User manual

MSSegregation

Macro package for Fiji-ImageJ

ver 1.2

by Iztok Dogsa*

*Iztok Dogsa, University of Ljubljana, Biotechnical Faculty, Večna pot 111, SI-1000 Ljubljana, Slovenia, EU.

Email: Iztok.dogsa@bf.uni-lj.si

Software and hardware requirements:

All MSSegregation macros run in Fiji-ImageJ application. Installation of Fiji-Image application before using MSSegregation macros is required (for details see the step-by-step guide of individual macros). The macros have been tested in Windows7/10, 64-bit, Fiji-ImageJ 1.53c

The MSSegregation package was run on two different notebook computers and desktop computers (8Gb RAM, 64-bit Intel/AMD processors). We believe typical modern desktop computer can successfully run our macros. However, handling large image sets requires more powerful computer.

Availability

The ImageJ distribution package FIJI is freely available online: https://imagej.net/Fiji

The MSSegregation package is freely available at https://github.com/xx

Package: MSSegregation

Macro 1: *Convert_to_bin_ver1.2.ijm*

Introduction

This script runs as a macro in Fiji (ImageJ) environment. Its purpose is to split and convert a digital image that depicts a mixture of two components, (two types of particles, particles and background…, called *ch1*, *ch2* particles) into two images in a 8-bit binary format (black and white), where each image represent one type of the particle. The separation is done via user assisted thresholding of pixel values. These images can serve as input for Multiscale spatial segregation analysis implemented in script *MSS_calc_ver1.2* and in *Sim_seg_extremes_ver1.2* for more details refer to part of this manual corresponding to these macros). The stacks of images are supported by the macro.

Step by step guide

1. If you do not already have Fiji, download and install it from: https://imagej.net/software/fiji/downloads
2. Run Fiji (ImageJ-win64.exe file, if you are running 64-bit version).
   If asked to make an update, you can chose »Remind me later«, the update is usually not needed to successfully run *Convert_to_bin_ver1.ijm*

A. Installation

3. In Windows explorer (or equivalent File browser) go to the folder where you have installed Fiji and find plugins folder within Fiji folder. In our case:
   C:\Users\iztokd\Downloads\fiji-win64\Fiji.app\plugins
   Create folder \MSSegregation within plugins folder

   In running Fiji app press Ctrl+shift+M (=Install Macro) and choose a file *Convert_to_bin_ver1.ijm* .

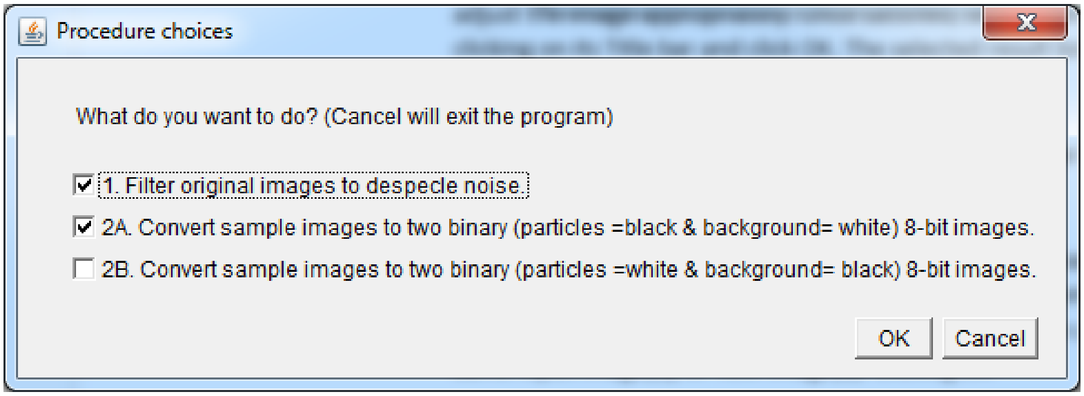
   Save it to directory \MSSegregation, created above.
   Close and restart Fiji app. If asked to save LUTMenuTool.ijm, just say “No”.

4. Now you can run the macro from the Menu Bar: Plugins> MSSegregation>Convert_to_bin_ver1.

B. Direct run of the Macro

3. Open Windows explorer and “drag and drop” the file *Convert_to_bin_ver1.ijm* to the running Fiji application.
4. New window with displaying the code of *Convert_to_bin_ver1.ijm* will open (script editor) and you can run it by pressing Ctrl+R or by choosing Run>Run from menu of the script editor.

5. First you will be asked, What do you want to do?



What do you want to do? (Cancel will exit the program)

☑ 1. Filter original images to despecle noise.

☑ 2A. Convert sample images to two binary (particles =black & background= white) 8-bit images.

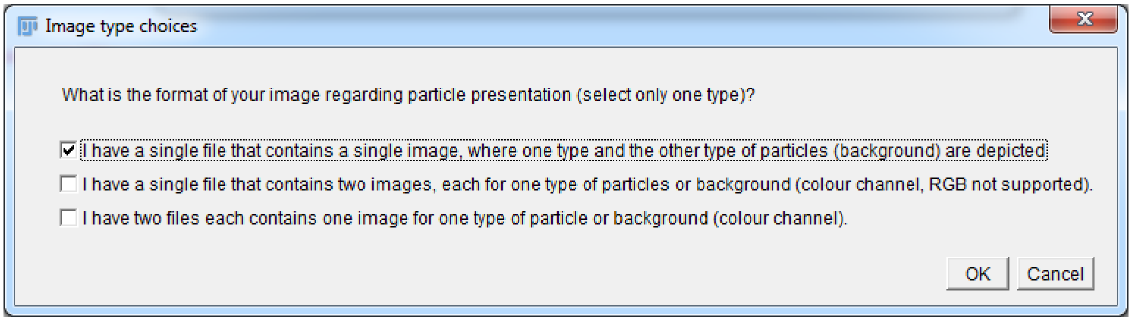
□ 2B. Convert sample images to two binary (particles =white & background= black) 8-bit images.

Option 1:
By default the filtering of original (input images) of pixel noise is selected. Usually this improves the output.

Option 2:
Also, because the main purpose of this macro is to obtain two binary images (8-bit), where on first image the particles of one type are shown (with bit value of 255) and on the second image the particles of other type are shown (with bit value of 255) the option 2 is selected by default. The background or anything which is not a particle of the particular type shown in the image will have bit value of 0. By (ImageJ) default the particles in these 8-bit images are displayed in black colour (option 2A), but you can also chose to display them in white (option 2B).

6. After you decided what you want to do, press “OK”. Now you will be asked to choose a directory to save the output binary images. Select the directory and press “Select”.

7. Now the menu “Image type choice” should appear:



What is the format of your image regarding particle presentation (select only one type)?

☑ I have a single file that contains a single image, where one type and the other type of particles (background) are depicted

□ I have a single file that contains two images, each for one type of particles or background (colour channel, RGB not supported).

□ I have two files each contains one image for one type of particle or background (colour channel).

Your input image(s) representing the mixture of two components (particles) can be of various formats:

Option 1:

user can have one file in which one digital image is decoded where both type of the particles are depicted (i.e. singe file, single image). As the result of the opening procedure, two identical images will be displayed (duplicates), if the image is not RGB. In case of an RGB image, the original image will be split to three grayscale images (red, green, blue).

Option 2:

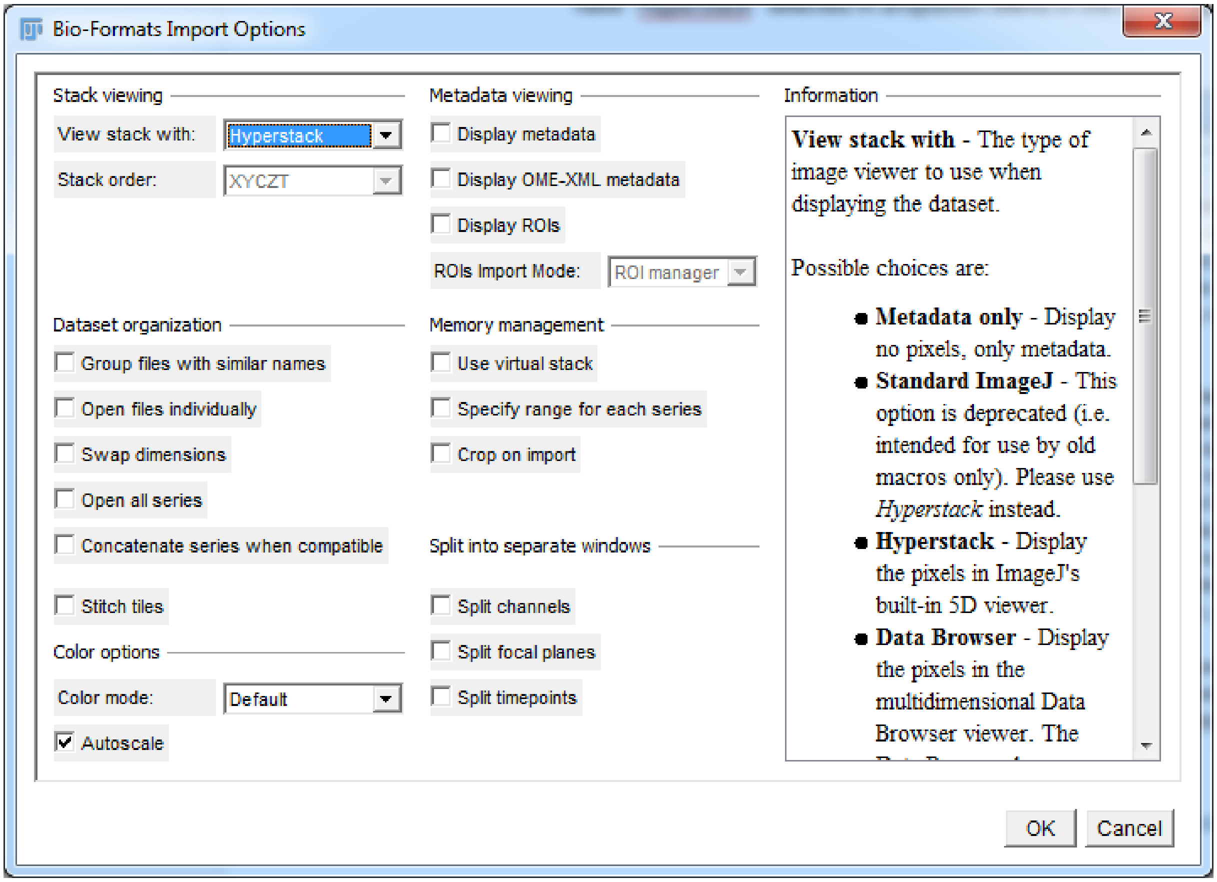
user can have one file in which two digital images are decoded, each for one type of the particles (i.e. singe file, two images). As the result of the opening procedure, two images will be displayed.

Option 3:

as last, you can have two files that each decode one image representing one type of the particle. As the result of the opening procedure, two images will be displayed.

In short, the user needs to know, which of the above holds for his input image and make the corresponding choice in the displayed menu and then click “OK”.

8. In next step select the corresponding image file and click “Open”. Note that varies file formats are supported by ImageJ (tif, jpg…) and many more in Fiji through the plug-in Bio-formats that automatically starts, when file of particular format is to be opened (like in the case of Zeiss .czi format). If the Bio-formats menu pops up, you should not thick anything (you can leave Autoscale), but you need to have “Hyperstack” selected in dropdown Menu in the upper left corner. Once you click “OK” the next Bio-Formats window may appear, offering you a selection of the size of the images that are going to be opened. By default, chose ONLY the biggest resolution series.



Bio-Formats Import Options

Stack viewing

View stack with: Hyperstack

Stack order: XYCZT

Dataset organization

□ Group files with similar names

□ Open files individually

□ Swap dimensions

□ Open all series

□ Concatenate series when compatible

□ Stitch tiles

Color options

Color mode: Default

☑ Autoscale

Metadata viewing

□ Display metadata

□ Display OME-XML metadata

□ Display ROIs

ROIs Import Mode: ROI manager

Memory management

□ Use virtual stack

□ Specify range for each series

□ Crop on import

Split into separate windows

□ Split channels

□ Split focal planes

□ Split timepoints

Information

View stack with - The type of image viewer to use when displaying the dataset.

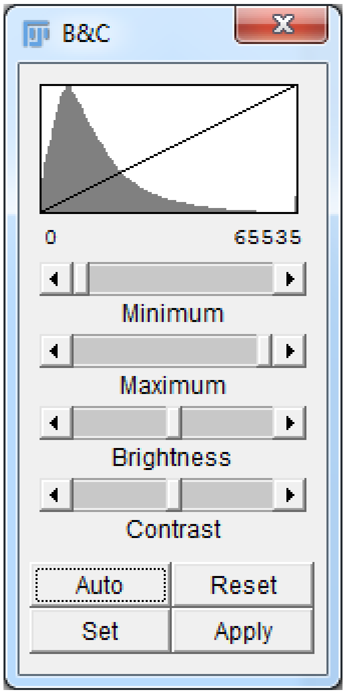
Possible choices are:

- **Metadata only** - Display no pixels, only metadata.
- **Standard ImageJ** - This option is deprecated (i.e. intended for use by old macros only). Please use *Hyperstack* instead.
- **Hyperstack** - Display the pixels in ImageJ's built-in 5D viewer.
- **Data Browser** - Display the pixels in the multidimensional Data Browser viewer. The

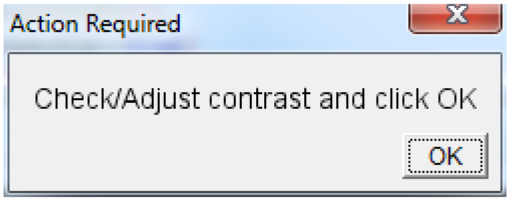
9. In the case the user has opened an RGB image (*Option 1 in step 7*) three images will be displayed, representing red, green, blue channels. (S)he will be asked to close the image with the worst contrast of the particles of one type against the other type. So, the remaining two images, should show, each separately, as good as possible one type of the particles.

   There are now two images opened (we will call them *image_ch1* & *image_ch2*), each representing one type of the particle and to observe them in best possible contrast the auto-contrast option is automatically applied by the macro. Each of the two images will be now sequentially converted to two binary images.

10. Let us start with the *image_ch1*. The user can manually improve the contrast through B&C menu that appears simultaneously on the right hand side next to the image in focus, if this is necessary.



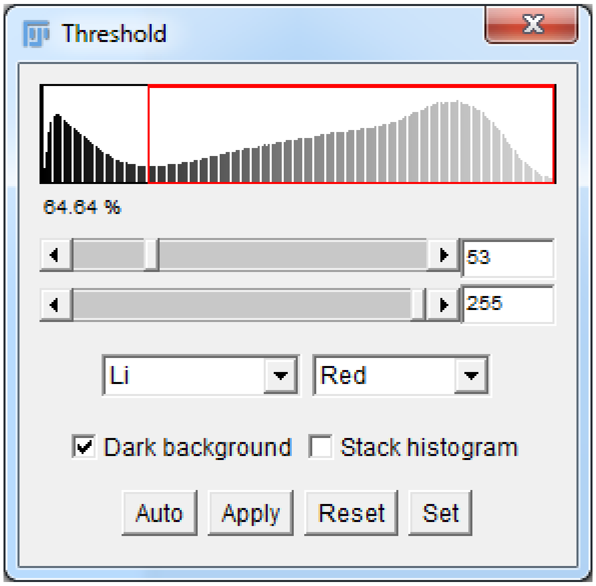
Unless the user is an expert, we suggest, only the sliders are adjusted in this menu if this is necessary at all. Once satisfied, the user should click “OK” in the simultanously displayed window:



Check/Adjust contrast and click OK

Note that contrasting is applied on selected window (*image_ch1*), which can be accidentally changed by clicking the other window (*image_ch2*). In the case you did accidentally click on the *image_ch2*, just click again on the *image_ch1* and then adjust contrast. Also, in the case you have two stacks of images (like in the case of CLSM images of biofilms composed from two bacterial strains, labelled with different fluorochromes), you can slide through the stack prior clicking “OK”, just to make sure you adjusted the contrast satisfactorily through all slices in stack.

11. After clicking OK in previous step, the Threshold menu will open. The red area in the *image_ch1* denotes the particles that are supposed to be extracted from the *image_ch1*. By default it is assumed that the background in *image_ch1* is darker than particles. If this is not the case the user can uncheck the “Dark background” choice. By using the sliders in the Threshold menu the user can adjust the lower and upper boundary of pixel values to be included in the selection to select the particles of interest in the best possible way.



53

255

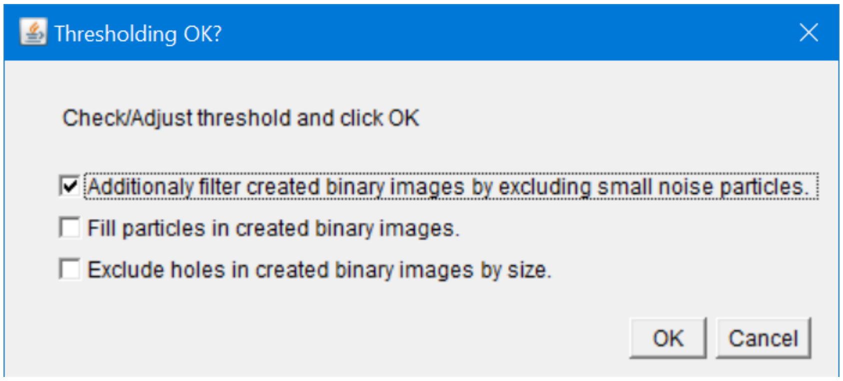
Li

Red

☑ Dark background □ Stack histogram

Unless the user is an expert, we suggest, only the sliders (and choice of Dark background) are adjusted in this menu if this is necessary.

Sometimes the thresholding cannot be set ideally- there are some parts of the particles that are not coloured red and some pixels or group if pixels that are not part of the particles of interest, but are coloured red. In this case it is a good idea to select Option 1 and 2 in the simultaneously displayed window:



Check/Adjust threshold and click OK

☑ Additionaly filter created binary images by excluding small noise particles.

□ Fill particles in created binary images.

□ Exclude holes in created binary images by size.

Option 1:
The obtained binary image may display particles (noise, like some shadows…) that are not part of the "true" particles that you want to obtain. Often they are small in size, and can be filtered out. Chose this option if needed.

Option 2:
The obtained binary image may display particles that are the "true" particles that you want to display. However, some pixels that are part of them are missing (displayed as background, due to some

shadows etc.). If the extend of missing pixels is not too high, you can fill the missing parts by choosing this option. However, all holes in the particles will be filled regardless of their size.

Option 3:

This option differs from the above option. Here, you can specify the maximum size of the hole that will be filled.

Once satisfied with set thresholds, you can click “OK” to continue.

12. If you checked *Option 1* in previous step you will be asked to encircle the biggest particle (that are not of your interest) you want to remove. You can use the Oval or Rectangle tool, which you can select from ImageJ-Fiji toolbar. All particles of the encircled and smaller size will be removed after clicking “OK”.

13. Now the result binary image is displayed. You will be asked if you are satisfied. If Yes, select the result binary image by clicking on its Title bar (the top of the image with title) and click OK. If Not, you can manually use additional built-in tools in ImageJ-Fiji to adjust the image appropriately. Once satisfied, select the result binary image by clicking on its Title bar and click OK. The selected result binary image will be saved to the save directory you have set in step 6 and closed.

14. The steps from 10 to 13 are now repeated, but this time the *image_ch2* is in the focus.

15. Once *image_ch1* and *image_ch2* had been converted to two binary images (*ch1_bin* and *ch2_bin*), where each image separately presents one type of the particle (or component, background) the user can chose either to process next image set (user can select again the same input image(s), or new image(s). Clicking OK brings the user again to step 5, while No or Cancel terminates the macro.

    The saved output images from step 13 can serve as input images in macro *sim_seg_extremes* and *MSS_calc.*

Macro 2: *Sim_seg_extremes_ver1.2.ijm*

Introduction

This script runs as a macro in Fiji (ImageJ) environment. Its purpose is to simulate and depict maximal and minimal segregation of two components system (or particles of two types, called *ch1* and *ch2* particles) depicted in input digital images. Two input images (8-bit binary format, black and white) are required, each representing one type of the particle (*ch1_bin_, ch2_bin_*) in the same space. The two images can be obtained by the macro *Convert_to_bin_ver1.2.ijm*, but the user can obtain them also in other ways (e.g. manually by splitting via ImageJ tools). The output of the simulation are images (8-bit) representing minimal and maximal segregation case for each particle type (component type), together four images (*segmin_ch1_bin_, segmax_ch1_bin_* and *segmin_ch2_bin_, segmax_ch2_bin_*) stored to the disk as corresponding files. The particles are simulated as circles of user defined diameter; their number is obtained by dividing particle area in an input image by the sphere area. In addition, the macro can also simulate segregation for only part of the image (ROI) i.e. by creating the user defined mask. The masked original images are also outputted (*M_orig_ch1_bin_, M_orig_ch2_bin_*), together with the image representing mask itself (*Mask_*) and stored to the disk as corresponding files. Also, the stacks of images are supported by the macro. *segmin* and *segmax* files together *ch1_bin, ch2_bin* serve as input for *MSS_calc_ver1.2*, where Multiscale spatial segregation analysis is performed.

Step by step guide

1. If you do not already have Fiji, download and install it from: https://imagej.net/software/fiji/downloads
2. Run Fiji (ImageJ-win64.exe file, if you are running 64-bit version).
   If asked to make an update, you can chose »Remind me later«, the update is usually not needed to successfully run *Sim_seg_extremes_ver1.2*, unless you are running really old version of Fiji-ImageJ.

A. Installation

3. In Windows explorer (or equivalent File browser) go to the folder where you have installed Fiji and find \plugins folder within Fiji folder. In our case:
   C:\Users\iztokd\Downloads\fiji-win64\Fiji.app\plugins
   Create folder (if it is not there already) \MSSegregation within plugins folder

   In running Fiji app press Ctrl+shift+M (=Install Macro) and choose a file *Sim_seg_extremes_ver1.2.ijm*

   Save it to directory \MSSegregation, created above.
   Close and restart Fiji app. If asked to save LUTMenuTool.ijm, just say “No”.

4. Now you can run the macro from the Menu Bar: Plugins>MSSegregation> *Sim_seg_extremes_ver1.2*

B. Direct run of the Macro

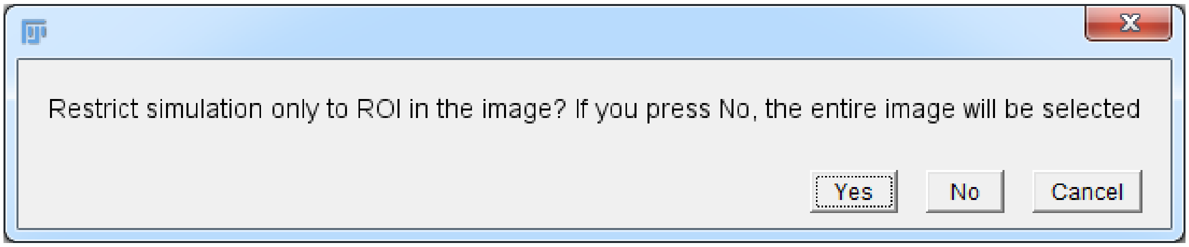
3. Windows explorer (or equivalent File browser) and "drag and drop" the *Sim_seg_extremes_ver1.2.ijm* to the running Fiji application.

4. New window with displaying the code of *Sim_seg_extremes_ver1.2.ijm* will open (script editor) and you can run it by pressing Ctrl+R or by choosing Run>Run from menu of the script editor.

5. Now you will be asked to choose a directory to save the output binary images, which are the result of the simulation. Select directory and press "Select".

6. Next you will be asked to select an image file that represent *ch1* particles. The image needs to be in 8-bit format. The particles must have pixel value of 255 the rest of pixels in the image must be 0. If you had used the macro *Convert_to_bin_ver1.2* than this is the file that ends with *_ch1_bin.tif* Click "OK" and select the file. Repeat the same for the image file that represent *ch2* particles.

7. Next the following question appears:



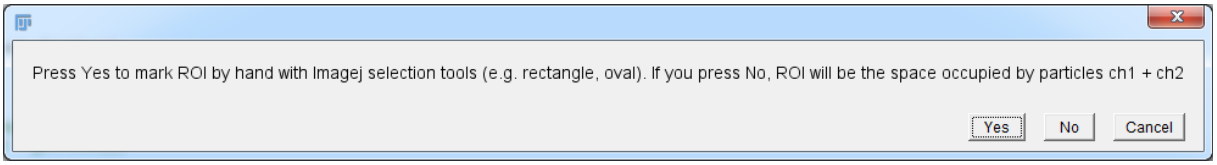
Restrict simulation only to ROI in the image? If you press No, the entire image will be selected

If you press "Yes" the macro will simulate segregation extremes only on the specified region (ROI = region of interest) of the input images from step 6. This can be handy, if the image has some background that you do not want to include into analysis and/or particles can occupy only one particular region in the image.

If you press "No" the macro will simulate segregation extremes on the entire area of input images from step 6. This is the most common situation. In this case you can go directly to step 9.

If you press "Cancel" the macro will terminate.

8. If you pressed “Yes” in previous step the following will appear on your screen:



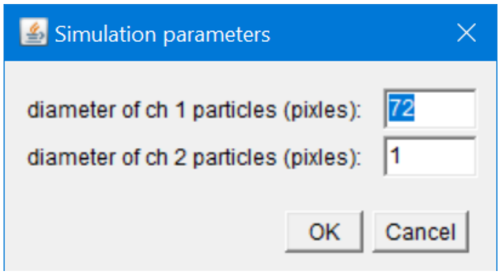
Press Yes to mark ROI by hand with Imagej selection tools (e.g. rectangle, oval). If you press No, ROI will be the space occupied by particles ch1 + ch2

If you press “Yes” you will be asked to mark ROI on ch1 image (*_ch1_bin*). Use ImageJ-Fiji selection tools from the toolbar (e.g. Oval, Rectangle…) to encircle the ROI. The same ROI will be automatically copied and applied to ch2 image (*_ch2_bin*).

If you press “No”, the ROI will become the space that is occupied by the particle *ch1* plus the space occupied by the particle *ch2*. This is useful, when two component or particles are restricted to certain area in the image. For example, oats and raisins can be restricted by the plate that occupies only part of the entire image. Oats or raisins cannot “move” out of the plate. Further example: you want to preserve the shape of two species biofilm in microscopy image and just simulate segregation within biofilm. You will be also asked to fill the holes in the common space. If they are not any holes present or the displayed holes are true empty space, just press No, otherwise press OK. If you are satisfied press OK, when asked about the result, No will undo fill.

If you press “Cancel” the macro will terminate.

9. In this step you need to specify the diameter of the particles (component).



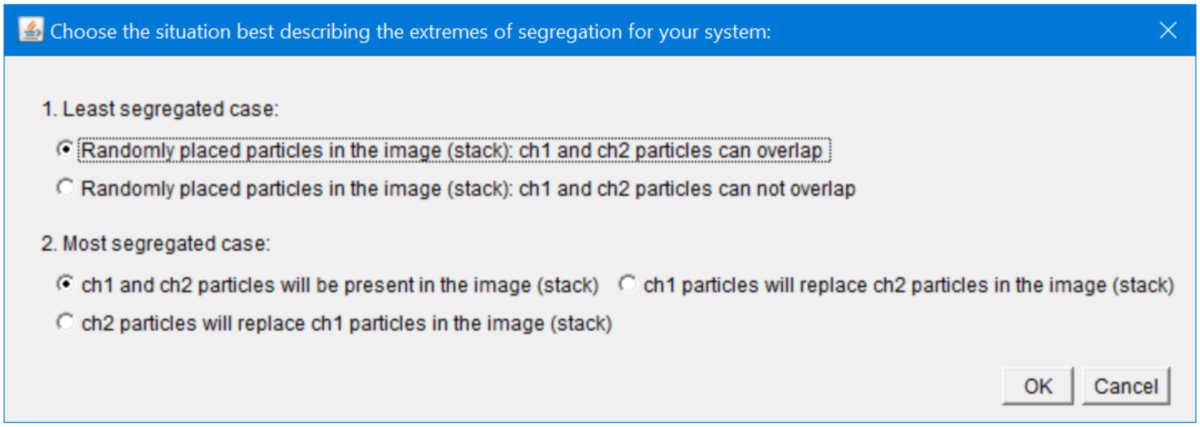
Simulation parameters

diameter of ch 1 particles (pixles): 72

diameter of ch 2 particles (pixles): 1

The particles can be only 1 pix in size, but can be also bigger. Only circular particles are supported at present time. It is advisable that user opens the image of particular particles in a separate instance of Fiji app and measures the particle size by, for example, by Straight tool from ImageJ-Fiji toolbar. However, if particles can be regarded as a continuous matrix than their size is the image resolution, i.e. 1 pixel. For example, in the case of mixture of oats and raisins, the oats can be considered as continuous matrix (diameter 1 pix) and raisins as circular particles(diameter 72 pix).

10. After you entered the diameter of respective particles and have pressed “OK” the following menu will appear:



Choose the situation best describing the extremes of segregation for your system:

1. Least segregated case:

● Randomly placed particles in the image (stack): ch1 and ch2 particles can overlap

○ Randomly placed particles in the image (stack): ch1 and ch2 particles can not overlap

2. Most segregated case:

● ch1 and ch2 particles will be present in the image (stack) ○ ch1 particles will replace ch2 particles in the image (stack)

○ ch2 particles will replace ch1 particles in the image (stack)

1. *Least segregated case:*

Here you decide, if the randomly placed particles of one type (*ch1*) can overlap (i.e. share the same location in an image) with particles of other type (*ch2*) in the image. By default particles of the same type cannot overlap.

2. *Most segregated case:*

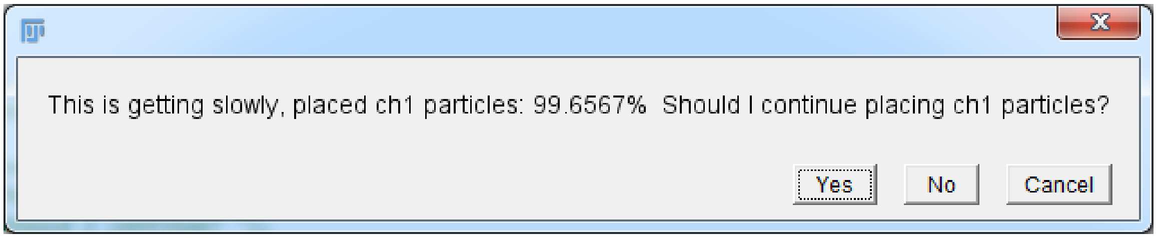
Here you decide about the type of maximal segregation:

The two type of particles (*ch1* and *ch2*) are restricted to the space of image (or ROI, step 8) and cannot “escape” from it. However, they can segregate to the sides of the image in order to maximize the distance from not alike particles. On the other hand, the alike particles will be placed maximally to each other, their individual shape will be ignored (the area of the particles is held constant).

If the most segregated case is such that particles *ch1* can completely exclude particles *ch2* then choose the second option.

If the most segregated case is such that particles *ch2* can completely exclude particles *ch1* then choose the third option.

11. After clicking OK in the previous step the simulation of placing the particles in the image will begin, without displaying anything, except in Log window “start of the simulation:” together with time will be printed.
However, you may get warnings that “Total area of simulated particles exceeds the available image space”. This can happen in the case you did not allow particle overlap in step 10 and the circle-approximated particles have slightly higher total area than the area of image or ROI. This is due to rounding errors that occur in approximation of particles composed of finite pixels to circle particles. You should click “OK” and soon you will likely encounter a message like this:



This is getting slowly, placed ch1 particles: 99.6567% Should I continue placing ch1 particles?

This occurred because the space available to particles become very scarce during simulation and not all of the particles have been placed. If the total area of simulated particles exceeds the available image space than the placing all of the particles without overlapping will never happen. If the percentage is small, you should press Yes and see in the next message if the percentage improved significantly. If it did, than you should press Yes again and continue pressing Yes as long as you see significant improvements. Once you have reached high enough number of the % of placed (typically >95 %) you can press No to finish simulation and check the result images. On the other hand, if you did not observe significant steps in increase in % of placed particles after clicking Yes, you can either abort the macro by clicking Cancel and rerun the simulation with different parameters (allowing overlap, or different size of particle) or press No to finish simulation and check the result images.

12. Once all the particles *(ch1* and *ch2*) have been placed or user agrees not to continue placing the particles (read step 11) the simulation finishes, prints to Log window "end of simulation:" together with time and displays the result images (three dots in the name means the name of the original image):

    1. Two Images showing minimally segregated cases (best mixed):
    segmin_ch1_bin_…, segmin_ch2_bin_…

    2. Two Images showing maximally segregated cases (worst mixed):
    segmax_ch1_bin_…, segmax_ch2_bin_…

    3. One Image showing the mask (i.e. the image that shows the area, where particles were allowed to be placed. The pixel value of 1 (black) indicates this area. If no restriction (ROI) of simulation space was selected (see step 7) then this image will be black:
    *Mask_*

    4. Two Images showing the original input images through the mask (if the mask was uniform black (no restrictions) than these images are the same as original input images):
    M_orig_ch1_bin_…, M_orig _ch2_bin_…

    5. One image showing overlap of *segmin_ch1_bin_…* and *segmin_ch2_bin_…* images (useful to see how severe overlap is, if it was permitted- black colour (pixel value 512) indicates overlapping of *ch1* and *ch2*, grey (pixel value 255) means particle is there, and white (pixel value 0) means empty space):
    overlap of ch1 min seg and ch2 min seg

6. In addition, Log window is displayed, where most important data of simulation is displayed.

   All of these images are stored as .tif files beginning with the above names in the save directory selected in step 5, together with Log window, stored as: *Log_Sim_seg_extremes_ver1.2.txt* **.** The image files from 1. and 2., together with *ch1_bin_, ch2_bin_* image files serve as input files for macro *MSS_calc_ver1.2*, where Multiscale spatial segregation analysis is performed.

Macro 3: *MSS_calc _ver1.2.ijm*

Introduction

This script runs as a macro in Fiji (ImageJ) environment. Its purpose is to calculate the measures of segregation in images of two components system (mixture of two particles, particle and background….). Two components are called *ch1* and *ch2* (particles). The main quantity to be calculated is spatial size scale dependent segregation level (*Sd^*), i.e. *Sd^* as a function of dimension, *d*, of field of view (FOV). This is obtained by scanning the image by randomly placing the square of size *d* x *d* (= FOV) and calculating the dominance of one particle over the other in each placed FOV. This is done *n* times and then averaged to calculate *Sd^* (for the details, please check the reference Dogsa et al., 2021, once it is published). To do this the two input images (8-bit binary format, black and white) are required, each representing one type of the particle (*ch1_bin_, ch2_bin_*) in the same space (like an image of oats-*ch1* and raisins-*ch2* in the same plate). The two images can be obtained by the help of macro *Convert_to_bin_ver1.2.ijm*, but the user can obtain them also in other ways (e.g. manually doing the splitting and converting via ImageJ tools). To compare *Sd^* of your sample to the positive and negative controls, i.e. to extremely segregated and extremely non-segregated cases in the same space by the same particles the images displaying these situations are required for *ch1* and *ch2* particles. These images (*segmin_ch1_bin_, segmax_ch1_bin_* and *segmin_ch2_bin_, segmax_ch2_bin_*) can be obtained by the macro *Sim_seg_extremes_ver1.2.ijm.* The stacks of images are supported by the macro. Besides calculating the segregation level (*Sd^*) as a function of dimension of field of view (FOV), the macro can also calculate:

1. multiscale spatial segregation level (*MSSL*), which is *Sd^* averaged over all *d* (from *dmin* to *dmax*) of FOV to give distance averaged segregation level for users (sample) image and for min and max segregation images.
2. multiscale spatial segregation distance (*MSSD*), which is *Sd^* integrated over all *d* of FOV
   It can be interpreted as the distance (dimension) over which segregation level is 1 (i.e. maximal segregation level), if we theoretically compact all segregation levels, *Sd^* , from *dmin* to *dmax* to the distances from *dmin* to *MSSD*. This distance roughly corresponds to dimension of particles in the hypothetical image, where these particles form homogenous aggregates of *MSSD* and contain either one or the other type of the particle.
3. Relative multiscale spatial segregation level (*rMSSL*). This segregation measure takes into account *MSSL* of users (sample) image and *MSSL* of min and max segregation images to calculate relative FOV independent segregation measure.

Step by step guide

1. If you do not already have Fiji, download and install it from: https://imagej.net/software/fiji/downloads
2. Run Fiji (ImageJ-win64.exe file, if you are running 64-bit version).

If asked to make an update, you can chose »Remind me later«, the update is usually not needed to successfully run *MSS_calc _ver1.2*, unless you are running really old version of Fiji_ImageJ.

A. Installation

3. In Windows explorer (or equivalent File browser) go to the folder where you have installed Fiji and find \plugins folder within Fiji folder. In our case:
C:\Users\iztokd\Downloads\fiji-win64\Fiji.app\plugins
Create folder (if it is not there already there) \MSSegregation within plugins folder

   In running Fiji app press Ctrl+shift+M (=Install Macro) and choose a file *MSS_calc _ver1.2.ijm*

   Save it to directory \MSSegregation, created above.
Close and restart Fiji app. If asked to save LUTMenuTool.ijm, just say “No”.

4. Now you can run the macro from the Menu Bar: Plugins> MSSegregation> *MSS_calc _ver1.2*

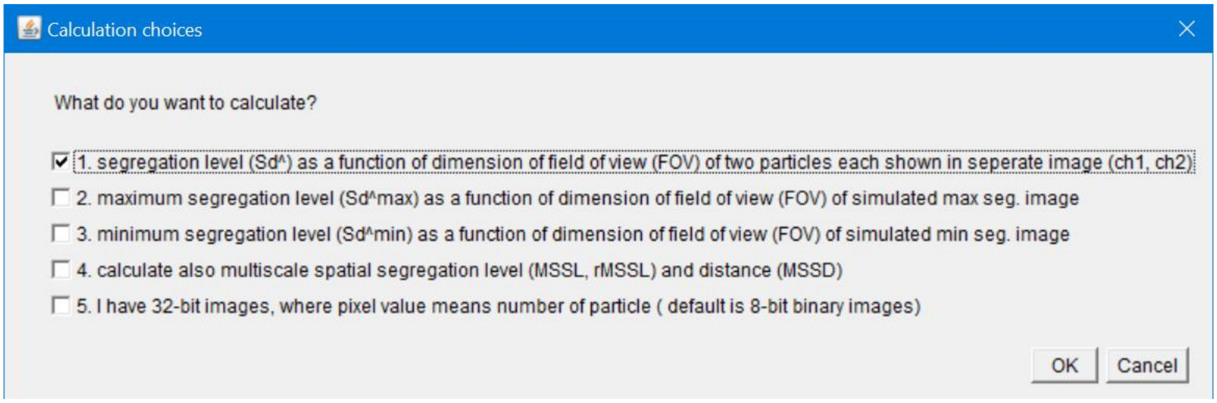
B. Direct run of the Macro

3. Open Windows explorer (or equivalent File browser) and “drag and drop” the *MSS_calc _ver1.2.ijm* to the running Fiji application.

4. New window with displaying the code of *MSS_calc _ver1.2.ijm* will open (script editor) and you can run it by pressing Ctrl+R or by choosing Run>Run from menu of the script editor.

5. Now you will be asked to choose a directory to save the output calculations, which are the result of the simulation. Select directory and press “Select”.

6. Then you will be asked what to calculate.



What do you want to calculate?

☑ 1. segregation level (Sd^) as a function of dimension of field of view (FOV) of two particles each shown in seperate image (ch1, ch2)
□ 2. maximum segregation level (Sd^max) as a function of dimension of field of view (FOV) of simulated max seg. image
□ 3. minimum segregation level (Sd^min) as a function of dimension of field of view (FOV) of simulated min seg. image
□ 4. calculate also multiscale spatial segregation level (MSSL, rMSSL) and distance (MSSD)
□ 5. I have 32-bit images, where pixel value means number of particle (default is 8-bit binary images)

Option 1:

This option is by default checked, as this is the most fundamental purpose of this macro. To calculate segregation level (*Sd^*), you will need two binary images, each representing one type of the particle or component in the common mixture, which can be obtained by splitting and converting the original sample image by *Convert_bin_ver1.2.ijm* macro.

Option 2:

This option is used to compare calculated segregation level of sample image to the positive control, i.e. to the segregation level of hypothetically maximally segregated sample image. This image can be obtained by *Sim_seg_extremes_ver1.2.ijm* macro.

Option 3:

This option is used to compare calculated segregation level of sample image to the negative control, i.e. to the segregation level of hypothetically minimally segregated sample image. This image can be obtained by *Sim_seg_extremes_ver1.2.ijm* macro.

Option 4:

If the Option 1 to 3 is checked the macro will calculate the segregation measures derived from segregation level, *Sd^*.

Option 5:

By default the input images need to be 8-bit with pixel value of 255 indicating the presence of the particle and 0 of its absence. If this option is checked, the user can also input 32-bit images, where pixel value represents the number of particles in the location of the pixel. This is useful in the case of choropleth maps.

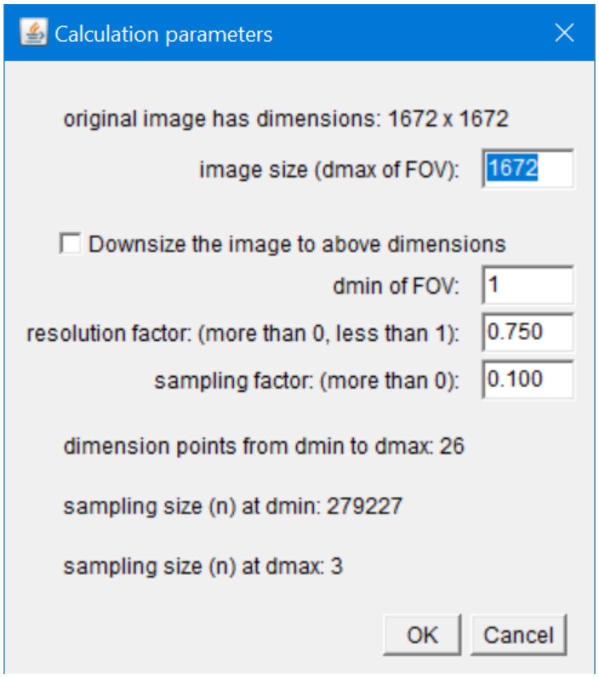
Pressing "OK" will get you to the next step, while pressing "Cancel" will exit the macro.

7. Now you will be asked to select a directory to save the calculation results. Click on the chosen directory and press "Select".

8. Next you will be asked to select an image file that represent *ch1* particles. The image needs to be in 8-bit format (unless you have checked Option 5 in step 6). If you had used the macro *Convert_to_bin.ijm* than this is the file that ends with *_ch1_bin.tif.* Click "OK" and select the file. Repeat the same for the image file that represents *ch2* particles.

9. If you have checked Option 3 and 4 in step 6, you will be asked to choose corresponding 8-bit (32-bit if you have checked option 5 in step 6) images, representing maximal and minimal segregation cases. If these images were obtained by the macro *Sim_seg_extremes_ver1.2.ijm.*, then the corresponding files are named as: *segmax_ch1_bin_*....tif, *segmax_ch2_bin_....tif* and *segmin_ch1_bin_....tif, segmin_ch2_bin_....tif*). For each image click "OK" and select the file.

10. In this step you can adjust calculation parameters.



Calculation parameters

original image has dimensions: 1672 x 1672

image size (dmax of FOV): 1672

□ Downsize the image to above dimensions

dmin of FOV: 1

resolution factor: (more than 0, less than 1): 0.750

sampling factor: (more than 0): 0.100

dimension points from dmin to dmax: 26

sampling size (n) at dmin: 279227

sampling size (n) at dmax: 3

Image size (dmax of FOV) [pixels]:

By default, image size in pixels is displayed corresponding to the maximum dimension (*dmax*) of FOV that will be applied to an image to calculate FOV dimension dependant segregation level, *Sd^*. The user can decrease the *dmax* of FOV for segregation level calculation, but in this case (s)he will not obtain the segregation level, *Sd^* of entire area of image.

Downsize the image to above dimensions:

If the time taken for the calculation is very long and the image has unnecessary high pixel resolution, the user can downsize the input images to new *dmax* [pixels]. In this case the *Sd^* of entire area of image can be obtained.

dmin of FOV [pixels]:

The user can set the smallest dimension (*dmin*) of FOV. This can improve the calculation speed significantly, if *Sd^* at *dmin* of 1 pix is not necessary to be calculated, than this number should be increased. For example, in the mixture of oats & raisins that are in size about 50 pix, there the *Sd^* at *dmin* of 1 pix will for sure be 1 as the two constituents cannot interpenetrate each other.

resolution factor:

This factor determines the number of different dimensions, *d* of FOV at which *Sd^* will be calculated and by which factor the values of *d* will be spaced, in the direction from *dmax* to *dmin*. For example factor of 0.99, means that starting from *dmax*, every next *d* will be smaller by factor of 0.99 in the direction to *dmin*. Unless the range from *dmax* to *dmin* is very low, this means a lot of *d* points.

sampling factor:

This factor determines the number, *n*, of local segregation levels will be calculated from randomly placed FOVs of dimension *d* in the image , which get then averaged to obtain *Sd^* at *d*. The more the better the statistics, but of course more time is required for the calculation. Maximal sampling size per *d* is 1 000 000, but often a few thousands will be enough. This factor impacts the duration of the calculation significantly, so try first with *n* about few thousands and check the estimated experimental error of the calculated quantities in the end of simulation, maybe they are small enough and you will not need to increase sampling factor.

Note that the total time of calculation depends on the number of points to be calculated (distance points from *dmax* to *dmin*), the number of sampling at each point (sampling size, *n*, and dmin and dmax) and the resolution of image.

If the changes were made to whatever parameters default values in this menu, the parameters (*distance points from dmax to dmin*), (*sampling size* at *dmin* and *dmax*) will get refreshed after clicking “OK”. Once no more changes to the parameters values (and parameter values are within their ranges) had been made, clicking “OK” will start the calculations and calculation start time will appear in the Log window.

During the calculations, the user can follow the progress in Log window. For each calculation point the data about the dimension, *d*, and number, *n*, of FOV (sampling size) will be displayed in Log window. Once the *Sd^* at certain *d* is calculated, the macro proceeds to calculate next *Sd^* at next *d*, until all *Sd^* from *dmax* to *dmin* are calculated. When this happens the calculation end time will appear in the Log window and the calculation results will be displayed in Results window, including all the segregation measures selected in step 6 together with their standard errors. The Result table and Log window are saved to the saving directory specified in step 6, having the names *Sd_MSSDL_ver1.2.txt* and *Log_Sd_ver1.2.txt*. In addition, the file *statistics_of_segregation_Sd_ver1.2.txt* containing additional segregation indices that can help very advanced users to further analyse their segregation cases is also stored to the saving directory.

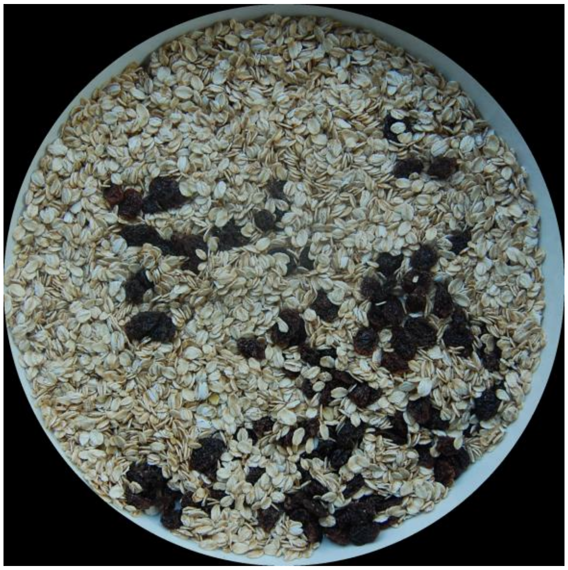
Step 11: As a last step, the user can chose that the segregation level as a function of *d* of FOV gets plotted, which will be than displayed.

Example 1: Oats &Raisins

Convert_to_bin_ver1.2.ijm

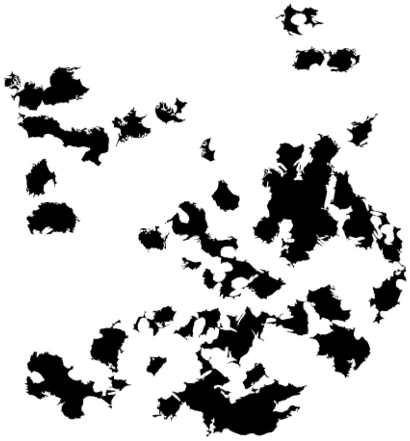
Parameters used: step 5 (*Option 1 and 2*), step 7 (*Option 1*), step 9: closed image (blue), step 11 image (red = ch1) (*Option 1, 2*); image (green) = ch2 (*Option 1, 3*)

Input images (step 8):

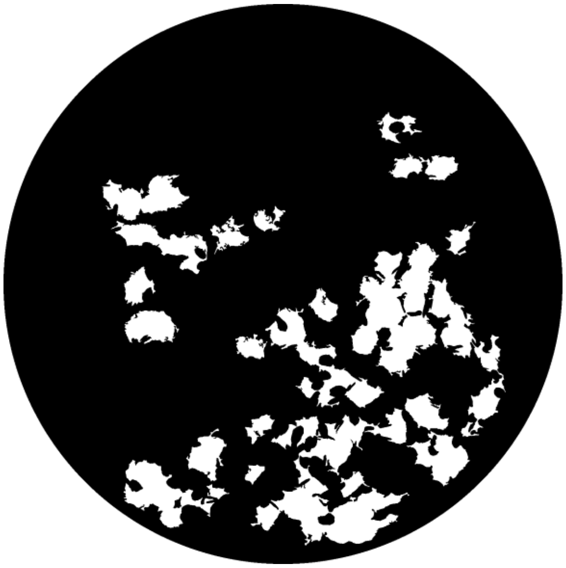


t3.tif (RGB image)

Output of images (step 13):



t3.tif (red)0_ch1_bin.tif



t3.tif (green)0_ch1_bin.tif

Sim_seg_extremes_ver1.2.ijm

Parameters used: step 7 (*Yes*)*,* step 8 (*No*), step 9: ch1 = *72* pix, ch2 = *1* pix, step 10: *Randomly placed particles in the image (stack): ch1 and ch2 particles can not overlap*, *ch1 and ch2 particles will be present in the image (stack)*, step 11, (*No,* after >99% ch2 partciles placed).

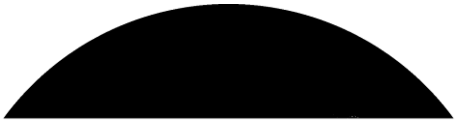
Input images (step 6):

Output of *Convert_to_bin_ver1.2.ijm :*

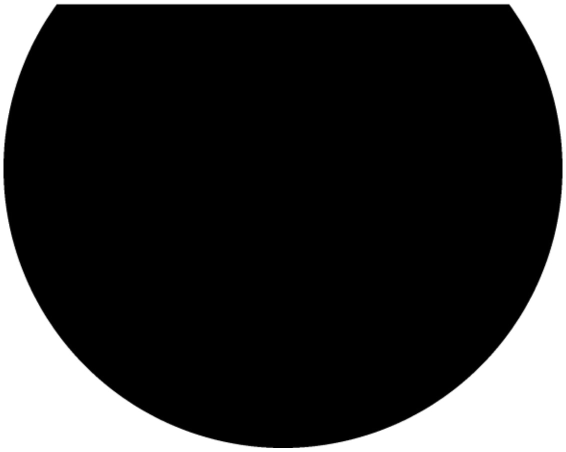
t3.tif (red)0_ch1_bin.tif

t3.tif (green)0_ch1_bin.tif

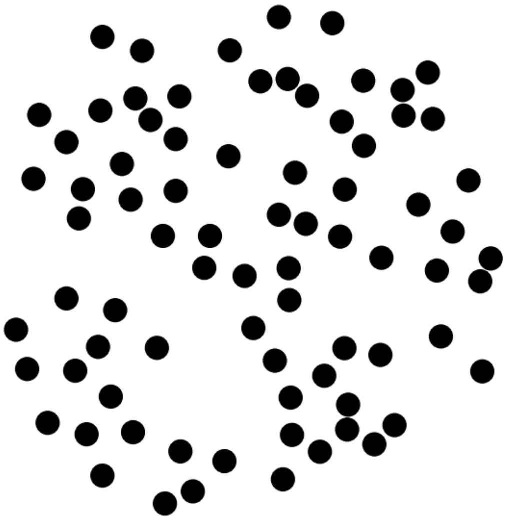
Output images (step 15):



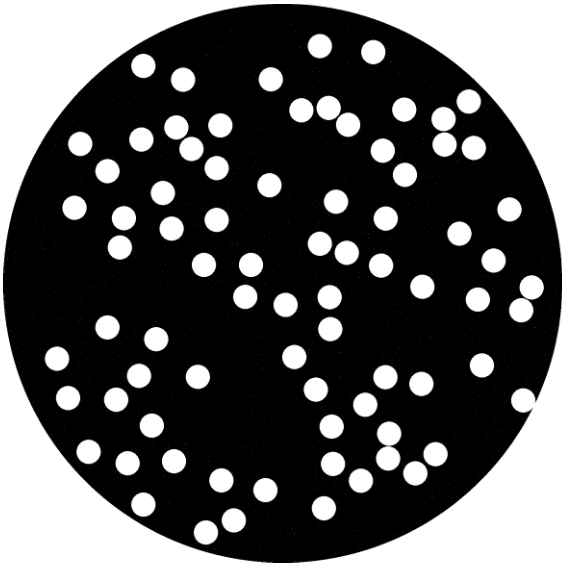
segmax_ch1_bin_t3.tif (red)0_ch1_bin



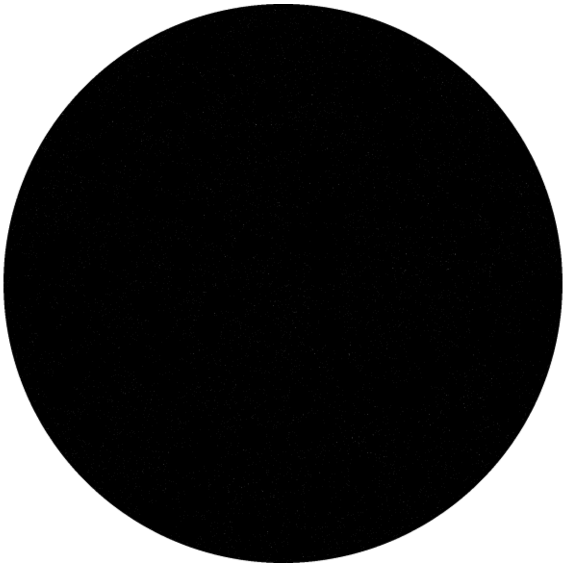
segmax_ch2_bin_t3.tif (green)0_ch2_bin



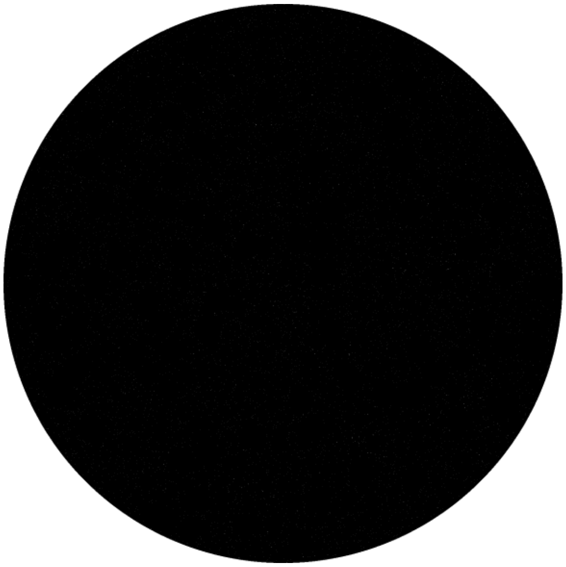
segmin_ch1_bin_t3.tif (red)0_ch1_bin.tif



segmin_ch2_bin_t3.tif (green)0_ch2_bin.tif



overlap of ch1 min seg and ch2 min seg.tif



Mask_t3.tif (red)0_ch1_bin t3.tif (green)0_ch2_bin.tif

M_orig_ch1_bin_t3.tif (red)0_ch1_bin.tif

M_orig_ch2_bin_t3.tif (green)0_ch2_bin.tif

MSS_calc _ver1.2.ijm

Parameters used: step 6 (*Option 1 to 4*), *step 10 (dmax =1663, dmin =5, res_factor =0.75, sampling_factor =0.01), step 11 (Yes)*

Input images (step 8 and 9):

1. Output of *Convert_to_bin_ver1.2.ijm :*

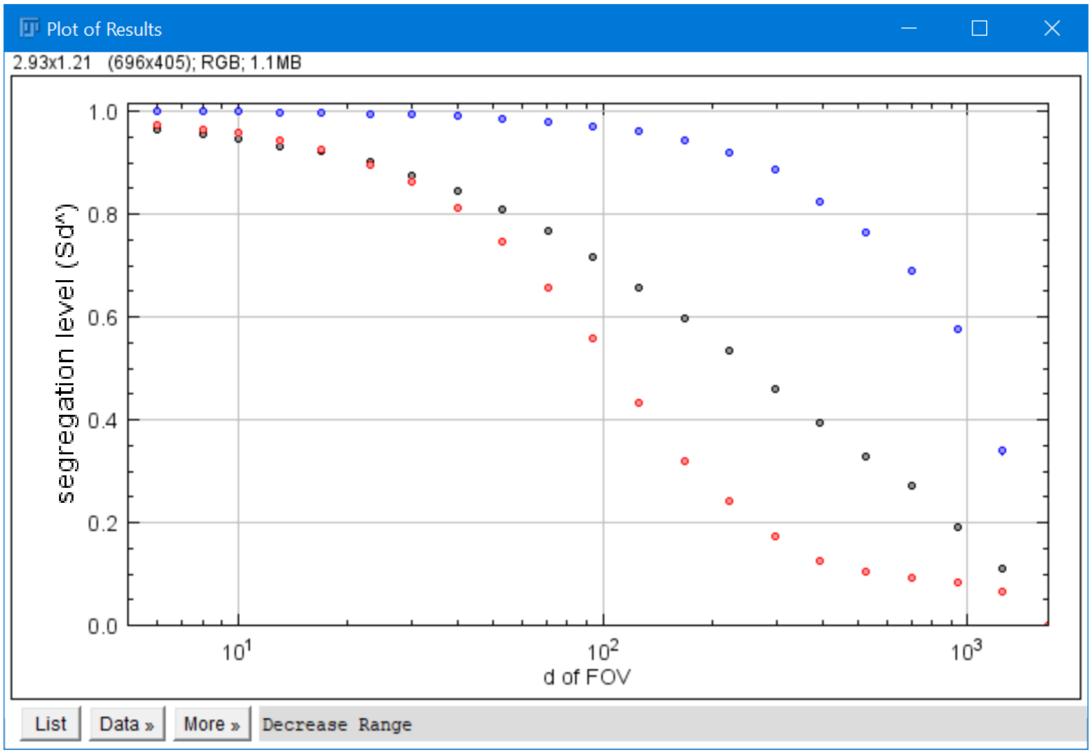
t3.tif (red)0_ch1_bin.tif t3.tif (green)0_ch1_bin.tif

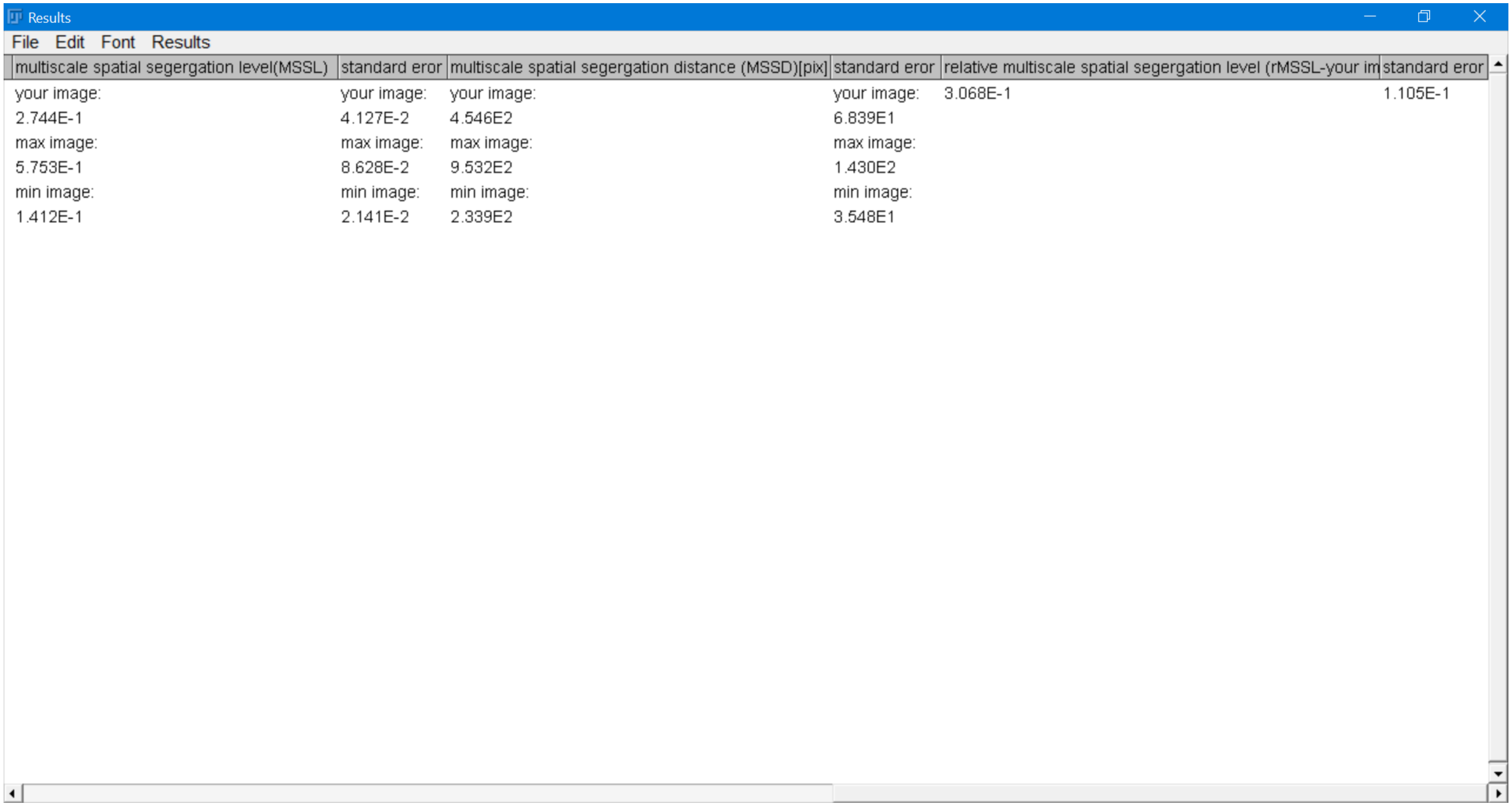
2. Output of *Sim_seg_extremes_ver1.2.ijm:*

segmax_ch1_bin_t3.tif (red)0_ch1_bin.tif segmax_ch2_bin_t3.tif (green)0_ch2_bin.tif
segmin_ch1_bin_t3.tif (red)0_ch1_bin.tif segmin_ch2_bin_t3.tif (green)0_ch2_bin.tif

Output result (most important):

-table with results of segregation level vs d of FOV for user's sample image (= Your image), max seg image and min seg image that get plotted in graph (step 11 , *Yes*) :





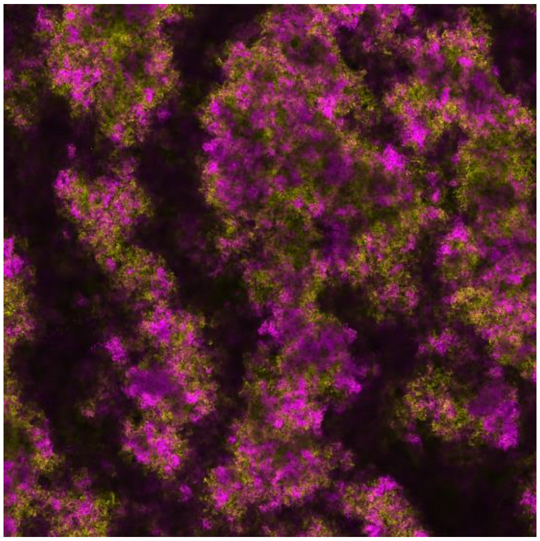
multiscale spatial segergation level(MSSL)	standard eror	multiscale spatial segergation distance (MSSD)[pix]	standard eror	relative multiscale spatial segergation level (rMSSL-your im	standard eror
your image:	your image:	your image:	your image:	3.068E-1	1.105E-1
2.744E-1	4.127E-2	4.546E2	6.839E1		
max image:	max image:	max image:	max image:		
5.753E-1	8.628E-2	9.532E2	1.430E2		
min image:	min image:	min image:	min image:		
1.412E-1	2.141E-2	2.339E2	3.548E1		

Example 2: bacterial biofilm composed of two strains (microscopy)

Convert_to_bin_ver1.2.ijm

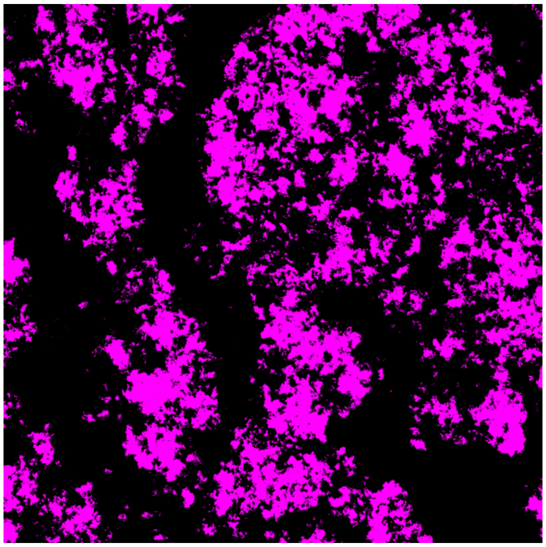
Parameters used: step 5 (*Option 1 and 3*), step 7 (*Option 2*), step 11 (images ch1 and ch2: default threshold of Li Method applied on middle slice) (No other *Options*).

Input images (step 8):

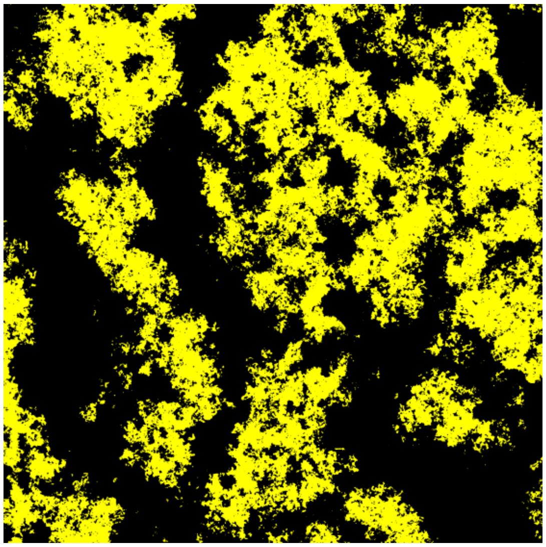


mk + yfp.czi (Zeiss. czi image stack of 17 slices, only middle slice (=9) is shown)

Output of images (step 13):



C1-mk + yfp.czi - mk + yfp.czi #10_ch1_bin.tif



C2-mk + yfp.czi - mk + yfp.czi #10_ch2_bin.tif

Sim_seg_extremes_ver1.2.ijm

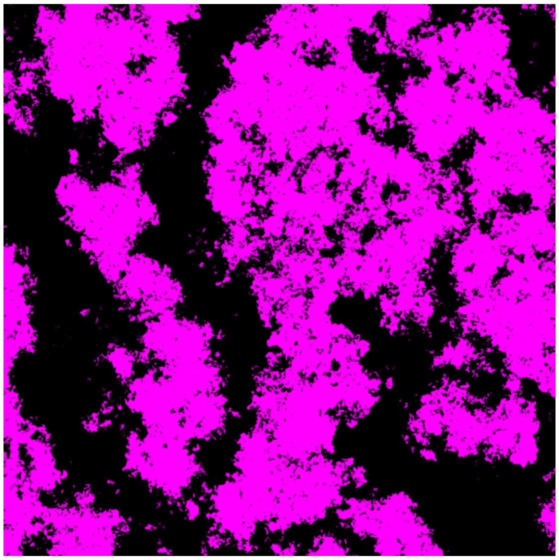
Parameters used: step 7 (*Yes*)*,* step 8 (*No*), step 9: ch1 = *3* pix, ch2 = *3* pix, step 10: *Randomly placed particles in the image (stack): ch1 and ch2 particles can overlap, ch1 particles will replace ch2 particles in the image (stack)*, step 11, (*Yes,* after cca. 75% ch1 particles placed, the rest can be placed by partial overlap; the same for ch2).

Input images (step 6):

Output of *Convert_to_bin_ver1.2.ijm :*

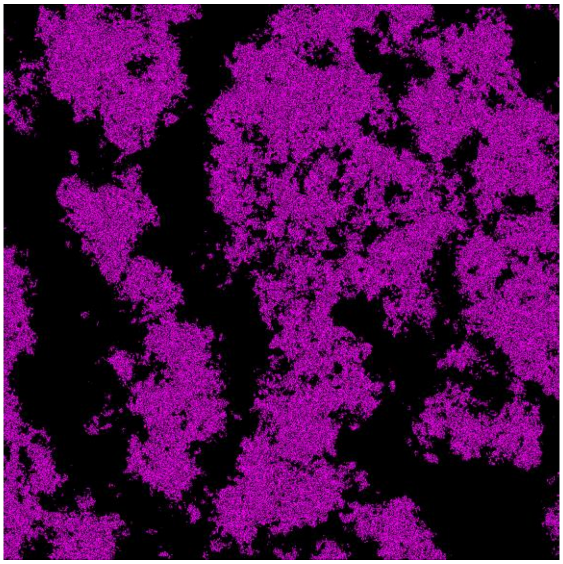
C1-mk + yfp.czi - mk + yfp.czi #10_ch1_bin.tif   C2-mk + yfp.czi - mk + yfp.czi #10_ch2_bin.tif

Output images (step 15):

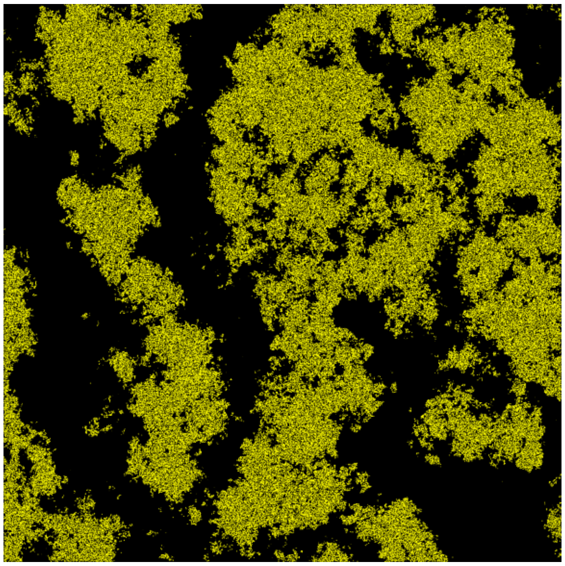


segmax_ch1_bin_C1-mk +yfp.czi -… _ch1_bin.tif

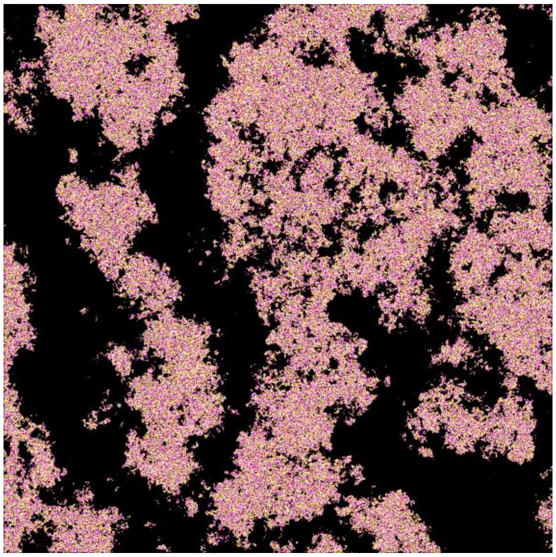
segmax_ch2_bin_C2-mk + yfp.czi - …_ch2_bin.tif



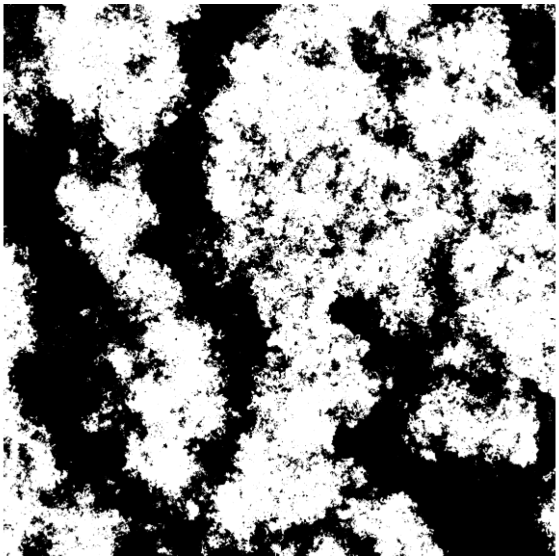
segmin_ch1_bin_C1-mk +yfp.czi -… _ch1_bin.tif



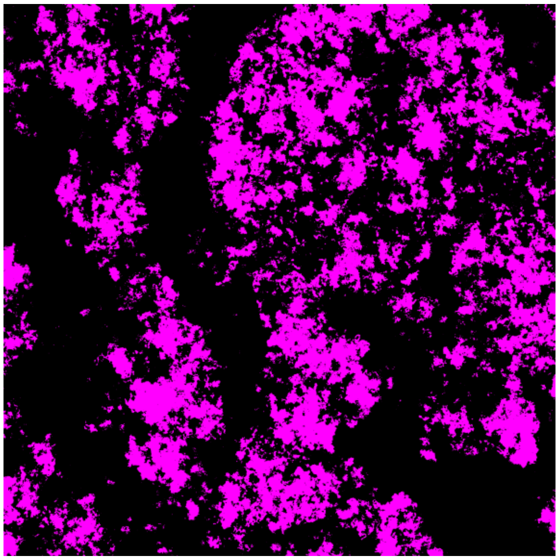
segmax_ch2_bin_C2-mk + yfp.czi - …_ch2_bin.tif



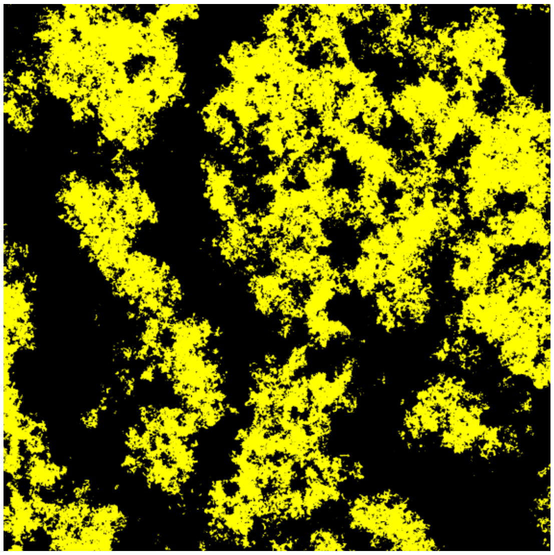
overlap of ch1 min seg and ch2 min seg.tif



Mask_C1-mk + yfp.czi -mk + yfp… _ch2_bin.tif



M_orig_ch1_bin_C1- mk + yfp.czi-…_ch1_bin.tif



M_orig_ch2_bin_C2- mk + yfp.czi -…_ch2_bin.tif

MSS_calc _ver1.2.ijm

Parameters used: step 6 (*Option 1 to 4*), *step 10 (dmax =1964, dmin =7, res_factor =0.75, sampling_factor =0.001), step 11 (Yes)*

Input images (step 8 and 9):

1. Output of *Convert_to_bin_ver1.2.ijm :*

C1-mk + yfp.czi - mk + yfp.czi #10_ch1_bin.tif

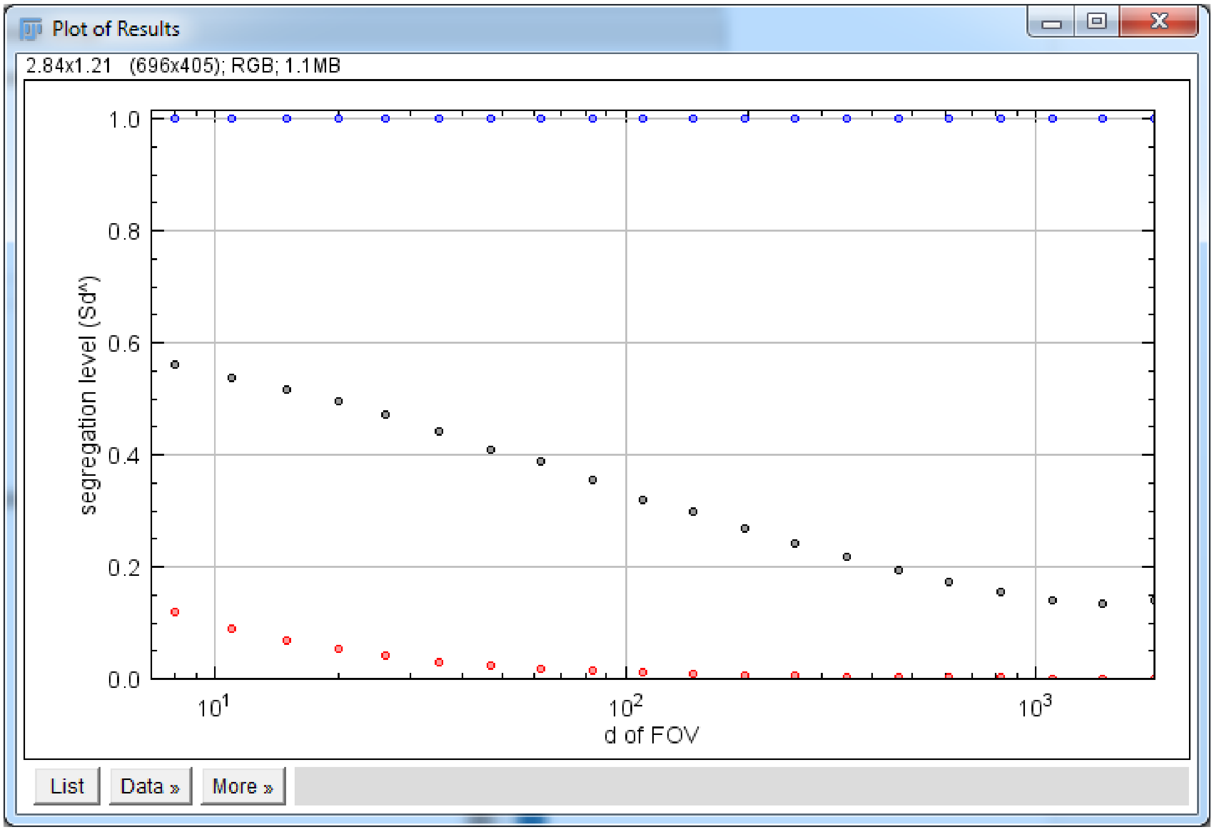
C2-mk + yfp.czi - mk + yfp.czi #10_ch2_bin.tif

2. Output of *Sim_seg_extremes_ver1.2.ijm:*

segmax_ch1_bin_C1-mk +yfp.czi -… _ch1_bin.tif
segmin_ch1_bin_C1-mk +yfp.czi -… _ch1_bin.tif

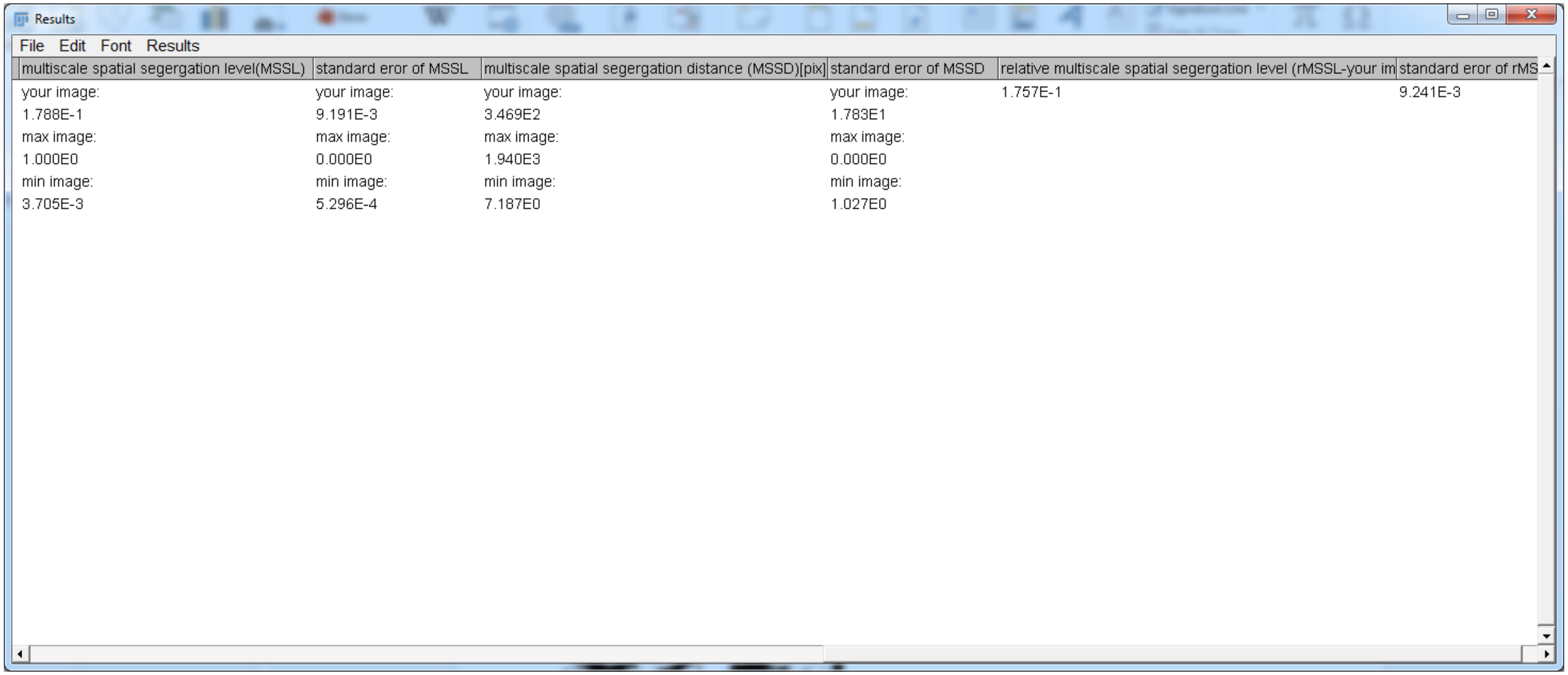
segmax_ch2_bin_C2-mk + yfp.czi - …_ch2_bin.tif
segmax_ch2_bin_C2-mk + yfp.czi - …_ch2_bin.tif

Output result (most important):
-table with results of segregation level vs d of FOV for users sample image (= Your image), max seg image and min seg image that get plotted in graph (step 11 , *Yes*) :



● Sd^min, ● Sd^ (user's sample image) and ● Sd^max

Part of the Results table showing average segregation measures:



multiscale spatial segergation level(MSSL)	standard eror of MSSL	multiscale spatial segergation distance (MSSD)[pix]	standard eror of MSSD	relative multiscale spatial segergation level (rMSSL-your im	standard eror of rMS
your image:	your image:	your image:	your image:	1.757E-1	9.241E-3
1.788E-1	9.191E-3	3.469E2	1.783E1		
max image:	max image:	max image:	max image:		
1.000E0	0.000E0	1.940E3	0.000E0		
min image:	min image:	min image:	min image:		
3.705E-3	5.296E-4	7.187E0	1.027E0		