CANCER RESEARCH UK | CAMBRIDGE INSTITUTE

Image Analysis with Fiji Exercises

Section 1: Basics

A. Opening and displaying microscopy file formats

- Open *singleChannelCell.ome.tif* using the Bio-Formats Importer Plugin (Plugins -> Bio-Formats -> Bio-Formats Importer). Change "Color Mode" to Grayscale and leave other inputs with default settings.

  - ***Note we have used the Bio-Formats plugin directly to ensure the .ome.tiff format is correctly recognised and loaded. For most file formats File -> Open… or simply dragging the image to the FIJI window will work.***

- Adjust the display range using Image -> Adjust -> Brightness/Contrast... Try the "Auto" and "Reset" buttons.

  - ***The display range can be specified using "Set". The "Apply" button will apply the current display range mapping function to the pixel data (careful this can modify pixel values!).***

- Try changing the color map using Image -> Lookup Tables -> Red. Experiment with different LUTs.

  - ***The HiLo LUT can be used to visualise saturated and undersaturated pixels if the display range is set to the full dynamic range of the detector (0-255 for 8-bit image). Are there any saturated pixels in this image?***

- The image Meta Data can be viewed using Image -> Show Info…. The pixel size should be automatically calibrated from this Meta Data. Check this has been done using Analyse -> Set Scale …

- Add a scale bar using Analyse -> Tools -> Scale Bar…

  - ***Ensure that the overlay box is NOT checked. This will ensure that the scale bar is "printed" on the image and will be visible in other software applications.***

- Once you are satisfied with the display settings save the image as a TIF using File -> Save As -> Tiff…

  - ***To save a color (RGB) image using the associated LUT first covert to RGB using Image -> Type -> RGB Color.***

B. Working with multi-channel data

- Open Rat_Hippocampal_Neuron.tif using File -> Open… or drag and drop.

  - ***The data has four fluorescent channels and one DIC image (see Image -> Show Info). These can be viewed using the slider at the bottom of the window.***

- Duplicate the data minus the DIC channel using Image -> Duplicate. Select channels 1-4 only and give the new duplicated data a sensible name. Close the original image window.

    - ***The duplicate function is a very useful feature!***

- Visualise the data as a composite image using Image -> Color -> Make Composite.

    - ***The display range of each channel can still be adjusted separately (Image -> Adjust -> Brightness/Contrast). The slider is used to select the channel. Try resetting the display range for each channel.***

- Split the data with separate windows for each channel (Image -> Color -> Split Channels). Change the LUTs for one or more of the channels before remerging the data (Image -> Color -> Merge Channels …).

- A standard RGB color image can be created using Image -> Color -> Stack to RGB.

    - ***This should only be used to create images for display. Information is lost in the conversion!***

C. Working with z-stacks

- Open z-stack.tif

    - ***The data has 2 fluorescent channels and 5 z-slices. Use the sliders to select a slice.***

- You can visualise the stack using Image -> Stacks -> Orthogonal Views

    - ***Use the crosshairs to select the corresponding XY, XZ and YZ ortho-slices. To exit the viewer close either or both of the XZ or YZ windows.***

- Project the data onto a single image using Image -> Stacks –> Z Project…

    - ***Try maximum and average projections.***

- Create a 3D rendering of the stack using Image -> Stacks -> 3D Project…

    - ***Use the "Brightest Point" projection method. The slice spacing is calibrated from the Meta Data. The rendering is rotated about an axis to create a movie. The axis and rotation increment can both be specified. The "Interpolate" option should be ticked to eliminate gaps in the z-dimension.***

D. Measurements

- Open fluorescentCells.tif. Plot a histogram of the first channel using Analyse -> Histogram.

- Select the third channel (blue) and draw a line across one of the nuclei (tool bar). Plot the intensity profile along this line using Analyze -> Plot Profile.

- Draw a rectangular region using the toolbar and measure some properties corresponding to the chosen region using Analyze -> Measure. The results are displayed in the Results Table.

- ***The properties measured can be set using Analyse -> Set Measurments …***

E. Challenge: The FigureJ plugin

- Use the FigureJ plugin [1] to create a panelled figure from Rat_Hippocampal_Neuron.tif suitable for use in a publication. There should be one grayscale image for each channel and a sixth composite color image. Include a scale bar in at least one of the images.

    - ***To use FigureJ you must first create and save the images for each panel. The FigureJ plugin is an optional upgrade which has been pre-installed on the virtual machine. Further documentation and installation instructions can be found at http://imagejdocu.tudor.lu/doku.php?id=plugin:utilities:figurej:start***

Section 2: Introduction to Image Processing and Analysis

A. Thresholding, the Watershed transformation and object counting

- Open HeLa.tif [2]. The aim of this exercise is to segment and count the number of nuclei in the image.

- Apply a threshold to the image (Image -> Adjust -> Threshold …). Ensure "Dark Background" is selected.

    - **The threshold value can either be selected manually or an automated method can be chosen from the dropdown menu (better!). Try both the Li and Otsu methods. Once you are satisfied click apply to create a binary image.**

- Any "holes" in the binary image can be filled using Process -> Binary -> Fill Holes

- Touching objects can be separated using the watershed transformation (Process -> Binary -> Watershed).

    - **The watershed transformation is appropriate for targets which are concave (such as nuclei). Objects with concavities (such as cellular membranes) tend to be over-split.**

- Use the Analyze Particles plugin (Analyze -> Analyze Particles…) to count the number of nuclei and record the size of each.

    - **Set a minimum size to exclude objects smaller than the given value. Check "Summarize" to display a summary, "Display Results" to show all the measurements in the Results Table and "Add to Manager" to add the object boundaries to the ROI manager / visualise the results. The Results Table and/or Summary can be saved to an excel document.**

B. Noise removal and image filters

- Open cellPoissonNoise.tif. This image has been corrupted with (primarily) Poisson noise.

- Duplicate the image (image -> Duplicate…) and rename as "Gaussian Filter" (or similar)

- Apply a Gaussian Filter (Process -> Filters -> Gaussian Blur…) to the duplicated image.

- **The radius of the Gaussian filter should be matched to the width of the point spread function (here try 120nm). Ensure "Scaled Units" is selected to enter sigma in micro-meters. Does this improve the image quality?**

- Duplicate the original image a second time and rename as "Median Filter". Apply a median filter (Process -> Filters -> Median…) with a radius of 2 pixels.

  - **Which filter (Gaussian or Median) do you think is best at removing the Poisson noise and preserving the true signal? Use the zoom tool and adjust the display range to look at the detail!**

- Repeat the above steps but with cellImpulseNoise.tif. This image has been corrupted with both impulse (salt and pepper) and Poisson noise.

  - **Which filter (Gaussian or Median) do you think is best at removing the noise? Change the display range and zoom to access this.**

C. Spot detection with Difference of Gaussians

- Open similuatedSpots.tif [3]. The purpose of this exercise is to enhance and detect the spots.

  - **The Laplacian of Gaussian (LoG) filter is very useful for spot detection. The LoG filter is approximated by a Difference of Gaussians with a ratio between the radii equal to 1.4.**

- Duplicate the image twice and rename the new windows smallGaussian and bigGaussian.

- Apply Gaussian filters to the duplicated images with sigma values of 2.5 and 3.5 pixels respectively.

  - **Note 3.5/2.5=1.4 hence a pixel wise subtraction of bigGaussian from smallGaussian approximates a LoG filter.**

- Perform smallGaussian – bigGaussian using Process -> Image Calculator.

  - **Check "Create New Window". Adjust the display range using the "Auto" setting (Image -> Adjust -> Brightness/Contrast). Has the LoG filter emphasised the spot like features?**

- Apply an Otsu threshold and count the number of spots with the Analyze Particles plugin.

  - **If appropriate use a Watershed Transformation to separate touching spots.**

D. Rolling ball background subtraction

- Open helaCellsTwoChannel.tif and split the channels into separate windows (Image -> Color -> Split Channels).

  - **Note there are areas of background signal which do not correspond to the structures we are interested in. This background signal could be due to various factors such as auto-fluorescence or non-specific binding of the fluorophore.**

- Estimate the width of the largest object in the red channel.
    - **This can be done by drawing a line though the largest feature (not designated background) and plotting a line profile (Analyze -> Plot Profile).**
- To remove the background in each channel try using a rolling-ball background subtraction approach (Process -> Subtract Background). Ensure "Light Background" is not checked.
    - **The size of the rolling ball radius should be at least as large as the largest feature of interest (convert to pixels manually using the image properties). Use the preview option to experiment with other values.**
- Repeat the above two steps for the green channel. Merge the two background subtracted images to make a color composite.

E. The PureDenoise Plugin

- Use the PureDenoise plugin [4] to remove noise from cellPoissonNoise2.tif.
    - **You first need to download and install the plugin. Go to http://bigwww.epfl.ch/algorithms/denoise/ and download PureDenoise_.jar. Place the .jar file in the plugins folder within the Fiji directory (/home/imgcam/Fiji.app/plugins/). When you restart FIJI the PureDenoise plugin should have appeared under the plugin menu.**
    - **The plugin can estimate the required parameters from the image. Run the plugin with 4 spin-cycles and automatic global noise estimation.**
    - **How does the result compare to a Gaussian filter?**

F. Challenge

- Use what you have learned to develop a workflow to count the number of spots in the second channel (red) of helaCellsTwoChannel.tif

Section 3: Workflow Automation with Recorded Macros

A. Developing a workflow

- The ND2H-SIM folder contains images of simulated nuclei from a public repository [2]. Open one of the images and develop a workflow to count the number of nuclei.
    - **The aim of this section is to automate this process so it is important to consider this when designing the workflow. For example manually selected threshold values should not be used.**

B. Recording and running a macro

- Close all windows and open the macro recorder (Plugins -> Macros -> Record…). Open an image (use File -> Open) and repeat your workflow to record the commands in ImageJ macro language. Once you have finished click create to convert to a macro.

  - **The macro recorder is very useful for automation as you don't need to learn the commands for specific plugins, you just record them!**

- Add comments to the macro using //. Comments are really important! A typical example would look like this:

```
// Open the Image
open("C:\\Users\\pike01\\Documents\\Reports\\FIJI Course\\Images\\ND2H-SIM\\I6.tif");

// Perform a Gaussian Blur with a radius of 4 pixels to de-noise the image
run("Gaussian Blur...", "sigma=4");

// Subtract the background with rolling ball radius of 60 pixels
run("Subtract Background...", "rolling=60");

// Perform an automated Otsu thresholding and convert to binary image
setAutoThreshold("Default dark");
setAutoThreshold("Otsu dark");
run("Convert to Mask");

// Fill any "holes" in the binary image
run("Fill Holes");

// Separate touching objects with a Watershed Transformation
run("Watershed");

// Count the number of particles with the Analyze Particles plugin.
// Minimum object size set to 9 micro-meters squared
// Only the "exclude on edges", "summarize" boxes were ticked
run("Analyze Particles...", "size=9-Infinity exclude summarize");
```

  - **Click "run" to run the macro. Try modifying the file name within the open command to run the macro on a different image. Does the workflow work on this image?**

C. Batch processing with for loops

- Add a "for loop" to your macro to batch process all the files in the ND2H-SIM directory:

```
for (i = 1; i < 7; i = i + 1) {

        // Open the Image
        open("C:\\Users\\pike01\\Documents\\Reports\\FIJI Course\\Images\\ND2H-SIM\\I" + i + ".tif");

        Remaining workflow commands go here

}
```

  - **When you run a batch operation you may want to stop windows opening for each image! To do this add the flowing command at the beginning of your macro:** setBatchMode(true);

D. Challenge

- Write a macro which labels each slice in mri-stack.tif with its position. For example the tenth stack should be labelled with “z=10”.
    - **You will need to loop through all the slices in the stack. A full list of built in macro commands can be found at http://rsb.info.nih.gov/ij/developer/macro/functions.html.**
    - **Hint: the nSlices(), setSlice(n), makeText(string, x, y) and run("Draw", "slice") commands could be useful!**

- Write a new macro to implement your protocol from the challenge in section 2.
    - **Modify the macro to take user input for the smaller of the two radii. You will need to use the getNumber("prompt", defaultValue) command.**
    - **Experiment with different input values for the small radius.**

[1] Mutterer, J., and E. Zinck. "Quick-and-clean article figures with FigureJ."Journal of microscopy 252.1 (2013): 89-91.

[2] Maška, Martin, et al. "A benchmark for comparison of cell tracking algorithms." Bioinformatics 30.11 (2014): 1609-1617.

[3] de Chaumont, F. et al. (2012) Icy: an open bioimage informatics platform for extended reproducible research, Nature Methods, 9, pp. 690-696

[4] F. Luisier, C. Vonesch, T. Blu, M. Unser, "Fast Interscale Wavelet Denoising of Poisson-corrupted Images", Signal Processing, vol. 90, no. 2, pp. 415-427, February 2010.