COMPARE
CENTRE OF MEMBRANE PROTEINS AND RECEPTORS

Image Analysis with Fiji Exercises

Section 1: Basics

A. Opening and displaying microscopy file formats

- Open metaphaseSpread.czi using the Bio-Formats Importer Plugin (Plugins -> Bio-Formats -> Bio-Formats Importer). Change "Color Mode" to Grayscale and untick "Autoscale". Leave other inputs with default settings.
    - **Note we have used the Bio-Formats plugin directly to load the Zeiss .czi file. 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!).**
    - **Note this is 16bit image file but the dynamic range of the detector was only 14bit!**
- 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 (0-16383 for a 14-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 Image -> Properties….
- Add a scale bar using Analyze -> 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.**
    - **Now change to a "Cyan" LUT. Does the scale bar stay the same color? Why can't you have a white scale bar on a cyan LUT? If you want to do this first convert to RGB (Image -> Type -> RGB Color).**
- Once you are satisfied with the display settings save the image as a TIF using File -> Save As -> Tiff…

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 diMaio_z_stack.tif

    - **The data has 2 fluorescent channels and 56 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.**

D. Measurements

- Open diMaio_airyscan.czi. Plot a histogram of the first channel (mitochondria) using Analyse -> Histogram.

- Select the second channel (tubulin) and draw a line across one a few of the microtubules (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 Analyze -> Set Measurments …**

E. Challenge: Volume Viewer

- Update Fiji and install the Clear Volume plugin; Help -> Update… then select Manage Updates Sites and tick ClearVolume, close update sites window and select Apply Changes. Finally restart Fiji.

- Open diMaio_z_stack.tif and/or image t1-head.tif (Image -> Open Samples -> T1 head (16 -bits)).

- Use ClearVolume to perform an interactive volume rendering (Plugins -> ClearVolume -> Open in ClearVolume). This may not work if your graphics card is incompatible with the plugin. The instructors can perform a demo if this is the case. Experiment with the various visualisation settings and options.

- You can also try the 3D Viewer and Volume Viewer plugins if you want which have similar functionality.

F. Challenge: The ROI Manager

- Multiple regions of interest can be handled simultaneously in FIJI with the ROI manager.

    - **Open an image and draw a rectangular region. Add this region to the ROI manager by pressing ctrl+t. Also add a line region and check "Show all" to see both the line and rectangle.**

    - **Experiment with the different features of the ROI manager including the "Measure" tool. More information can be found at https://imagej.nih.gov/ij/docs/guide/146-30.html.**

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 [2]. 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.
    - **Ensure "Scaled Units" is NOT checked. 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. 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 (http://www.codesolorzano.com/Challenges/CTC/Datasets.html) [4]. Open one of the images and develop an analysis pipeline 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:

```
// Loop through the 6 files in the directory
for  (i = 1; i < 7; i = i + 1)  {

        // Open each 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);

- Change one of the file names in the ND2H-SIM directory. If you run you macro now, will it crash? Why?

- Use the getFileList(directory) function to retrieve all file names within a folder. Store the directory and file names as variables:

```
// Turn on batch processing
setBatchMode(true);

// Store the directory as a variable
directory = "C:\\Users\\pike01\\Documents\\Reports\\FIJI Course\\sample data\\ND2H-SIM\\";

// Get the names of all files in given directory
fileNames = getFileList(directory);

// Loop through the number of files in the directory, regardless of how many there are
for  (i = 0; i < fileNames.length; i = i + 1)  {

        // Open each image
        open( directory + fileNames[i] );

        Remaining workflow commands go here

        // Close all windows
        run("Close All");
}
```

  - **Does this new macro work with the modified file name?**

  - **Note the run("Close All"); command has been added to prevent build-up of data in the computer's memory. This is need even if batch mode is on.**

D. Conditional Statements

- Create a text document and save it in the ND2H-SIM directory. Try running the macro from the previous section. Why does it crash?

- Modify the macro using an conditional if statement such that the macro only opens files with the .tif extension:

```
// Turn on batch processing
setBatchMode(true);

// Store the directory as a variable
directory = "C:\\Users\\pike01\\Documents\\Reports\\FIJI Course\\sample data\\ND2H-SIM\\";

// Store the file extension as a variable
extension = ".tif";

// Get the names of all files in given directory
fileNames = getFileList(directory);

// Loop through the number of files in the directory, regardless of how many there are
for  (i = 0; i < fileNames.length; i = i + 1)  {

        // Only open if it's a .tif file open and process it
        if (endsWith(fileNames[i], extension) == true) {

                // Open each image
                open( directory + fileNames[i] );

                Remaining workflow commands go here

                // Close all windows
                run("Close All");
        }
}
```

- **Note this is a very useful framework for batch processing files. Just record and insert the commands you want and specify the directory ☺**

E. Challenge

- Write a macro which labels each slice in a z-stack (for example 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. The functions you need can be found using Help -> Macro functions.**
    - **Hint: the getDimensions(width, height, channels, slices, frames), Stack.setSlice(n), makeText(string, x, y) and run("Draw", "slice") commands could be useful!**
- Write macros to automate your pipelines from section 4.

Section 4: Macros for Colocalization Analysis

A. Calculating the Manders' coefficients with a macro

- Open and inspect coloc_simulated/Set1_ColocalizationSimulation_0.tif. This is a two channel 3D volume of spot like objects where roughly 70% of the objects are perfectly colocalizing. We know this because the data is simulated.

- Open the coloc_simulated/simple_coloc_macro.ijm macro within the script editor. This macro will ask the user to specify a two-channel file. The macro will then segment the signal in each channel (user a Gaussian filter and Otsu thresholding) and use these segmentations to calculate the Manders' coefficients. Have a look through the code and see if you can understand all the steps.

- Run the macro on Set1_ColocalizationSimulation_0.tif and inspect the results. Do the values calculated for the Manders' coefficient seem reasonable and match the data (roughly 70% colocalized).

B. Calculating the Manders' coefficients with the Coloc 2 plugin

- Open coloc_simulated/Set1_ColocalizationSimulation_0.tif and split the channels (Image -> Color -> Split Channels).

- Run the Coloc 2 plugin channels (Analyze -> Colocalization -> Coloc 2). Select the two channels and use the Costes' method for threshold regression. Select the Manders' correlation. For out purposes select all other tick-boxes are optional and can be deselected.

- Inspect the Manders' coefficients calculated by Coloc 2. Are these consistent with those calculated by the macro, do they seem reasonable and match the data? If not why might this be the case?

C. Adapting the macro for batch processing

- Open the simple_coloc_macro.ijm macro within the script editor. Use what you learned from the previous section to adapt the macro for batch processing. The template found at Templates->Image J1.x -> Batch -> Process Folder (ImageJ macro) may help to speed this process up. You can find a solution on the course website.

- Use this macro to batch process all images within the coloc_simulated directory.

References

[1] Royer, Loic A., et al. "ClearVolume: open-source live 3D visualization for light-sheet microscopy." Nature methods 12.6 (2015): 480-481.

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