

Requirements Document

*Biomembrane Team - Nick Klose, Richard
Leung, Cameron Mann, Glen Nelson, Omar
Qadri, James Wang*

Project Overview

The Biomembrane Project encompasses the redevelopment of the current MATLAB program, which computes auto/cross/triple-correlations on images of biological membranes retrieved from microscopes. Currently, this program can only analyze singular sets of RGB images, and the processing time is substandard. Analysis of the MATLAB algorithm has led to the discovery of various bottlenecks. It has been decided that the development team will port the code to the Python programming language and refactor it to improve run-time speed. Batch processing is a major feature that will be added to handle more than one set of images at once. The main goal of this project is to improve the efficiency of computation, and these improvements will satisfy that goal.

In addition to an increase in processing speed, there is a demand for a separate web based interface. Batches of images will be uploaded to a central server to be processed. This server can be accessed by individual researchers or students globally via the internet, when given a login name and password.

The Biomembrane program will be used by university level students and professors, as well as researchers. Therefore the user interface design will not be ultra simplistic, but it will maintain efficient input of data and parameters. The redesigned local interface can be downloaded as a package from the server, and it will contain everything necessary to perform both singular and batch tests. The local version of this program will not require connection to the server and works as a standalone means of biomembrane analysis.

Requirements

Image Inputs (High)

Requirement #: 1

Description: The program will handle either single or separate RGB images in a variety of lossless formats.

Rationale: Ease of use for the end user.

Originator: Nils Petersen

Fit Criterion: Users will be able to upload either singles or triples of images in various formats.

Image Processing (High)

Requirement #: 2

Description: The program will process the images using known algorithms.

Rationale: The resulting data from the image processing is the primary function of the program.

Originator: Nils Petersen

Fit Criterion: The program will run the following algorithms:

- FFT
- Auto and cross correlation
- Reverse FFT
- 2D gaussian fitting

Batch Processing (High)

Requirement #: 3

Description: The program will accept multiple images/sets of images as input.

Rationale: Allowing users to process multiple sets of data in one step is more efficient.

Originator: Nils Petersen

Fit Criterion: Users will be able to upload multiple sets of independent data in a single step.

Performance (High)

Requirement #: 4

Description: The program will perform the computations in a reasonable timeframe.

Rationale: The computations currently take a significant amount of time; improvements would greatly improve usability for users.

Originator: Nils Petersen

Fit Criterion: The program will perform the computations faster than the existing MATLAB solution.

Portability (High)

Requirement #: 5

Description: The program will run on a variety of platforms.

Rationale: Make it easier for users to access the program from their own machines.

Originator: Nils Petersen

Fit Criterion: The web frontend does not need to be portable, but allows users to access the program from anywhere with an internet connection. The local GUI will run on Windows, OSX and Linux.

Authentication (High)

Requirement #: 6

Description: The web frontend should have an authentication system to control who has access.

Rationale: Prevent unauthorized and/or wasteful use of resources.

Originator: Nils Petersen

Fit Criterion: An administrative system will be available to allow the addition and removal of user accounts.

Automatic Parameter Detection (Medium)

Requirement #: 7

Description: Parameters for the image processing algorithms can be determined from the input data.

Rationale: Remove the need for users to manually enter this information.

Originator: Nils Petersen

Fit Criterion: Parameters that must currently be manually entered will be automatically determined instead.

Microscope Metadata (Low)

Requirement #: 8

Description: The program will be able to determine parameters for the image processing algorithms using metadata from the microscope.

Rationale: Remove the need for users to manually enter this information.

Originator: Nils Petersen

Fit Criterion: Users will be able to upload microscope metadata files which can be used to automatically configure parameters.

Storyboard

Image Correlation Spectroscopy

Input Parameters

Xi1

Xi2

Xi3

Xi4

Xi5

Xi6

Scheme

Nb0

Ng0

Nr0

Nrg0

Nrb0

Ngb0

Nrgb0

Nb

Ng

Nr

Nrg

Nrb

Ngb

Nrgb

Poisson

ICS Fitting Parameters

☒ Single Correlation

☐ All Correlations

☐ Coincidence Analysis

Channels to correlate:

☐ Red

☒ Green

☒ Blue

Selected: GB XCS

Range:

Iterations:

Res. Norm.:

g0:

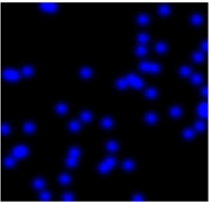
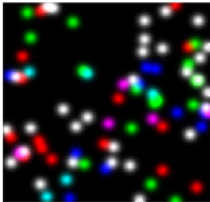
w:

ginf:

Start

Stop

Image Settings



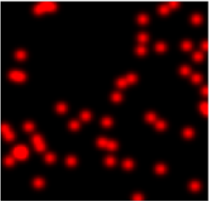
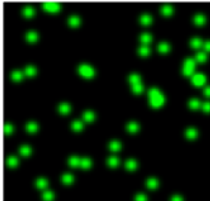


Image size: 512 x 512

☐ Single RGB Image

☒ Three Channel Images

Load Red Image

Load Green Image

Load Blue Image

Beam radius:

10 pixels

Channels

Channel	Brightness	bg	N/image	N/beam
<input checked="" type="checkbox"/> Red	1.0	0.0	0.0	0.0
<input type="checkbox"/> Green				
<input type="checkbox"/> Blue				

- The input parameters section is largely unchanged from the original interface due to outstanding questions regarding their purpose and function. This will be revamped upon meeting with the client to determine the best way of implementing them. The function of the Scheme button is currently unknown.
- In ICS Fitting Parameters, the labels g_0 , w , and g_{inf} need to be replaced with more meaningful ones.
- The inner box in the ICS Fitting Parameters section will have a different layout depending on which radio button is selected.
 - For *single correlation*, the user can select only one color (self-correlation), two colors (cross-correlation), or three colors (triple-correlation). Red will be selected by default, then afterwards the user's last choice should be saved. The user will be unable to deselect a checkbox if it is the only checkbox selected (since it doesn't make sense to correlate without any colors).
 - For *all correlations*, the program will execute all possible combinations of colors rather than a specific correlation. The inner box will simply be grayed out for this option.
 - For *coincidence analysis*, optional parameters can be shown in the inner box.
- The currently loaded image(s) and their size are previewed in the Image Settings panel. The user can choose either to use a single RGB image, or three separate images containing red, green, and blue channels. For the former selection, only one button will be shown allowing the user to load the chosen RGB image. If the user chooses to use three images, three separate loading buttons are shown.
- The user can also select which channels to use in the Image Settings panel. For any channel which has been selected, the user can configure parameters specific to that channel.
- If the user presses the Start button, the correlation begins. Relevant graphs and data are shown similar to the old interface, except that they will all be shown in one organized window rather than in separate windows.
- The user can press the Stop button to end the correlation calculations.
- If the user has not started the process yet, the Stop button will be grayed out. Otherwise, the Start button will be grayed out. This will make it clear to the user whether or not the process has been started.

Intellectual Property

The Biomembrane project will be licensed under the "Intellectual Property Agreement" created for the 401 class.

This Agreement, effective the 1st day of April 2013, is entered into by and between the Client (hereinafter "client") and the students of the Biomembrane team (hereinafter "the development team"), in order to establish terms and conditions concerning the completion of the

Biomembrane Application (hereinafter “The Application”) which is limited to the application domain of application-domain (hereinafter “the domain of use for the application”)

It is agreed by the client and the development team that all domain specific knowledge and compiled research is the intellectual property of the client, regarded as a copyrighted collection. The framework and code base created by the development team is their own intellectual property, and may only be used for the purposes outlined in the documentation of the application, which has been provided to the client. The development team agrees not to use their framework for, or take part in the development of, anything that falls within the domain of use for the application, for a period of 6 (six) months after the signing of this agreement.

Project Glossary

- **Application-Domain** - The domain of use for the “Biomembrane” application.
- **Application, the** - The “Biomembrane” application, based off of the ICS software.
- **Client** - The client of the project (Nils Peterson)
- **ICS Software** - Image Correlation Spectroscopy Software.
- **MATLAB** - A high-level language and interactive environment for numerical computation, visualization, and programming.

Resources

Image Correlation Spectroscopy Software. “ICS Soft” and ICS with Poisson N
<https://github.com/UniversityOfAlberta/BioMembrane/tree/master/ICS>

Research Material

Petersen, N. O., Hoddellius, P. L., Wiseman, P. W., Seger, O., & Magnusson, K. (1993). Quantitation of Membrane Receptor Distributions by Image Correlation Spectroscopy: Concept and Application. *Biophysical Journal*, Volume 65, 1135-1146.

Keating, E., Brown, C. M., & Petersen, N. O. (2002). Mapping Molecular Interactions and Transport in Cell Membranes by Image Correlation Spectroscopy. *Mapping Molecular Interactions by Image Correlation Spectroscopy*.

Heinze, K. G., Jahnz, M., & Schwille, P. (2004). Triple-Color Coincidence Analysis: One Step Further in Following Higher Order Molecular Complex Formation. *Biophysical Journal*, 86, 506–516.

Kolin, D. L., & Wiseman, P. W. (2007). *Advances in Image Correlation Spectroscopy: Measuring*

Number Densities, Aggregation States, and Dynamics of Fluorescently labeled Macromolecules in Cells. Cell Biochem Biophys, 49, 141-164.

Nohe, A., & Petersen, N. O. (2003). Analyzing protein–protein interactions in cell membranes.

Ridgeway, W. K., Millar, D., & Williamson, J. (0). Accompanying: The spectroscopic basis of Fluorescence Triple Correlation Spectroscopy, S1-S7.

Ridgeway, W. K., Millar, D. P., & Williamson, J. R. (2012). The Spectroscopic Basis of Fluorescence Triple Correlation Spectroscopy. Journal of Physical Chemistry B, 116, 1908–1919.

All material can be found at

<https://github.com/UniversityOfAlberta/BioMembrane/tree/master/docs>

External Python Libraries Used:

NumPy (2012) Numpy developers (Version 16.2) [Open Source Python Library]

<http://www.numpy.org/>

SciPy (2012) Community Project sponsored and supported by Enthought (Version 0.11.0) [Open Source Python Library] <http://www.scipy.org/>

Matplotlib (2012) John Hunter (Version 1.20) [Python Plotting Library] <http://matplotlib.org/>

Django (2012) Django Software Foundation (Version 1.43) [Web Framework]

<https://www.djangoproject.com/>

NumPy

Is an extension for python which provides support for scientific computations of large n-dimensional arrays and matrices. Also has libraries to calculate correlation and Fourier transformations functions we need. Numpy also has tools to integrate Fortran and C/C++ code.

SciPy

Is an open source library of mathematical tools and algorithms designed for Python. It is used perform the curve fitting in our functions.

Matplotlib

Used to plot graphs and shares the same look and feel as graphs generated in Matlab.

Django

An open source python web framework that has powerful backend and frontend features.