BCH 571 Final Paper

ROI.gen

In image analysis regions of interest (ROIs) are used to denote a subset of pixels containing desired information that are assessed together as a unit. For high magnification images of biological samples one ROI often corresponds to one whole cell. If the sample size is low and/or the phenotype of interest displays substantial heterogeneity within one cell, subdivision into smaller regions of interest can be useful in describing the observed variation. Laying out 20+ ROIs per cell is time consuming and tedious to do manually, and achieving a good random distribution of the ROIs is improbable. The goal of this program was to automatically generate random ROIs within the confines of a cell outline.

The automatic generation of ROIs can be integrated into a program which also opens image files, splits signal channels, takes relevant readings and stores acquired data. If membrane dyes were included in the image acquisition process it would be possible to obtain numerical data from an entire batch of images without human intervention during the process. Tailoring an automatic analysis pipeline like this to the parameters of a protocol drastically increases potential for attainable high-throughput experiments.

This version of ROI.gen is designed to accept a directory filled with .TIFF files. These must contain an overlay layer with outlines of the depicted cells, and must also be splittable into multiple signal channels. There are also two easily customizable variables within the code: size of the ROIs (roi\_width) and number of ROIs generated inside each cell (num\_roi). Running the program will result in generation of two .txt files containing a table of measurements. The table consists of row numbers, area, mean gray value, integrated pixel intensity and raw pixel intensity. Save location for these files can be set by changing a file path within the code.

All of these manipulations are made possible by integration with the popular image analysis software ImageJ. The FIJI distribution of ImageJ is a biology-applications specific package that includes support for running jython scripts that utilize the ij API. To function ROI.gen must be run within the FIJI interface of an installation that contains ij API files. These files include functions and methods for controlling various ImageJ features, such as ‘ROI manager’, ‘measure’ and ‘results table’. When these methods are called by a jython script running within FIJI, the ImageJ components respond the same way as they would if you called them manually using menus or buttons.

Achieving the basic functionality of ROI.gen using the ij API requires many steps:

1. Specify input files, and for each file access overlay features
2. Sort overlay shapes into background spots (for calculating and subtracting background fluorescence) and cell outlines.
3. Save background spot ROIs in an array for later measurement.
4. Store the number of unique cell outlines, because that is the number of times we will need to loop through the ROI generation procedure.
5. Add the first cell outline to the ROI manager
6. Create a bounding box for the cell
7. Generate a random point within the bounding box
8. Create an ROI rectangle with upper lefthand corner at that point
9. Add rectangle to a new overlay (changes it from an array of points to a shape)
10. Add rectangle to ROI manager from the overlay (relates the shape to properties of the active image)
11. Use the ROI manager ‘OR’ function to see if rectangle is contained within the cell outline. If it is, accept it, otherwise remove it. To determine this, an area measurement is taken of the cell outline ROI. Then another area measurement is taken including all pixels that are in the cell ROI ‘OR’ the rectangular ROI. If the ‘OR’ area is larger, then part of the rectangular ROI hangs out of the cell outline.
12. Once the desired number of ROIs are placed (or 20 consecutive attempts to generate acceptable placement fail) save accepted ROIs to an ROI array and clear the ROI manager.
13. Repeat as many times as the number of cells in the image, each time using a different cell outline bounding box.
14. When all ROIs are generated split the image into channels.
15. If there are background spots, add all of them to the ROI manager and measure them on each channel.
16. Retrieve the measurement results from the results table as an array, then save an average of those values.
17. Next clear the ROI manager before adding all of the generated ROIs to it.
18. Measure all ROIs for each relevant channel and retrieve the results from the results table as an array
19. Subtract the appropriate average background value from each of these if applicable.
20. Save the results in a .txt file. File name includes the .TIFF name, the channel identity, and a time stamp.
21. Close images and files, then open the next .TIFF file.

ROI.gen can be customized to fit disparate experimental requirements. The measurements taken can be changed within the code. The ‘relevant’ channels can also be changed. Code pertaining to the ‘background spots’ can be commented out if none are included in the overlay. The initial overlay with manually traced cell outlines could be done away with all together if a membrane dye was used in collecting the images. Achieving this would require downloading a script designed to automatically make ROIs from cell outlines, and calling that script from ROI.gen.