



- Motivation
- The Batch Processor
- Custom Batch Process

In this video, we will cover how to process a folder of images, or how to do batch processing. We will discuss a bit about the motivation, though it should hopefully be clear by now. Then we will cover Fiji's built in Batch Processor. Finally we will see some limitations of the batch processor and how we could potentially overcome them with a custom script.

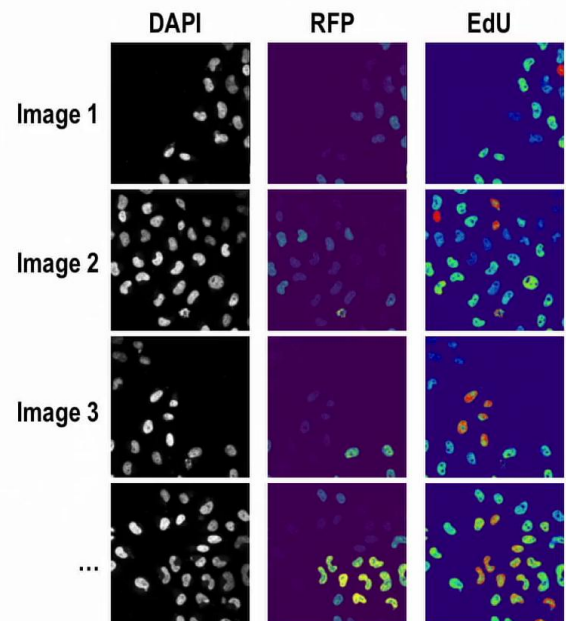
Notes

Summary



0m 08s

It's Never Just One Image



Over the last weeks we covered how to get objects and measurements from images. Now, suppose we have a nice and clean workflow that was setup for the images we tested. But let's face it. An experiment is not just one image. Depending on what you are analysing, you might need 10, 50, 100 images. And I can imagine that the prospect of manually running your gorgeous workflow on each image by hand does not sound too appealing.

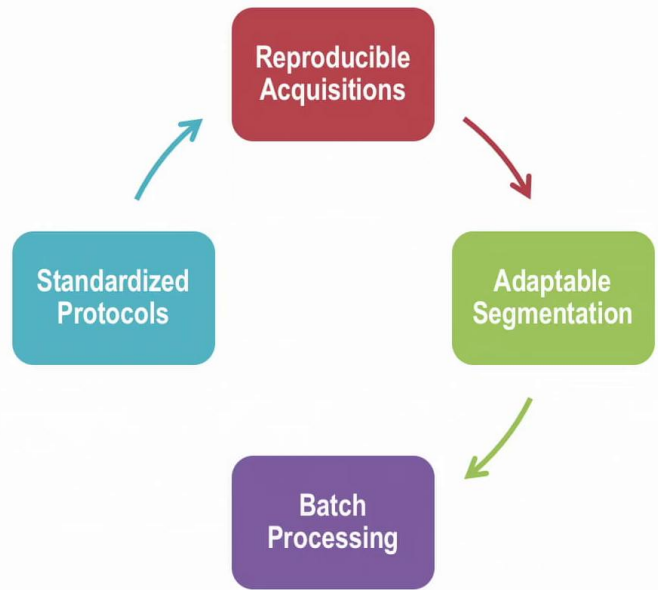
Notes

Summary



0m 30s

The Need For Standardization



So, this is why we need batch processing. However, we need to setup a few preconditions for it to work seamlessly. Having strict protocols for your samples, and knowing your instruments allows for reproducible acquisitions. These can, in turn, be processed automatically by building segmentation workflows that are adaptable to the unavoidable variations that will occur. We make these algorithms robust enough that they can be applied in batch to a whole series of images. In turn, this allows for other people to reproduce, not just your results, but the workflow that allowed you to reach those results. And then, you can only do batch processing when these other criteria are met, because otherwise they might fail.

Notes

Summary

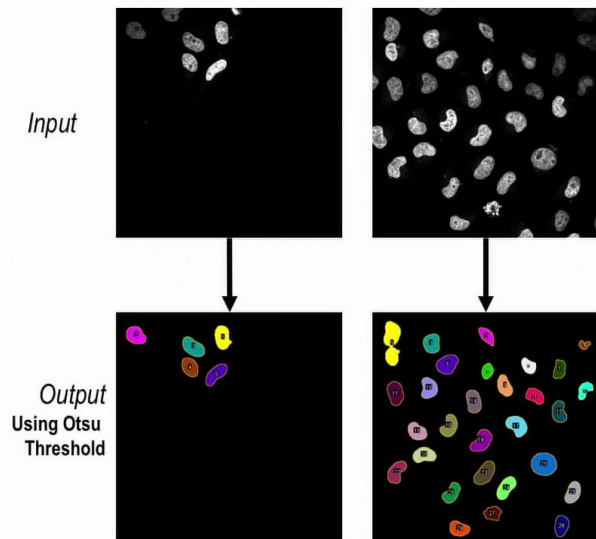


0m 56s

Batch: What do we Need?



• Output for Quality Control



Supposing we have met these conditions, let's see what we need to make our general analysis script batch compatible. We need to ensure that our process will have an output for visual inspection. Otherwise, having only tabular data will not tell us whether the segmentation worked or not. In this example, the workflow uses the MaxEntropy auto threshold method, which seems to work very well for the image on the left. But if we take another image and run it through the same workflow, we can clearly observe that something, somewhere has gone terribly wrong. Thanks to the visual output we produced here, quality control is simple, and we can correct the issue by, for example, selecting another threshold method that would be more adapted for what we need. Such as the Otsu method here.

Notes

Summary



1m 44s

Batch: What do we Need?



- Image Name
- Image Channel
- Measurements

Label	Channel	Area	Mean
Image 1: ROI 1	1	23.2	210.2
Image 1: ROI 1	2	23.2	158.3
Image 1: ROI 1	3	23.2	53.7
Image 1: ROI 2	1	34.5	123.2
Image 1: ROI 2	2	34.5	173.0
...

So, that's it for the image output. But we have some criteria that we need for our output results tables. We need to make sure that for each result row, we can find the corresponding image that created this result, and the ROI we would also like a single result table for the whole analysis instead of one for each image. If we set up our results table to contain the image name, the ROI name, the image channel and the measurements our script will be able to append the results of new images as it goes through them.

Notes

Summary



2m 35s

Batch: What do we Need?



- Image Name
- Image Channel
- Measurements

Label	Channel	
Image 1: ROI 1	1	2
Image 1: ROI 1	2	2
Image 1: ROI 1	3	2
Image 1: ROI 2	1	3
Image 1: ROI 2	2	3
...

Set Measurements

☒Area

☐Standard deviation

☐Min & max gray value

☐Center of mass

☐Bounding rectangle

☒Shape descriptors

☐Integrated density

☐Skewness

☐Area fraction

☒Mean gray value

☐Modal gray value

☐Centroid

☐Perimeter

☐Fit ellipse

☐Feret's diameter

☐Median

☐Kurtosis

☒Stack position

☐Limit to threshold

☒Display label

☐Invert Y coordinates

☐Scientific notation

☐Add to overlay

☐NaN empty cells

Redirect to:

None

Decimal places (0-9):

3

OK

Cancel

Help

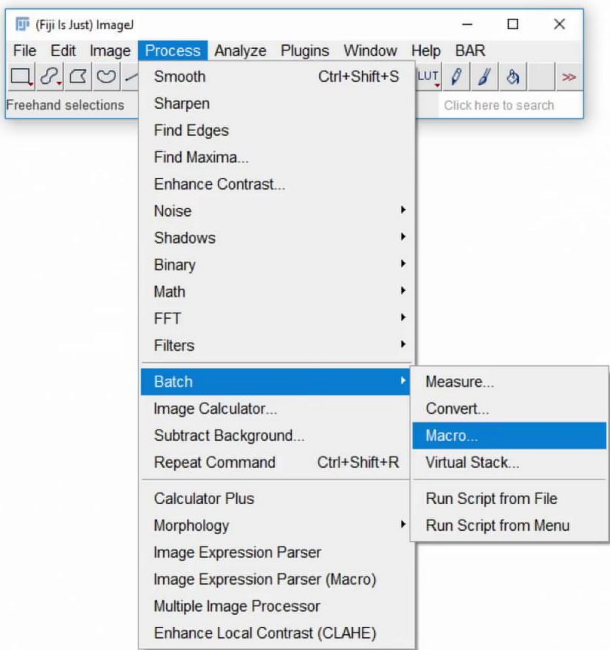
We've seen this before when talking about measurements. We can achieve this by simply checking the stack position and display label boxes from the Set Measurements Command. In regards to the regions of interest, the ROI manager will append the name of the region to the label automatically when we measure it. So, just like before, now all our conditions are met.

Notes

Summary



Batch Process Macro



In Fiji, the basic batch processor can be found in the Process > Batch menu. We are interested in the batch processor that runs a macro. The layout is quite simple.

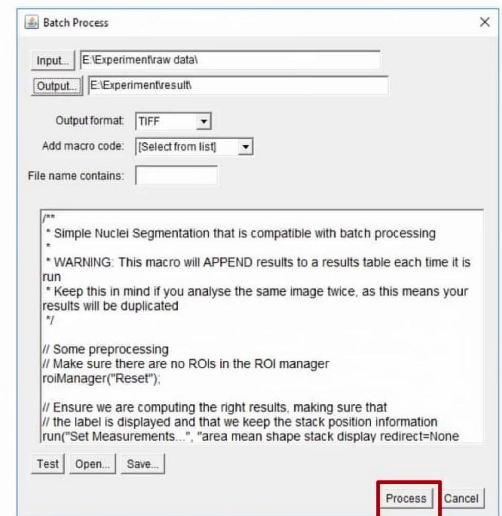
Notes

Summary

3m 34s



Batch Process Macro



We need to provide an input folder and an output folder. Note that the input folder should only contain your images and not other files, and that if your output folder doesn't exist, you're going to have to create it before starting. After you've defined the input folders and output folders, your macro code can be pasted in the text area here or imported using the open button. We are now ready to run this code on the folder with simply pressing the Process button.

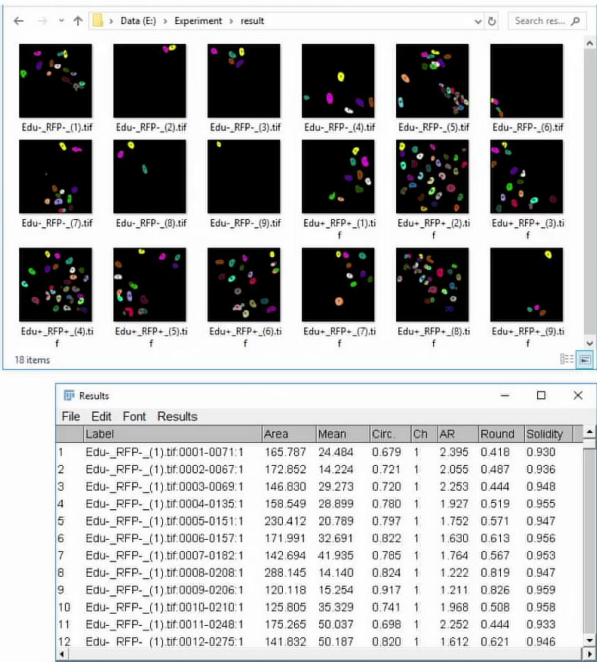
Notes

Summary



3m 46s

Batch Process Macro



After a few minutes of processing, our output folder, in this case, will contain the segmented particles, which we can use to visually inspect the results. In Fiji, the results table has all the measurements for all the images, which we now need to save manually.

Notes

Summary

4m 18s



Batch Process Macro Limitations



- Will not save your results table
- Will open ALL files in the input directory.
 - Might cause bugs
- **Overwrites** your images if input and output folders are the same!!

Unfortunately this process has some limitations. As we saw, your results will not be saved automatically. It also tries to open all the files on the input folder, regardless of the extension, which might cause errors, or bugs. More importantly, if you accidentally set your input and output folder to be the same, it will overwrite your data, because it resaves the file with exactly the same name! So be careful.

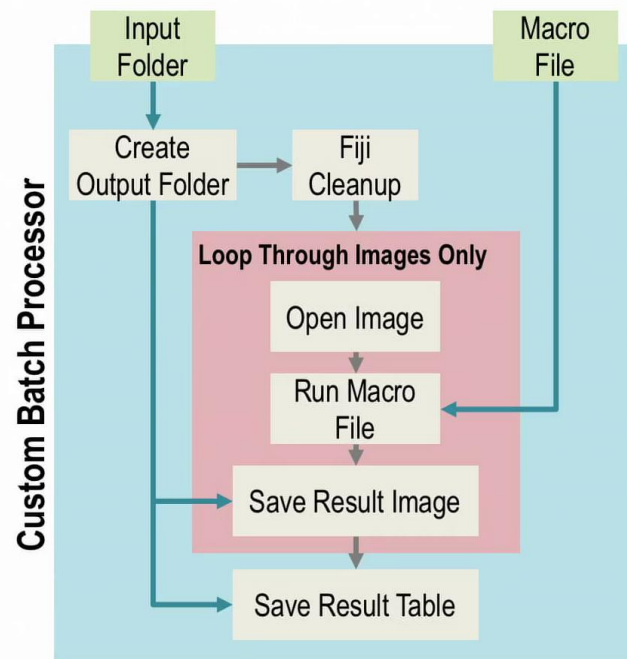
Notes

Summary



4m 35s

Creating Our Own Batch Processor



To overcome this, we can try to use our own custom batch processing script. It's got a few improvements over the standard one. Now, the only input we would like would be simply the input folder, and the macro file that we want to run on each image. We can make the script create the output folder automatically, and do some pre-processing cleaning steps, like resetting the ROI manager, closing all open images, or clearing any previous result tables. After that, we'll make the script loop through all the image files. It opens one image, and will then apply the macro on that current image save the result in the output folder, and, finally, after processing all the images, it can make sure to save the results as well.

Notes

Summary



5m 02s

Conclusion



- Motivation
 - Processing Large Amounts of Images
- The Batch Processor
 - Limitations
- Custom Batch Process
 - To Help Overcome Limitations

So, in today's video, we've seen how to perform batch measurements in Fiji using the built-in batch processor as well as how to do it with a custom script. This is one of the most powerful aspects of image analysis and the reason we've gone through all this work to be reproducible and consistent. Now that our workflow is established, we can easily apply it to your entire dataset in just a few clicks. Thank you.

Notes

Summary



5m 46s