

pomBseen and MATLAB

pomBseen is a program written in MATLAB to automate the detection and measurement of *S. pombe* cells in multi-channel images. pomBseen was written specifically for images taken on a DeltaVision fluorescence microscope. pomBseen automatically measures cell length and mean nuclear and cellular fluorescence, and then saves the data for each segmented cell.

Downloading MATLAB and toolboxes necessary to run pomBseen

Go to the following website to download the latest version of MATLAB:

<https://www.mathworks.com/downloads/>

Follow the standard installation process for your operating system.

During MATLAB installation you will have the opportunity to add toolboxes, which are packages that allow you to do specific types of analyses.

pomBseen requires that you select the “Image Analysis” toolbox. If you miss it or are not given the option during the installation, you can go here and add it after the fact:

<https://www.mathworks.com/products/image.html>

One more custom package you will need is called Bio-formats – which is a tool developed by a microscopy standards group to work with various microscopy image formats (companies often develop their own format). Go here to download the Bio-formats package for MATLAB:

<https://www.openmicroscopy.org/bio-formats/downloads/>

Click on the Download icon labeled: “MATLAB Toolbox”

Keep track of where your MATLAB and Bio-formats files in the “bfmatlab” folder are downloaded to – you might need to tell MATLAB where these files are located to run pomBseen. pomBseen works best if the “bfmatlab” folder is saved in the same folder as the MATLAB folder, and the path is set to that location. For example, I have the following MATLAB folders:

/Users/makotoohira/Documents/MATLAB/bfmatlab

When I open the MATLAB environment, I want to set the path to that folder location:

- Go to the “Home” tab
- In the “Home tab go to the “Environment” menu
- Under “Environment” click on “Set Path”
- In the “Set Path” dialog box click on “Add Folder”
- Navigate to your “bfmatlab” folder location and click on that folder
- Click “Open”
- Click “Save”

Now you should be ready to run pomBseen.

If you do not have MATLAB and still want to use pomBseen, we have a MATLAB-generated standalone app which installs a limited version of MATLAB (called MATLAB Runtime) onto your machine along with the standalone pomBseen app. Rather than downloading the program files, click on pomBseenAppInstaller_web_for_Macs.app.zip or pomBseenAppInstaller_web_for_Windows.app.zip depending on which OS you have. The installer walks you through the installation of MATLAB Runtime and pomBseen.

Getting the pomBseen code

Go to Github and download the latest version of pomBseen here:
<https://github.com/makotojohira/pomBseen/releases>

Standalone app

If you do not have MATLAB and still want to use pomBseen, we have a MATLAB-generated standalone app which installs a limited version of MATLAB (called MATLAB Runtime) onto your machine along with the standalone pomBseen app. Rather than downloading the program files, click on pomBseenAppInstaller_web_for_Macs.app.zip or pomBseenAppInstaller_web_for_Windows.app.zip depending on which OS you have. The installer walks you through the installation of MATLAB Runtime and pomBseen.

Currently, one needs to launch the standalone app for each file to be analyzed. We are working to allow multiple files to be analyzed without quitting and relaunching.

Microscopy

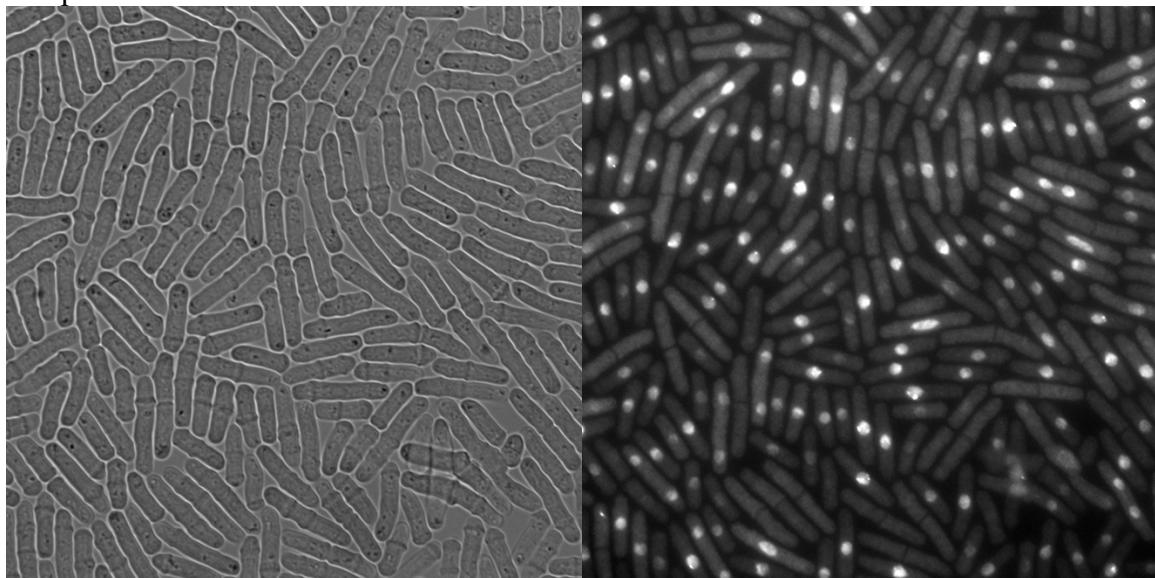
pomBseen was written for DeltaVision fluorescent micrographs. The DeltaVision system can take up to four channels of images but pomBseen requires that you take two or three. Channel number 1 must be assigned to the bright field (BF) image, and channels 2 and 3 must be fluorescent images. The software controlling the scope will save images from multiple channels in a single DV file.

The prescribed order of channels is important because pomBseen will split the DV file into as many separate data files as channels used, and will save and analyze them in order.

Note: When you save microscopy data, make sure you save both DV and DV log files for each image. The DV log file contains information on the parameters with which your image was taken and you may need that information later. One key piece of information in the DV log file is the range of intensity output. Images that contain the maximum intensity value (65,534 for 16-bit images) have truncated your intensity range, and therefore will not accurately reflect your intensity statistics. Discard such images from your analysis pipeline.

pomBseen requires that the BF image must be slightly de-focused in such a way that each cell is defined by a bright halo – a thin outline which is brighter than either the cell or the background. This halo is important since it is specifically what PomBseen detects and uses to identify each cell. Cells which are not fully outlined will eventually be eliminated during various filtering operations that occur in pomBseen. A few cells are expected to be lost to these filtering operations, more if the image quality is not good (such as excessive background variability or noise, or poor halo continuity or focus).

The following (left) is an example of how the cells and halos in the BF channel should look when you save the image. The FITC channel of the same field of view is shown on the right for comparison.



Overview of PomBseen

The flow of pomBseen is described as follows and is illustrated in Figure 1 (numbers correspond to those in the figure):

1. import user-selected DeltaVision file and save TIFF files of each channel
2. extract bright-field image (channel 1) and sharpen the focus
3. apply Otsu's thresholding method and convert the gray-scale bright-field image to a black-and-white image where the halo around each cell is emphasized
4. invert the BW image so the interior of each cell is white
5. eliminate cells touching the border since length measurements may not be accurate for those cells; segment and number each cell
6. apply a convexity filter to eliminate any shapes which are not likely to be pombe cells
7. re-segment and re-number cells
8. allow user select cells for deletion
9. extract fluorescent image (channel 2) and calculate the background
10. apply Otsu's thresholding to nuclei

11. segment and number nuclei
12. filter cells based on nucleus:cell ratio; if there is a second fluorescent channel repeat above analysis for channel 3, starting at step 9
13. calculate output data

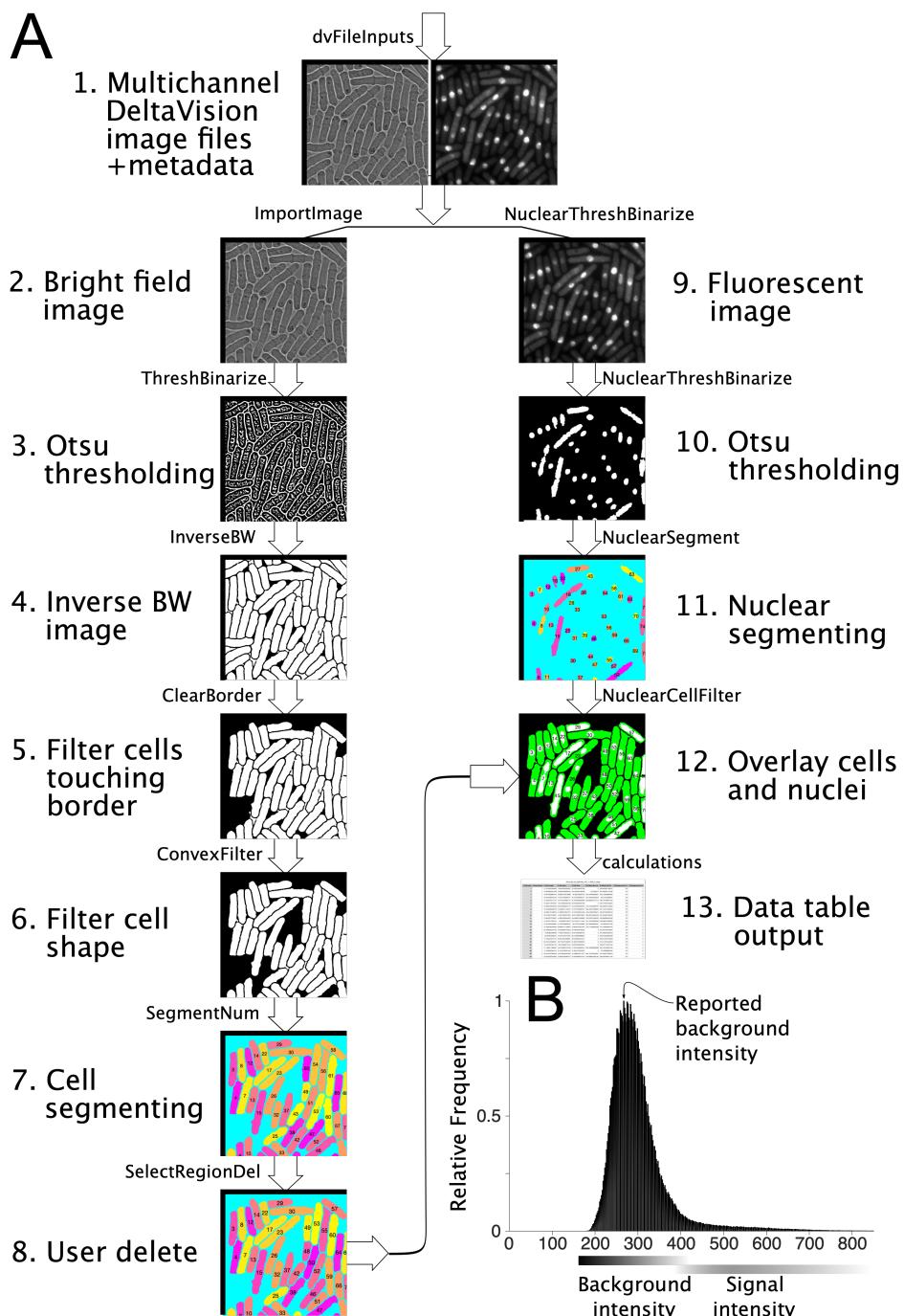


Figure 1. A) Workflow of pomBseen. Only top left quadrant of each figure is shown for clarity. B) Histogram of image intensity showing how background was calculated.
pomBseen outputs a figure at almost every step to verify its functionality and to provide the user with a useful reference if needed. All the output figures will be deleted upon the next run of pomBseen. You are welcome to save any of the figures before pomBseen deletes them. The

only data saved by pomBseen are the TIFF files extracted from the initial DV file, and a final CSV file containing the output data.

You are welcome to comment out the output figures in each function where you do not require them... the figure code looks like the following:

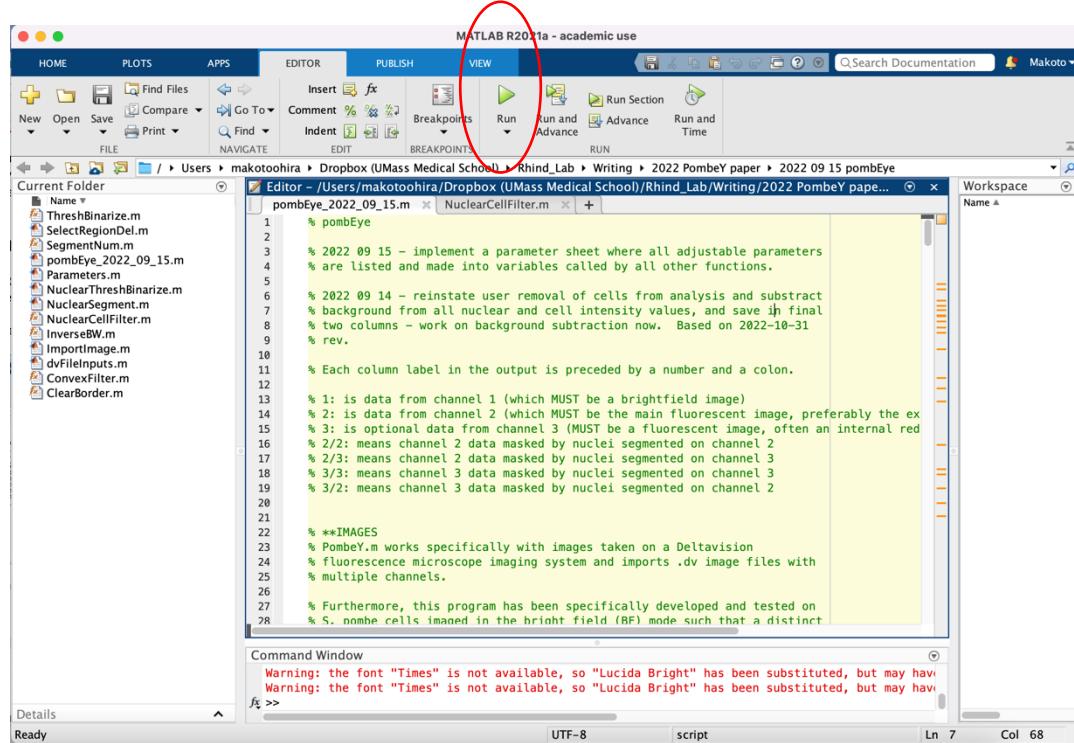
```
figure('Numbertitle', 'off','Name','Function: InverseBW.m and bwareaopen');
imshow(BW1b);
title(tiffFilename, 'Interpreter', 'none');
```

To comment out a section of code, type a % in front of each line that you want to disable, yet retain in case you want to reverse the operation. The line will turn green, indicating that it will no longer be read as MATLAB code.

Working with PomBseen

If you have installed MATLAB and the toolboxes and the pomBseen code, you should be able to click on the file “pomBseen_2022_09_15.m” and you will pop right into the MATLAB editor.

You will see the actual code for PomBseen and your screen should look similar to what is shown below:

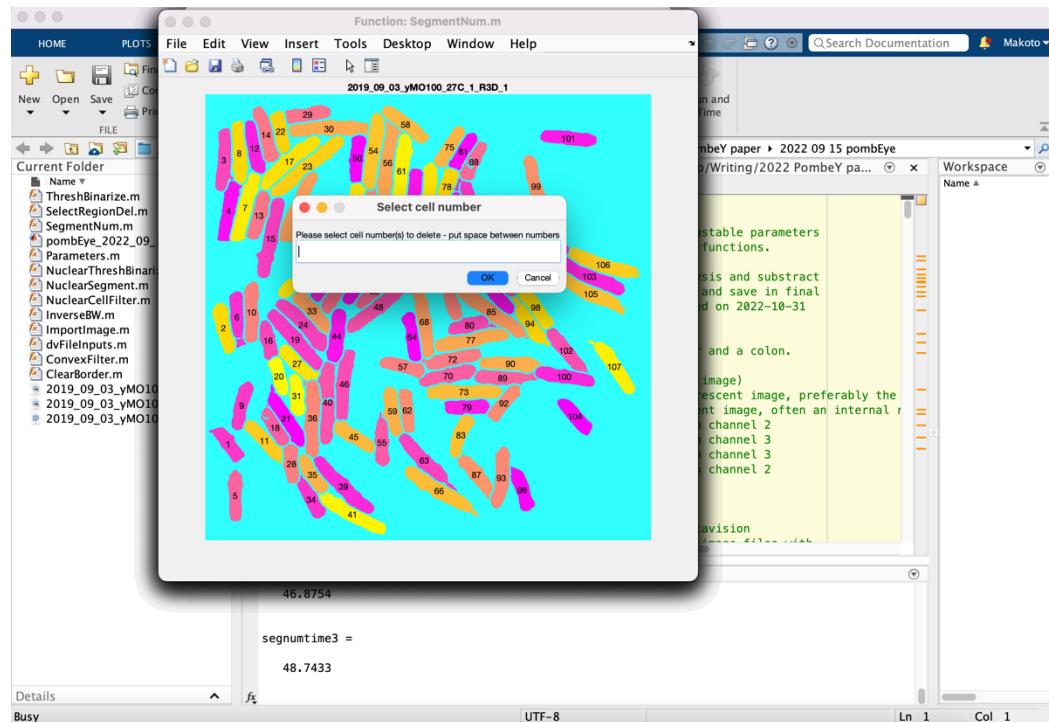


Before you run the program you will have to set the folder with the location of your DV files. The window to the left in the screenshot is the “Current Folder” window. If the folder does not

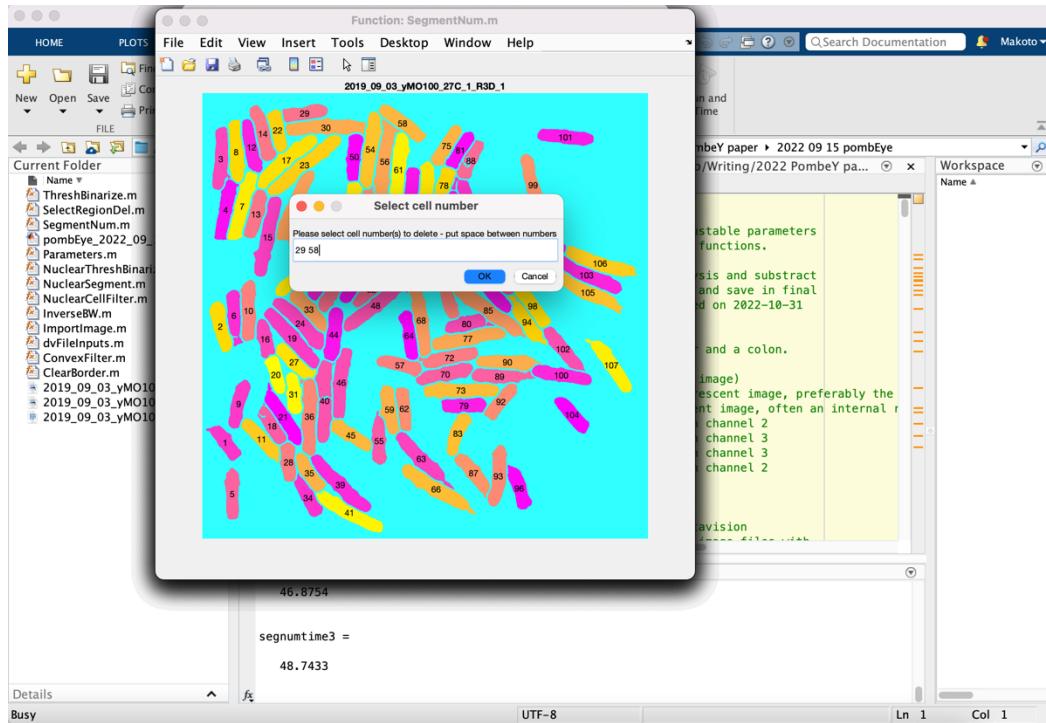
contain your files, use the folder icons above the window to navigate to your folder and click on it.

To run the program, merely click the green “Run” button circled in red above in the “Editor” tab. The whole program will run automatically with a couple points where the user is prompted for input.

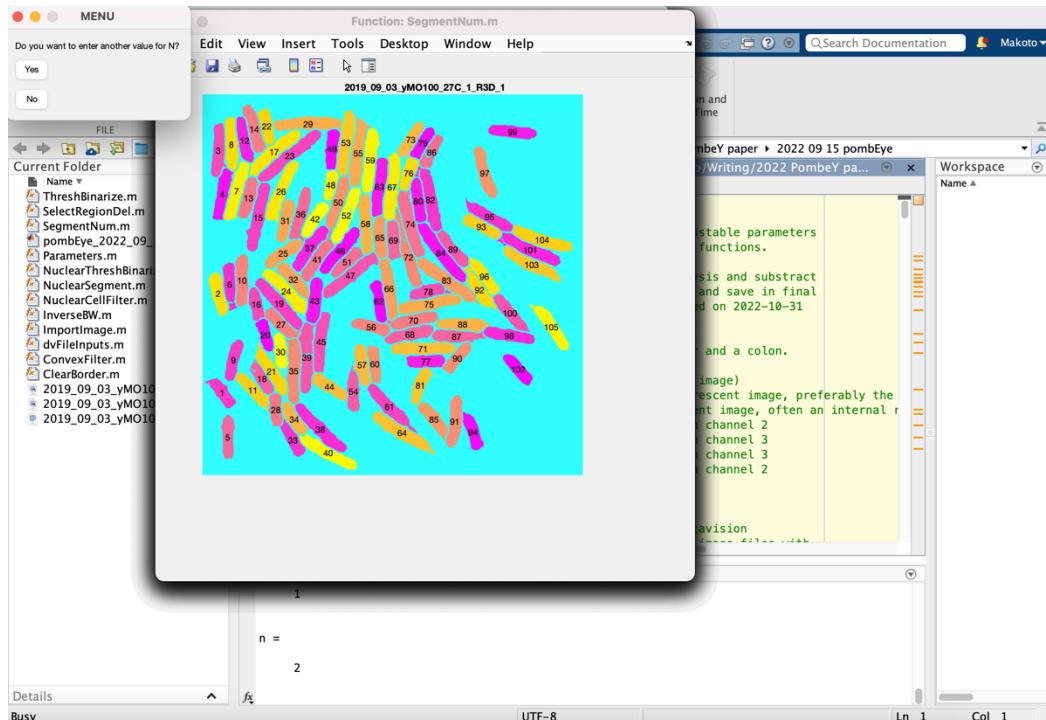
pomBseen runs automatically, but asks the user at one point whether or not to delete any cells from further analysis, and a dialog box appears as shown below:



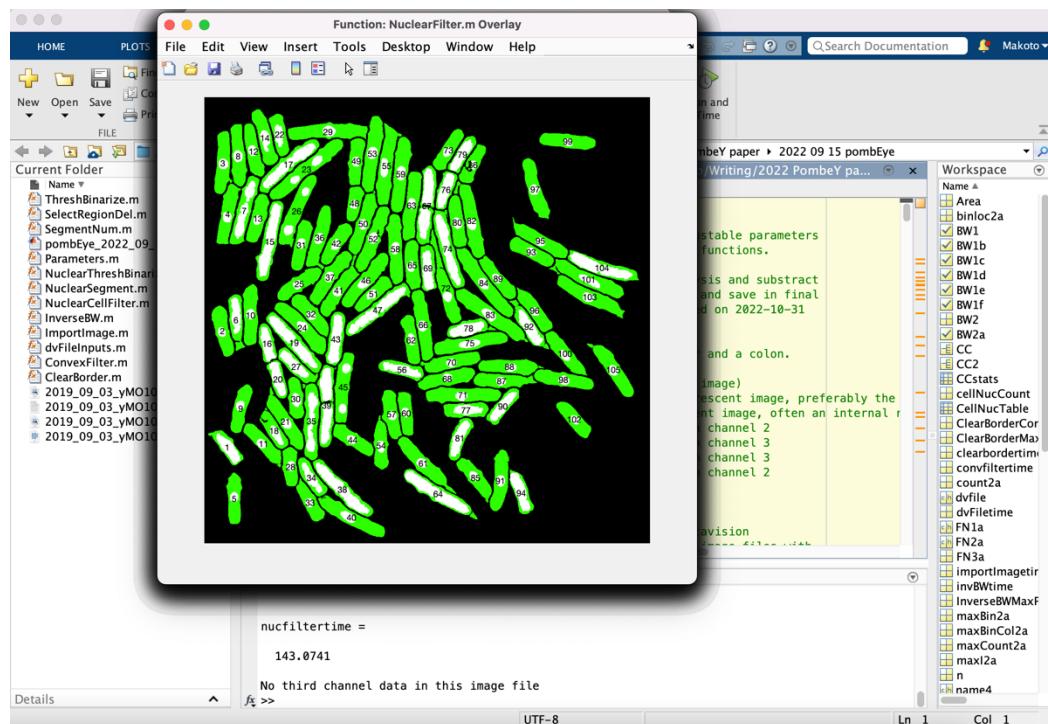
The image immediately behind the dialog box shows the last segmentation and numbering of each cell. If the user, in this example, decides for some reason to delete cells 29 and 58, the user enters those numbers as shown below:



When the user clicks on OK, pomBseen re-segments and re-numbers the cells, and a new dialog box appears asking if the user wants to delete any other cells. Note that cells 29 and 58 above are no longer in the figure below and cells 25 and above have been re-numbered:



If the user selects No, then pomBseen continues the remaining analysis until it saves the output data in a CSV file. The final image will be an overlay of segmented cells and nuclei. In this case, the image was a 2-channel image, and no third channel was analyzed. A message appears in the MATLAB command window (below the figure) indicating that there was no image in the 3rd channel, as shown below:



In this example, as seen in the figure above, there were a total of 105 cells with segmented nuclei. Many of the cells, such as numbers 2, 3, 4, and 5 show a small, round well-defined nucleus (white) near the center of the cell (green), as expected. Some cells, such as numbers 1, 6, and 7, show a much larger irregular region segmented as a nucleus (white). These regions are filtered out if the segmented nuclear area is greater than 20% of the segmented cell area. The intensity of those regions is reported as zero and the user may sort and discard that data. If the user opts to use whole cell fluorescence instead of nuclear fluorescence, that data is available and is not reported as zero. A few cells, such as numbers 23 and 26, have two nuclei as expected in post-mitotic cells.

The CSV file looks similar to the example shown below:

The first column shows the index or number of the cell, corresponding to the numbers reported in the final figure with superimposed nucleus and cell (see Figure 1, insert number 12). The data in a given row belong to the cell whose number is reported in column 1 of the same row. The user may choose to save that final figure (or two figures for three-channel DV files) for reference.

The second column reports the number of nuclei segmented in each cell. Most cells should have a single nucleus per cell, but post-mitotic cells will have two nuclei. pomBseen should successfully segment multiple nuclei if the fluorescent protein is highly expressed and evenly distributed throughout the nucleus and minimally in the cytoplasm (see Figure 1, for example cell numbers 23 and 26).

The third column reports the major axis length of the cell, the fourth reports the nuclear area, the fifth the cell area, the sixth the mean nuclear intensity, the seventh the whole cell intensity, and so on (Table 1).

The header for each column shows a numerical prefix followed by a colon and the title of the data in the column. The numerical prefix shows the channel from which the data is generated. For example, in the first column, the cell index always derives from the brightfield image in channel 1. The number of nuclei in the second column is culled from the nuclear fluorescence image in channel 2.

Several columns have two numbers arranged like a fraction. For example, the title of the 6th column is: 2/2: Mean Nuc Int. The first number denotes the channel from which mean nuclear intensity is calculated. The second number denotes the channel by which the nucleus was

segmented. So, the 6th column title refers to: mean nuclear intensity of nuclei imaged in channel 2, and segmented from channel 2.

The reason for this approach is that nuclei may not be well-segmented in both fluorescent channels. pomBseen calculates and reports each channel's mean nuclear fluorescence using the nuclei segmented from each channel, for a total of four permutations of data. The user can select the combination which is most appropriate.

Sometimes, pomBseen is unable to segment nuclei for some fluorescently labeled proteins (perhaps due to very low nuclear expression or excessive cytoplasmic concentration). For this reason, we also report whole cell fluorescence intensity. In this case, masking is done using the more reliable brightfield image of the whole cell in channel 1.