Practice : In the “Q5/rawdata” folder, there are 7 .tif files, each containing 3 channels (Green/Red/Blue). Our goal: write a macro processing those 7 files sequentially. For each file, we discard green and red channels, segment blue channel (DAPI for nucleus), use “Analyze Particles” and “ROI Manager” to measure size and mean intensity for each detected nuclear region and save the result file into “Q1/result folder”. We will do it in 5 steps.

Step I : Start FIJI, open one file (1.tif) in the FIJI, separate channels, delete green and red channel. Examine the blue channel, figure out a good protocol to process the data. Background subtraction? Mean/median/gaussian blur filter? What level of thresholding? Watershed after thresholding? Use “Analyze” -> “analyze particles” to segment the nuclear regions. See if the ROI selection did a *reasonable* job. Write down your protocol.

Lai Ding’s protocol is : open file; split channels, delete green/red channels; apply “Mean” filter with radius of 2 on blue channel; thresholding the blue channel at intensity level of 22; convert to binary mask; fill holes; watershed the mask; apply “analyze particles”, set size as 600 – infinity in pixel unit, add to ROI manager; Lai Ding feels the ROI result is quite good. Save the ROI selection in “Q1/result folder“, delete all the ROI selections; Finally, close blue channel, only left ROI manager open.

Step II. Record your protocol using “macro recorder”

Lai Ding’s code “Lai Ding code1” is saved in the “Q1/Lai Ding Code” folder

Note that:

1. in Lai Ding’s code, since after the channel split, the blue channel show up at the top. It is kind of tricky to delete the green/red channel in macro. So Lai decided to delete all 3 channels at the end of the macro (the 3 “close()” commands).

2. the macro is recorded on Lai’s notebook, so the file path lines are different from what you would see.

3. Close the ROI Manager window may not show up in macro recorder. The command is roiManager(“delete”);

try adjusting the file lines (open(…) and save command) and open 2.lsm or other files, see if the ROI selection is reasonable. Compare results from your protocol and Lai’s. After testing on several .lsm files, if you believe the ROI selection works fine, save the code as code1 in “Q1/code” folder, not in “Q1/Lai Ding Code” folder.

Step III: now we feel comfortable on our protocol. We will let it batch process the 7 files in “Q1/rawdata” folder. We will write our code2 macro based on the code1 macro. Keep a copy of your code1 macro in case you need to start again (most of the time you DO).

The command for assigning a folder to a variable is getDirectory(“ …. “); (case sensitive) We can store the rawdata and result folder into two variables as below.

```
rawDir=getDirectory("Choose Raw Data Folder");
resultDir=getDirectory("Choose result Data Folder");
```

To get file list under one folder and process them sequentially we use the code below:

```
list=getFileList(rawDir);
for(i=0;i<list.length;i++)
 {

   // you code1 here       // makes comment lines

}
```

Remember: you need to change your code1 so that the folder/filename variables replace the absolute file path.

```
list=getFileList(rawDir);
for(i=0;i<list.length;i++)
 {
  //open("D:\\presentation\\ImageJ mini course\\Oct 14th\\Lab\\Q1\\rawdata\\1.tif");
    open(rawDir+list[i]);
   …
   //roiManager("Save", "D:\\presentation\\ImageJ mini course\\Oct
14th\\Lab\\Q1\\result\\1_roi.zip");
   roiManager("Save", resultDir+list[i]+“_roi.zip");
 }
```

save the macro as code2 in “Q1/code”. Run the macro and check the result ROI selections in “Q1/result” folder. Lai Ding’s code 2 is in “Q1/Lai Ding Code” folder.

Be aware : When running the macro repeatedly, the old files in the result folder will be overwritten by the new result files each time you run.

Step IV: by now you have practiced the macro recorder and batch processing files under one folder. Now we will working on getting the measurements.

Save a copy of the your code2 macro.

In code2, after we get the ROI selections, we would like to measure the size and mean intensity of each selection.

We will adjust code2 and insert code sections to do the following: set the measurement parameters; do the measurement; save the measurement as .xls file; close the measurement file. Use the macro recorder to get corresponding command lines and adjust them (use folder/filename variables to replace absolute file path …)

To clear the result window, use the line below
run("Clear Results");
You may need to run the clear result command just before the measurement, and once again after the measurement.

Run code3. You should see 7 .xls files show up in the result folder. Check the result file to see if there is a problem. Lai Ding's code3 is in "Q1/Lai Ding Code" folder.

Step V: There is a problem. The mean values are all 255. Why? Because we are measuring the "MASKED" blue channel. We should measure the original blue channel.

How to do that? There are many methods. Here we use a simple one. We duplicate the original blue channel before the processing starts (before the thresholding, and before any filtering occurs). Get the duplicate command through macro recorder.

In ImageJ macro language, we use selectImage(….) to choose which image to operate on. In our case, we should give separated names to the original blue channel and the duplicated one. The command is rename(..)

In code3, after we split the channels, the blue channel is on top. We will rename it to "blue0": rename("blue0"); then we duplicate the blue channel and name the duplicate one "blue1" : run("Duplicate...", "title=blue1");

You can acquire these two command lines through the macro recorder. "rename" and "duplicate" function is under "Image" tab.

We process "blue1" to get the ROI selections, then close "blue1" (already a binary mask file), apply ROI selection on "blue0" to get measurement.

Lai Ding's code4 is in "Q1/Lai Ding Code"

Additional step: we may want to get a file that summaries the results. The text file will be like the following:

Filename total_cell_count

1.lsm 193
2.lsm 200
….

Use “print(..) “ to write text on Log window. Before the for loop starts we will add
print(“filename total_cell_count”);
Inside the loop, after analyzing particles, get total cell count by
cell_count=roiManager(“count”); then add line print(list[i]+” “+cell_count);

Save the log window text by
selectWindow(“Log”);
saveAs("Text","D:\\presentation\\ImageJ mini course\\Oct 14th\\Lab\\Q1\\result\\summary.xls");

Run the code5, check the result summary.xls file in “Q1/result” folder

Finally, we will try to retrieve information from result window.

After we generate the result window with area and mean intensity of each nuclei, we would like to calculate the total intensity in all detected nuclei in the image.

The code is list as below, also check Lai Ding code6.

```
//code6 section
  total_intensity=0;  // initialize
  for(j=0;j<cell_count;j++) // i is already been used, we use j now
   {
    tmpx=getResult("Area",j);  //store jth area in tmpx
    tmpy=getResult("Mean",j);  //store jth mean in tmpy
    total_intensity=total_intensity+tmpx*tmpy;  // add jth intensity to total intensity

   }
 //
```

Lai's final code6 is in "Q1/Lai Ding Code".