum_{i\in I_{/0}}^{n} G_i - \overline{G}_0 \sum_{i\in I_{/0}}^{n} R_i + \sum_{i\in I_{/0}}^{n} \overline{R}_0\overline{G}_0 \iff$, (4)

$0 \leq -\overline{R}_0 \sum_{i\in I_{/0}}^{n} G_i - \overline{G}_0 \sum_{i\in I_{/0}}^{n} R_i + \sum_{i\in I_{/0}}^{n} \overline{R}_0\overline{G}_0 \iff$, (7)

$0 \leq -\overline{R}_0 \frac{1}{n}\overline{G}_0 - \overline{G}_0 \frac{1}{n}\overline{R}_0 + n\overline{R}_0\overline{G}_0 \iff$

$0 \leq \overline{R}_0\overline{G}_0(n-2)$

[expansion 3]

For $i \in I_{/0}$:

$\sum^n R_i^2 - 2\overline{R}_0 \sum^n R_i + \sum^n \overline{R}_0^2$

$= \sum^n R_i^2 - 2\overline{R}_0 x + n\overline{R}_0^2$, (5)

$= \sum^n R_i^2 - 2n\overline{R}_0^2 + n\overline{R}_0^2$, (7)

$= \sum^n R_i^2 - n\overline{R}_0^2$

For $i \in I$:

$\sum^{n+1} R_i^2 - 2\overline{R} \sum^{n+1} R_i + \sum^{n+1} \overline{R}^2$

$= \sum^{n+1} R_i^2 - 2\frac{n}{n+1}\overline{R}_0 x + (n+1)(\frac{n}{n+1})^2 \overline{R}_0^2$, (5) ,(7)

$= \sum^{n+1} R_i^2 - \frac{2n^2\overline{R}_0^2 + n^2\overline{R}_0^2}{n+1}$

$= \sum^{n+1} R_i^2 - \frac{n^2\overline{R}_0^2}{n+1}$

Therefore

$\sum_{i\in I}(R_i - \overline{R})^2 = \sum^n R_i^2 + \frac{n}{n+1}(\sum_{i\in I_{/0}}(R_i - \overline{R})^2 - \sum^n R_i^2)$

$= \frac{\sum_{i\in I} R_i^2}{n+1} + \frac{n}{n+1}\sum_{i\in I_{/0}}(R_i - \overline{R})^2$

And likewise for G.

[expansion 4]

$(r_{denom})^2 = (\frac{\sum_{i\in I} R_i^2}{n+1} + \frac{n}{n+1}\sum_{i\in I_{/0}}(R_i - \overline{R})^2)(\frac{\sum_{i\in I} G_i^2}{n+1} + \frac{n}{n+1}\sum_{i\in I_{/0}}(G_i - \overline{G})^2)$, (10)

$= \frac{1}{(n+1)^2}(\sum_{i\in I} R_i^2 + n\sum_{i\in I_{/0}}(R_i - \overline{R})^2)(\sum_{i\in I} G_i^2 + n\sum_{i\in I_{/0}}(G_i - \overline{G})^2)$

$= \frac{1}{(n+1)^2}(\sum_{i\in I} R_i^2 \sum_{i\in I} G_i^2 + n(\sum_{i\in I_{/0}}(R_i - \overline{R})^2 + \sum_{i\in I_{/0}}(G_i - \overline{G})^2) + n^2 \sum_{i\in I_{/0}}(R_i - \overline{R})^2 \sum_{i\in I_{/0}}(G_i - \overline{G})^2)$

$= \frac{1}{(n+1)^2}(\sum_{i\in I} R_i^2 \sum_{i\in I} G_i^2 + n(\sum_{i\in I_{/0}}(R_i - \overline{R})^2 \sum_{i\in I} G_i^2 + \sum_{i\in I_{/0}}(G_i - \overline{G})^2 \sum_{i\in I} R_i^2) +$

$n^2(\widetilde{r}_{denom})^2)$, [[expansion 5]]

$= \frac{1}{(n+1)^2}(\sum_{i\in I} R_i^2 \sum_{i\in I} G_i^2 + n(2(\widetilde{r}_{denom})^2 + c) + n^2(\widetilde{r}_{denom})^2)$

$= \frac{1}{(n+1)^2}(n^2(\widetilde{r}_{denom})^2) + 2n(\widetilde{r}_{denom})^2 + 1 + cn + \sum_{i\in I} R_i^2 \sum_{i\in I} G_i^2 - 1)$

$= \frac{1}{(n+1)^2}(n^2(\widetilde{r}_{denom})^2) + 2n(\widetilde{r}_{denom})^2 + 1) + \frac{1}{(n+1)^2}(cn + \sum_{i\in I} R_i^2 \sum_{i\in I} G_i^2 - 1))$

$= (\widetilde{r}_{denom})^2 + \frac{1}{(n+1)^2}(cn + \sum_{i\in I} R_i^2 \sum_{i\in I} G_i^2 - 1)$

$(r_{denom})^2 \geq (\widetilde{r}_{denom})^2 \iff$

$(\widetilde{r}_{denom})^2 + \frac{1}{(n+1)^2}(cn + \sum_{i\in I} R_i^2 \sum_{i\in I} G_i^2 - 1) \geq (\widetilde{r}_{denom})^2 \iff$

$\frac{1}{(n+1)^2}(cn + \sum_{i\in I} R_i^2 \sum_{i\in I} G_i^2 - 1) \geq 0$

$cn + \sum_{i\in I} R_i^2 \sum_{i\in I} G_i^2 - 1 \geq 0$

[expansion 5]

$\sum_{i\in I_{/0}}(R_i - \overline{R})^2 \sum_{i\in I} G_i^2 =$
$\sum_{i\in I_{/0}}(R_i^2 - 2\overline{R}R_i + \overline{R}^2) \sum_{i\in I} G_i^2 =$
$(\sum_{i\in I_{/0}} R_i^2 - 2\overline{R} \sum_{i\in I_{/0}} R_i + n\overline{R}^2) \sum_{i\in I} G_i^2 =$
$\sum_{i\in I_{/0}} R_i^2 \sum_{i\in I} G_i^2 - 2\overline{R} \sum_{i\in I_{/0}} R_i \sum_{i\in I} G_i^2 + n\overline{R}^2 \sum_{i\in I} G_i^2$

And

$\sum_{i\in I_{/0}}(G_i - \overline{G})^2 \sum_{i\in I} R_i^2 =$
$\sum_{i\in I_{/0}}(G_i^2 - 2\overline{G}G_i + \overline{G}^2) \sum_{i\in I} R_i^2 =$
$(\sum_{i\in I_{/0}} G_i^2 - 2\overline{G} \sum_{i\in I_{/0}} G_i + n\overline{G}^2) \sum_{i\in I} R_i^2 =$
$\sum_{i\in I_{/0}} G_i^2 \sum_{i\in I} R_i^2 - 2\overline{G} \sum_{i\in I_{/0}} G_i \sum_{i\in I} R_i^2 + n\overline{G}^2 \sum_{i\in I} R_i^2$

Therefore

$\sum_{i\in I_{/0}}(R_i - \overline{R})^2 \sum_{i\in I} G_i^2 + \sum_{i\in I_{/0}}(G_i - \overline{G})^2 \sum_{i\in I} R_i^2 =$
$\sum_{i\in I_{/0}} R_i^2 \sum_{i\in I} G_i^2 - 2\overline{R} \sum_{i\in I_{/0}} R_i \sum_{i\in I} G_i^2 + n\overline{R}^2 \sum_{i\in I} G_i^2 + \sum_{i\in I_{/0}} G_i^2 \sum_{i\in I} R_i^2 -$
$2\overline{G} \sum_{i\in I_{/0}} G_i \sum_{i\in I} R_i^2 + n\overline{G}^2 \sum_{i\in I} R_i^2$, (4)

$= 2\sum_{i\in I_{/0}} R_i^2 \sum_{i\in I} G_i^2 + \overline{R}^2 \sum_{i\in I} G_i^2 (n - 2\sum_{i\in I} R_i^2) + \overline{G}^2 \sum_{i\in I} R_i^2 (n - 2\sum_{i\in I} G_i^2)$
$= 2(\widetilde{r}_{denom})^2 + \overline{R}^2 \sum_{i\in I} G_i^2 (n - 2\sum_{i\in I} R_i^2) + \overline{G}^2 \sum_{i\in I} R_i^2 (n - 2\sum_{i\in I} G_i^2).$
$= 2(\widetilde{r}_{denom})^2 + c$
where $c = \overline{R}^2 \sum_{i\in I} G_i^2 (n - 2\sum_{i\in I} R_i^2) + \overline{G}^2 \sum_{i\in I} R_i^2 (n - 2\sum_{i\in I} G_i^2)$

B Gantt Charts

Figure B.1: Gantt Chart: Term 1

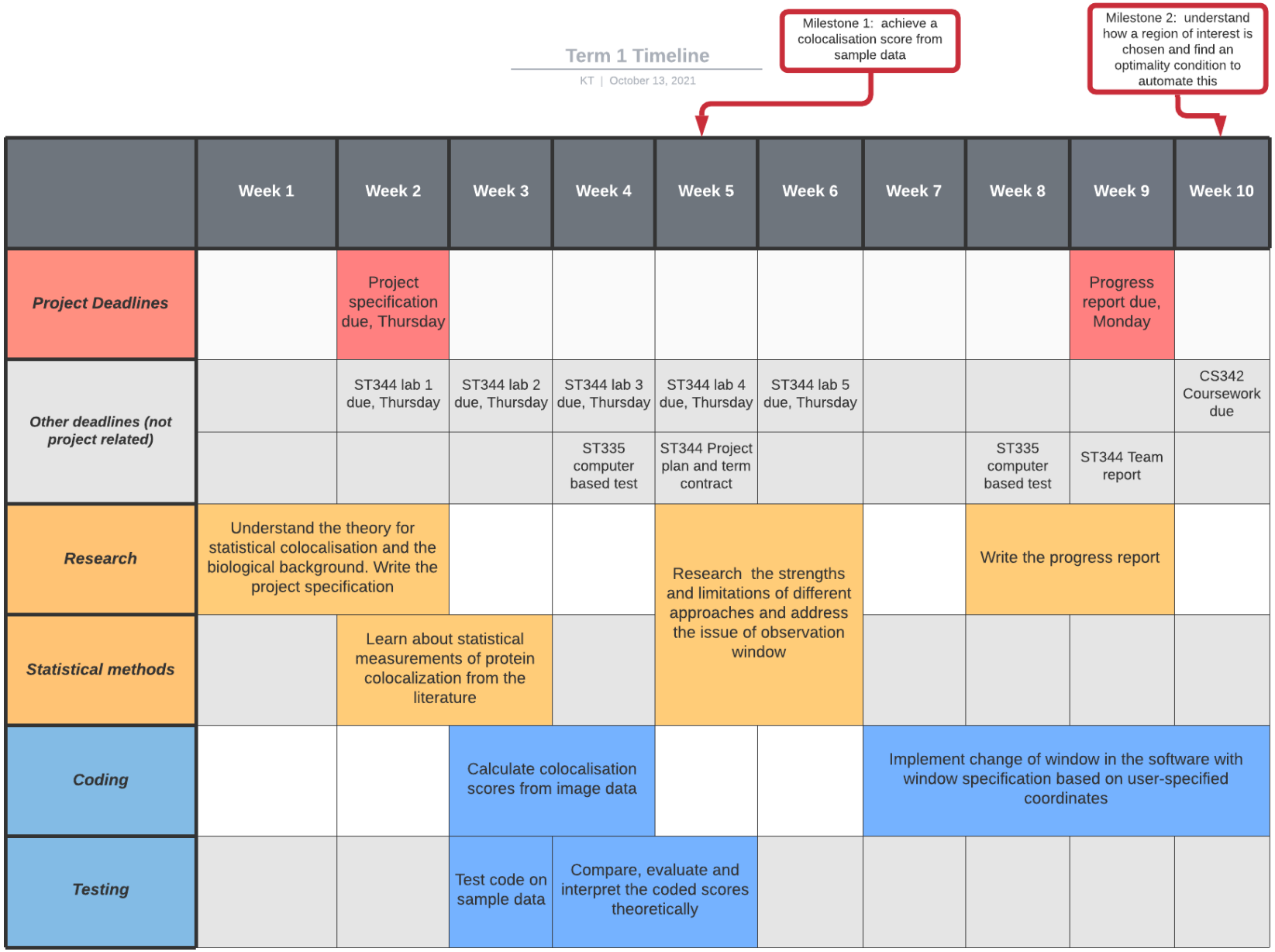


Figure B.2: Gantt Chart: Term 2