Developing a High-Throughput Image Analysis Tool

Gerd Bizi
Principal Investigator

Peter Kim
Faculty Advisor

September 2022

University of Toronto, Department of Biochemistry
SickKids, Department of Cell Biology
Laidlaw Scholars Leadership and Research Programme

I. Acknowledgements

I would like to give many thanks to my faculty advisor, Dr. Peter Kim, for providing me with excellent support, feedback, and guidance. This project would not have been possible without his unwavering dedication to helping me succeed.

When Dr. Kim wasn't helping me progress my project, I was lucky enough to have received mentorship from Victoria Riccio of the Kim Lab. Victoria taught me how to be a better biochemist during my time at the Kim Lab, and always pushed the envelope forward in how I could think regarding experimental technique. I am forever grateful to have had her as a mentor.

Next, I'd like to thank Dr. Miriam Ginzberg and Dr. Ran Kafri of the Kafri Lab at SickKids. Dr. Ginzberg assisted me with learning how to use MATLAB and how to do basic cell segmentation, an indispensable component of this project. They truly elevated the multi-disciplinary aspect of my project.

I'd like to also thank every member of the Kim Lab for assisting me in day-to-day tasks, providing me with feedback at lab meetings, and for helping me develop as a young scientist.

Shraddha Prasad, the Laidlaw Scholars Programme Coordinator at the University of Toronto, has been of immense help during my time during research. Shraddha was instrumental in the running of the program over the summer, and I am eternally grateful for her selecting me to be a Laidlaw Scholar.

This leads me to also express my gratitude for the Laidlaw Scholars Foundation. Through the generosity of Lord Laidlaw, I was able to undertake novel research, and I will forever cherish this moment in my life.

Finally, my family and friends deserve their flowers as well. Whether it be sparring with my good friend about how to implement a certain algorithm for my program, or explaining what a peroxisome was to my grandmother, my solid support network allowed me to seamlessly move from conducting research to pursuing my hobbies and generally having fun during the busiest summer of my life.

II. Introduction

In biological research, the use of computer analysis has increased immensely for its ability to analyze large data sets efficiently, economically, and most importantly, correctly. Computer analysis, through implementations such as AI, is primarily used in bioinformatics for genome studies. In recent years, however, it has gained traction for its use in image analysis, correlating with the improvements in image processing capabilities.

In cell biology research, a common way of analyzing potential interactions and presence of various macromolecules is through various types of imaging. A common example is through fluorescence, where a confocal fluorescence microscope excites fluorescent protein tags attached to targets of interest in order to visualize them in three dimensions. Images can be readily taken, with a variety of factors such as the intensity of the laser that excites the protein tags, the gain of the image, and the 'depth' of the images (which corresponds to how many images are taken in the Z-dimension and how far apart the images are).

The relative ease of image acquisition does not typically match the difficulty of image analysis. Often, and especially in the context of this paper, image analysis consists of manually drawing out the regions of interest (ROI) to specifically analyze an object like a cell. While seemingly trivial, this becomes a large time-sink when a dataset of thousands of cells need to be analyzed.

Thus, to reduce the amount of time doing menial image analysis, computational techniques have been introduced to automate and to remove bias from image analysis.

During the first half of the summer, I was studying the translocation of protein between the mitochondria and peroxisome. The mitochondria is an organelle that is involved in metabolic pathways that lead to the production of ATP, a molecule that provides energy for most cellular processes. The peroxisome performs tasks such as the oxidation of very long and branched-chain fatty acids, which are essential to the body. For example, the myelin sheath, the protective outer membrane for neurons, brain cells, is composed of fatty acids. Problematic peroxisome function can damage the myelin sheath, leading to adrenoleukodystrophy, a severe brain disorder. Peroxisomes also contain reactive oxidative species such as hydrogen peroxide which oxidizes and destroys cell waste.

The localization results of these experiments were analyzed using Perkim-Elmer Volocity, an image analysis software that uses experimenter-drawn ROIs to analyze cells.

For my research as a part of the Laidlaw Scholars Program, I began to create my **High-Throughout Image Processor for Colocalization**. This program aims to automate the colocalization analysis of different cell markers.

III. Background Knowledge and Terminology - Biochemistry

My experiment during the summer was two analyze the localization of two mitochondrial proteins, OCIAD1 and Bcl-Rambo, to the peroxisome.

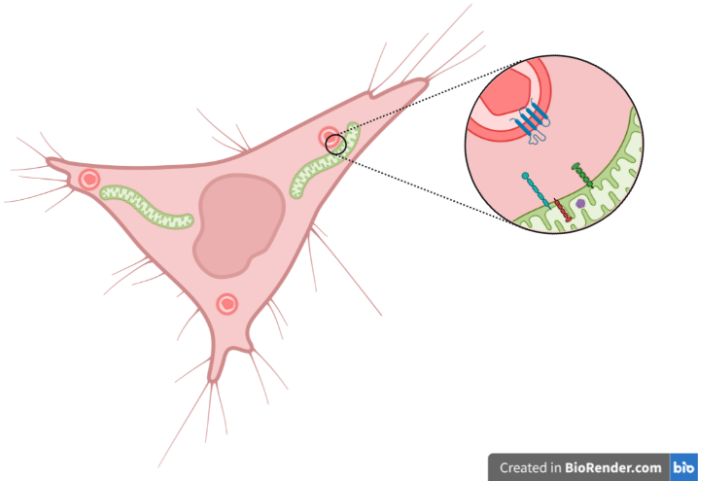
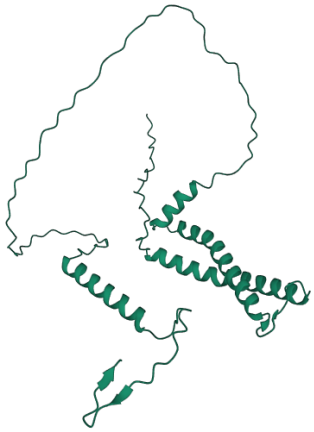
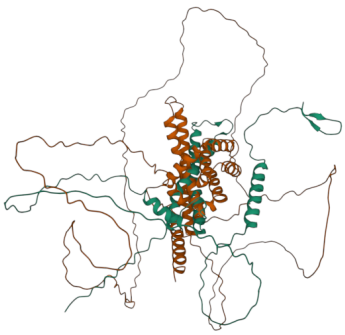


Figure 1: Schematic of HeLa cell and one of it mitochondria and peroxisomes under basal conditions. The blue multi-pass protein is PMP70, the red is OCIAD1, the purple is HSP60, and the other two are OMP25 and Bcl-Rambo.



OCIAD1 (Ovarian Cancer Immunoreactive Antigen Domain Containing 1) is an inner mitochondrial-membrane protein that is responsible for the integration of the catalytic subunit, CYC1, into CIII of the mitochondrial electron transport chain (mETC). This protein is essential in ensuring the proper flow of electrons in the mETC, which is the primary source of ATP production in the body. (Vasseur *et al*, 2021)

Figure 2: AlphaFold's predicted ribbon structure of OCIAD1 (AlphaFold, 2022; PDB Bank, 2022)



Bcl-Rambo is a mitochondrial pro-apoptotic, cell-death-inducing, protein. Through a mechanism that remains to be elucidated, Bcl-Rambo releases a member of the mETC chain, cytochrome C, which causes rapid cell death. (Kataoka, 2001)

Figure 3: AlphaFold's predicted ribbon structure of OCIAD1 (AlphaFold, 2022; PDB Bank, 2022)

Bio-ID is a proximity-based assay that can reveal protein interactions in an *in-vitro* setting. It does this by having a protein of interest (PoI) bonded to a BirA molecule. When in

close contact to other proteins, it causes the biotinylation of close proteins, which can be used to quantify interactions.

Localization describes where a protein goes in a cell. Dual-localization means that protein is going to two different places of significance. Co-localization describes the presence of two proteins in the same general vicinity. This shows as an overlap of signals in imaging. An example of this dual-localization and colocalization is between PEX2 and USP30. PEX2 is a peroxisomal signalling protein responsible for adding ubiquitin, a protein marker, to the peroxisome, which signals it for pexophagy, destruction of the organelle. Conversely, USP30 is a deubiquitinase protein that was originally thought to exist only on the mitochondria to remove ubiquitin. It was found that USP30 also dual-localizes in small numbers to peroxisomes, showing that on peroxisomes, USP30 and PEX2 co-localize and function in equilibrium so to maintain basal peroxisome count and quality. (Riccio, 2019)

An immunofluorescence experiment is where certain certain targets in cells are visualized using fluorescent microscopy. Two common approaches to visualizing protein are to either use specific antibody staining or to transfect cells with a construct containing a fluorescent construct, such as green fluorescent protein (GFP). With staining, a typical procedure involves using a primary antibody to bind to the protein of interest, while a secondary antibody containing the fluorophore binds to the primary antibody. With exogenous expression, DNA is transfected into live cells, and the its protein construction machinery use the DNA to produce the fluorescent protein which can be imaged.

Under an appropriate microscope, images can be taken where certain wavelengths of light specifically excite their respective fluorescent tags. The resultant fluorescence is then captured in an image at a set resolution. Multiple different tags can be imaged in a single photo, allowing for the analysis of protein proximity which can suggest possible protein interaction in spatial dimensions.

IV. Introduction to Wet Lab

In a BioID experiment conducted at the Kim Lab, both OCIAD1 and Bcl-Rambo were key hits for multiple key peroxisomal proteins, suggesting that these proteins might be dual-localizing to both the mitochondria and peroxisome.

Table 1: Kim Lab BioID Results

Gene	PMP Import	E3 Ubiquitin Ligase	PTS I Import	PTS II Import	Peroxisomal AAA ATPase	Misc. PMPs	Peroxisome Divison Machinery	DUBs	MCS
OCIAD1	PEX3 PEX19	PEX2 PEX10 PEX12	PEX13 PEX14	—	—	ABCD PMP70 PMP34	PEX11	USP30	ACBD
Bcl-Rambo	—	—	PEX14	—	—	—	—	USP30	ACDB

The results of this Bio-ID readout provided the basis for the experiment. Thus, the question became: **do OCIAD1 and Bcl-Rambo interact with the peroxisome and what is their mechanism of interaction?**

To investigate, a colocalization assay was performed, looking at the colocalization of these protein with mitochondrial and peroxisomal protein marker, HSP60 or OMP25 and PMP70, respectively.

V. Wet Lab Methodology

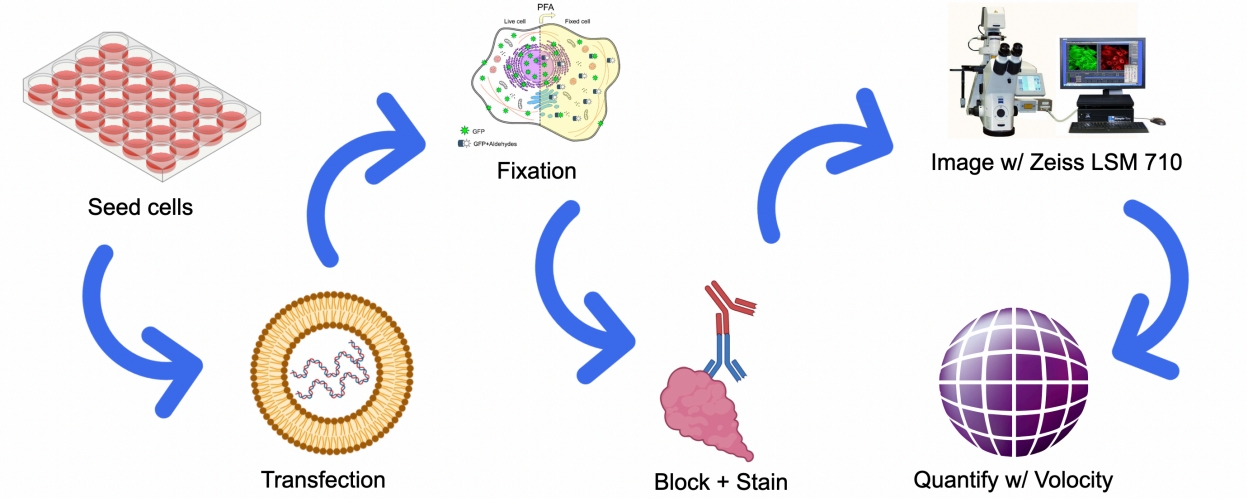


Figure 4: Flowchart describing the wet lab methodology

The experiment's setup is summarized in the table below for a total of 6 test conditions and two negative control conditions.

Table 2: Conditions for OCIAD1 and Bcl-Rambo Fluorescent Imaging Assay

	OCIAD1 Stain	OCIAD1 Overexpression	Bcl-Rambo Overexpression	Negative Control for Staining	Negative Control for Overexpression
Mitochondrial Marker	α-OCIAD1 α-HSP60	OCIAD1-GFP α-HSP60	GFP-Bcl-Rambo α-HSP60	α-HSP60 α-PMP70	OMP25-GFP α-PMP70
Peroxisomal Marker	α-OCIAD1 α-PMP70	OCIAD1-GFP α-PMP70	GFP-Bcl-Rambo α-PMP70		

VI. Wet Lab Results

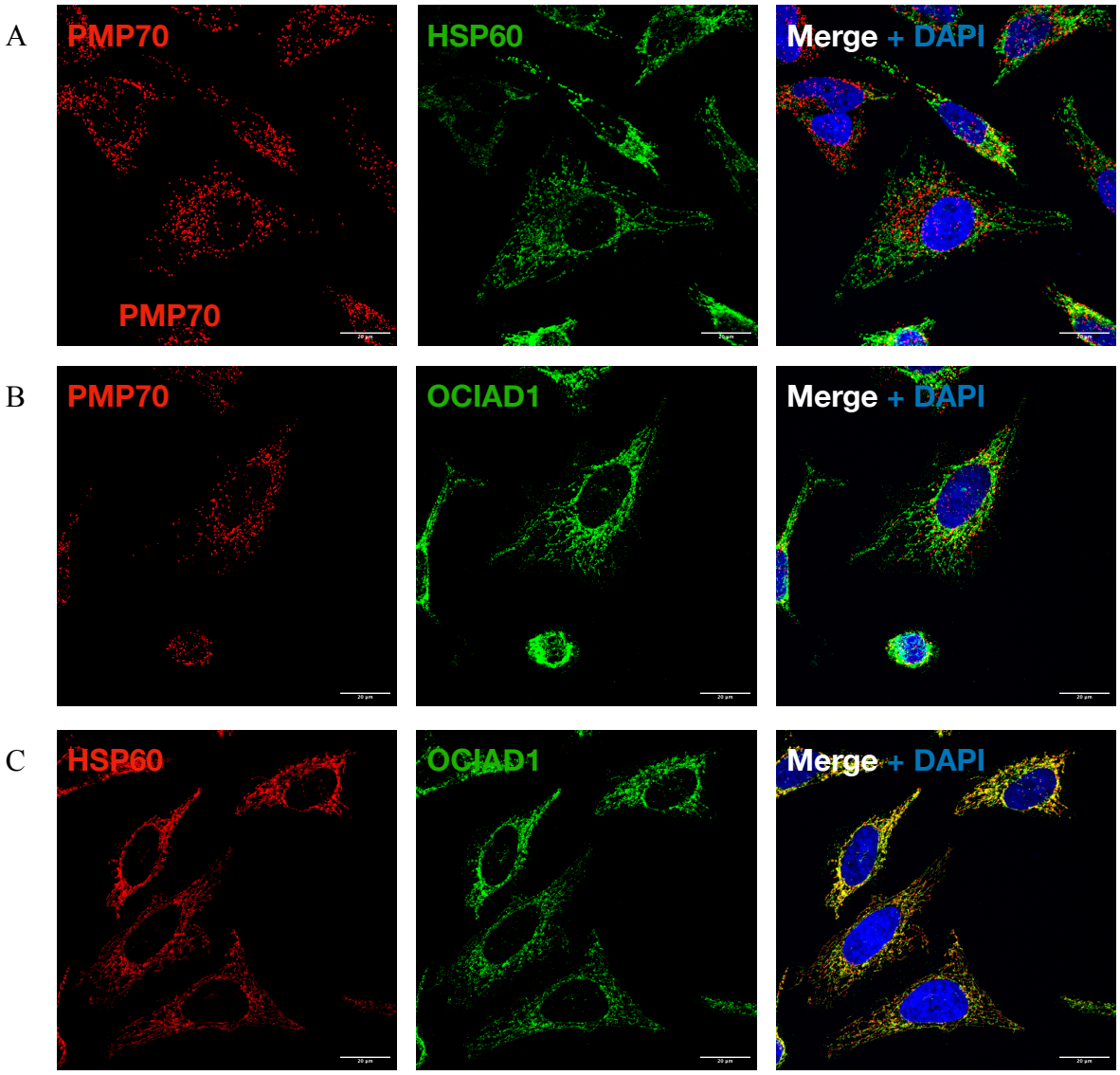
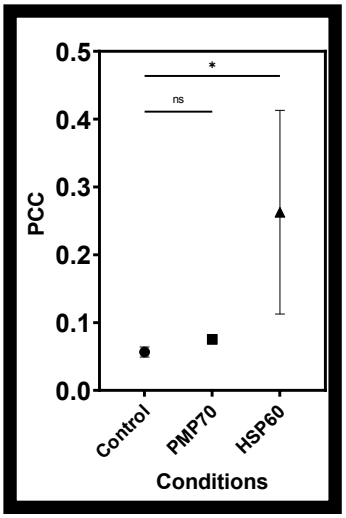


Figure 5: Representative images of OCIAD1 stain experiment. A is the negative control, HSP60 and PMP70. B shows the colocalization of PMP70 and OCIAD1. C shows the colocalization of HSP60 and OCIAD1. The graph on the right compares the Pearson's Correlation Coefficient, a measure of correlation in the data. ANOVA test w/ Dunnett's used.



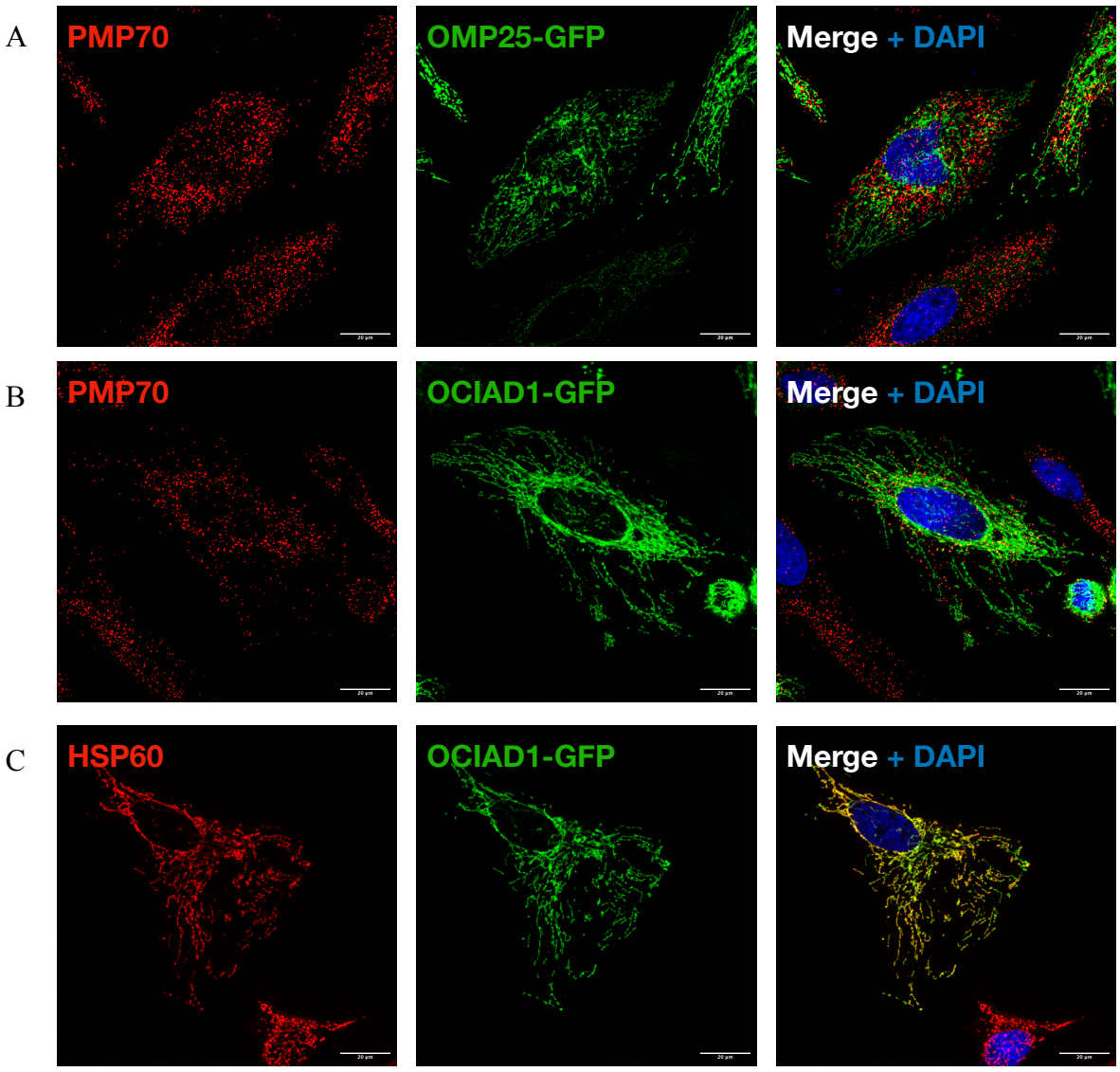
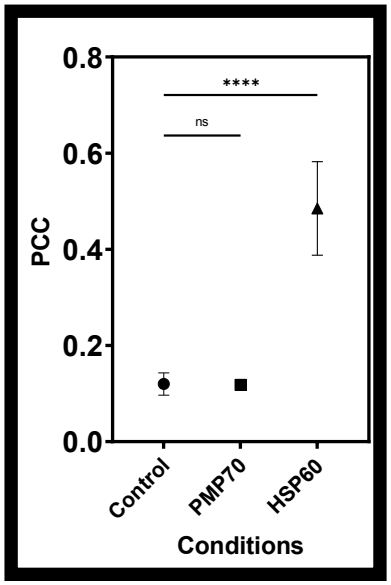


Figure 6: Representative images of OCIAD1 overexpression experiment. A is the negative control, OMP25-GFP and PMP70. B shows the colocalization of PMP70 and OCIAD1-GFP. C shows the colocalization of HSP60 and OCIAD1-GFP. The graph on the right compares the Pearson's Correlation Coefficient. ANOVA test w/ Dunnett's used.



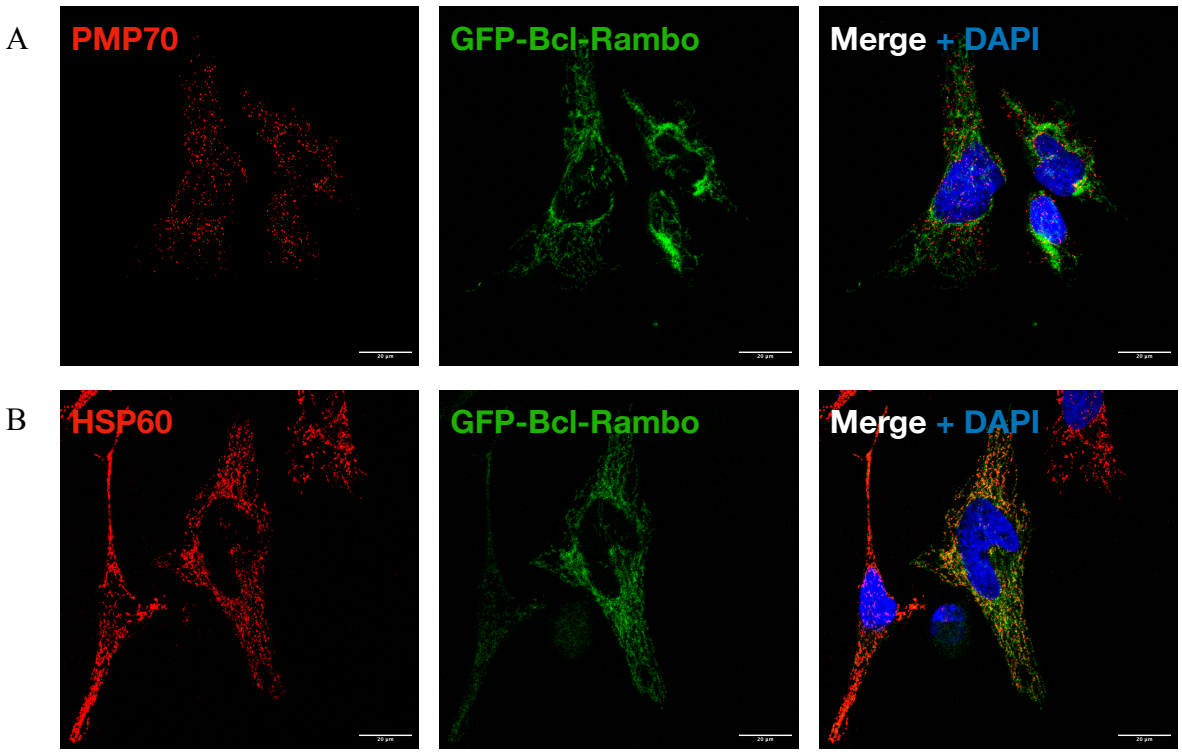
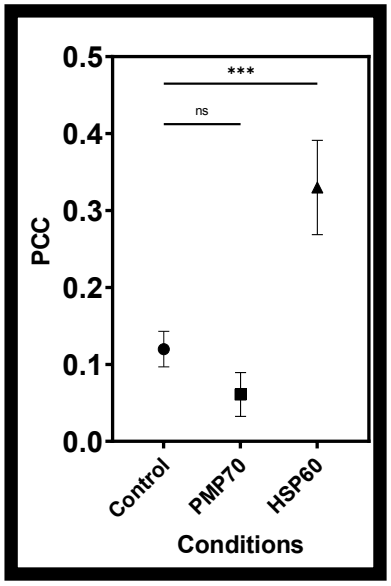


Figure 7: Representative images of GFP-Bcl-Rambo overexpression experiment. This condition uses the same negative control described in figure 6. A shows the colocalization of PMP70 and GFP-Bcl-Rambo. B shows the colocalization of HSP60 and GFP-Bcl-Rambo. The graph on the right compares the Pearson's Correlation Coefficient. ANOVA test w/ Dunnett's used.



VII. Wet Lab Discussions and Conclusion

Statistical analyses of the data showed that there was no significant difference in PCC comparing co-localization of OCIAD1 and Bcl-Rambo with PMP70 to their respective controls. The data also showed that OCIAD1 and Bcl-Rambo had a statistically significant increase in PCC with HSP60 compared to the control, implying co-localization. This was expected since these are mitochondrial proteins.

These quantitative co-localization studies shows that OCIAD1 and Bcl-Rambo are mainly localized to mitochondria and not on the peroxisome in HeLa cells. Analysis in other cells lines might yield different results, however, due to certain qualities

Therefore, it is possible that OCIAD1 and Bcl-Rambo may interact with peroxisomal proteins through other mechanisms such as mediating contact sites.

VIII. Background Knowledge and Terminology - Mathematical Morphology and Filtering

The basis of my approach to image analysis was to use a branch of mathematics called mathematic morphology.

Any image in binary morphology, a subset of the field that only involves black and white, is represented in $\mathbb{Z}^2$ and $\mathbb{Z}^3$ with two key operations.

Operations done on these images require another element called a structuring element. This is also a binary image. Any operation is done by somehow applying the structuring element onto the image.

The two key operations in mathematical morphology are dilation and erosion.

Dilation is essentially enlarging the image by the size and shape of the structuring element. It does this by linearly going through the pixels of the binary image, and for every positive hit, 1, it will enlarge the image.

Erosion is the opposite of dilation. It will shrink the image by the size and shape of the structuring element.

Dilation and erosion are simply inversions of each other. In other words, dilating an eroded image will not necessarily revert the image back to its previous state. In fact, these types transformations are so common that they even have names.

An opening is defined as the dilation of the erosion of some image with the same structuring element. Such transformation is used to remove boundary elements, like lines or sharp corners.

A closing is the opposite, defined as the erosion of the dilation of some image with the same structuring element. This transformation is good for closing holes found in images.

Regarding images, more information is stored in a grayscale image compared to a binary image, since every individual value of a pixel can span for some desired range, like from 0 to 255, which corresponds to an 8 bit integer value. There are versions of the aforementioned transformations for grayscale images, but alternatively, filters can be used instead for different reasons.

The Gaussian filter is one of such filters that is extensively used in image analysis. The equation for the filter is as follows:

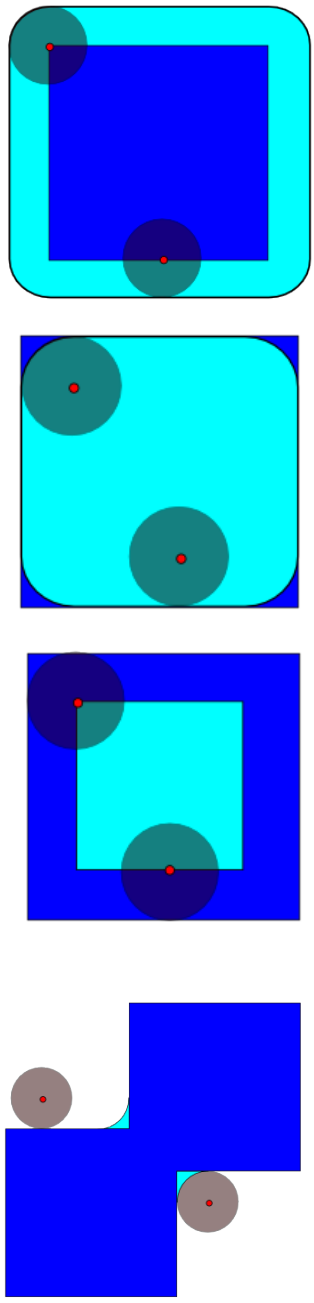


Figure 8: Schematics showing dilation, erosion, opening, and closing (Keshet, 2008)

$$G(x,y) = \frac{1}{2\pi\sigma^2} \; ^{-\frac{x^2+y^2}{2\sigma^2}}$$

where σ is the standard deviation of the Gaussian distribution. In the context of image filtering, this value can be changed to modify how strong the blurring of the image was.

Blurring is used to decrease the amount of noise in the selected image. This can allow for more accurate characterization of what is being observed in a given image.

IX. Background Knowledge and Terminology: Noise in Images

Noise can defined as data that appears in a given dataset that is not part of the actual intended signal. Noise in biological images can arise for a multitude of different reasons.

Firstly, there is Gaussian noise. The sensor used to take images that can contribute to noise. There exist random disturbances, like the temperature of the sensor and the illumination of the image.

Secondly, shot noise can also contribute. This type of noise is associated with the particle-like characteristics of light, and when the light level of a photo is sufficiently low, this causes an observable amount of randomness. When light levels are increased, the amount of shot noise becomes indistinguishable from Gaussian noise.

Fluorescent microscopy can be prone to producing a large amount of noise in the photos that it takes. A variety of factors can contribute to noise in image acquisition. Having too intense of a laser power can misrepresent the quantity of a given signal at a certain pixel. Having too high of a gain, which can be thought of as a global multiplier for signal per each pixel can also increase the amount of noise in an image. (Takeo, 2020)

Autofluorescence is a phenomenon that affects confocal fluorescent microscopy. The probes used to image targets of interest have highly specific absorbance spectra, meaning that they need a specific wavelength of light to fluoresce. In cells, there are other compounds that have broadband absorbance and thus fluoresce. This results in erroneous signals. (Van de Lest *et al*, 1995)

X. Introduction to Dry Lab

This project was motivated by the long amount of time it takes to quantify the images taken during the image acquisition step for single cell experiments. The aim is to use the different fluorescent signals to construct an outline of the cell, shown on the right, to be able to calculate the colocalization of two different signals in the same pixel vicinity and if applicable, the peroxisomal density of the cell. Moreover, the goal was to also not use a cell membrane marker such as CellBrite™, so that there was a maximal number of laser channels open for additional data.

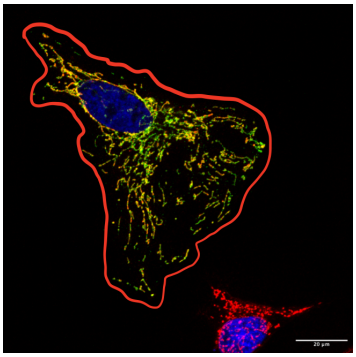


Figure 9: Proposed segmentation mechanism.

XI. Overview of the Program

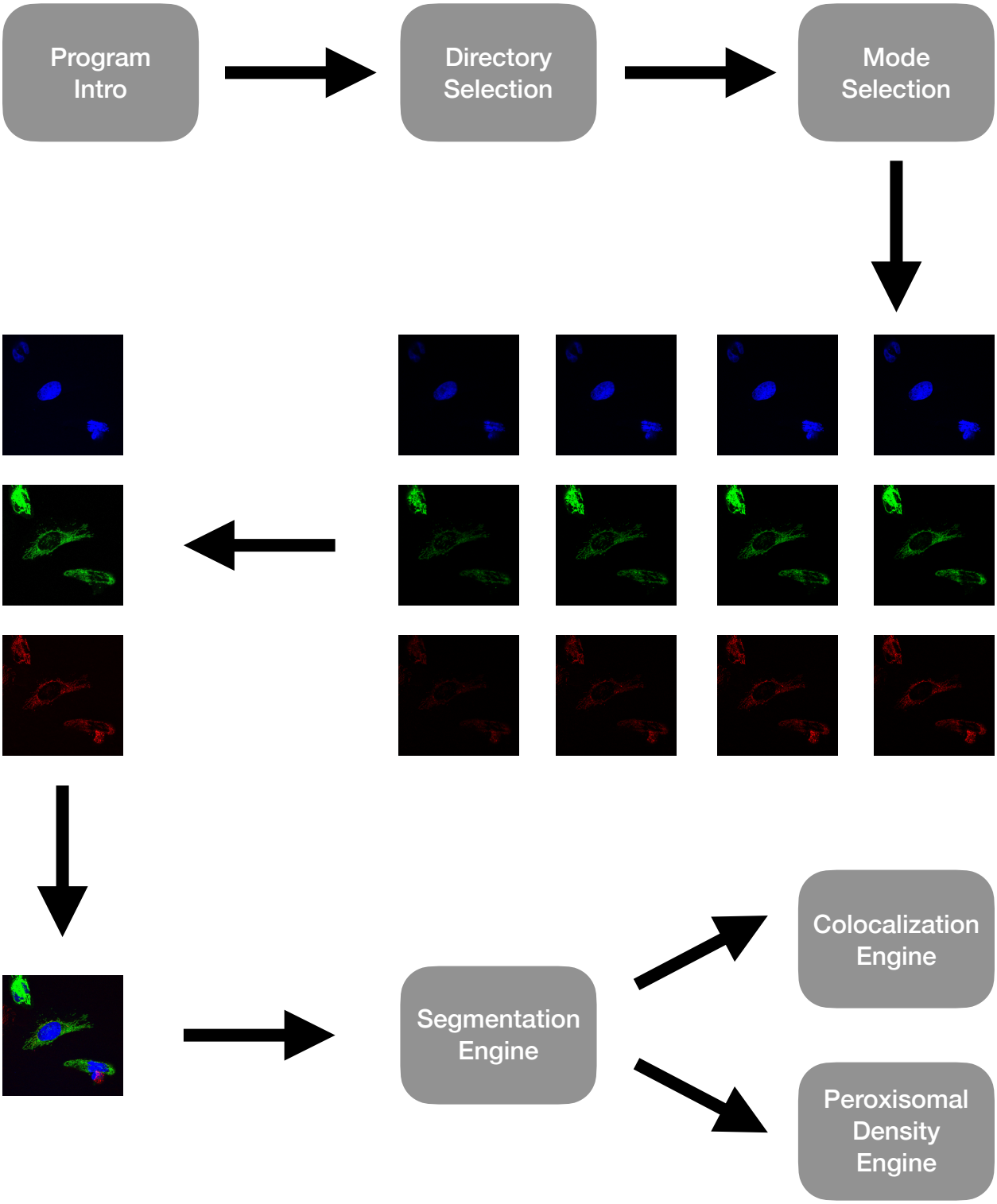


Figure 10: Flowchart depicting the High-Throughout Biological Image Processor's Function

Different microscopes use different image formats that all typically build on a standard lossless image, *tiff*. In addition to having multiple images within the single file, since every channel produces an image and every slice of a stack has that many images.

The LSM 710 produces .lsm files, which is one of Carl Zeiss' enhanced image formats. Within the image's metadata is information like the laser power, gain, spectra, and lens used to acquire the photo. This image format is designed to be opened on proprietary software developed by Zeiss called Zen.

Despite this, open-source software has been developed to read proprietary image formats. Allowing for freer use of image files paves way for more specialized analysis that scientists can conduct, rather than just use the pre-built functions that proprietary software might provide.

Open Microscopy Environment has developed Bio-Formats, an open-source Java library containing portable methods for reading and analyzing biological images of over 100 file formats. Its methods can be used in languages like Java, Python through a wrapper, and MATLAB, our language of choice. (OME, 2022)

While Bio-Formats allows for a suite of free and open-access tools to image analysis, a huge win for creating equity within research and academia, the documentation for the software seems to be hit-or-miss at times. Many webpage and documentation links are broken, and popular .jar files like loci_tools.jar are deprecated. Trial and error was required to figure out how to use the Bio-Formats methods.

In regards to the flow of the diagram, a directory on a computer is selected containing all of the image files to be quantified. Then the mode, be it colocalization, or peroxisomal density is picked. After, a loop begins, going through every single image and creating a composite maximum. This is known as a max Z composite.

The algorithm for taking the max Z involves comparing a pixel in the $x\,y$ plane across all the z slices, and taking whichever is the maximum as the value for the projection. After doing this for all channels, a composite of the three channels is generated by taking the max Z of the other max Z projections.

The max Z composite was chosen to be used for the segmentation engine as it contains the signal from all the channels used during imaging. Having the important biological structures, like the mitochondria, peroxisomes, and the nucleus all help in identifying the cell's borders.

The segmentation engine generates a Vornoi diagram of all the predicted cell boundaries. These boundaries are then used as the parameters along with the max Z composite to determine either colocalization or peroxisomal density for the dataset.

The results are then documented into a CSV file, which is a common file format for viewing tabular data. This can be opened in programs like Excel or Prism GraphPad for further data analysis.

XII. The Segmentation Engine

Perhaps the crux of the project, the segmentation engine uses the various types of morphological operations and filters to create the cell segmentation image.

The outline for the segmentation procedure was adapted from a segmentation tutorial graciously provided by Dr. Miriam Ginzberg of the Kafri Lab at SickKids. This approach to image segmentation is a seed-based approach.

The adapted procedure is as follows:

1. Convert the image to gray-scale
2. Apply a Gaussian filter to smooth the image
3. Close the image, so to remove background noise
4. Binarize the image
5. Open the image with the approximate size of the largest cell
6. Fill the holes in the background
    - Thus, a BW mask is created
7. Shrink the image to the tenth of its original size
8. Get the regional maxima of the shrunken image
9. Resize the image, and confirm the regional maxima against the BW mask
    - Thus, we have seeds
10. Perform a morphological watershed on the seeds to get maximal borders for cells

This method was chosen as a way to reduce the computational complexity of the calculation, and for the fact that precision wasn't prioritized.

After conversation with my PI, a realization arose that we care more about the program being able to determine statistically significant differences within datasets than it being able to completely and accurately quantify the intended values.

Figure 11: Before and after re-sizing the image. This demonstrates the smoothing effect of losing data by shrinking the image

A morphological watershed is a processing technique akin to real life watersheds. In a grey-scale and binary image, it use lesser values to create depressions and greater values to create maxima that can then be separated with borders according to the relative shapes of the depressions and maxima. This turns into a watershed network that is then used to segment cells. (Preim *et al*, 2014)

There were quite a few issues with this approach, and alternative methods are suggested in part XVI.

The seed-based algorithm was designed to be able to differentiate cells that were almost in direct contact with each other. For example, looking at figure x, if a larger opening or Gaussian filter was used, the two nuclei (albeit, the bottom is necrotic) could have appeared as one. The seeds allowed for specific cell identification, and the border would have allowed for separation.

The primary issues with this approach include the magnification of the cells and the need to incorporate multiple channel signals that aren't uniformly distributed in the cell.

The magnification of the images, being at 63x for mine, made using just the nucleus a non-viable option since it would often cut around the edges of cells. The ideal for my experiment was to only image relatively circular and triangular cells, but this was impossible to do.

My images were at a 63 times magnification, meaning the greater resolution in my images could have affected the seed-based attempt at segmenting the images.

The other primary issue with my method was the non-uniformity of signal distribution in my sample images. Looking at the images from the Kafri Lab, the protein mass channels were relatively evenly distributed within the cell. In my cells, specifically staining for mitochondrial and peroxisomal proteins, this was not the case, and the high dynamic range of the signal impacted the segmentation potential.

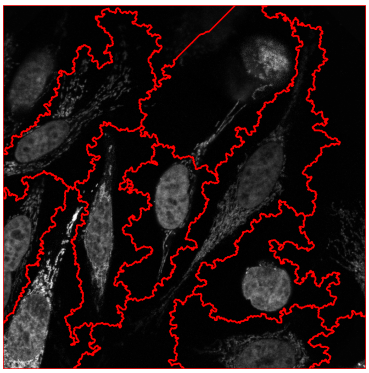


Figure 12: Example result from the segmentation engine. The watershed's segmentation shows up as the red ridges.
The different channels' signals were combined into single grayscale values.

To somewhat counteract the latter issue, I used the max Z projection as the segmentation template, but this also introduced a lot of noise in the image, which likely had an impact on the accuracy of the segmentation.

The engine had a tendency to cross through nuclei and to highlight backgrounds as cells. This artificially increased the number of regions of interest, both by splitting cells and by highlighting the background. The effect of this will be shown in the next section.

XIII. The Colocalization Engine

The colocalization engine used the segmentation engine's results to find Pearson's Correlation Coefficient (PCC) for colocalization of channel signals.

It did this by linearizing the pixels in each region of interest and then finding the coefficient between each pixel, treating the intensity value at each pixel as a datapoint.

The PCC is defined as:

$$r = \frac{\Sigma(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\Sigma(x_i - \bar{x})^2(y_i - \bar{y})^2}}$$

PCC values are in the range $[-1,1]$. Values close to 1 imply perfect positive correlation, zero implies no correlation, and -1 implies perfect negative correlation.

The engine averaged the PCC for each region in each slice to produce a final PCC for each region of interest.

There were two key issues with the colocalization engine: no threshold system was used, and erroneous cells were propagated in the overall calculation of the PCC.

By having no threshold system, background noise is accounted for in every region of interest, which would artificially lower the PCC.

Additionally, having regions of no cells would have a similar effect.

Despite these issues, this issue would be applied to the entire dataset, which could end up being a lesser issue when conducting statistical analyses.

XIV. The Peroxisomal Density Engine

This was unfortunately not completed during this research term.

XV. Results of the Colocalization Engine

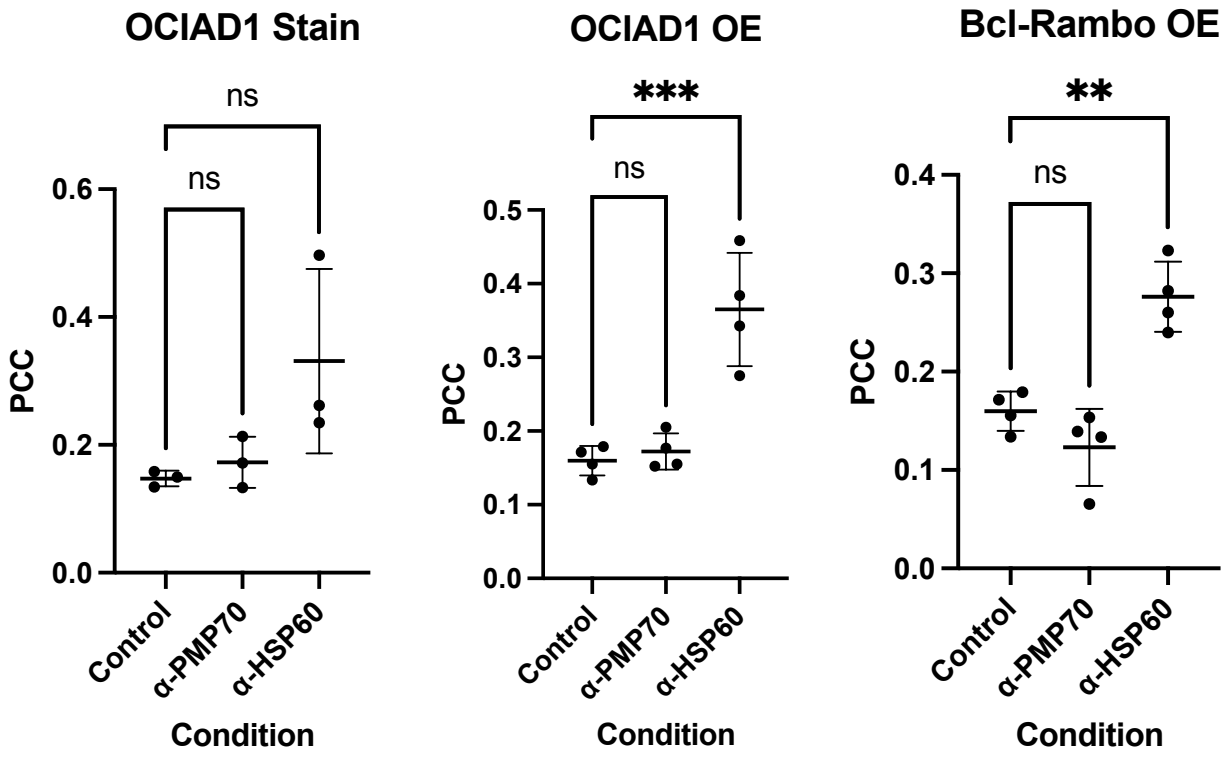


Figure z: Graphs with stated statistically significant differences similar to those presented in figures 5, 6, and 7. $*** \implies P < 0.002, ** \implies P < 0.0021$. ANOVA test w/ Dunnett's used.

XVI. Dry Lab Conclusion and Next Steps

My program was able to accurately identify the statistically significant differences between the control and HSP60 for both overexpression experiments. It does however report lower statistically significant differences than what was manually calculated. It does accurately identify *n s* for the control compared to PMP70.

The program however fails at finding the statistically significant difference for the control compared to HSP60 in the OCIAD1 stain experiment, as previously described. This might be due to the shortcoming of the current iteration of my segmentation engine.

Currently, the program is clearly able to identify very obvious differences in colocalization, but more work is needed to fine-tune it.

A few considerations I could start to make is to try to use a machine learning approach to tag specific regions of interest in an example dataset so to create strong parameters based on different type of protein markers.

Next, it would be to treat this data as three-dimensional rather than flat. If the data was analyzed in terms of PCC at different voxels, a much more accurate result could hypothetically be produced. Treating data as three dimensional could also account for changes in shape at different z-heights of cells.

Finally, a manual threshold ROI selector could be added so to remove the background signal.

I could also end up creating a nicer-looking GUI, or potentially moving to a more open platform so that scientists without MATLAB licenses could still use the program.

While still in its infancy, I view my project as a potentially powerful tool for biologists around the world to immensely improve their image analysis workflow

XVII. References

AlphaFold Protein Structure Database. (n.d.). Alphafold.ebi.ac.uk. https://alphafold.ebi.ac.uk

Bank, R. P. D. (n.d.). *RCSB PDB: Homepage*. Www.rcsb.org. https://www.rcsb.org

Biorender. (2019). *BioRender*. Home; biorender-marketing-site. https://biorender.com
Mathematical morphology. (2022, April 13). In *Wikipedia*. https://en.wikipedia.org/wiki/Mathematical_morphology

Kataoka, T., Holler, N., Micheau, O., Martinon, F., Tinel, A., Hofmann, K., & Tschopp, J. (2001). Bcl-rambo, a Novel Bcl-2 Homologue That Induces Apoptosis via Its Unique C-terminal Extension. *Journal of Biological Chemistry*, *276*(22), 19548–19554. https://doi.org/10.1074/jbc.m010520200

Ogama, Takeo. (2020). A beginner's guide to improving image acquisition in fluorescence microscopy. The Biochemist. 42. 22-27. 10.1042/BIO20200075.

OME. "Reading Files — Bio-Formats 5.8.2 Documentation." Docs.openmicroscopy.org, 18 Apr. 2018, docs.openmicroscopy.org/bio-formats/5.8.2/developers/file-reader.html. Accessed 30 Sept. 2022.

Preim, B., & Botha, C. (2014). Image Analysis for Medical Visualization. *Visual Computing for Medicine (Second Edition)*, 111-175. https://doi.org/10.1016/B978-0-12-415873-3.00004-3

Salemi, M. R., Phinney, B. S., Weissman, J. S., & Nunnari, J. (2021, May 26). *Genome-wide CRISPRi screening identifies OCIAD1 as a prohibitin client and regulatory determinant of mitochondrial Complex III assembly in human cells*. ELife. https://elifesciences.org/articles/67624/figures#content

Van de Lest, C. H., Versteeg, E. M., Veerkamp, J. H., & Van Kuppevelt, T. H. (1995). Elimination of autofluorescence in immunofluorescence microscopy with digital image processing. *The journal of histochemistry and cytochemistry : official journal of the Histochemistry Society*, *43*(7), 727–730. https://doi.org/10.1177/43.7.7608528

Vasseur, M. L., Friedman, J., Jost, M., Xu, J., Yamada, J., Kampmann, M., Horlbeck, M. A.,

XVIII. Code

https://github.com/gerd-bizi/High-Throughput-Biological-Image-Processor