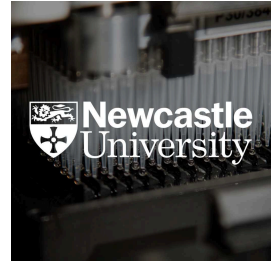


Jul 29, 2023

Co-culture leukemia high-content image analysis using supervised machine learning

DOI

<https://dx.doi.org/10.17504/protocols.io.rm7vzxy52gx1/v1>



Hayden Bell¹

¹University of Newcastle-upon-Tyne



Hayden L Bell

University of Newcastle-upon-Tyne, Dana-Farber Cancer Instit...

Create & collaborate more with a free account

Edit and publish protocols, collaborate in communities, share insights through comments, and track progress with run records.

Create free account

OPEN  ACCESS



DOI: <https://dx.doi.org/10.17504/protocols.io.rm7vzxy52gx1/v1>

Protocol Citation: Hayden Bell 2023. Co-culture leukemia high-content image analysis using supervised machine learning. protocols.io <https://dx.doi.org/10.17504/protocols.io.rm7vzxy52gx1/v1>

License: This is an open access protocol distributed under the terms of the **Creative Commons Attribution License**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited



Protocol status: Working

We use this protocol and it's working

Created: July 26, 2023

Last Modified: July 29, 2023

Protocol Integer ID: 85502

Keywords: image analysis, co-culture, leukemia, acute leukemia, machine learning, cell profiler, Ilastik, cell counting, content image analysis of acute leukemia cell, fluorescence microscopy image of cell, basic skeleton pipeline for image analysis, cell profiler, acute leukemia cell, microscopy image, content image analysis, image analysis, mesenchymal stromal cell, supervised machine, cell, cell class, absolute cell numbers for each cell class, supervised machine learning, using supervised machine, absolute cell number, basic skeleton pipeline, using supervised machine learning, fluorescence

Disclaimer

Not intended for medical purposes. This protocol is not intended to diagnose or treat any medical condition and should not be used for any medical purpose. This protocol is intended for use only in a research capacity.

The author/s accept no responsibility for the accuracy of data resulting from this protocol. The author/s further assume no responsibility or liability for any errors or omissions in the content of this protocol. The information contained in this protocol is provided on an "as is" basis with no guarantees of completeness, accuracy, usefulness or timeliness and without any warranties of any kind whatsoever, express or implied.

Abstract

This protocol describes a non-informatics based approach to high-content image analysis of acute leukemia cells in co-culture with mesenchymal stromal cells (MSCs) using supervised machine learning. The analysis pipeline leverages two powerful, open-source software applications - Cell Profiler and Ilastik. The aim of this protocol is to provide a basic skeleton pipeline for image analysis to, at minimum, determine absolute cell numbers for each cell class from a fluorescence microscopy image of cells stained with a DNA dye.

This protocol is a detailed companion walkthrough for the Github repository available at https://github.com/hayden-bell/Image_Analysis.



Guidelines

This example pipeline uses images from patient-derived xenograft (PDX) acute leukemia cells (AML/ALL) in co-culture with human bone marrow-derived mesenchymal stem cells (MSCs). The images are raw grayscale TIF images of cells assayed across a wide range of experimental conditions and live cells stained with the nucleic acid dye CyQUANT™.

In principle, any fluorescence microscope with a stable illumination source that can acquire images of DNA stained cells can be used for this analysis pipeline. Appropriate high-content imaging systems include the Zeiss CellDiscoverer 7 and the PerkinElmer Opera.

As default, quantitative data will be exported as an SQLite database and will require software to handle this file format. If working with low numbers of images at any one time, data can be alternatively exported as a CSV file format using the ExportToSpreadsheet module which is compatible with Microsoft Excel software.

Materials

CyQUANT™ Direct Cell Proliferation Assay, ThermoFisher Scientific, #C35011

Before start

Two open-source applications are required for this image analysis pipeline: Ilastik (<https://github.com/ilastik/ilastik>) and Cell Profiler (<https://github.com/CellProfiler>).

This protocol uses the project files from the Github repository available at https://github.com/hayden-bell/Image_Analysis. Download the *BaseProject.ilp* and *BaseProject.cppipe* files before starting.

Use high-quality fluorescence microscopy images in a lossless high-resolution file format such as TIF.

Training a supervised machine learning model (semantic segmentation)

- 1 Open the Ilastik software and load the *BaseProject.ilp* project.
- 2 In the *Input Data* tab, load several different images (up to ~10) for training which are representative of different experimental conditions.

For example, images from positive and negative controls whereby cell number is maximised/minimised.

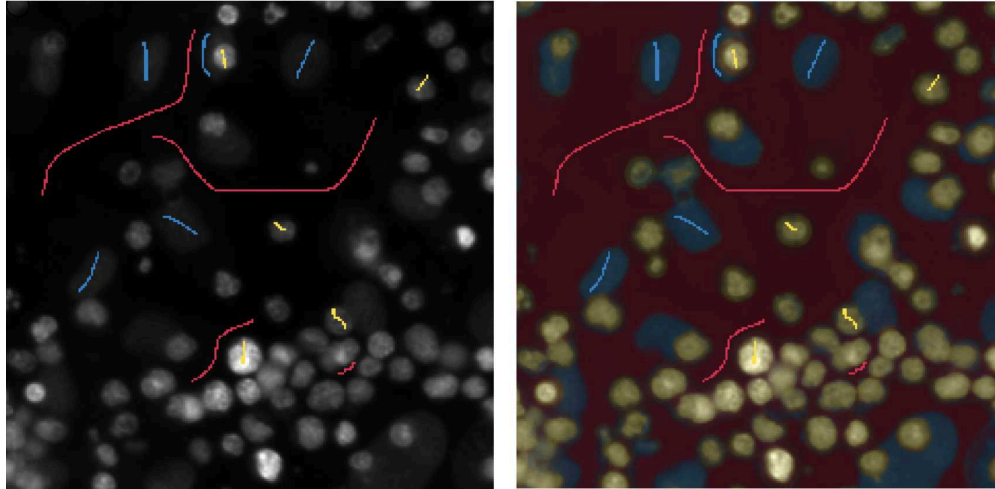
- 3 In the *Feature Selection* tab, click *Select Features...* and ensure all 37 features are selected.

- 4 In the *Training* tab, ensure there are three separate Labels/classes in order as:

1. PDX
2. MSC
3. Bg

- 4.1 Using the Brush Cursor, manually annotate within several nuclei of each class using the respective Label class. Use the zoom to view the image large to ensure precision in annotation.

Errors can be corrected using the Eraser Cursor and the image contrast can be changed using the Window Leveling tool to better visualise dimmer nuclei.



Example annotation for each semantic segmentation class. PDX, yellow; MSCs, yellow; background, bg.

- 4.2 Use the *Live Update* feature to view a real-time overlay of the probability map for each class over the original training images.
- 4.3 Iteratively refine the annotations across the training dataset images until performance is good.

Note

Avoid over-annotating the training dataset in Ilastik as this will result in poorer performance on unseen image data sets since the model will learn characteristics of the test data set and compromise generalisability of the model.

- 5 In the *Prediction Export* panel, select *Source: Probabilities*.
- 5.1 Click '*Choose Export Image Settings...*' and ensure the output file is *tif* format with the axis order *ycx*.
- 5.2 Choose the *Output File* destination as
{dataset_dir}/probabilities/{nickname}_{result_type}.tif
- 6 In the *Batch Processing* tab, click '*Select Raw Data Files...*' to import all of the test image data files.



- 7 Click 'Process all files'.

Quantifying individual cell nuclei (instance segmentation)

- 8 Open the Cell Profiler software and import the *BaseProject.cppipe* pipeline (*File > Import > Pipeline from File...*).
- 9 In the *Images* module, load the **probability map images** generated from the Ilastik project.

Note: do not load the original images at this step.
- 10 *Optional: In the Metadata tab, regular expressions (regex) can be used to extract meaningful data from each image filename such as plate id, well id, etc.*

By default, the pipeline will attempt to extract the well id of each image in the format A1 or A01.
- 11 *Optional: Outlines of how well the pipeline identifies individual PDX or MSC nuclei can be visualised using the OverlayOutlines module. Select the checkpoint to enable this module and save the output by using the SaveImages module.*
- 12 In the *ExportToDatabase* module, modify the *Experiment name* and SQLite database filename to better identify the experimental data output.

Note

The default output location can be modified by clicking the 'Output Settings' button.

- 13 Click *Analyze Images* to process the imported dataset and export data as SQLite database format.

Reading the data output

- 14 Data can be read using any database software application which can open SQLite file format.



Data can be retrieved from the *[Experiment name]_Per_Image* data table.

Recorded data include:

- Predicted PDX nuclei counts (Image_Count_LeukaemicNuclei)
- Predicted MSC nuclei counts (Image_Count_MSCNuclei)
- Image file name (Image_FileName_CyQ)
- Image well location (Image_Metadata_Well)
- Plus any additional data exported from separate modules or metadata extractions.