1. Strategy of making a customized macro
2. Figure out the workflow of image processing in the user interface of imageJ/Fiji.
3. Use the “macro record” function to log the procedures that are used to generate the mask, and save it as a temporary macro code.
4. Modify the temporary macro code into a generic code.

Please refers to the instructions of macro language on

<https://imagej.nih.gov/ij/developer/macro/macros.html>

The built-in function can be deployed in imageJ macro can be found as below:

<https://imagej.nih.gov/ij/developer/macro/functions.html>

1. The structure of the macros provided.

* For instance, “donut.ijm” contains three macros which can be loaded at once and be called independently. They are “toxoDonut”, “cleanWindows”, and “EMTmask”.

1. The procedures in the macro “toxoDonut”

|  |  |
| --- | --- |
| Lines 9-20 | Read the information of the original image |
| Line 21-122 | Try all built-in auto-thresholding functions in imageJ/Fiji. Sort the auto-thresholding values. Study the images with manual thresholding and define an ideal mask with its relative thresholding value. Figure out a linear equation by imputing one or two auto-thresholding values to match its output with the ideal thresholding value from the manual thresholding. This step is kind of subjective. But choosing a model by mixing the auto-thresholding values is much more reasonable than designating a fixed thresholding value to every image. |
| Line 123-144 | Use the built-in function “analyze particles…” in imageJ/Fiji to exclude extreme small and large particles. A binary image will be generated. |
| Line 145-166. | Dilating on the binary image by several pixels (approximately ~ 1 μm) and generate the “dilate” image. Then subtract the dilate image by the binary image and create a “donut” mask image. |