MicroPICell

Introduction to Bioimage Analysis with Fiji/ImageJ

Pasteur NEUBIAS Course on Bioimage Analysis
Early Career Investigators Track
Mélodie Ambroset, MicroPICell core facility, Nantes
Assisted by Heloïse Monnet, Collège de France

cnrs BioCore Nantes Université

Plan

I/ Bioimage Analysis (BIA)

II/ Fiji

III/ Fundamentals

IV/ Image ethics

V/ Quantification

Hands-on 1: Count nuclei and determine which ones are green

Hands-on 2: Quantify the number of red spots per nucleus

VI/ Intro to automation

V/ Quantification: Overview

Concepts we are going to go over:

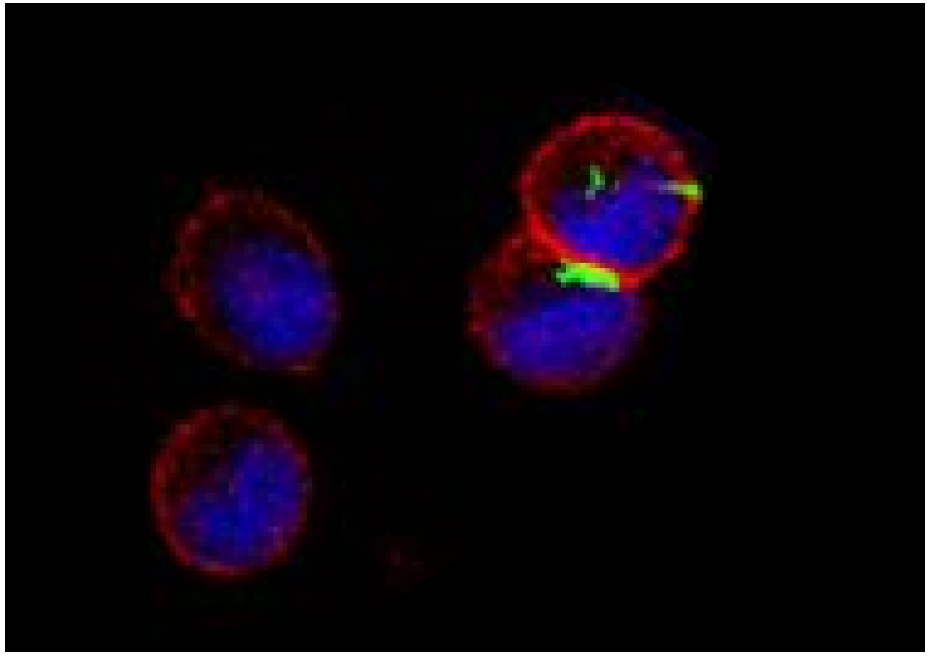
- Segmentation
- Thresholding
- Mask/ binary image
- ROI: Region of Interest
- Binary operations on masks
- Filters
- Maths on images

Tools we are going to use:

- Threshold
- Analyze Particles
- Watershed
- Z Projection
- Find Maxima

V/ Quantification: Why do we need computers?

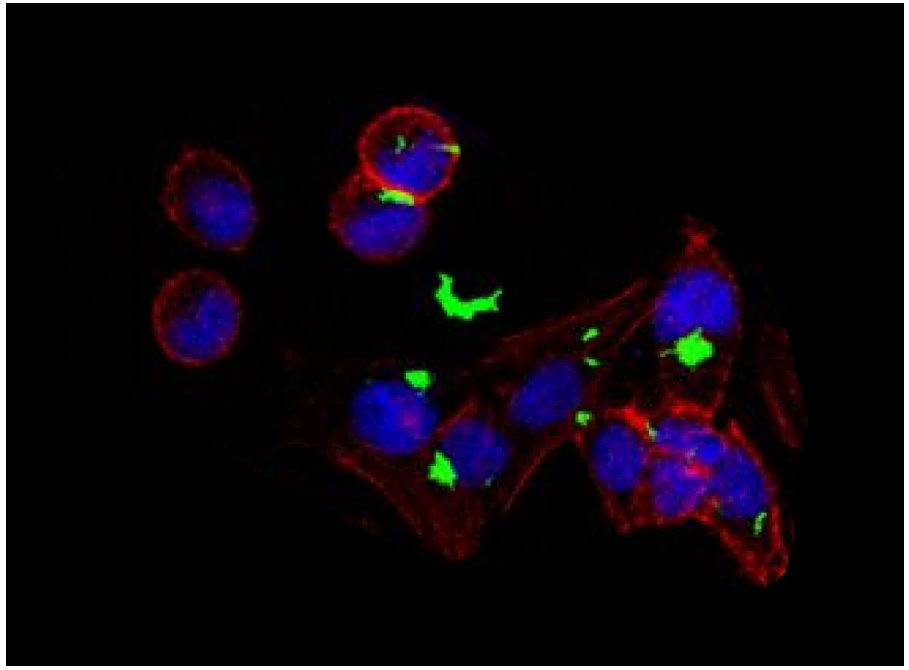
How many cells do you see?



4 cells
10 ms

V/ Quantification: Why do we need computers?

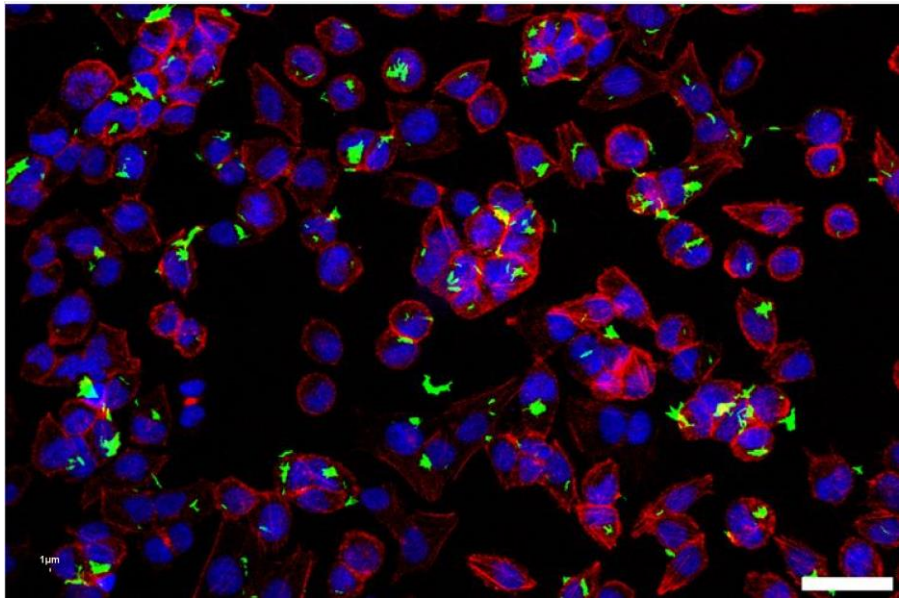
How many cells do you see?



12 cells
10 ms

V/ Quantification: Why do we need computers?

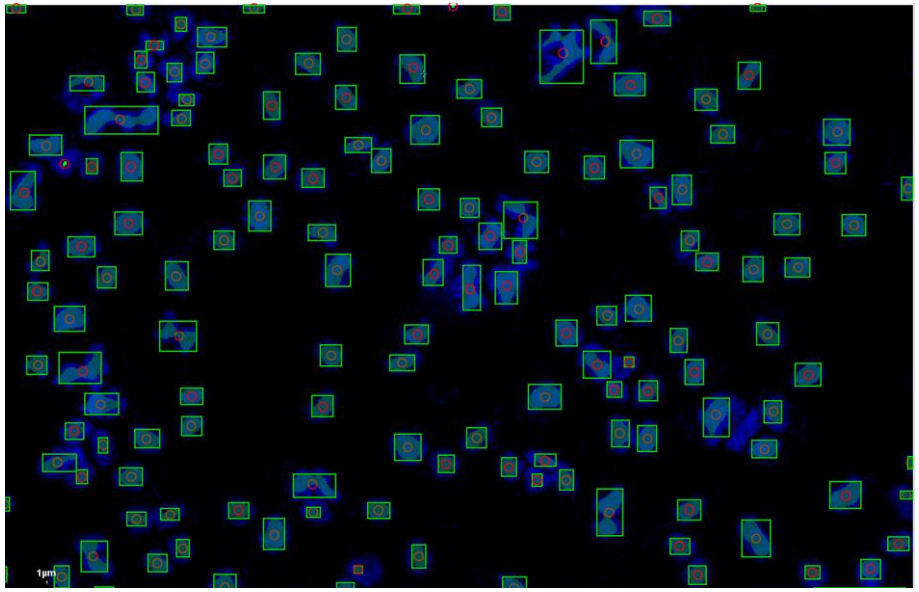
How many cells do you see?



153 cells
100 ms

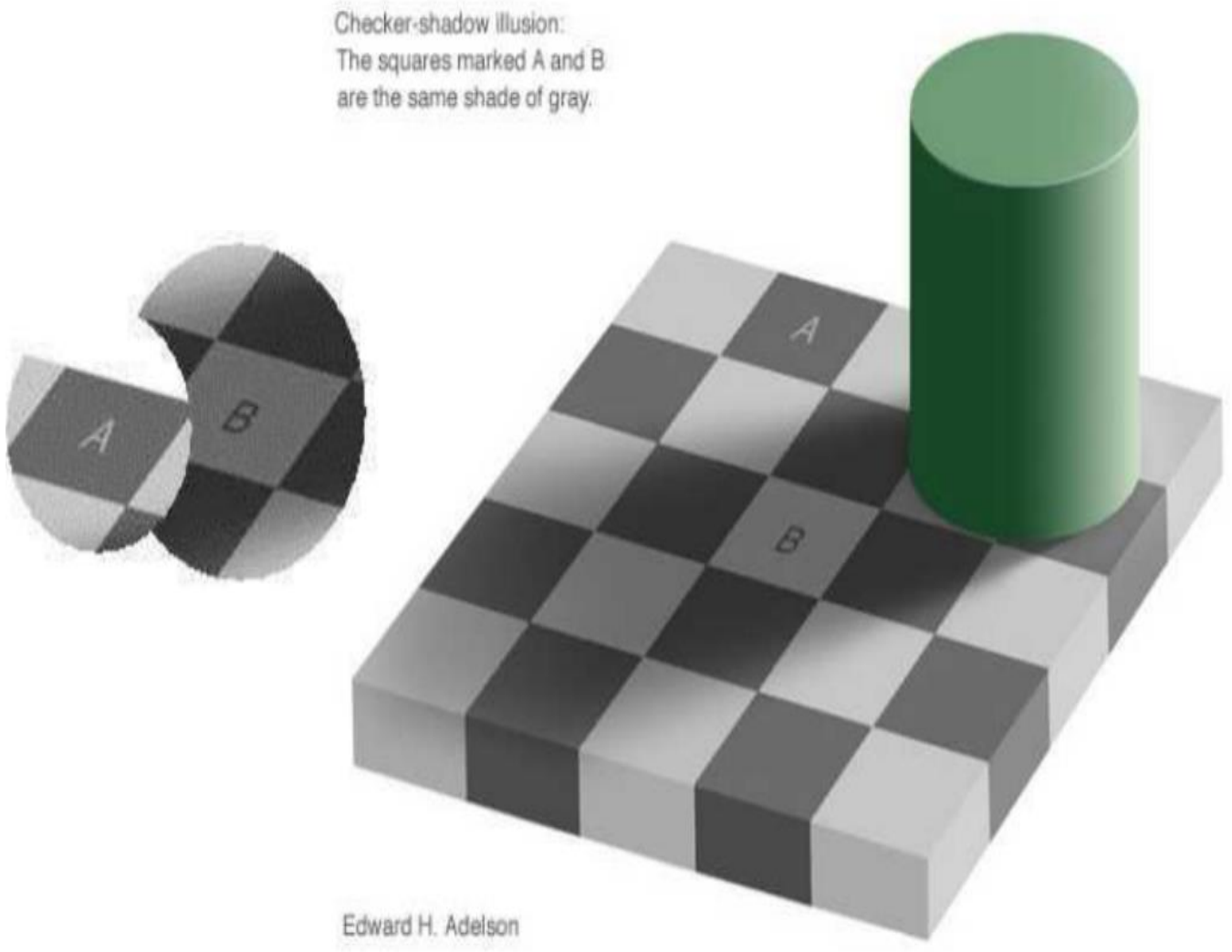
V/ Quantification: Why do we need computers?

How many cells do you see?

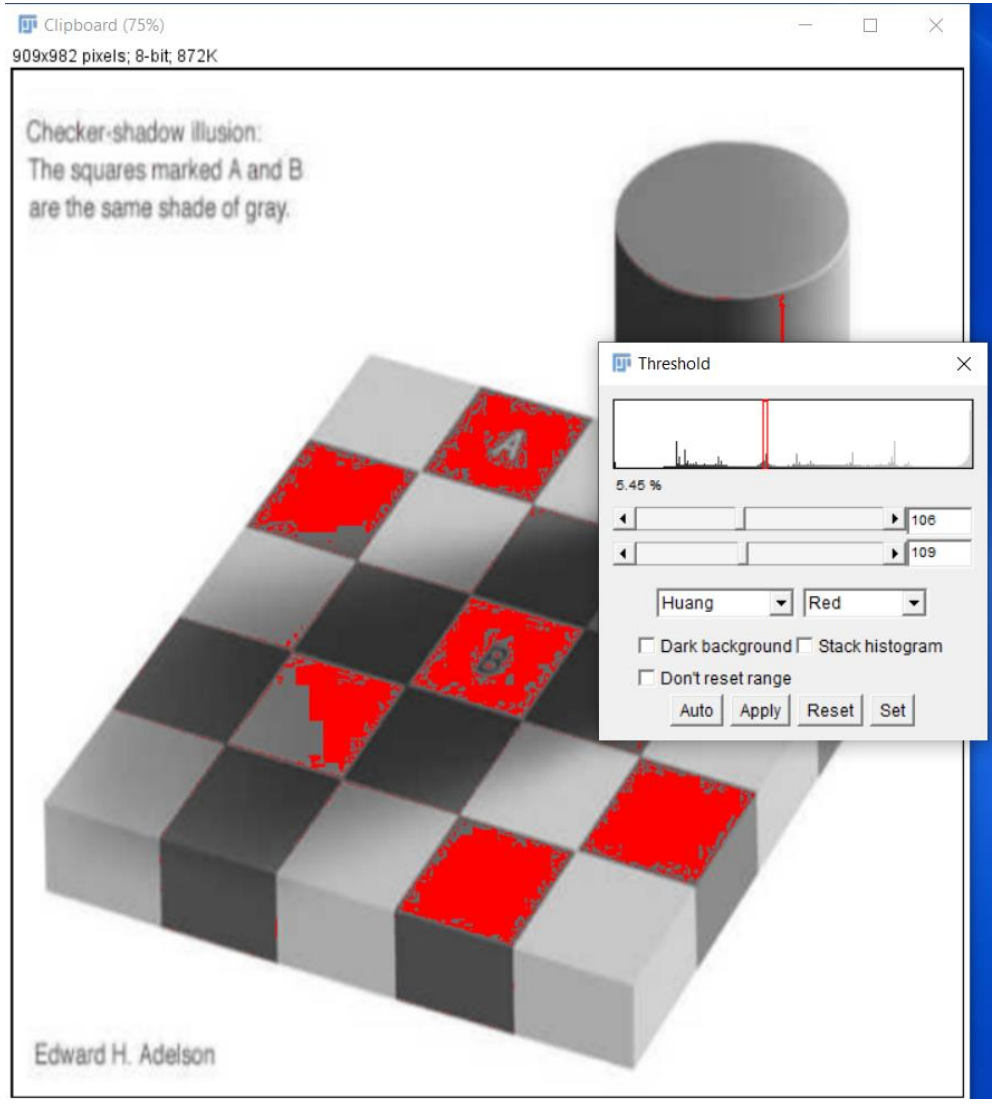


153 cells
100 ms

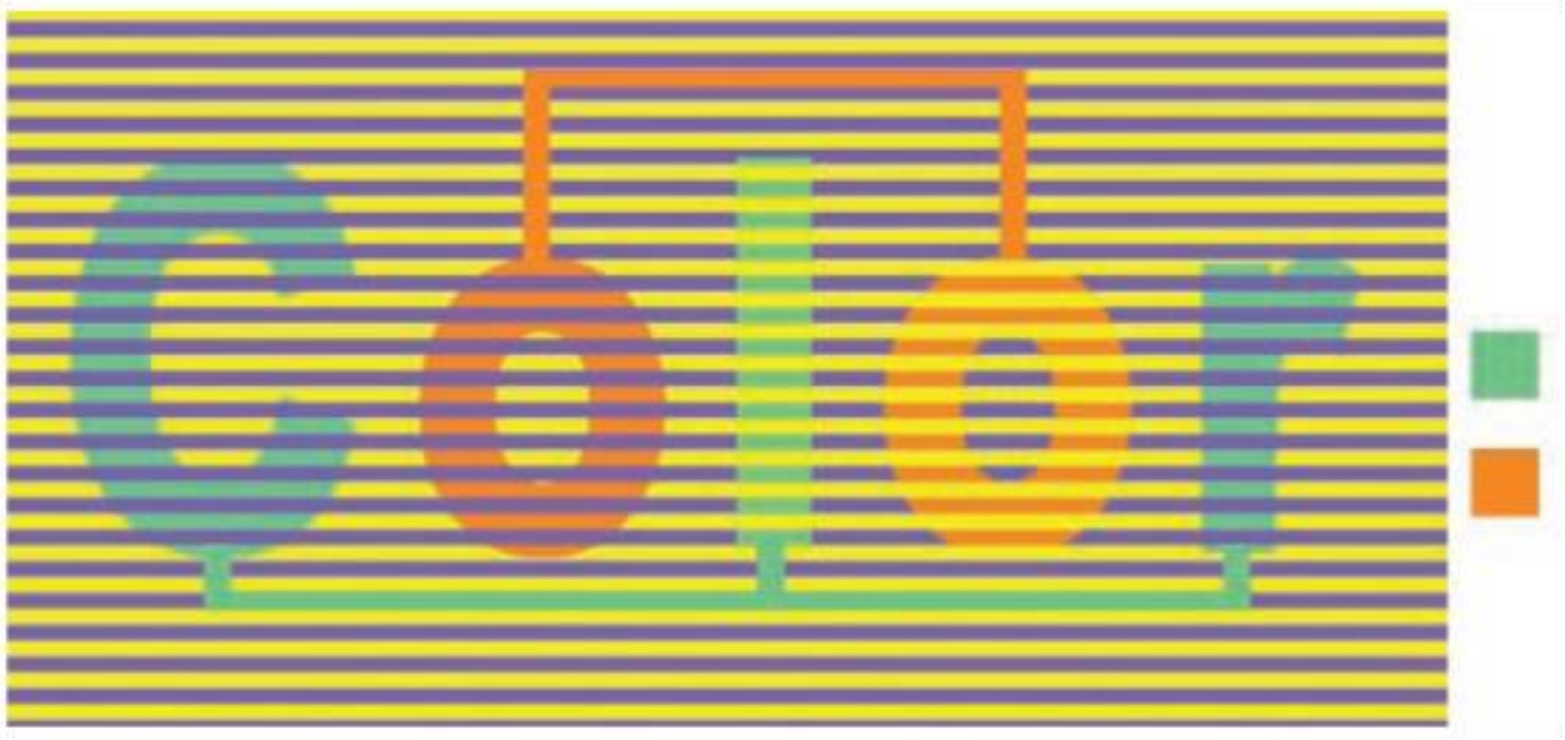
V/ Quantification: Why do we need computers?



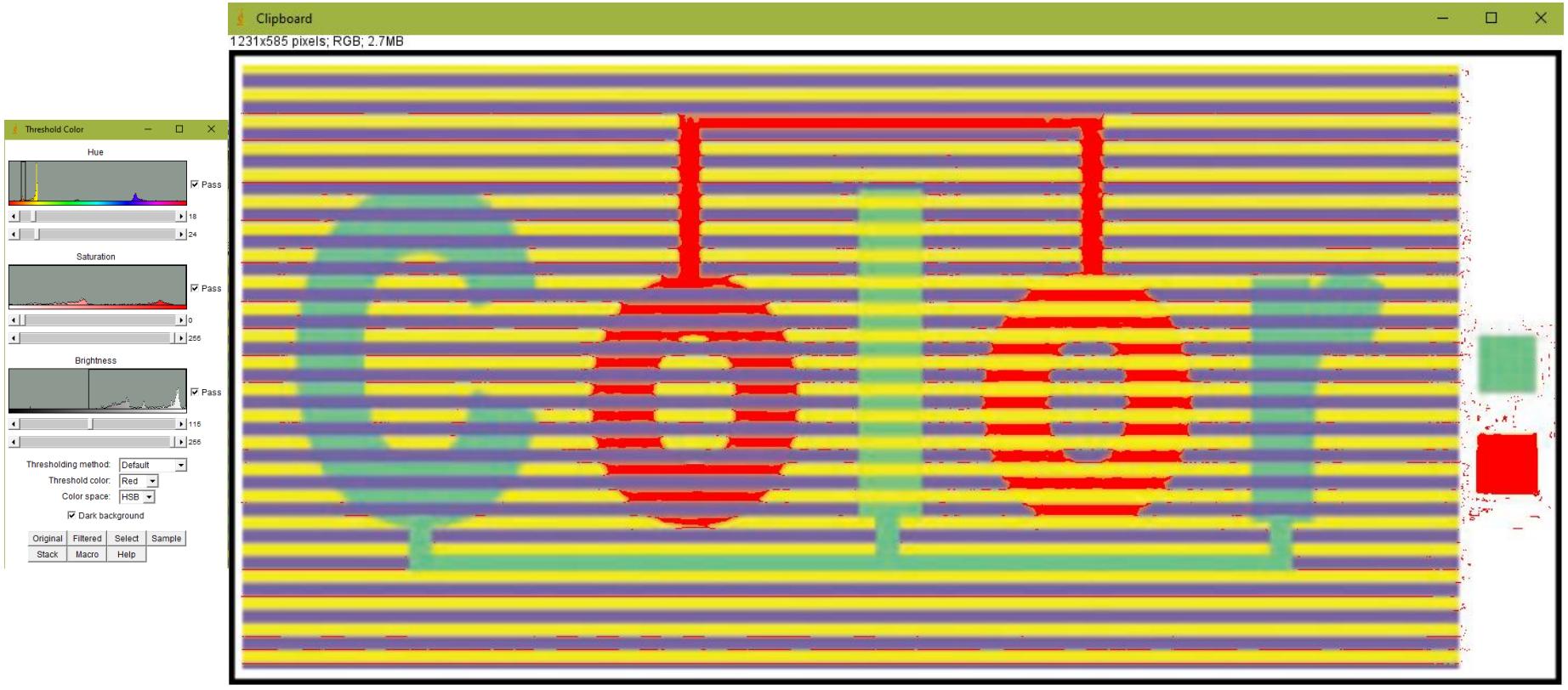
V/ Quantification: Why do we need computers?



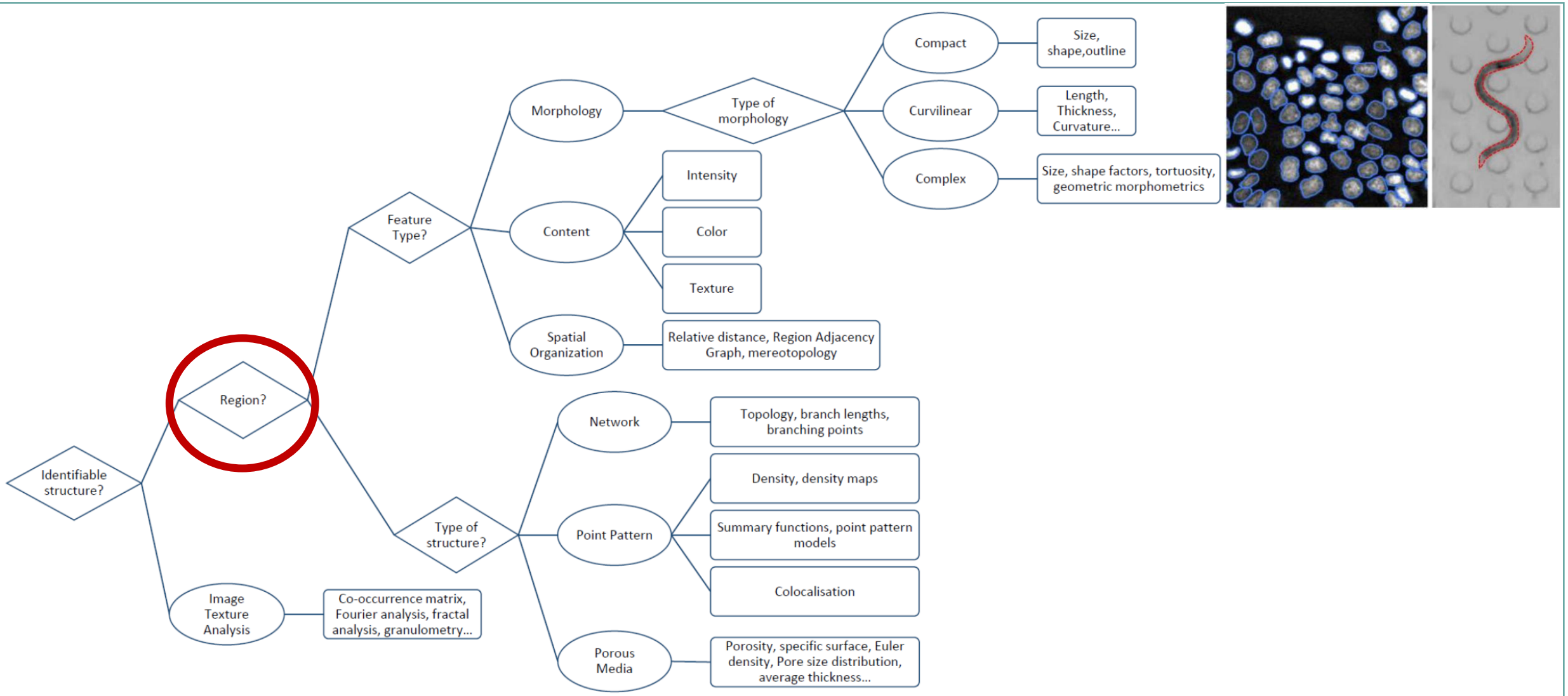
V/ Quantification: Why do we need computers?



V/ Quantification: Why do we need computers?



V/ Quantification: Segmentation

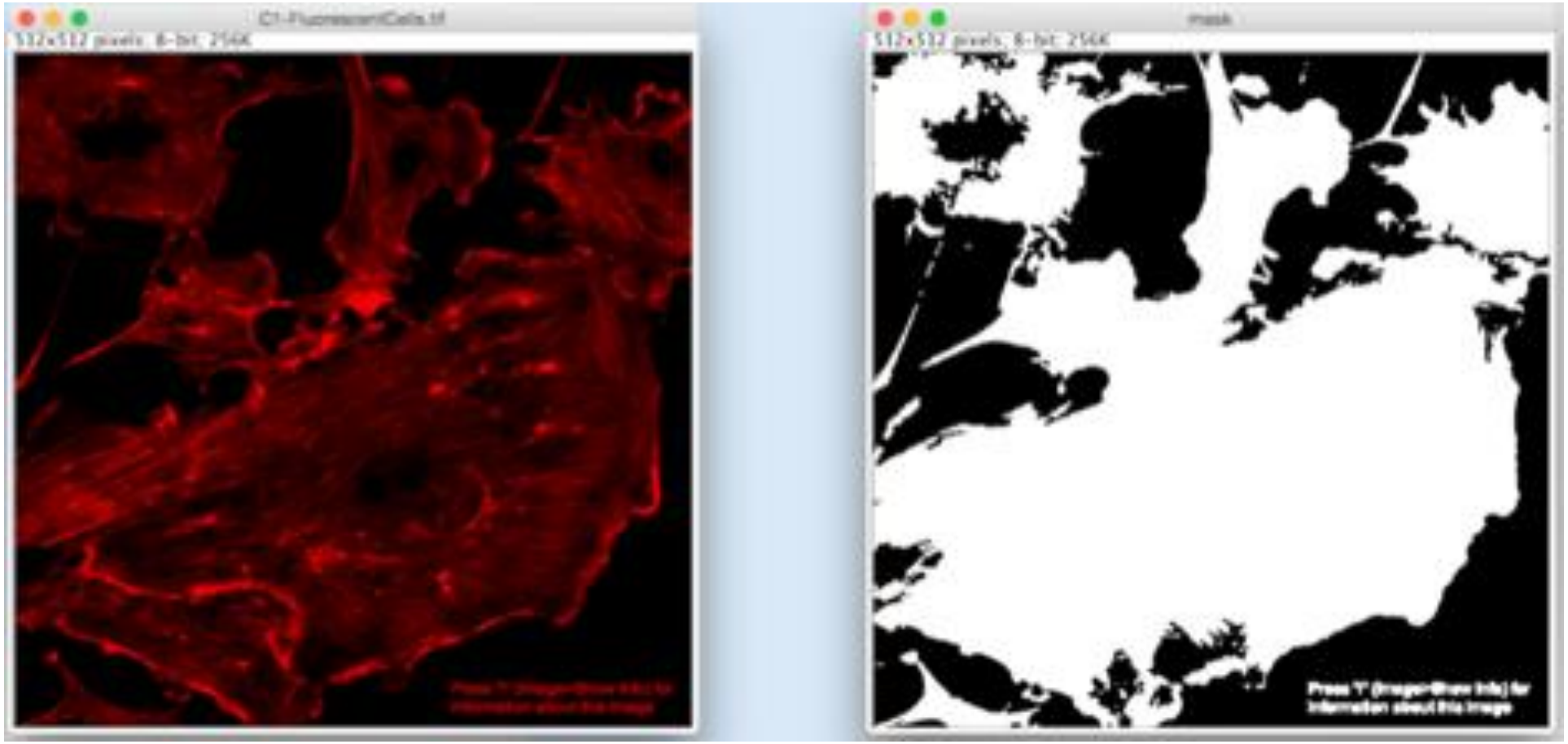


Fazeli, Elnaz, Robert Haase, Michael Doube, Kota Miura, et David Legland. « From Cells to Pixels: Bridging Biologists and Image Analysts Through a Common Language », 2024.

V/ Quantification: Segmentation

Segmentation = division of an image into several parts, generally what we are interested in and what we are not

Binary image/Segmentation mask



V/ Quantification: Segmentation: Thresholding

Segmentation = division of an image into several parts, generally what we are interested in and what we are not

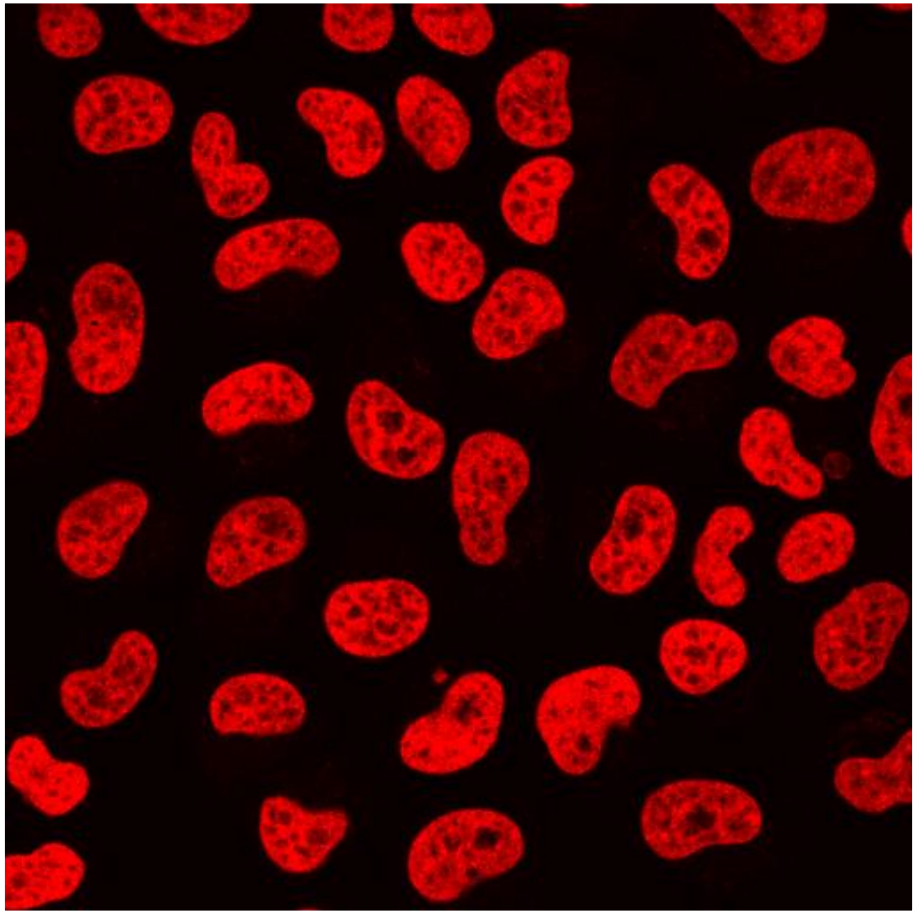
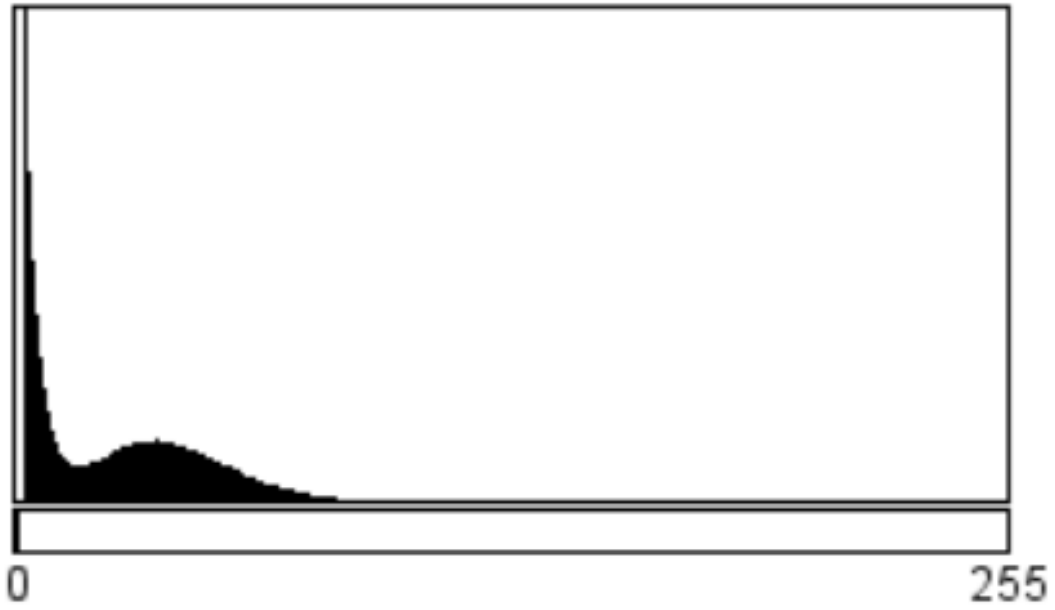


Image > Adjust > Threshold… (**Ctrl + t**)

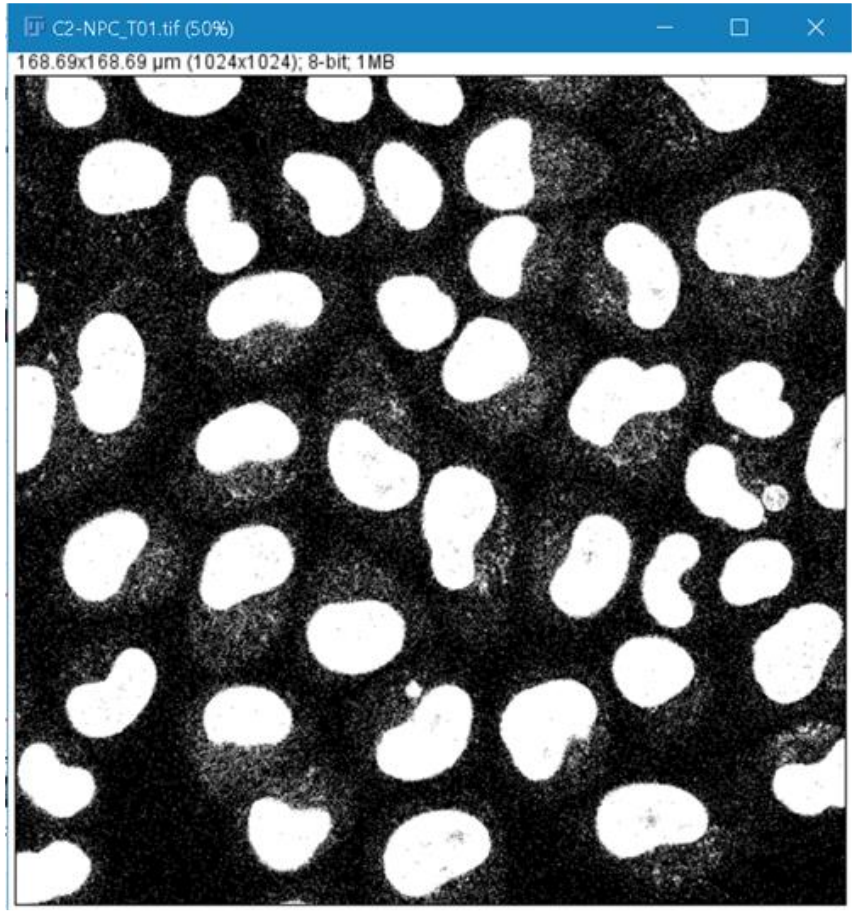


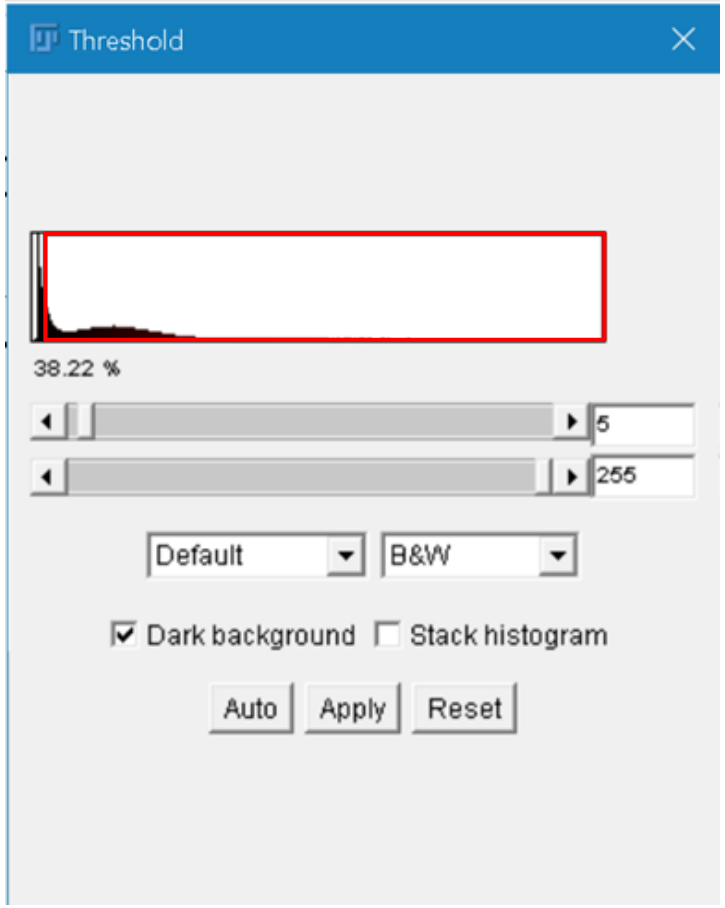
Count: 1048576
Mean: 13.777
StdDev: 19.322

Min: 1
Max: 168
Mode: 2 (578261)

V/ Quantification: Segmentation: Thresholding

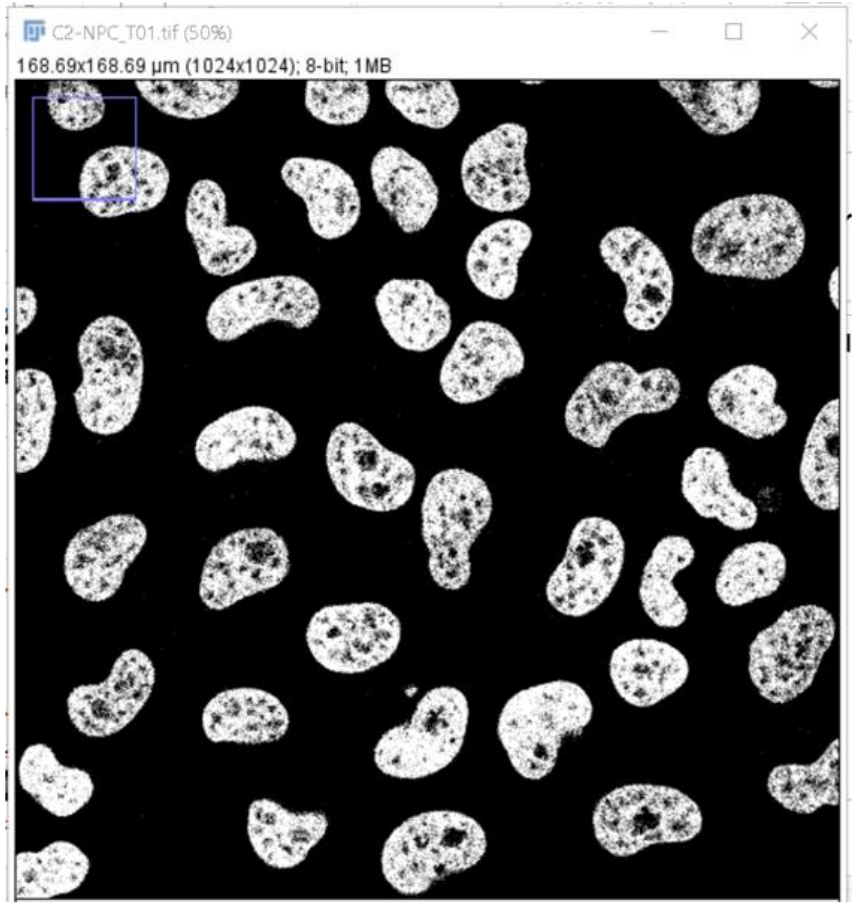
Segmentation = division of an image into several parts, generally what we are interested in and what we are not

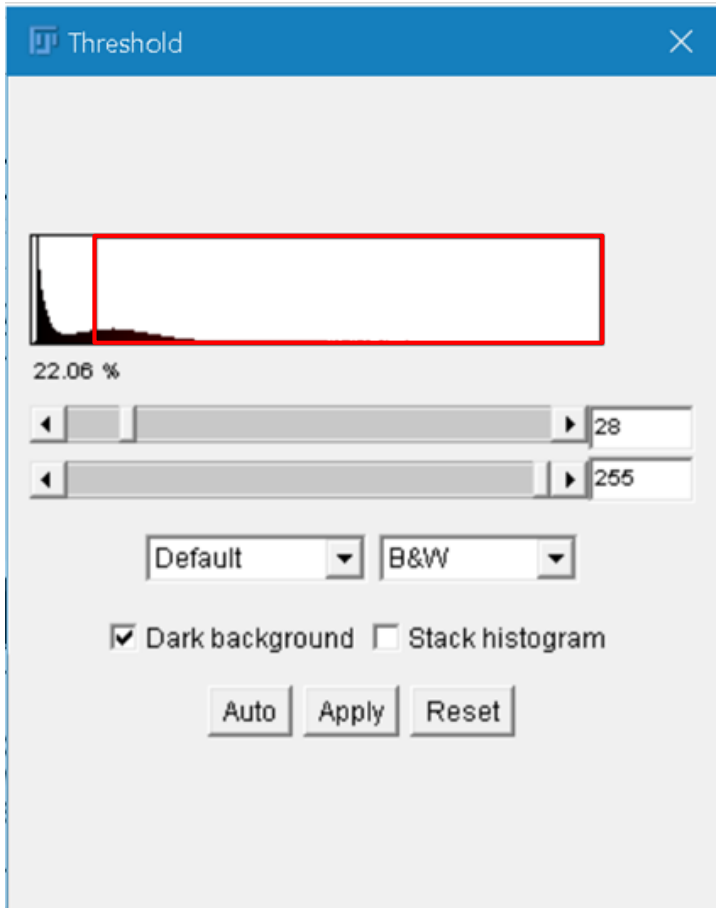




V/ Quantification: Segmentation: Thresholding

Segmentation = division of an image into several parts, generally what we are interested in and what we are not





V/ Quantification: Segmentation: Thresholding

Exercice 1a

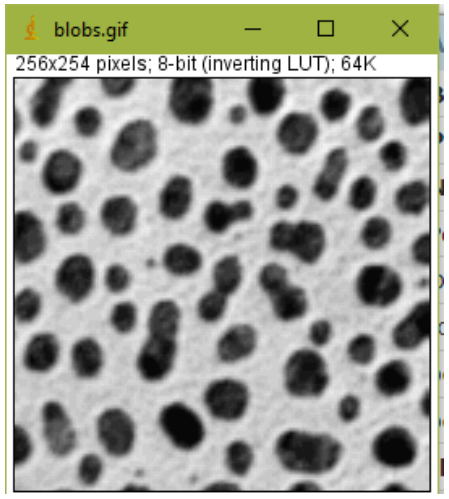
Step 1 File>Open Samples > Blobs

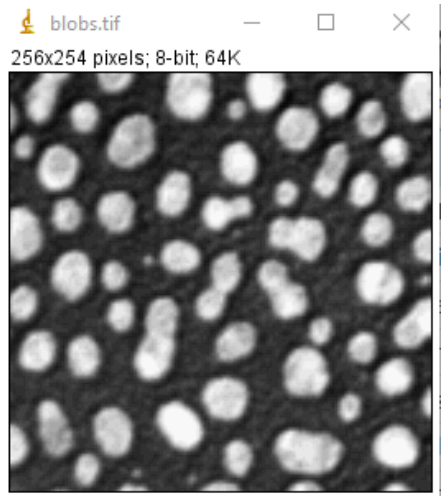
Step 2 Images> Look Up Tables > Invert LUT

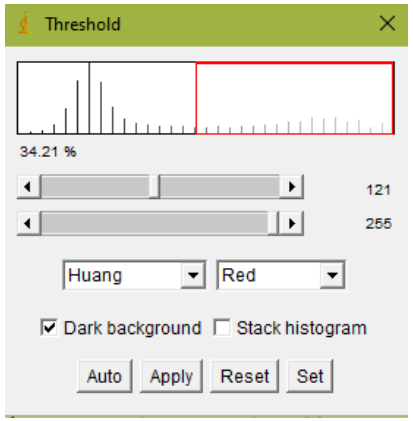
Step 3 Image> Duplicate

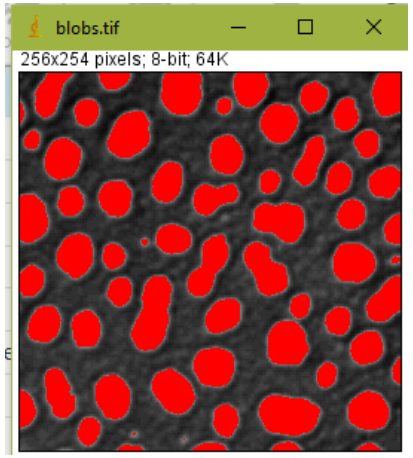
Step 4 Image> Adjust> Threshold…

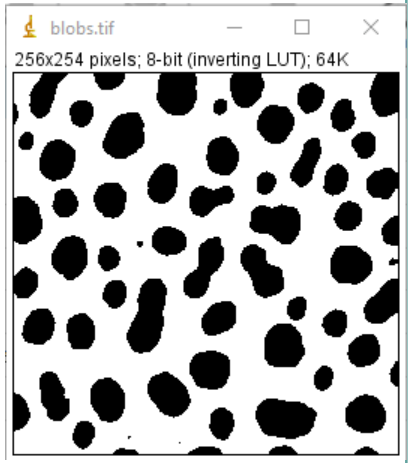
Step 5 **Choose Threshold method and click on Apply**



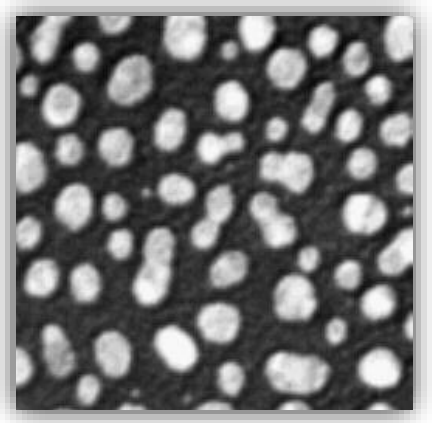




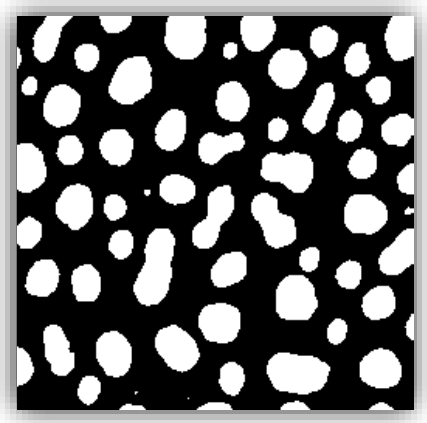




V/ Quantification: Segmentation: Outputs

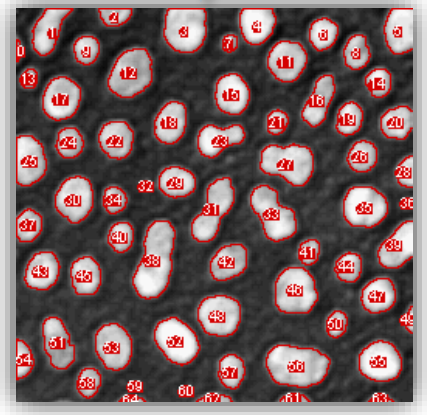


Original image



Binary image

Labelled image

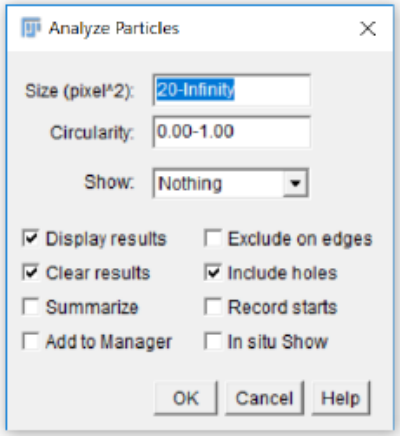


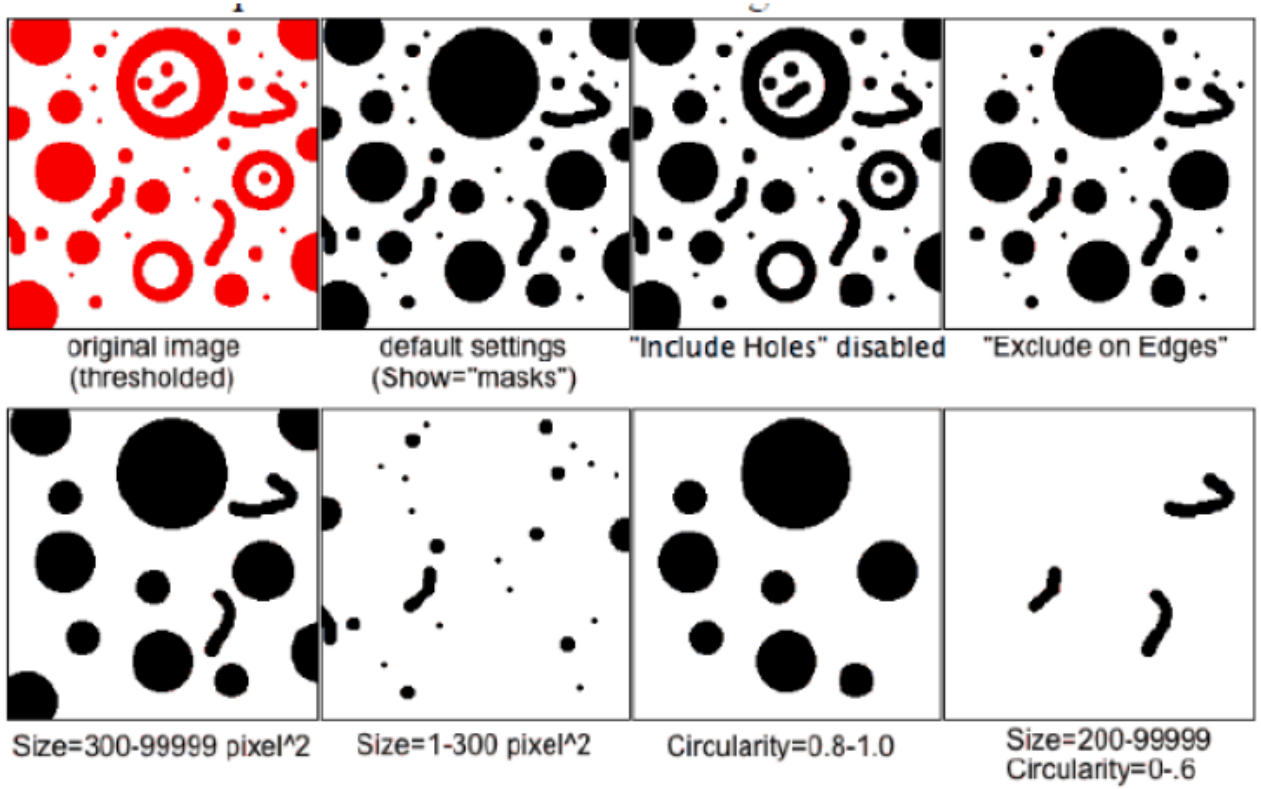
Regions of Interest

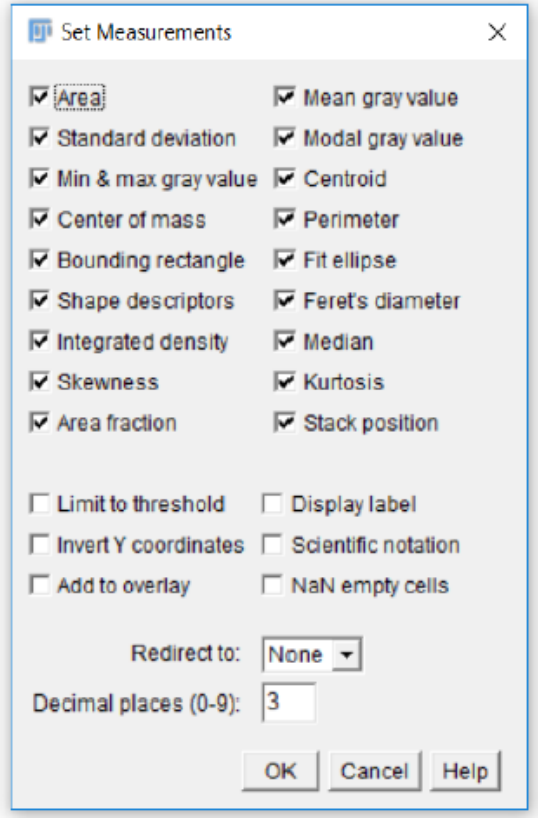
V/ Quantification: Analyze Particles

Analyze > Set measurements

Analyze > Analyze Particles…

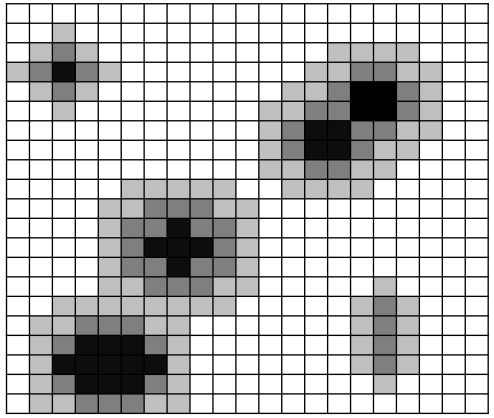




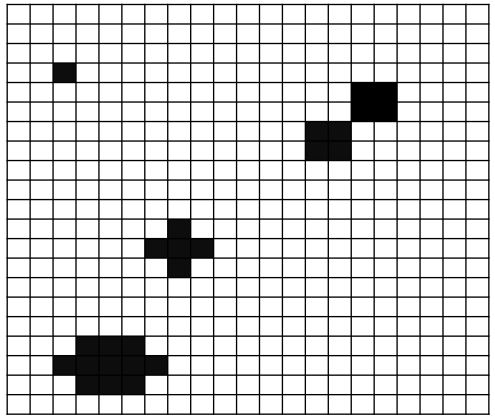


V/ Quantification: Analyze Particles: Connected components analysis

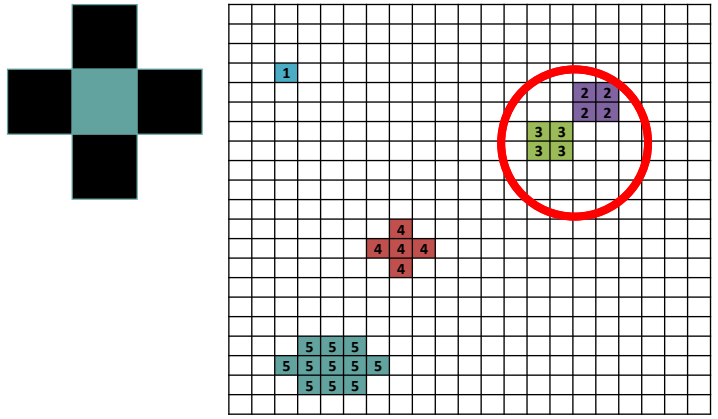
Original

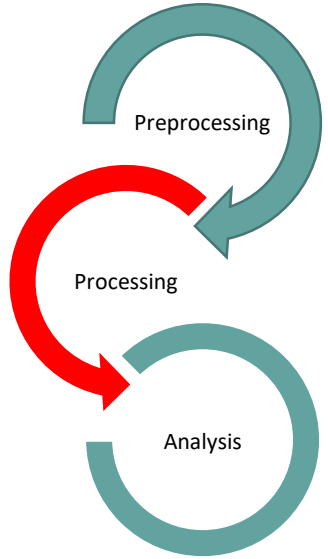


High threshold

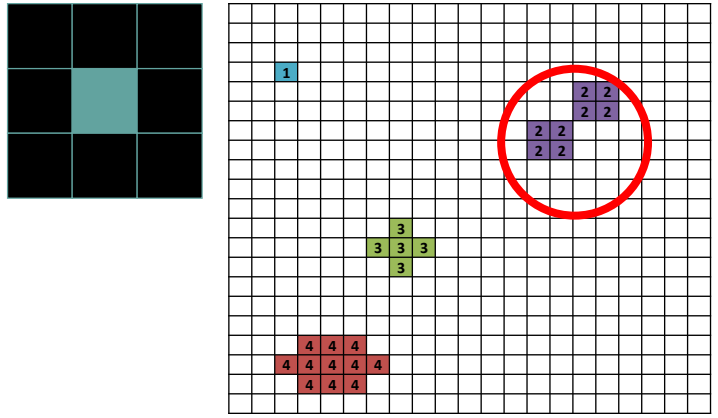


4-connected



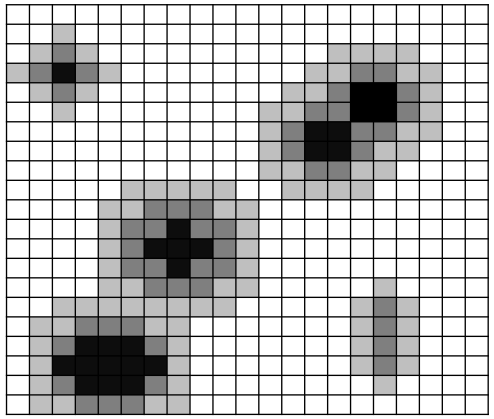


8-connected

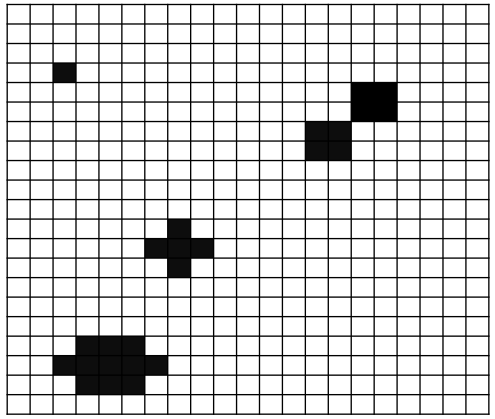


V/ Quantification: Analyze Particles: Connected components analysis

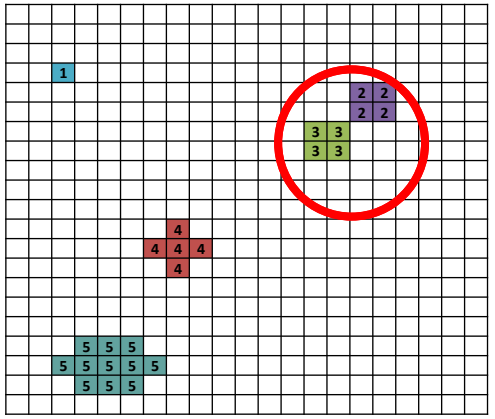
Original

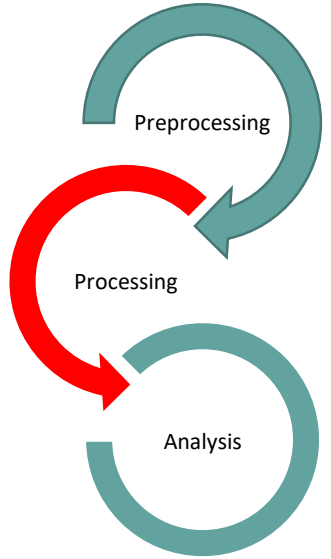


High threshold



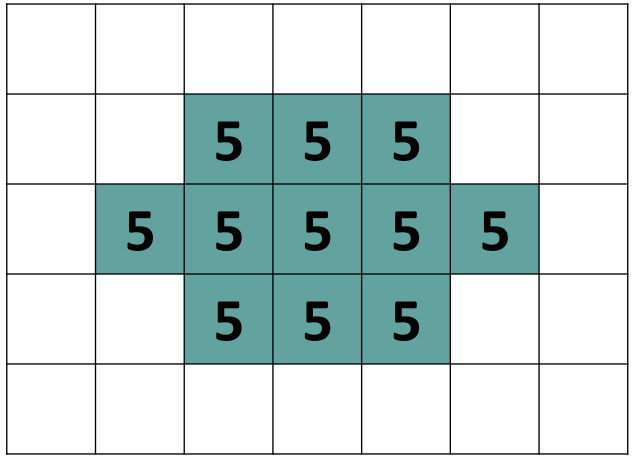
4-connected





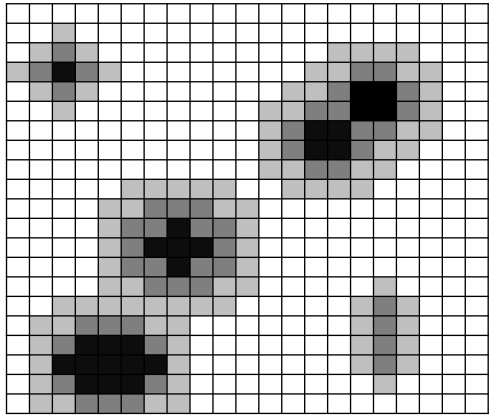
Example of parameters derived from label images:

Area: number of pixels carrying the same label (e.g.: **11 pixels "5"**)
Perimeter: number of pixels carrying the same label, and missing at least one neighbour (e.g.: **8 pixels with 4-connected**)

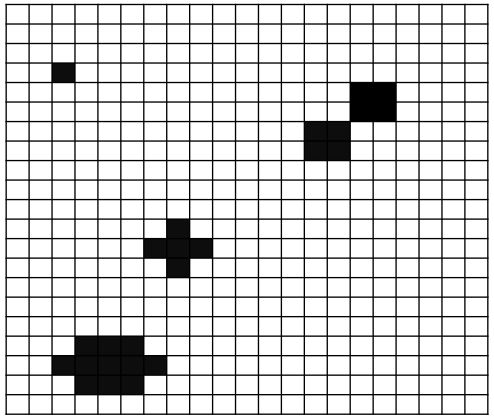


V/ Quantification: Analyze Particles: Connected components analysis

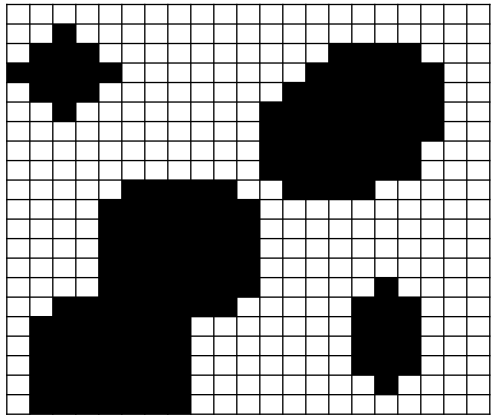
Original



High threshold



Low threshold



Always a bias!

But we should try to always have the same

V/ Quantification: Analyze Particles: Measurements

Code: *Red = Geometry based*, *Blue = Intensity based*, *Orange = Mixed*

Area Area of selection in square pixels or in calibrated square units (e.g., mm^2, μm^2, etc.) if Analyze▷Set Scale…↓ was used to spatially calibrate the image.

Mean gray value Average gray value within the selection. This is the sum of the gray values of all the pixels in the selection divided by the number of pixels. Reported in calibrated units (e.g., optical density) if Analyze▷Calibrate…↓ was used to calibrate the image. For RGB images, the mean is calculated by converting each pixel to grayscale using the formula gray = (red + green + blue) / 3 or
gray = 0.299 × red + 0.587 × green + 0.114 × blue if *Weighted RGB Conversions* is checked in Edit▷Options▷Conversions…↑

Standard deviation Standard deviation of the gray values used to generate the mean gray value. Uses the Results Table↑ heading **StdDev**.

Modal gray value Most frequently occurring gray value within the selection. Corresponds to the highest peak in the histogram. Uses the heading **Mode**.

Min & max gray level Minimum and maximum gray values within the selection.

Centroid The center point of the selection. This is the average of the x and y coordinates of all of the pixels in the image or selection. Uses the **X** and **Y** headings.

V/ Quantification: Analyze Particles: Measurements

Center of mass This is the brightness-weighted average of the x and y coordinates all pixels in the image or selection. Uses the **XM** and **YM** headings. These coordinates are the first order spatial moments.

Perimeter The length of the outside boundary of the selection. Uses the heading **Perim.**. With IJ 1.44f and later, the perimeter of a composite selection is calculated by decomposing it into individual selections. Note that the composite perimeter and the sum of the individual perimeters may be different due to use of different calculation methods.

Bounding rectangle The smallest rectangle enclosing the selection. Uses the headings **BX**, **BY**, **Width** and **Height**, where **BX** and **BY** are the coordinates of the upper left corner of the rectangle.

Fit ellipse Fits an ellipse to the selection. Uses the headings **Major**, **Minor** and **Angle**. **Major** and **Minor** are the primary and secondary axis of the best fitting ellipse. **Angle** is the angle between the primary axis and a line parallel to the X-axis of the image. The coordinates of the center of the ellipse are displayed as **X** and **Y** if Centroid is checked. Note that ImageJ cannot calculate the major and minor axis lengths if *Pixel Aspect Ratio* in the Analyze▷Set Scale…↓ dialog is not 1.0. There are several ways to view the fitted ellipse:

- The Edit▷Selection▷Fit Ellipse↑ command replaces an area selection with the best fit ellipse.
- The DrawEllipse macro draws (destructively) the best fit ellipse and the major and minor axis.
- Select *Ellipses* from the *Show:* drop-down menu in the particle analyzer (Analyze▷Analyze Particles…↑) and it will draw the ellipse for each particle in a separate window.

V/ Quantification: Analyze Particles: Measurements

Shape descriptors Calculates and displays the following shape descriptors:

Circularity $4\pi \times ([Area])/([Perimeter]^2)$ with a value of 1.0 indicating a perfect circle. As the value approaches 0.0, it indicates an increasingly elongated shape. Values may not be valid for very small particles. Uses the heading **Circ**.

Aspect ratio The aspect ratio of the particle's fitted ellipse, i.e., ([*Major Axis*])/([*Minor Axis*]). If ***Fit Ellipse*** is selected the **Major** and **Minor** axis are displayed. Uses the heading **AR**.

Roundness $4 \times ([Area])/(\pi \times [Major\ axis]^2)$ or the inverse of *Aspect Ratio*. Uses the heading **Round**.

Solidity ([*Area*])/([*Convex area*]); Note that the Edit▷Selection▷Convex Hull↑ command makes an area selection convex.

Feret's diameter The longest distance between any two points along the selection boundary, also known as maximum caliper. Uses the heading **Feret**. The angle (0--180 degrees) of the Feret's diameter is displayed as **FeretAngle**, as well as the minimum caliper diameter (**MinFeret**). The starting coordinates of the Feret diameter (**FeretX** and **FeretY**) are also displayed (*see also* Feret's Diameter macro and Chamfer distances and Geodesic diameters plugin).

V/ Quantification: Analyze Particles: Measurements

Integrated density The sum of the values of the pixels in the image or selection. This is equivalent to the product of **Area** and **Mean Gray Value**. With IJ 1.44c and later, **Raw integrated density** (sum of pixel values) is displayed under the heading **RawIntDen** when **Integrated density** is enabled. The Dot Blot Analysis tutorial demonstrates how to use this option to analyze a dot blot assay.

Median The median value of the pixels in the image or selection.

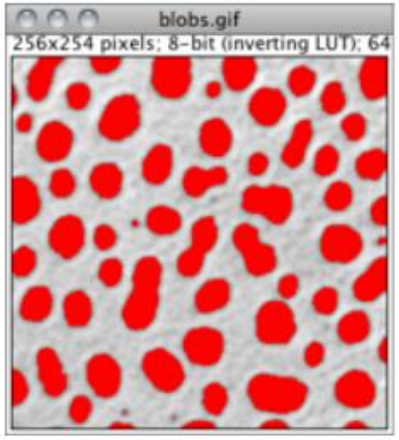
Skewness The third order moment about the mean. The documentation for the Moment Calculator plugin explains how to interpret spatial moments. Uses the heading **Skew**.

Kurtosis The fourth order moment about the mean. Uses the heading **Kurt**.

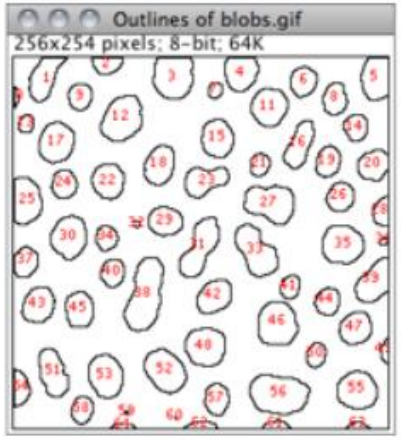
Area fraction For thresholded images is the percentage of pixels in the image or selection that have been highlighted in red using Image▷Adjust▷Threshold... [T]↑. For non-thresholded images is the percentage of non-zero pixels. Uses the heading **%Area**.

V/ Quantification: Analyze Particles: Outputs

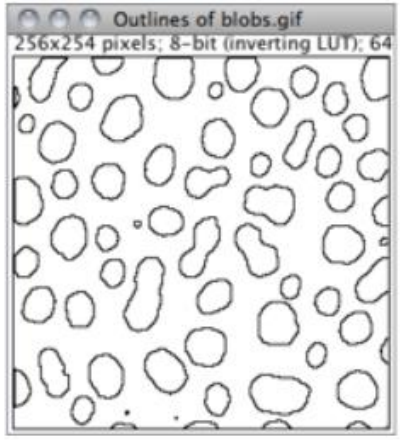
Original (thresholded)



Outlines

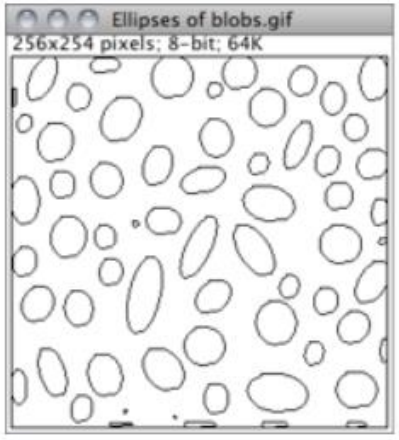


Bare Outlines

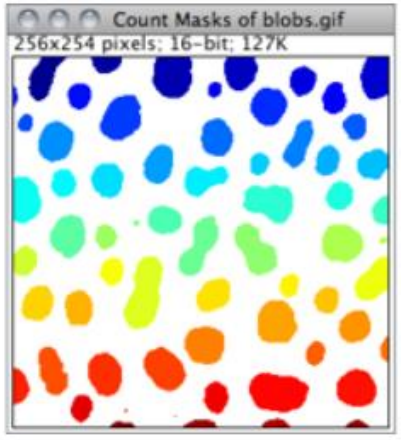


Masks

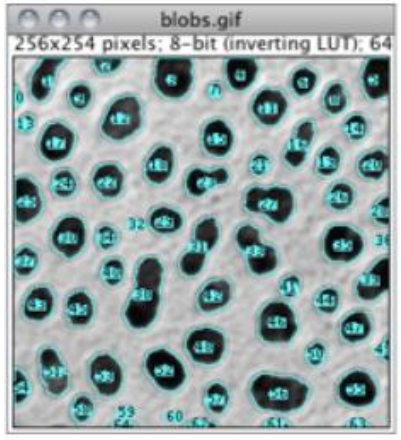
Ellipses



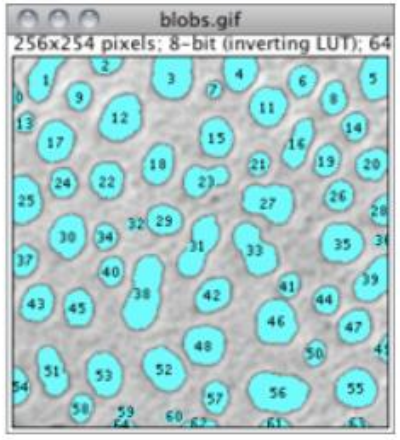
Count Masks



Overlay Outlines



Overlay Masks



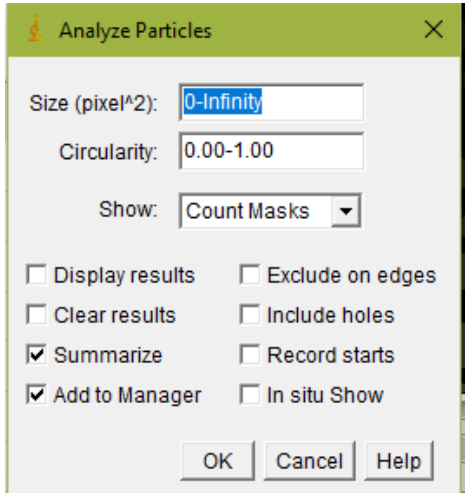
V/ Quantification: Analyze Particles: *Hands-on!*

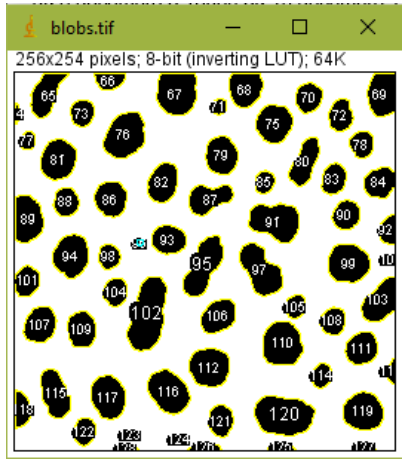
Exercice 1b

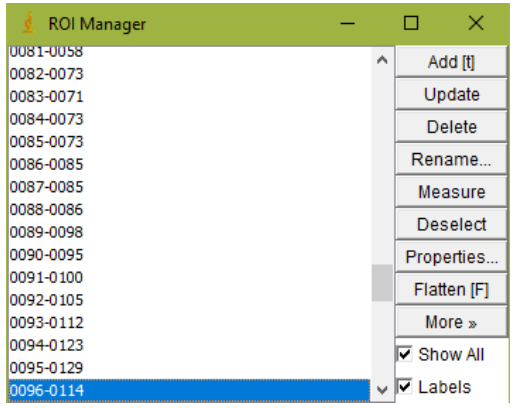
Step 6 — Analyze > Analyze Particles

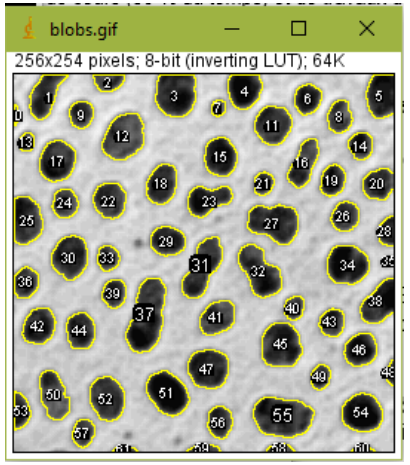
Step 7 — Do it again but exclude objects that are too small → Click on a too small ROI and click Measure in the ROI Manager

Step 8 — Move the ROIs on another image → Open Blobs, then click on Show all in the ROI Manager

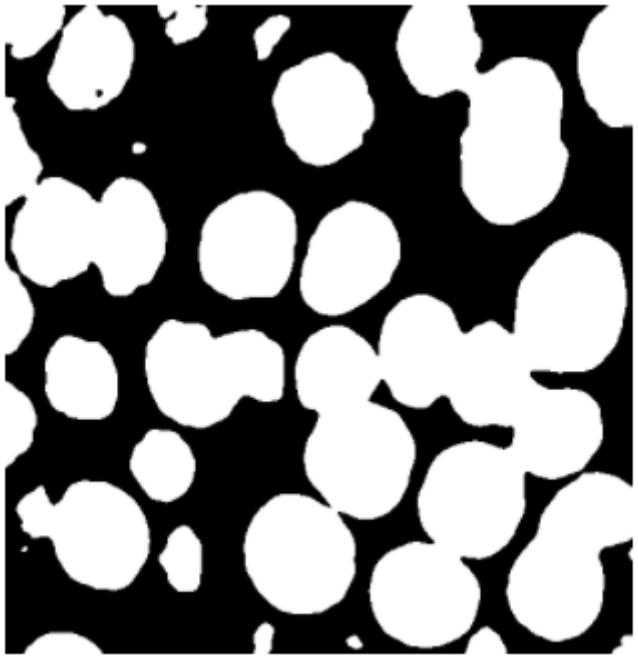


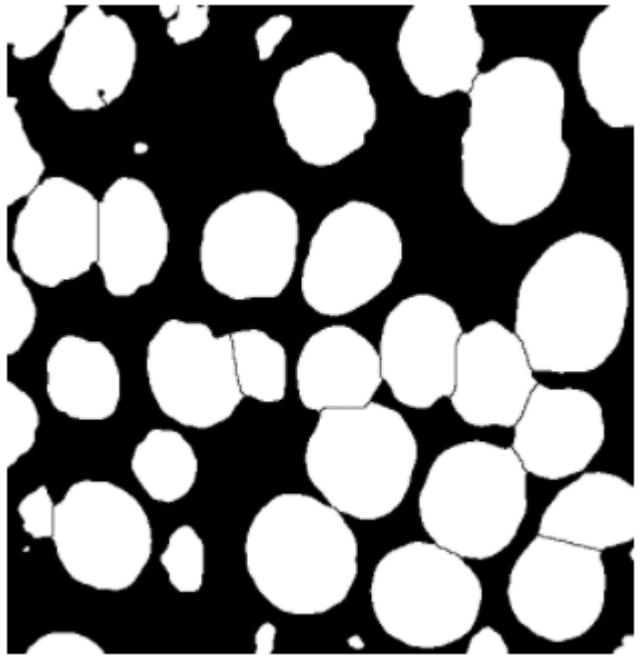






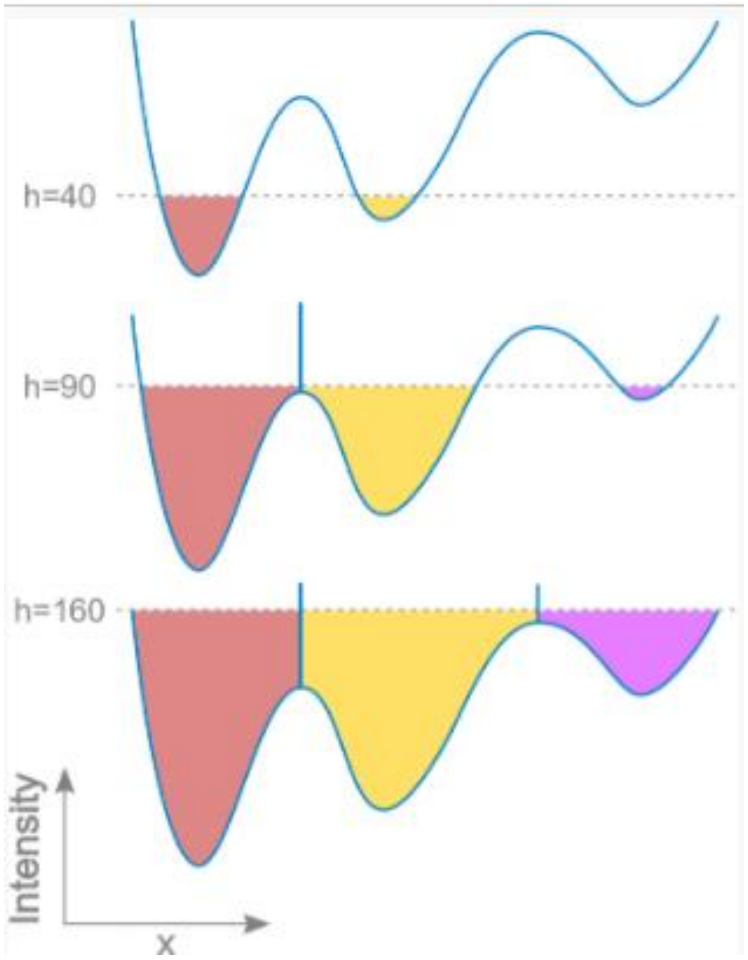
V/ Quantification: Watershed

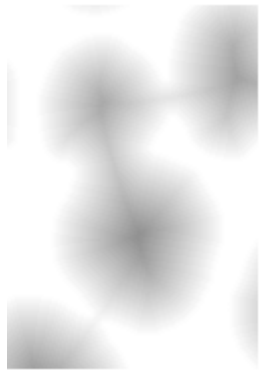


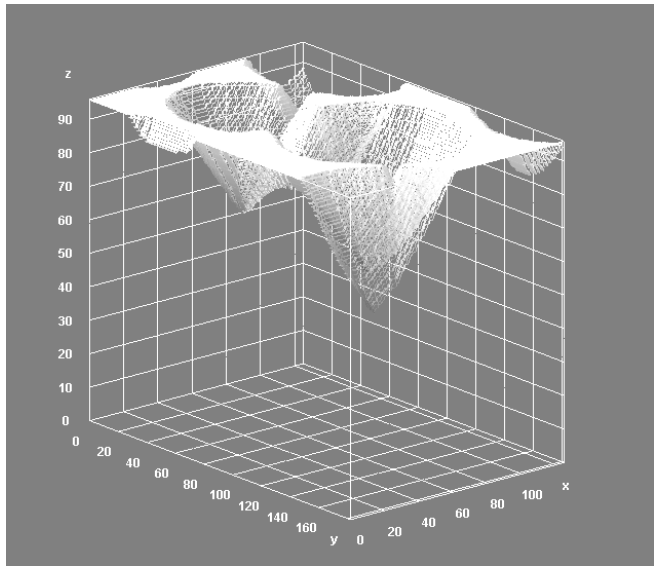


Process > Binary > Watershed

V/ Quantification: Watershed







How many objects before and after?

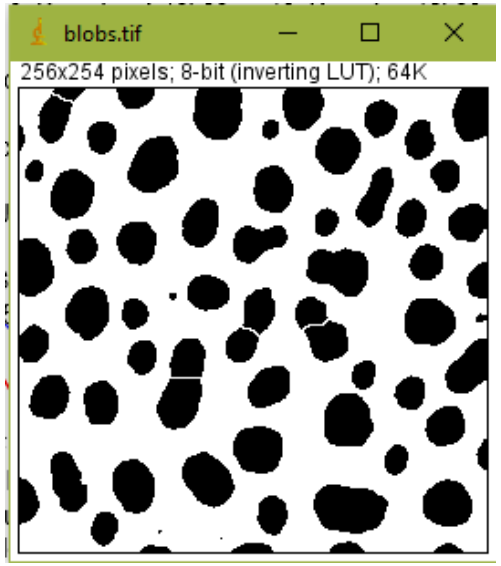
V/ Quantification: Watershed

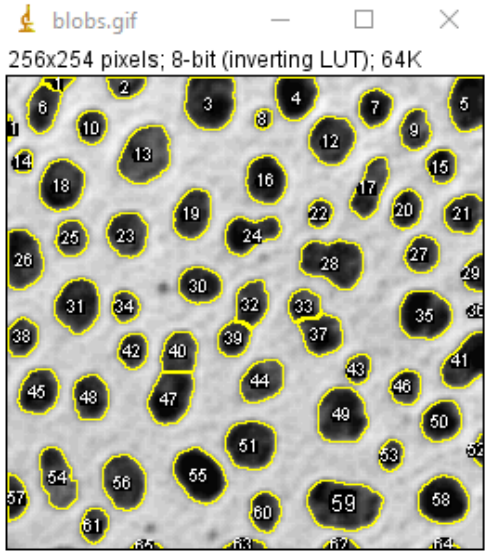
Exercice 1c

Step 9 Process > Binary > Watershed

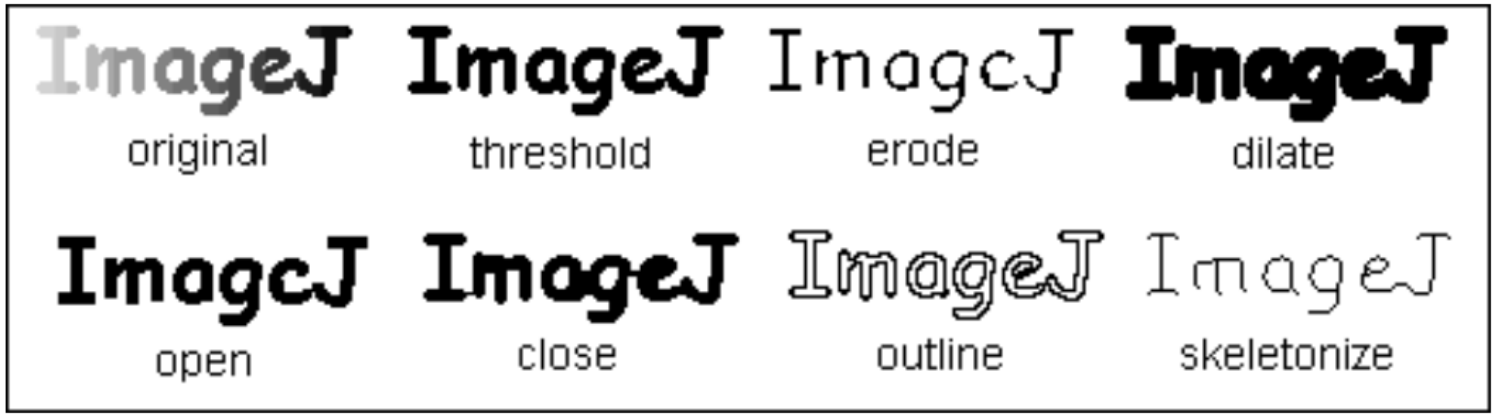
Step 10 Redo Analyze > Particles and visualize the result on the original image

Step 11 Now try to exclude objects touching the borders

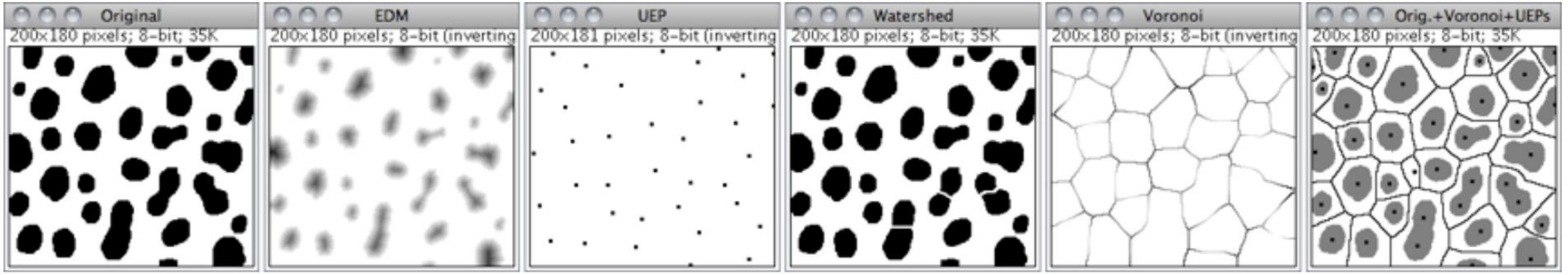




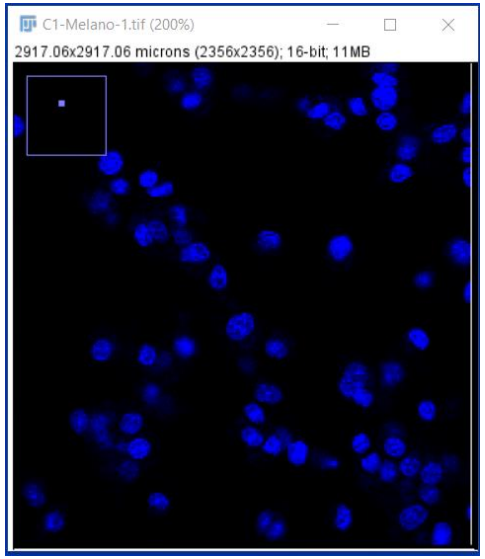
V/ Quantification: Binary operations

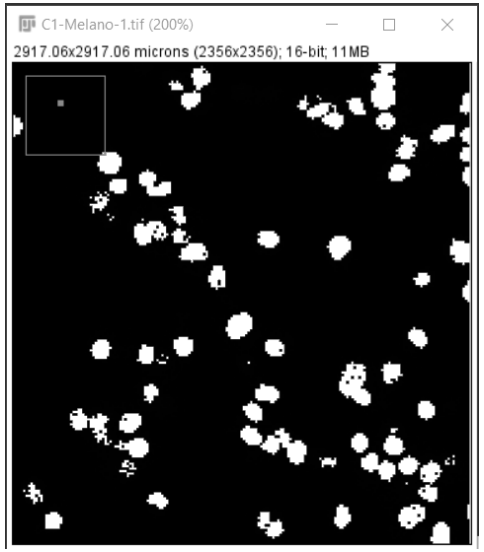


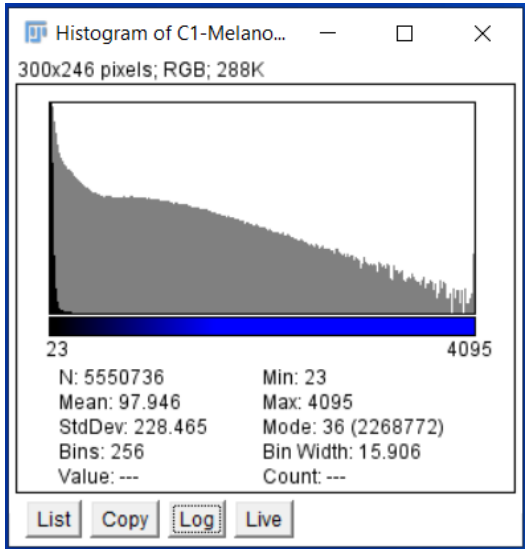
Original | Distance Map↓ | ULTIMATE POINTS↓ | Watershed↓ | Voronoi↓ | Original + Voronoi↓ + ULTIMATE POINTS↓

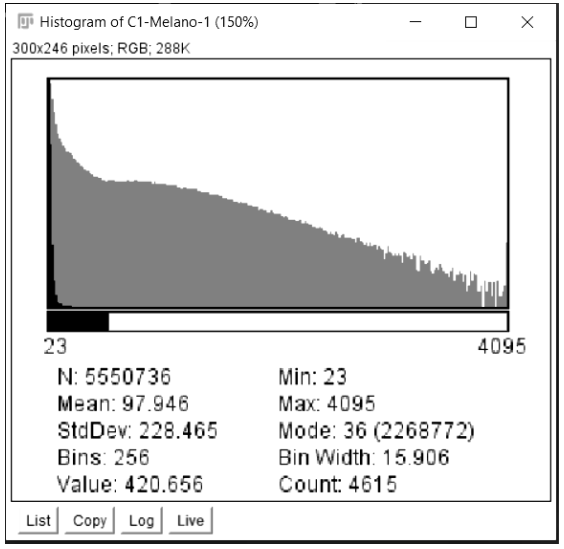


V/ Quantification: Filters





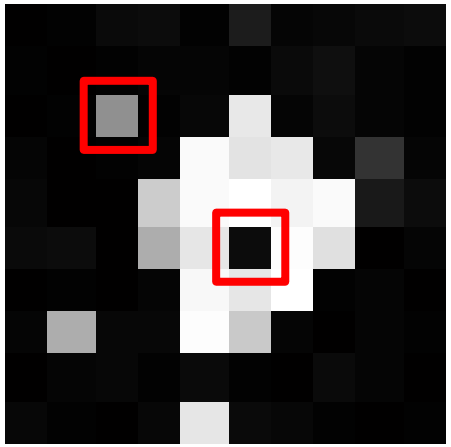




V/ Quantification: Filters

Application of a filter to smooth/denoise the image

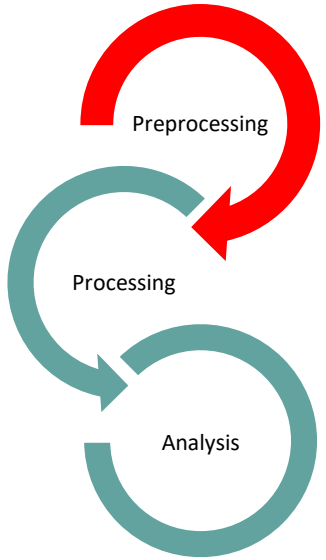
0	1	4	5	1	11	2	3	4	5
1	0	1	2	2	1	4	6	2	1
0	1	56	1	3	90	2	5	2	1
2	0	1	2	97	88	90	3	20	2
3	0	0	79	97	99	94	97	10	5
4	5	0	67	89	5	98	87	0	2
0	1	0	2	96	89	99	1	2	0
2	67	3	3	98	78	2	0	2	1
0	2	3	1	4	1	0	4	2	0
3	1	0	3	89	4	3	1	0	1



*

Filter Kernel

0.11111	0.11111	0.11111
0.11111	0.11111	0.11111
0.11111	0.11111	0.11111



0 * 0.1111 + 1 * 0.1111 + 2 * 0.1111 + 1 * 0.1111 + 56 * 0.1111 + 1 * 0.1111 + 0 * 0.1111 + 1 * 0.1111 + 2 * 0.1111

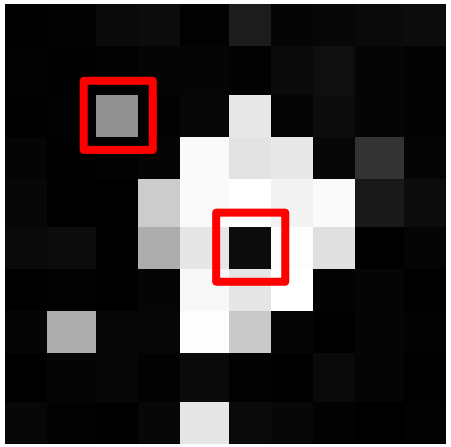
= 7.11

CONVOLUTION

V/ Quantification: Filters

Application of a filter to smooth/denoise the image

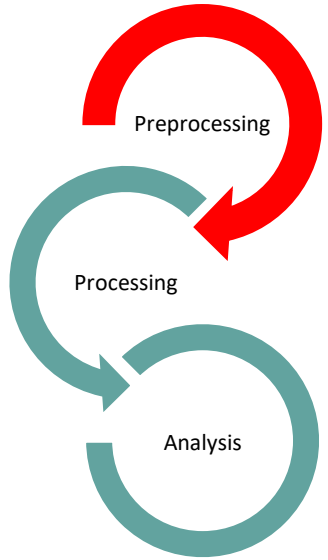
0	1	4	5	1	11	2	3	4	5
1	0	1	2	2	1	4	6	2	1
0	1	**56**	1	3	90	2	5	2	1
2	0	1	2	97	88	90	3	20	2
3	0	0	79	97	99	94	97	10	5
4	5	0	67	89	**5**	98	87	0	2
0	1	0	2	96	89	99	1	2	0
2	67	3	3	98	78	2	0	2	1
0	2	3	1	4	1	0	4	2	0
3	1	0	3	89	4	3	1	0	1



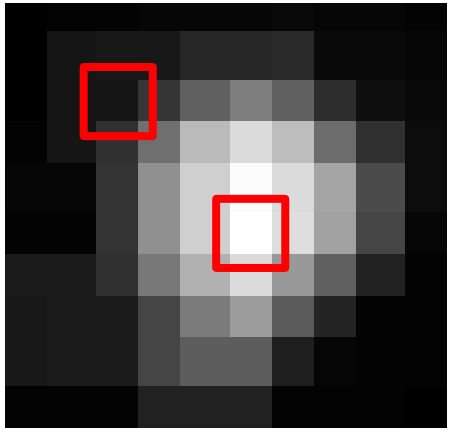
*

Filter Kernel

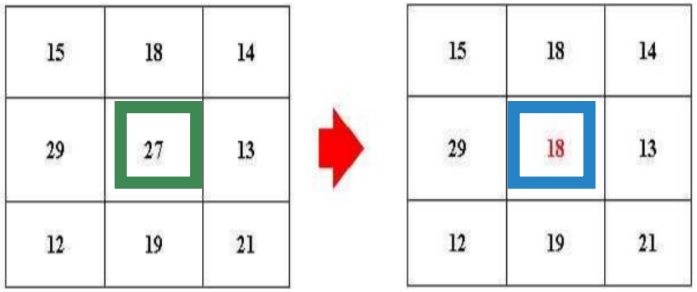
0.11111	0.11111	0.11111
0.11111	0.11111	0.11111
0.11111	0.11111	0.11111



0	1	1	2	2	2	3	2	2	1
0	7	8	8	13	13	14	3	3	2
0	7	7	18	32	42	32	15	5	3
1	7	16	37	62	73	63	36	16	4
2	2	17	48	69	84	73	55	25	4
1	1	17	48	69	85	74	54	23	2
9	9	16	40	59	73	51	32	11	1
8	9	9	23	41	52	30	12	1	1
8	9	9	23	31	31	10	2	1	1
1	1	1	11	11	11	1	1	1	0



Median Filter



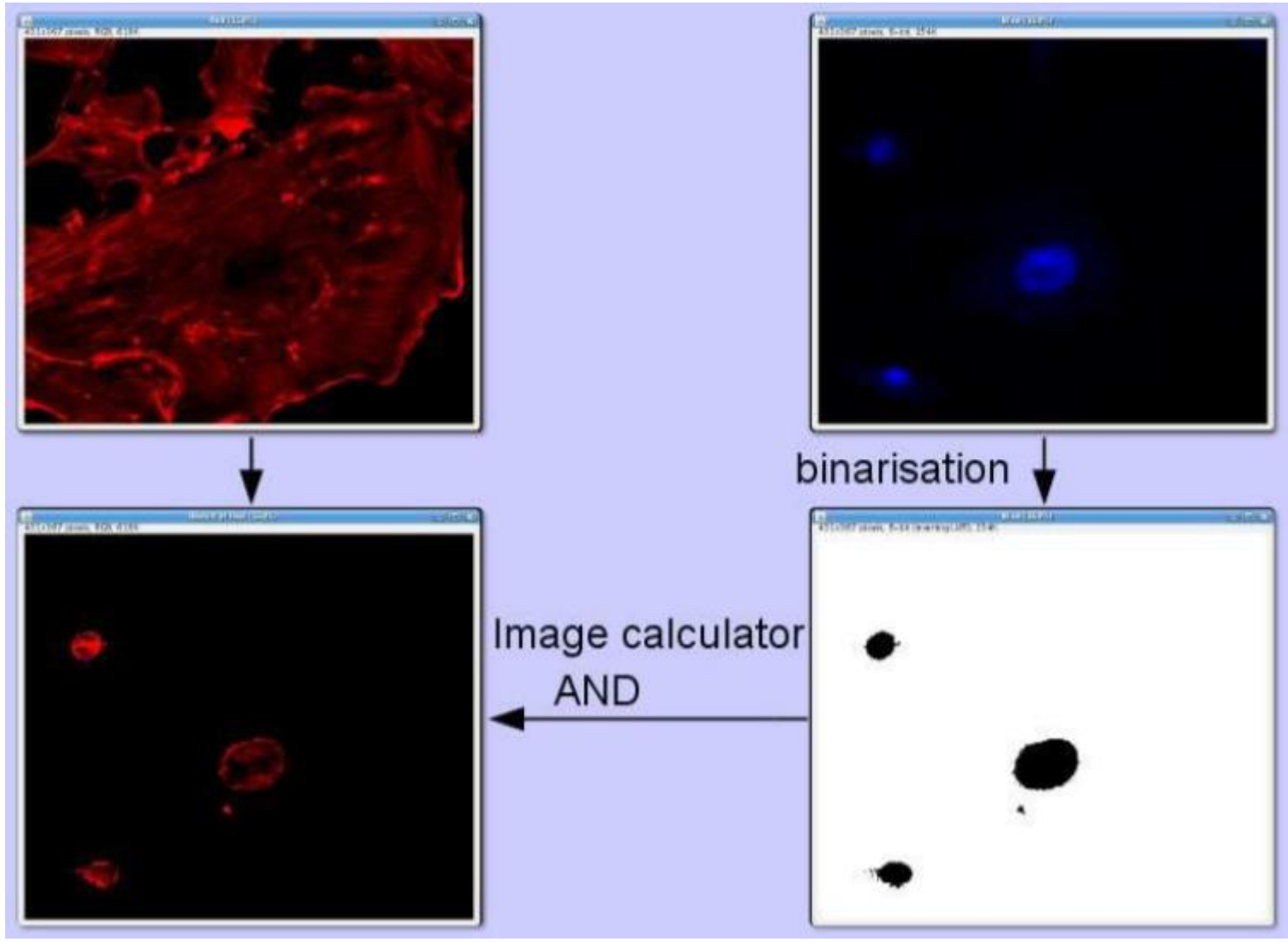
V/ Quantification: Maths on images

Source image (*img*1): 

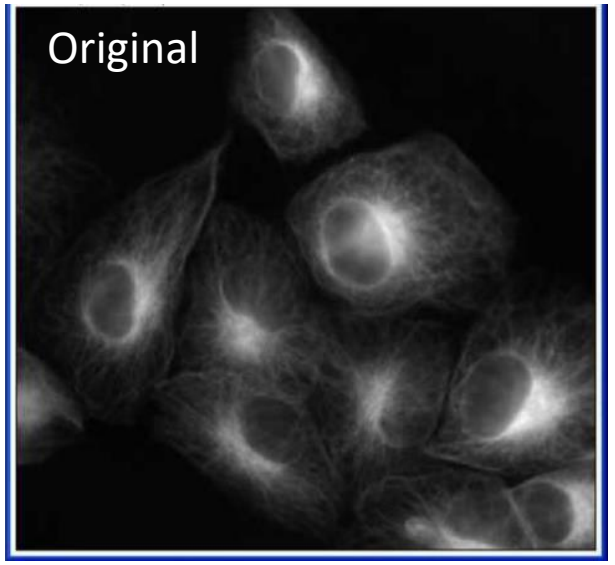
Destination image (*img*2): 

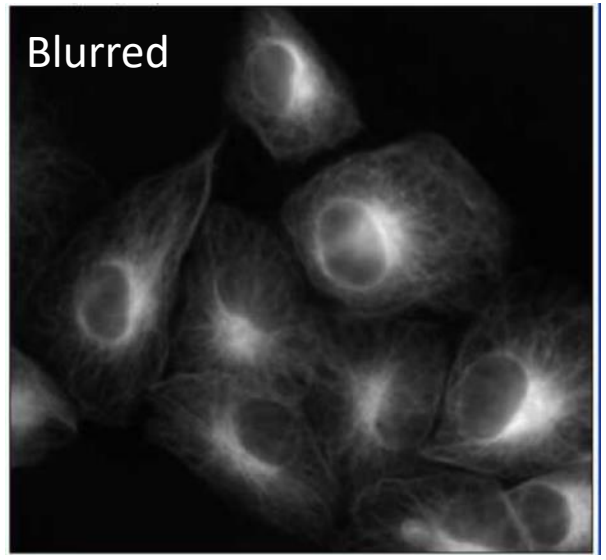
Operator	Result	Operator	Result
Add: $img1 = img1 + img2$		**Min**: $img1 = \min(img1, img2)$	
Subtract: $img1 = img1 - img2$		**Max**: $img1 = \max(img1, img2)$	
Multiply: $img1 = img1 \times img2$		**Average**: $img1 = (img1 + img2)/2$	
Divide: $img1 = img1 \div img2$		**Difference**: $img1 = \lvert img1 - img2 \rvert$	
AND: $img1 = img1 \wedge img2$		**Copy**: $img1 = img2$	
OR: $img1 = img1 \vee img2$		**Transparent--zero**	
XOR: $img1 = img1 \oplus img2$			

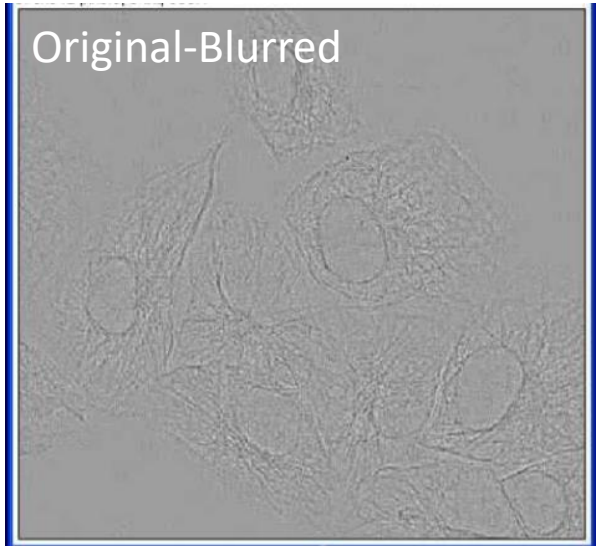
V/ Quantification: Maths: Masking images

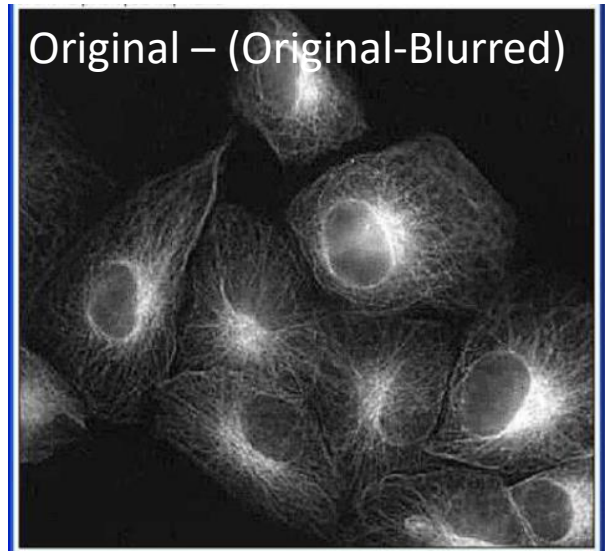


V/ Quantification: Maths: Sharpening an image

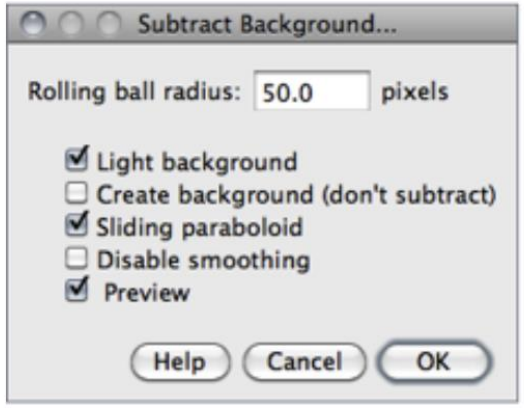


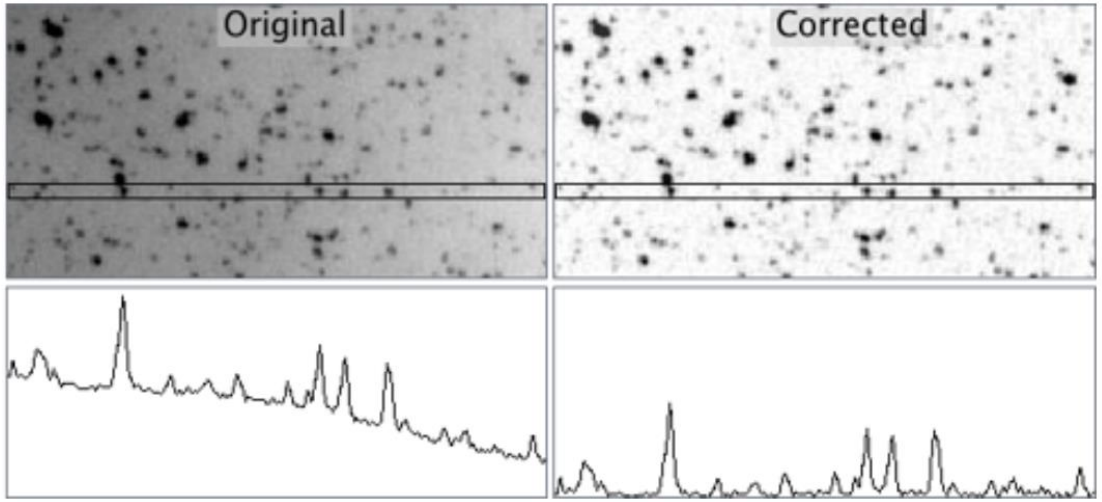


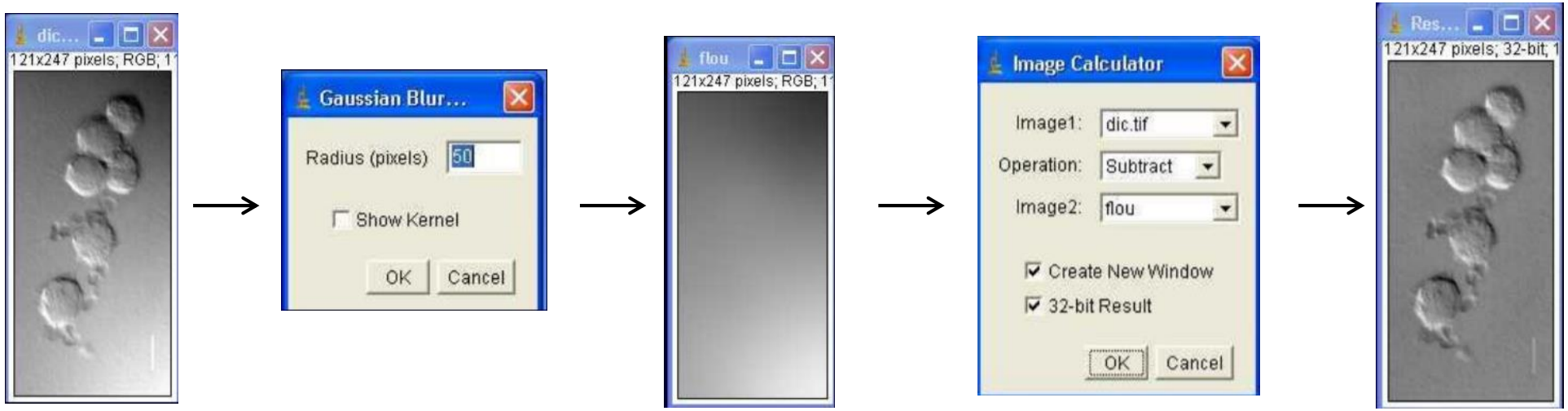




V/ Quantification: Maths: Subtract background

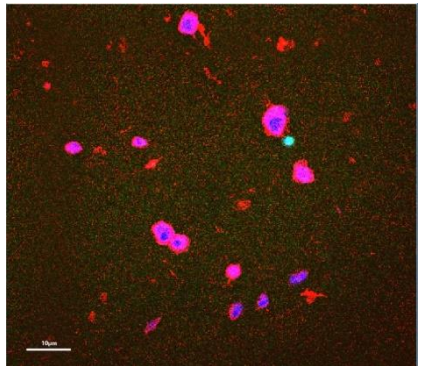


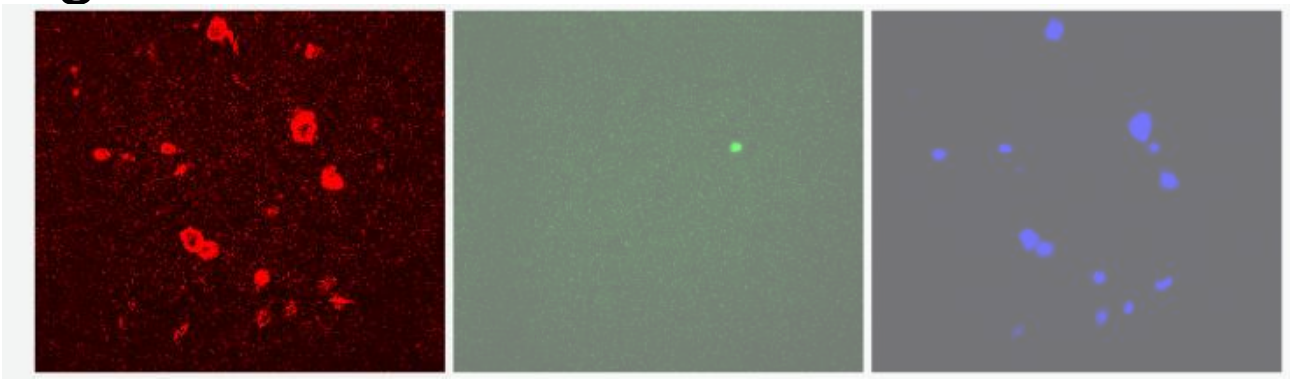


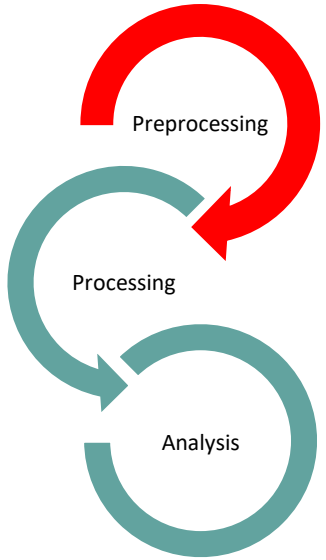


V/ Quantification: Typical segmentation workflow

Typical workflow: segment cells in the first channel

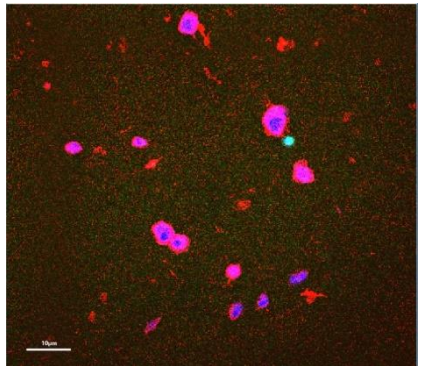


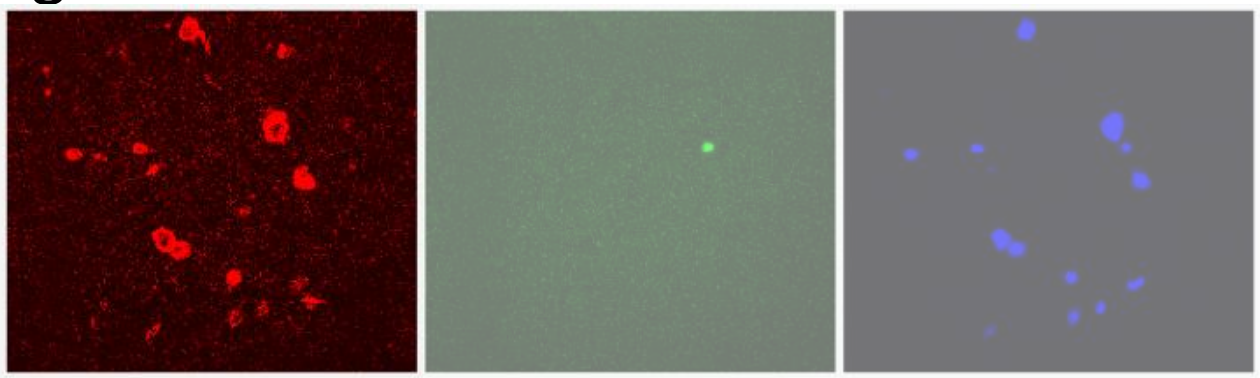


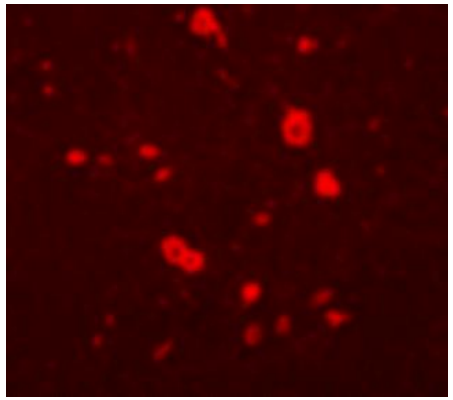


V/ Quantification: Typical segmentation workflow

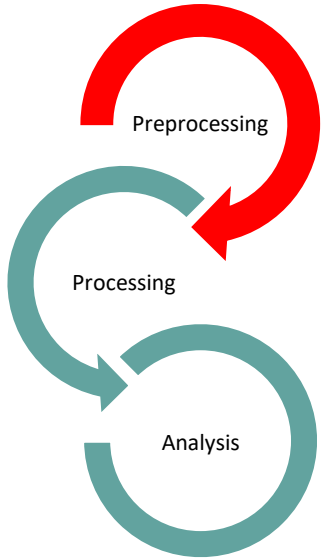
Typical workflow: segment cells in the first channel





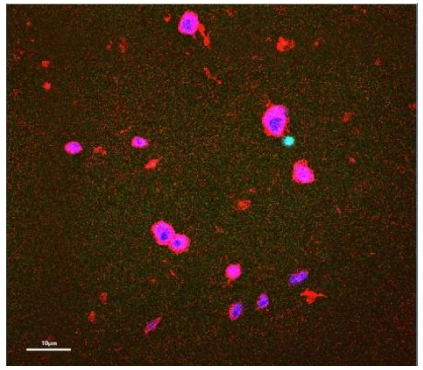


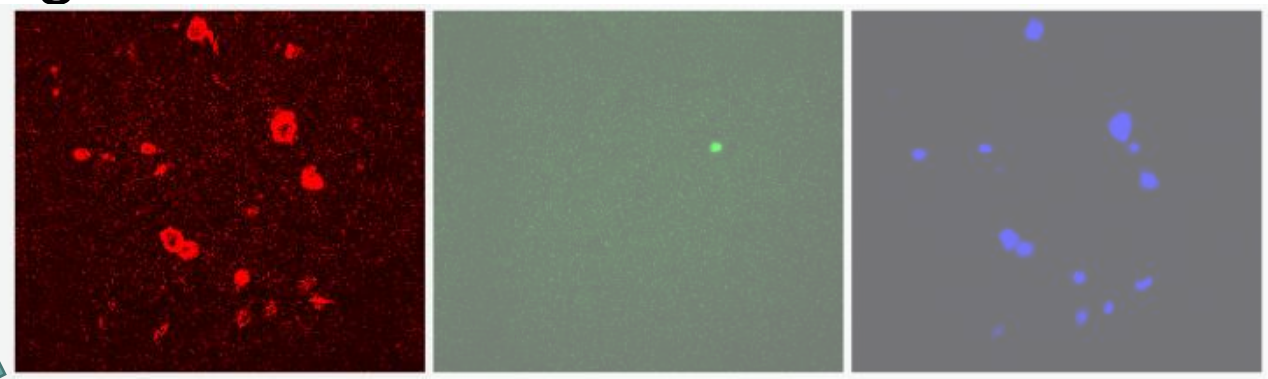
Smooth Image
(Gaussian Filter Sigma 3= convolution with gaussian kernel sigma 3)

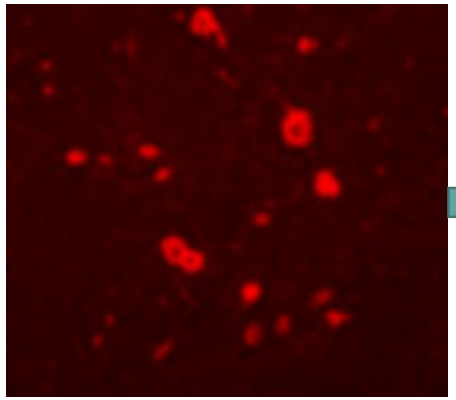


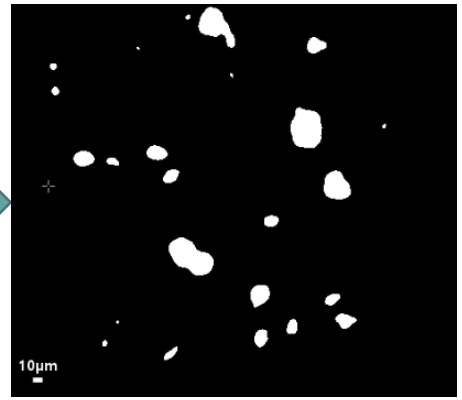
V/ Quantification: Typical segmentation workflow

Typical workflow: segment cells in the first channel



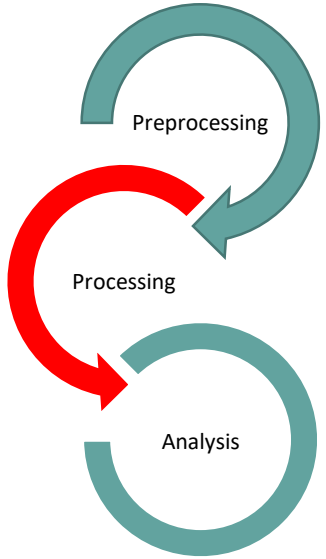






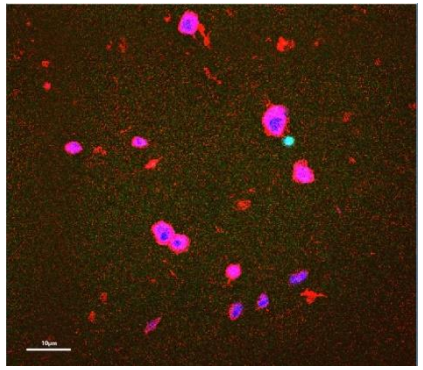
Smooth Image
(Gaussian Filter Sigma 3= convolution with gaussian kernel sigma 3)

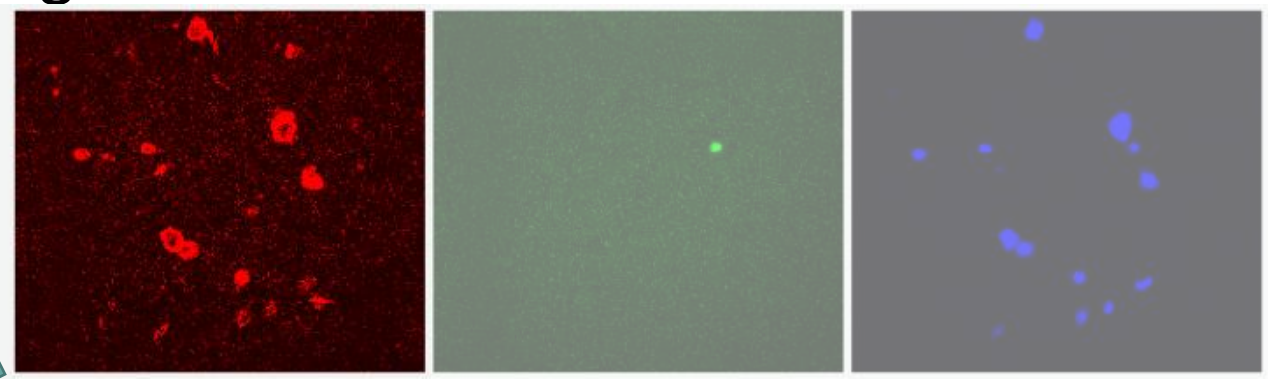
Select a threshold or threshold method
Ex: Otsu

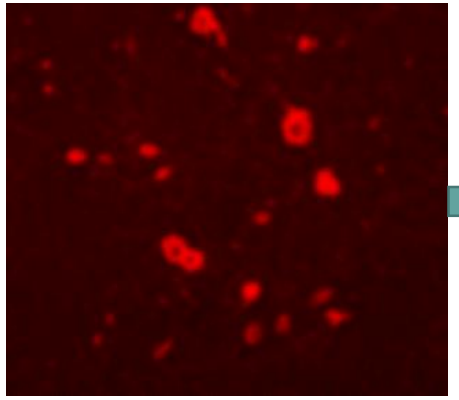


V/ Quantification: Typical segmentation workflow

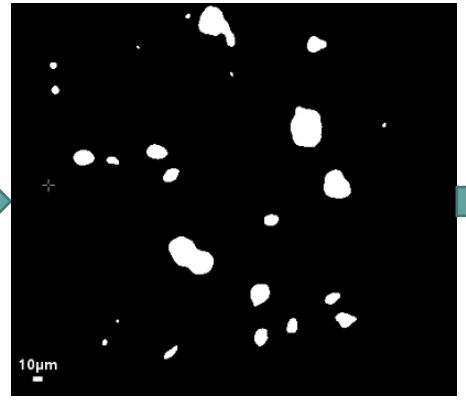
Typical workflow: segment cells in the first channel



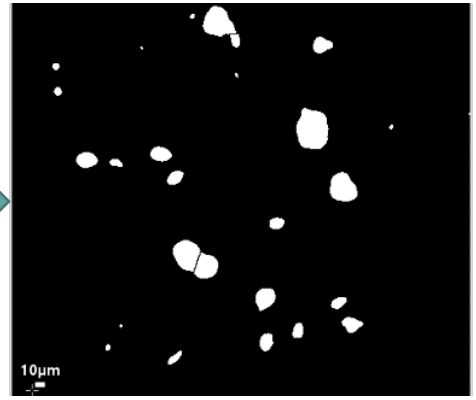




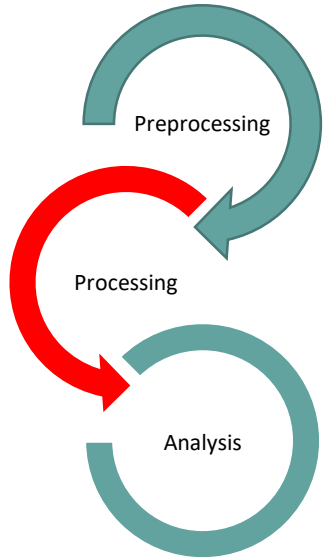
Smooth Image
(Gaussian Filter Sigma 3= convolution with gaussian kernel sigma 3)



Select a threshold or threshold method
Ex: Otsu

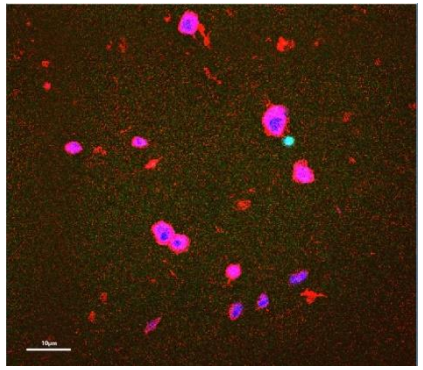


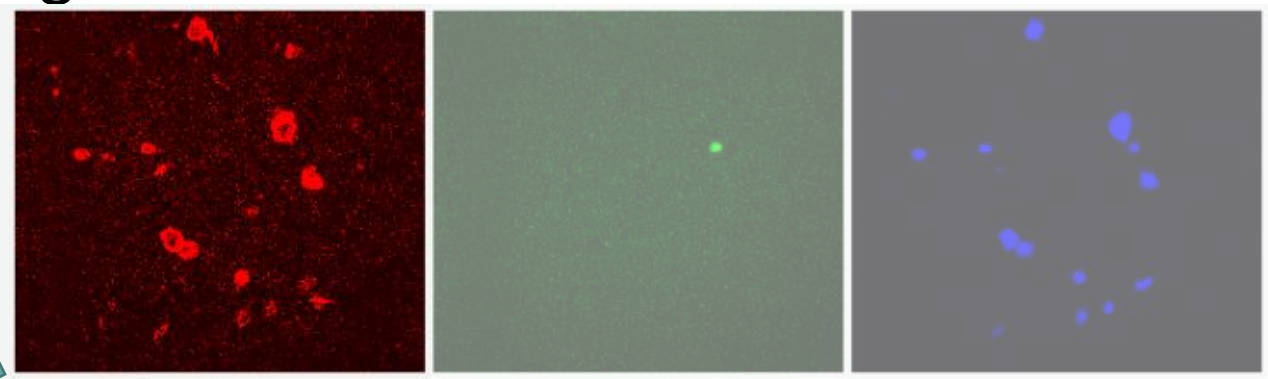
Separate touching objects: Watershed

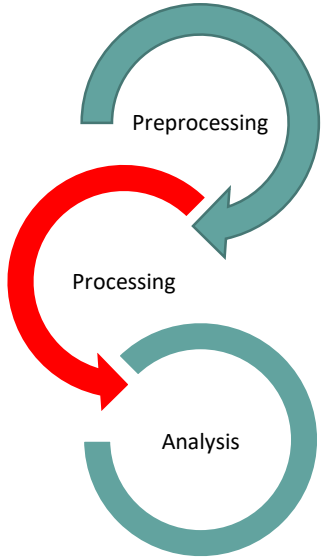


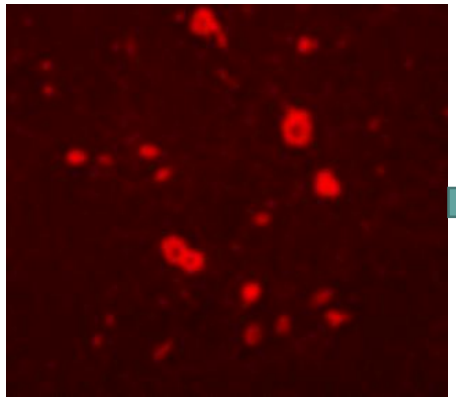
V/ Quantification: Typical segmentation workflow

Typical workflow: segment cells in the first channel

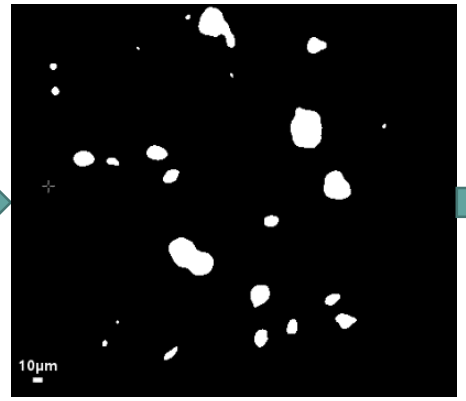




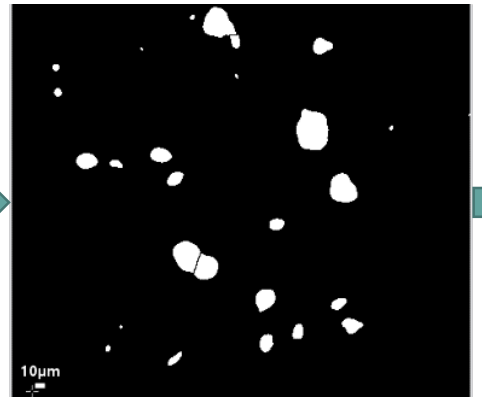




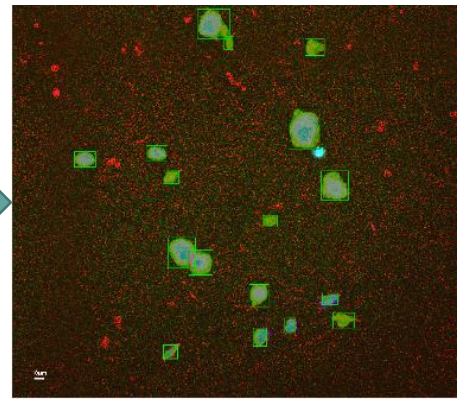
Smooth Image
(Gaussian Filter Sigma 3= convolution with gaussian kernel sigma 3)



Select a threshold or threshold method
Ex: Otsu



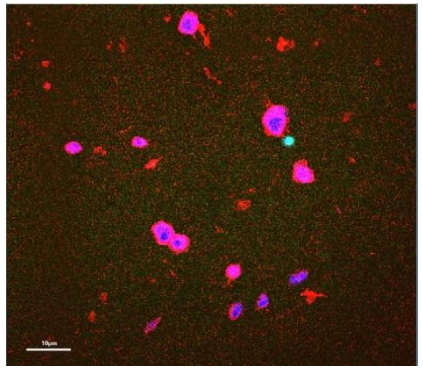
Separate touching objects: Watershed

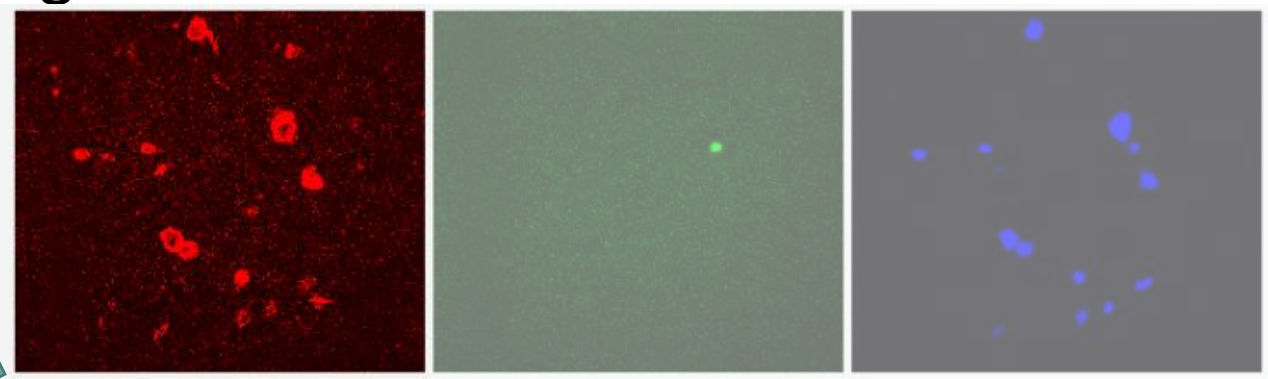


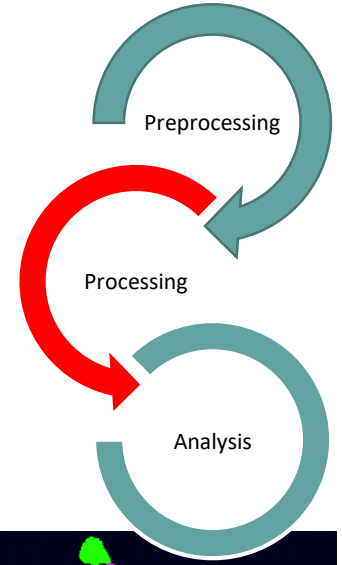
Remove small objects
Minimum size of object to keep: set threshold for size to 60 pixels

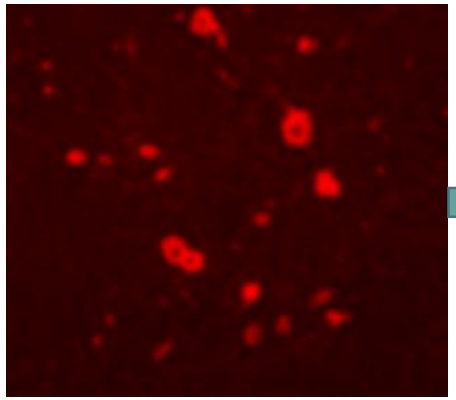
V/ Quantification: Typical segmentation workflow

Typical workflow: segment cells in the first channel

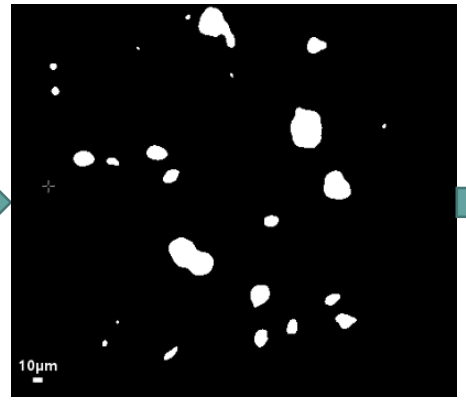




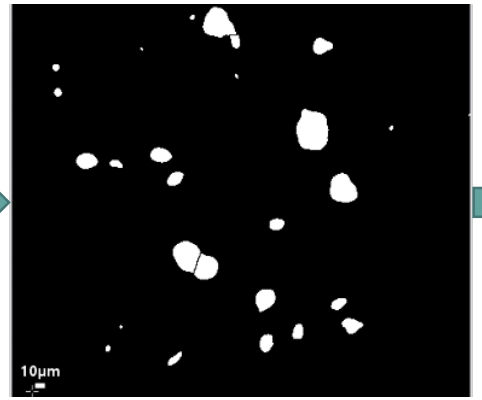




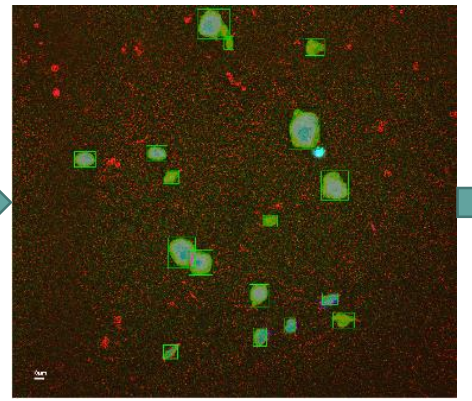
Smooth Image
(Gaussian Filter Sigma 3= convolution with gaussian kernel sigma 3)



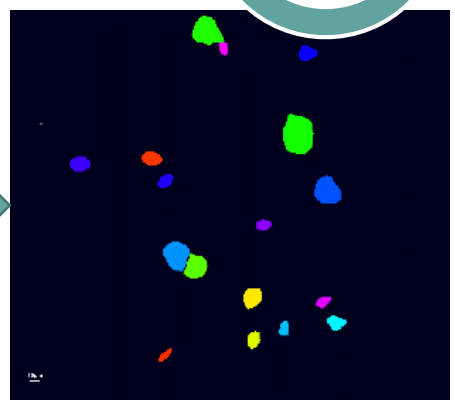
Select a threshold or threshold method
Ex: Otsu



Separate touching objects: Watershed



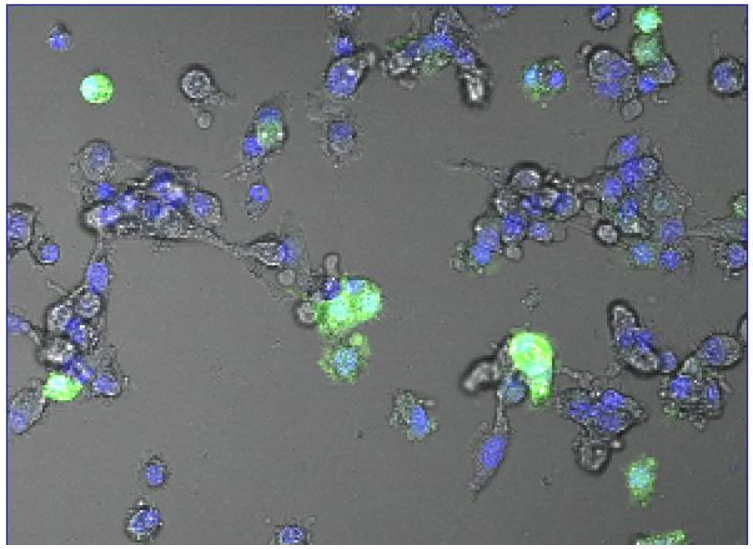
Remove small objects
Minimum size of object to keep: set threshold for size to 60 pixels

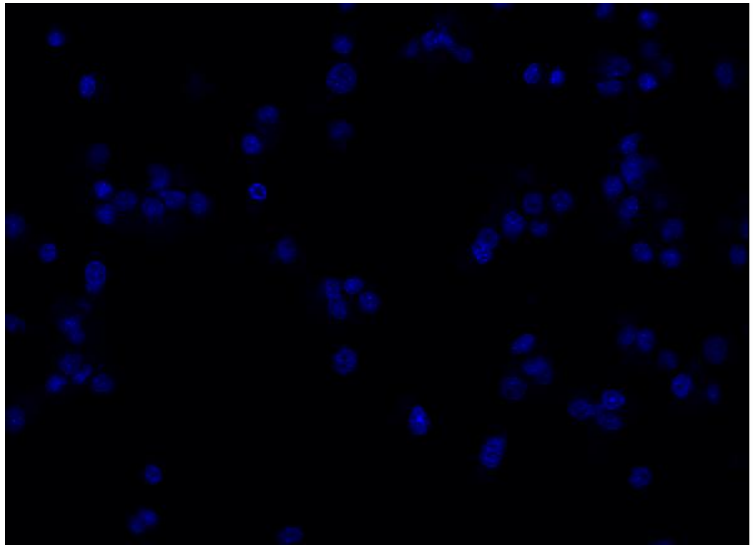


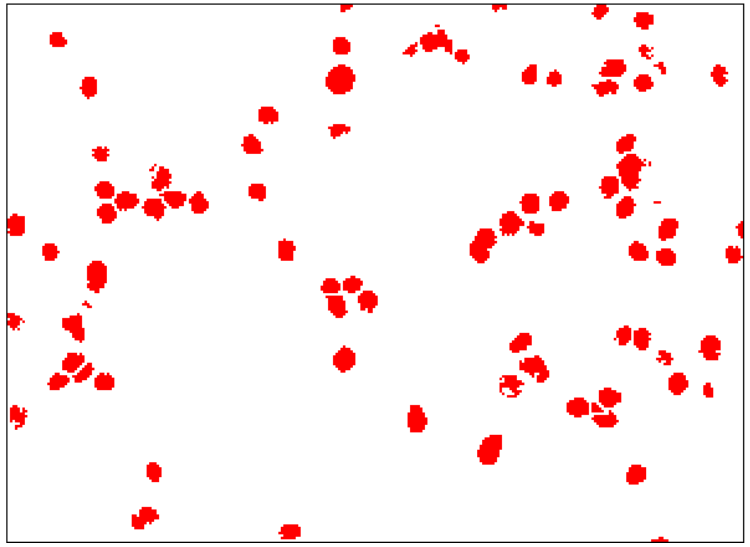
Label image

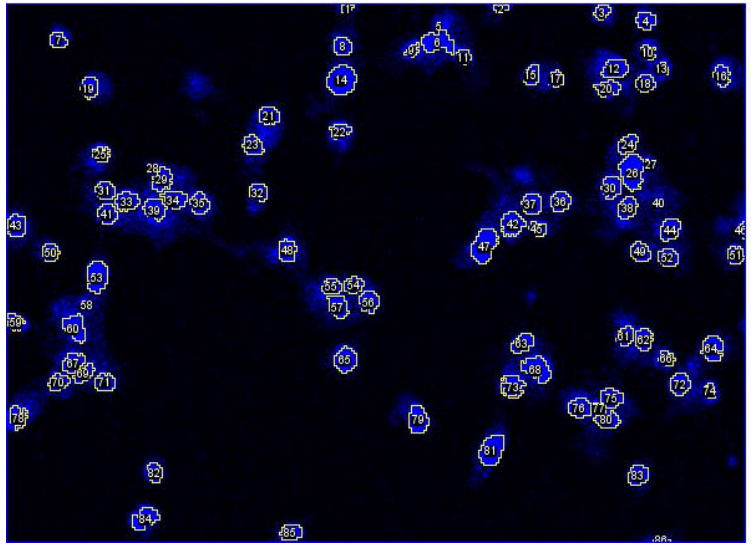
Hands-on 1: Count nuclei and determine which ones are green

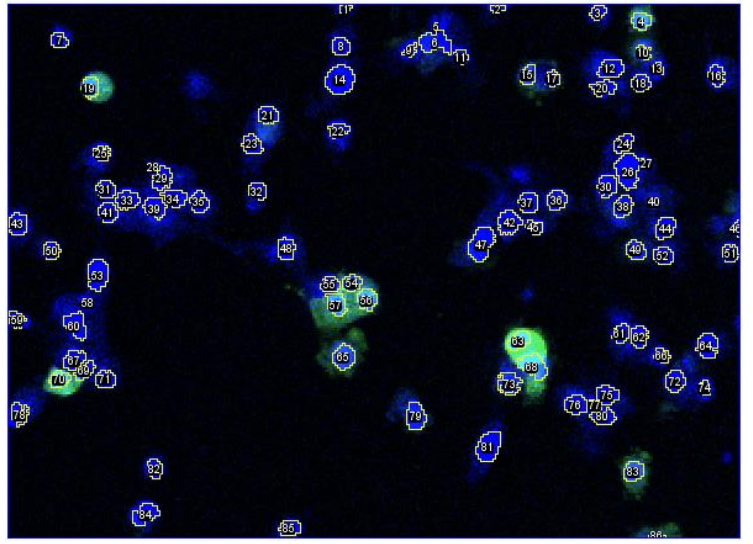
Hands-on 1: Count nuclei and determine which ones are green





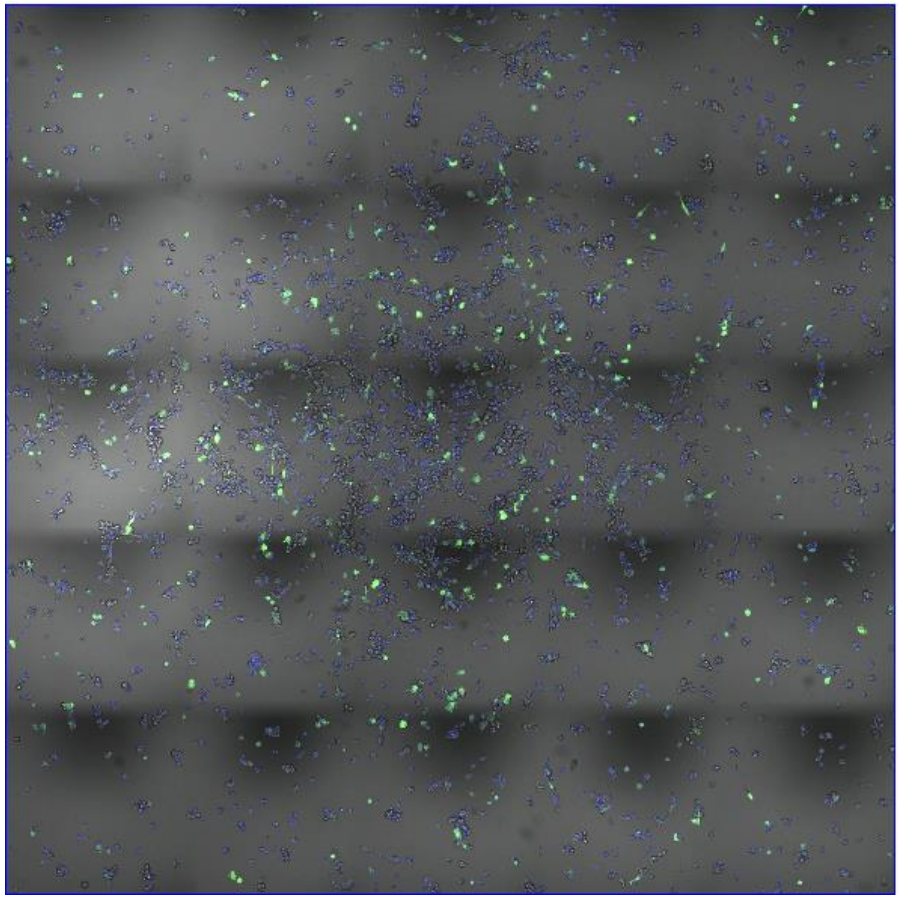






Hands-on 1: Count nuclei and determine which ones are green

Image: Melano-1.tif



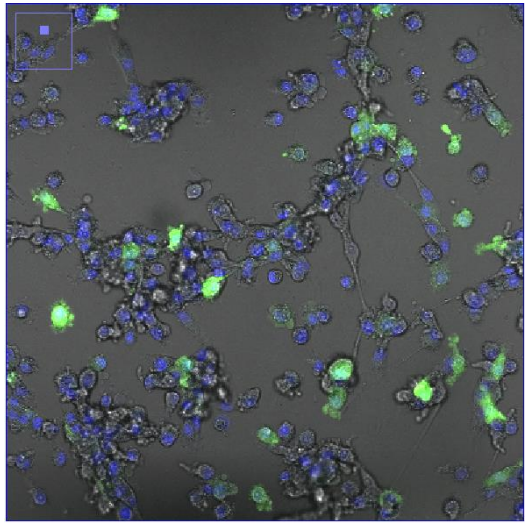
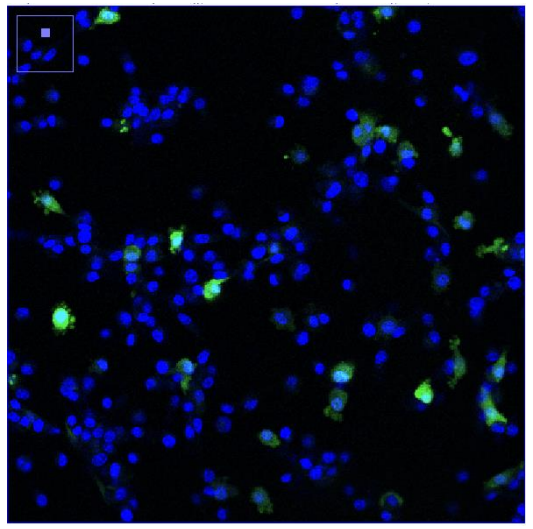


Image › Color › Channels Tool...

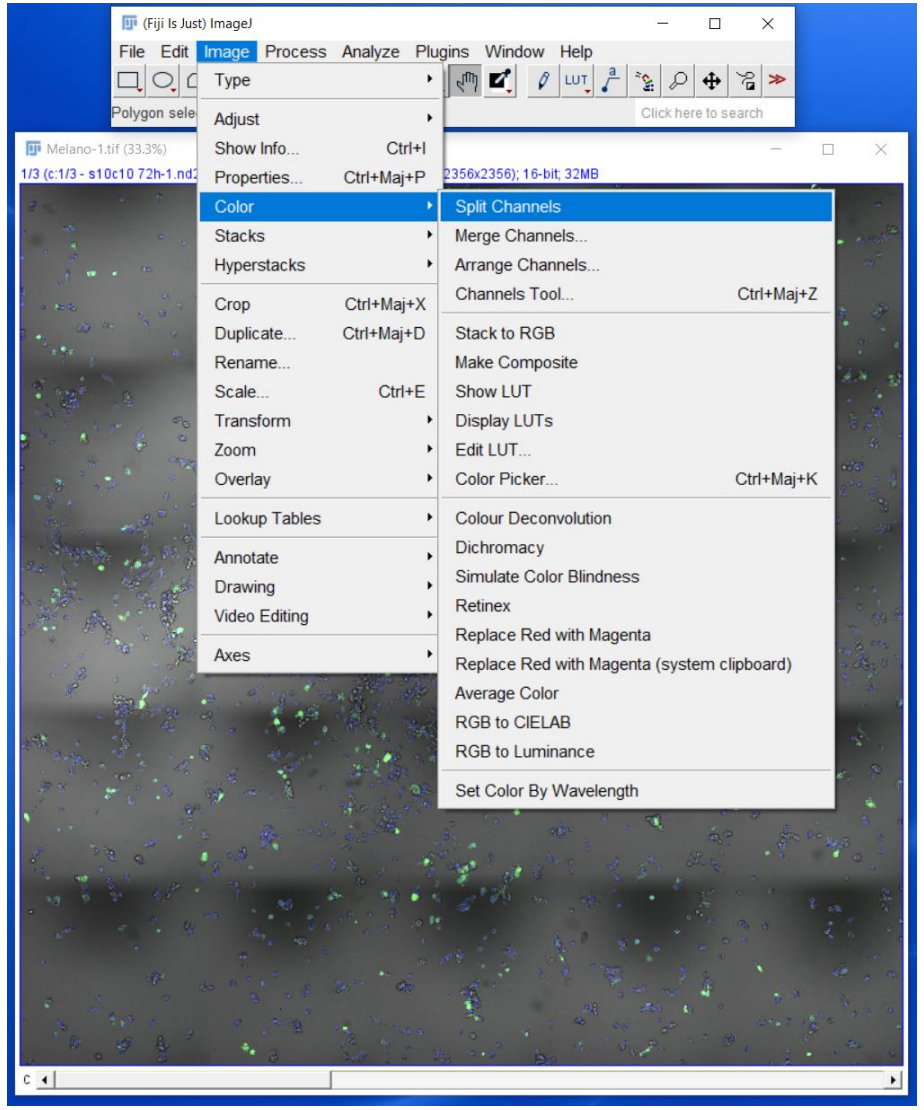


Hands-on 1: Count nuclei and determine which ones are green

Image: Melano-1.tif

Step 1 Split Channels

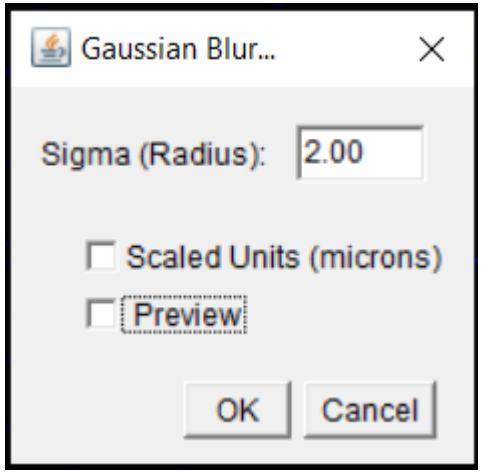
Step 2 Close C3

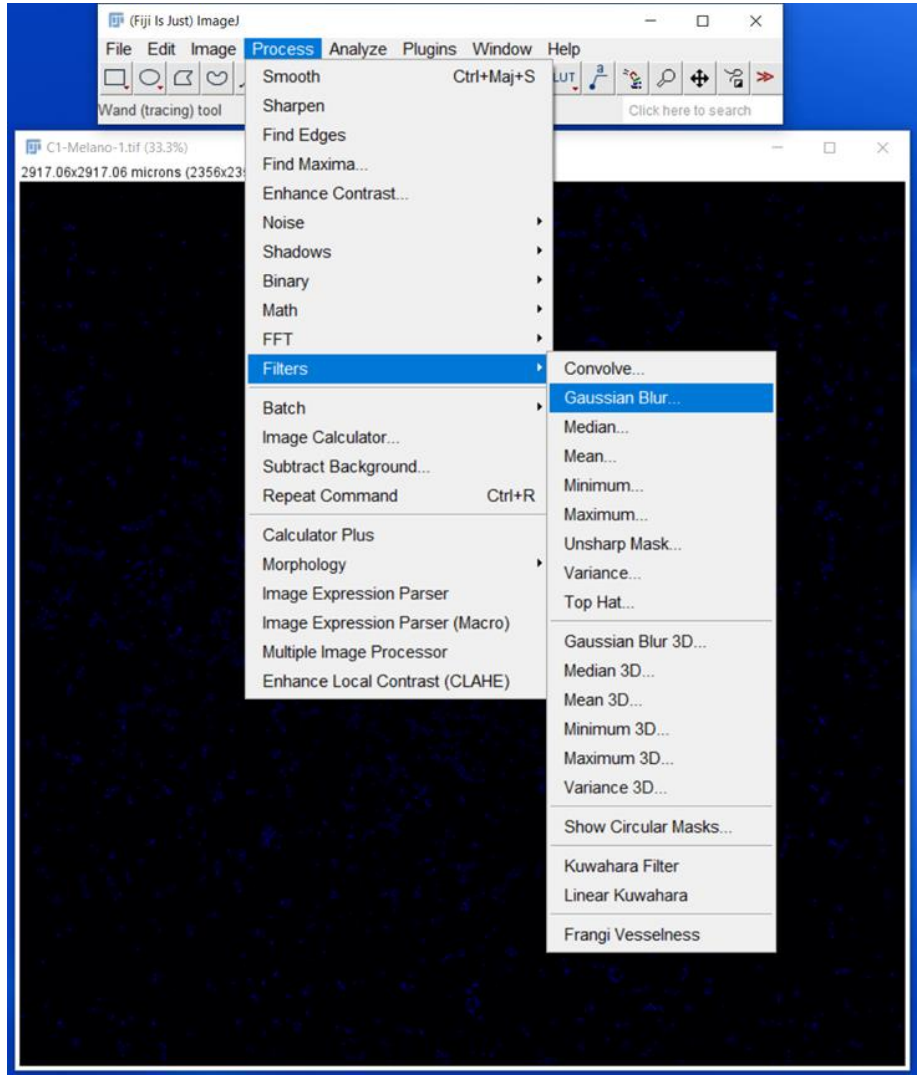


Hands-on 1: Count nuclei and determine which ones are green

Image: Melano-1.tif

Step 3 On C1, Gaussian filter

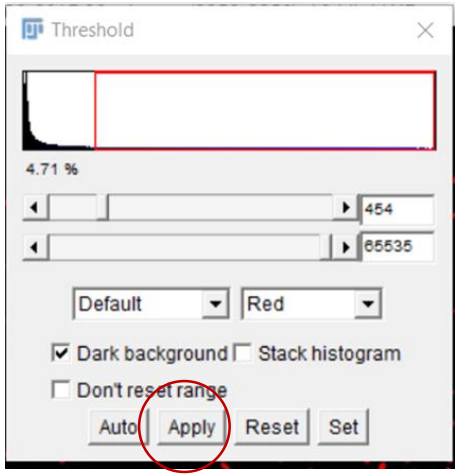


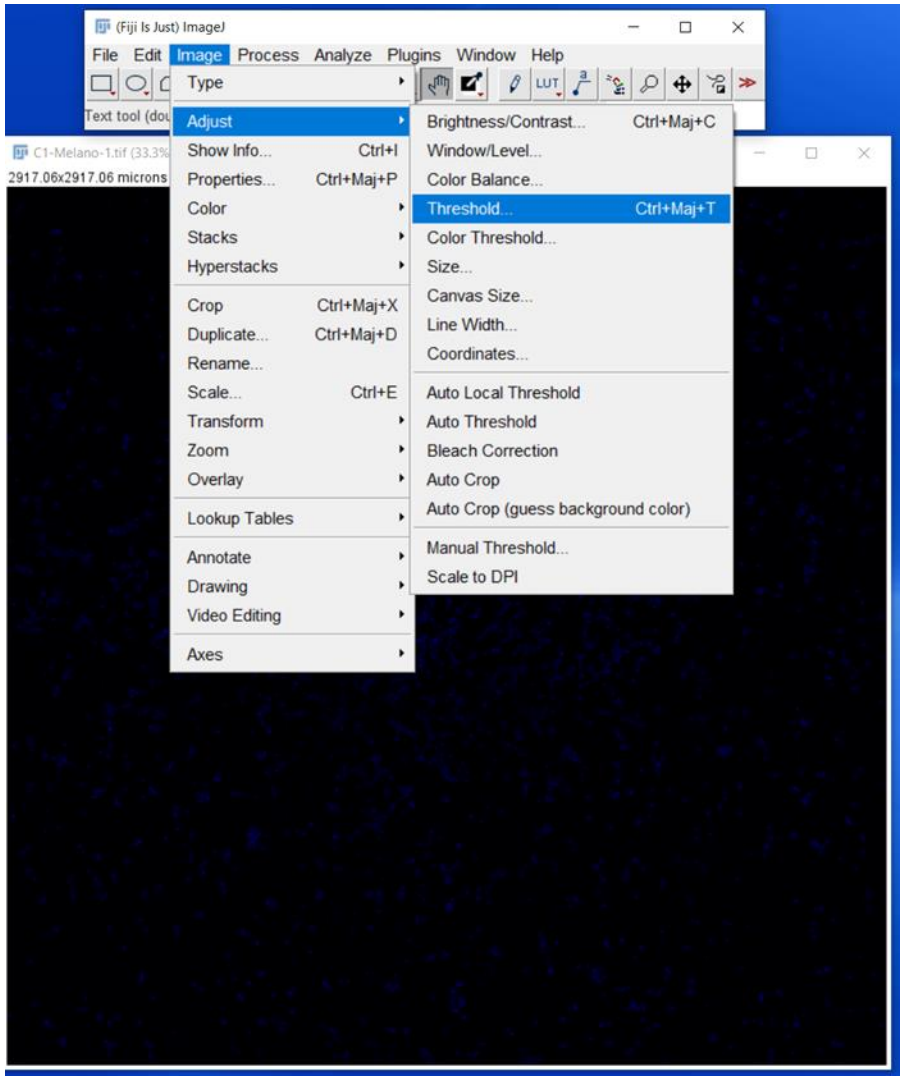


Hands-on 1: Count nuclei and determine which ones are green

Image: Melano-1.tif

Step 4 Thresholding



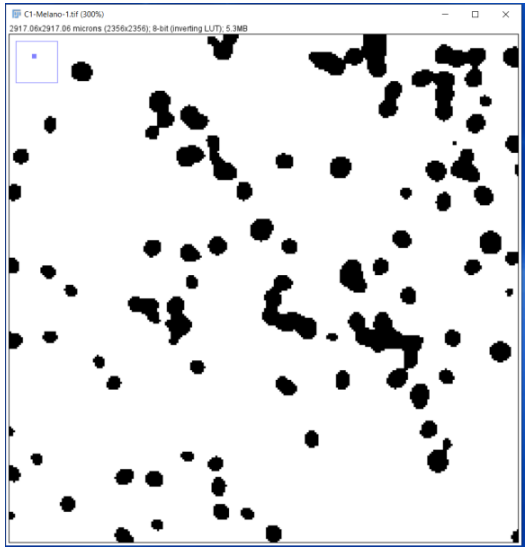


Hands-on 1: Count nuclei and determine which ones are green

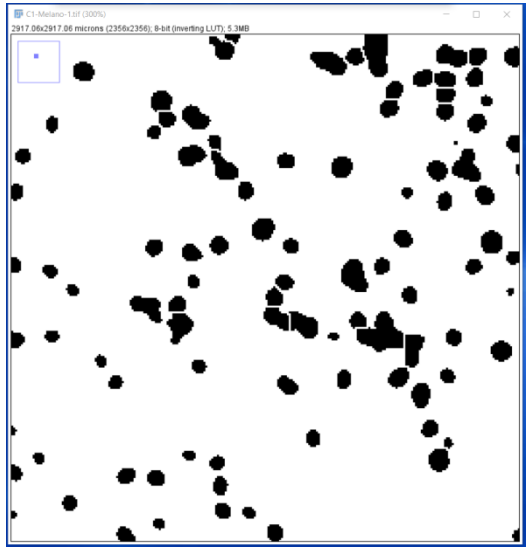
Image: Melano-1.tif

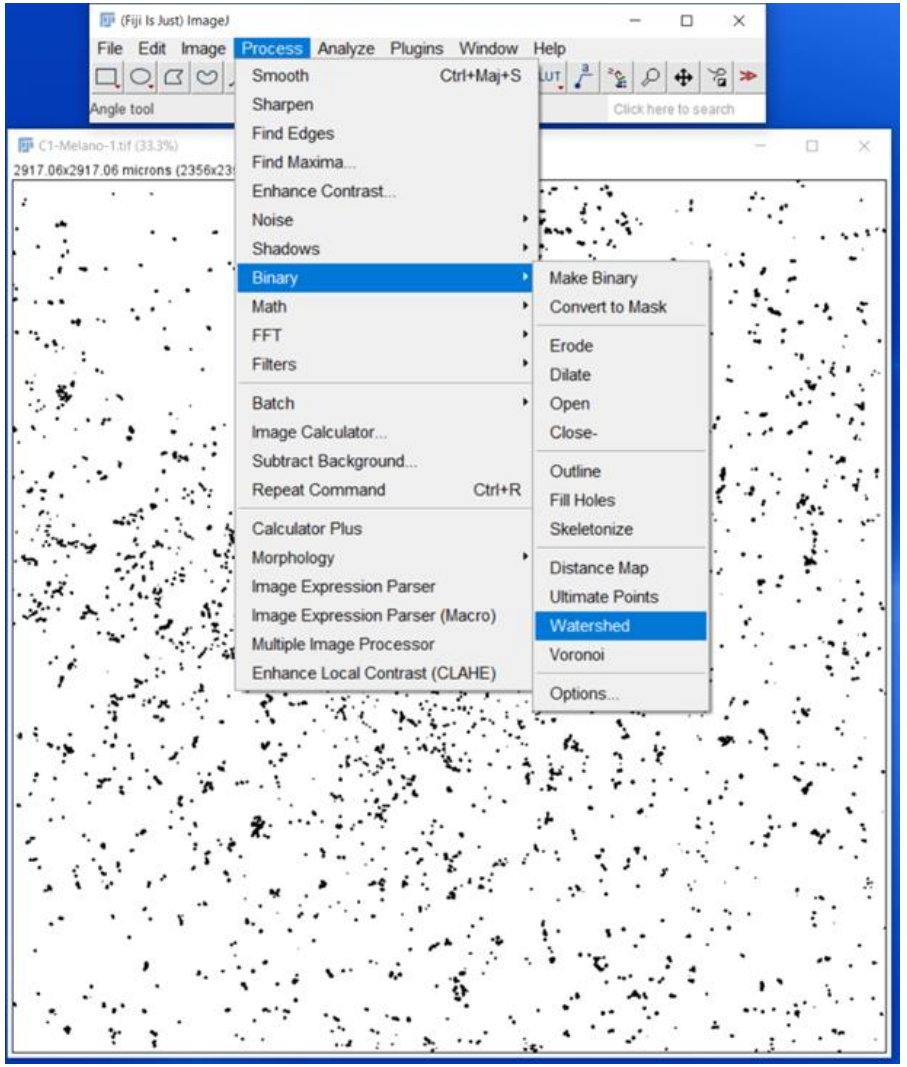
Step 5 Watershed

Before



After

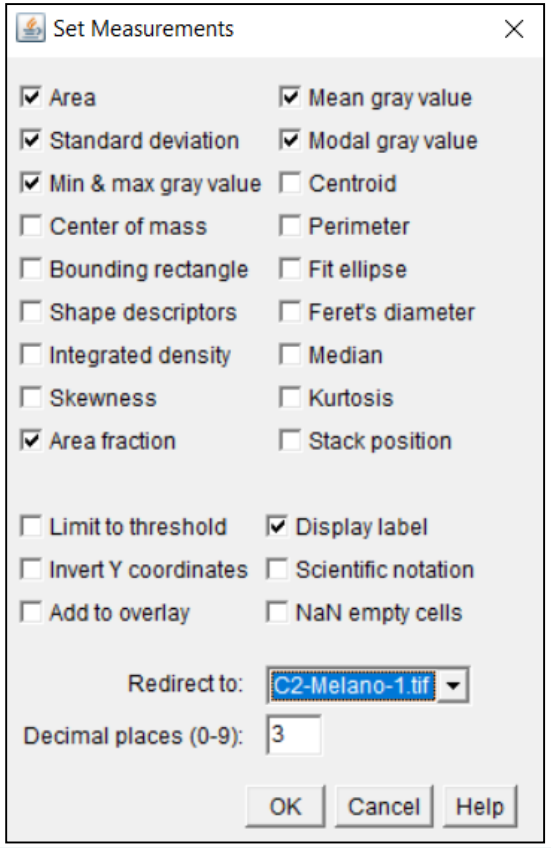


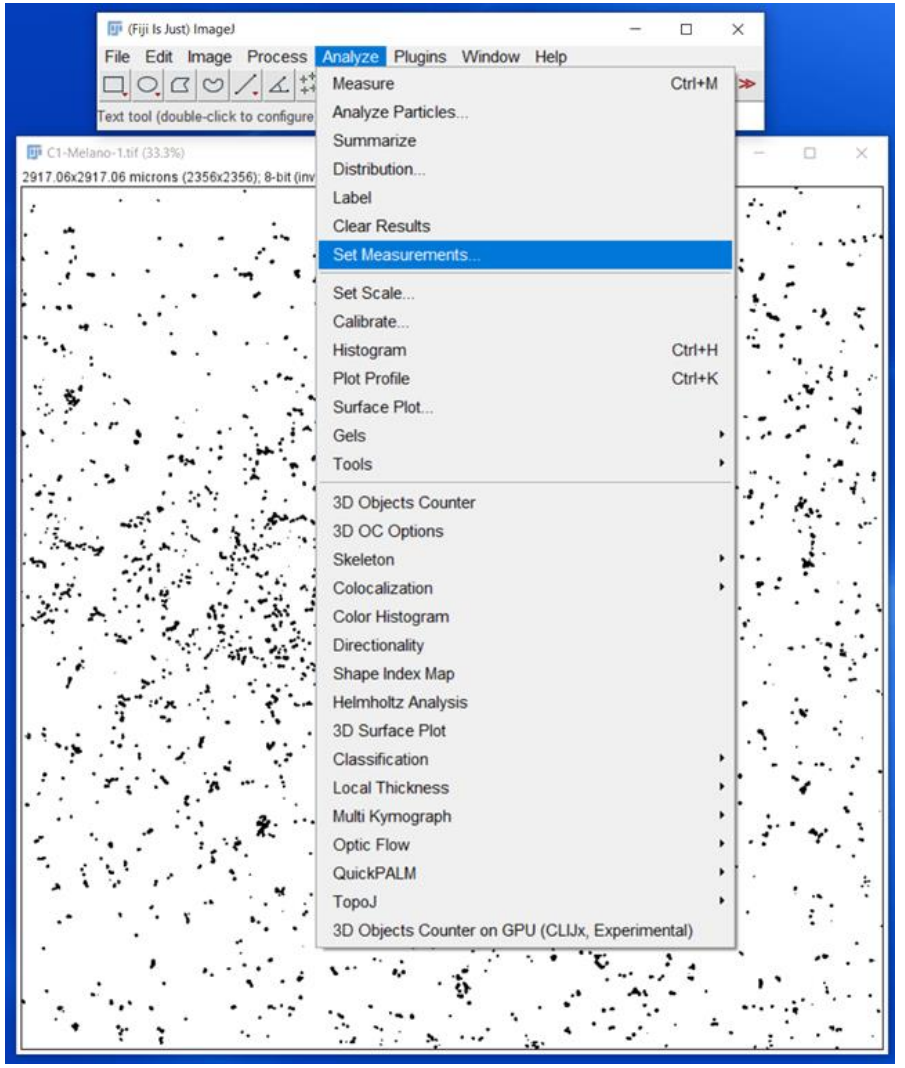


Hands-on 1: Count nuclei and determine which ones are green

Image: Melano-1.tif

Step 6 Set Measurement options, redirect to C2

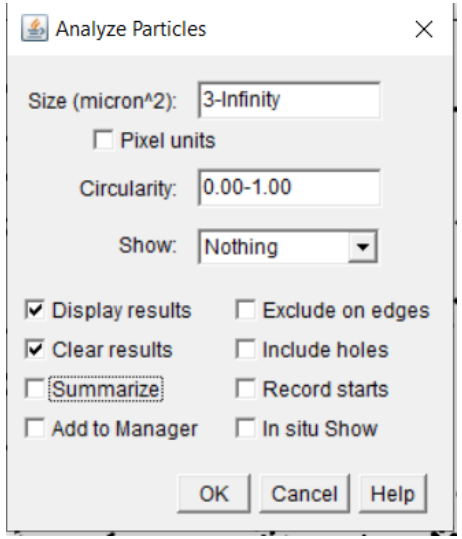


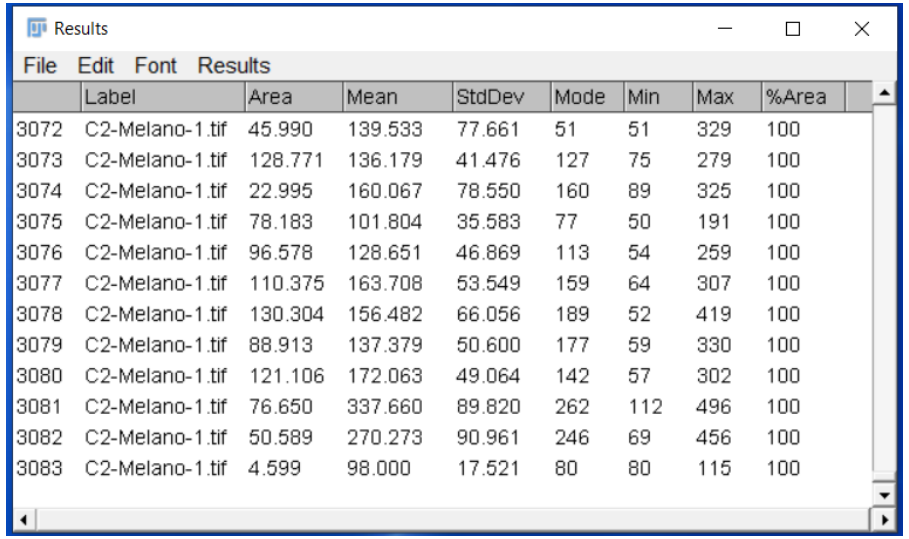


Hands-on 1: Count nuclei and determine which ones are green

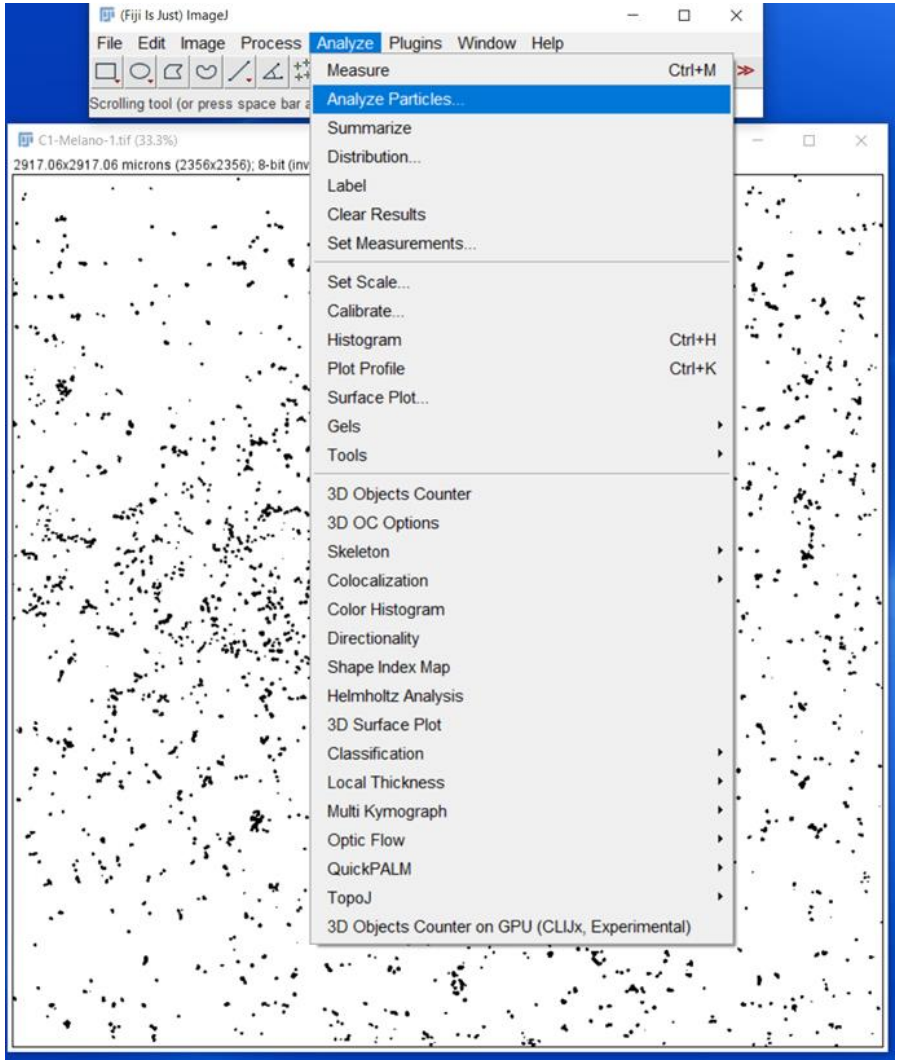
Image: Melano-1.tif

Step 7 — Analyze Particles





	Label	Area	Mean	StdDev	Mode	Min	Max	%Area
3072	C2-Melano-1.tif	45.990	139.533	77.661	51	51	329	100
3073	C2-Melano-1.tif	128.771	136.179	41.476	127	75	279	100
3074	C2-Melano-1.tif	22.995	160.067	78.550	160	89	325	100
3075	C2-Melano-1.tif	78.183	101.804	35.583	77	50	191	100
3076	C2-Melano-1.tif	96.578	128.651	46.869	113	54	259	100
3077	C2-Melano-1.tif	110.375	163.708	53.549	159	64	307	100
3078	C2-Melano-1.tif	130.304	156.482	66.056	189	52	419	100
3079	C2-Melano-1.tif	88.913	137.379	50.600	177	59	330	100
3080	C2-Melano-1.tif	121.106	172.063	49.064	142	57	302	100
3081	C2-Melano-1.tif	76.650	337.660	89.820	262	112	496	100
3082	C2-Melano-1.tif	50.589	270.273	90.961	246	69	456	100
3083	C2-Melano-1.tif	4.599	98.000	17.521	80	80	115	100



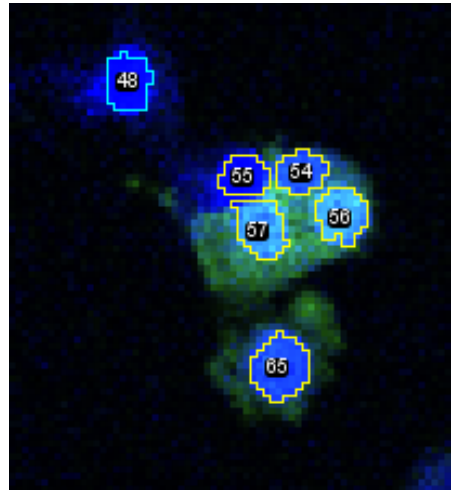
Hands-on 1: Count nuclei and determine which ones are green

Optional: play with the ROIs

Results

File Edit Font Results

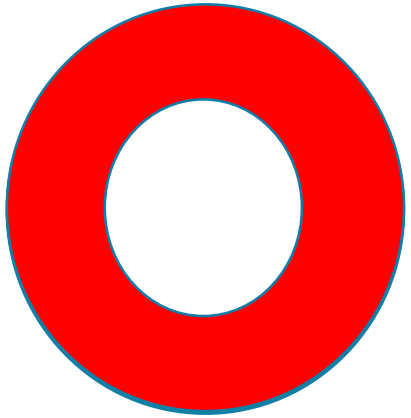
	Label	Mean
1	Melano-1.tif:c:2/3 - s10c10 72h-1.nd2 (series 1)	176.37780
2	Melano-1.tif:0065-0142:c:2/3 - s10c10 72h-1.nd2 (series 1)	1278.13043
3	Melano-1.tif:0048-0098:c:2/3 - s10c10 72h-1.nd2 (series 1)	160.32000

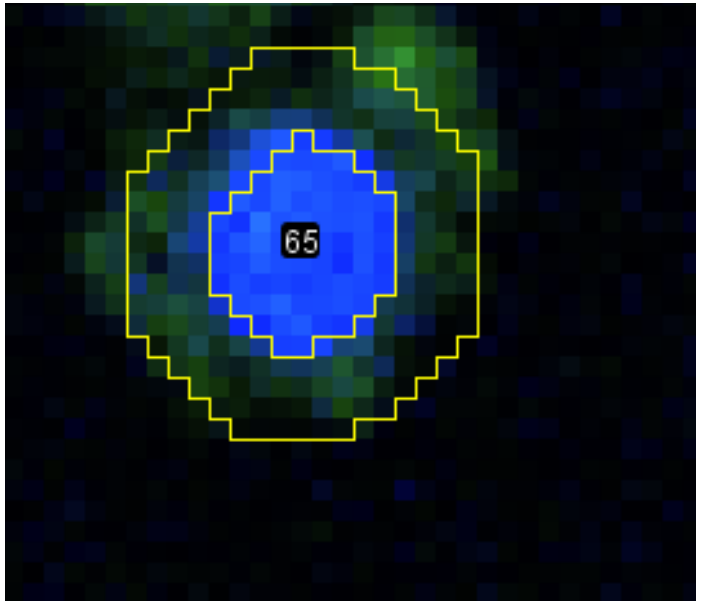


Tools to play with ROIs:
Select Original ROI

Edit > Selection > Enlarge: +xx dilates the ROI, -xx shrinks it

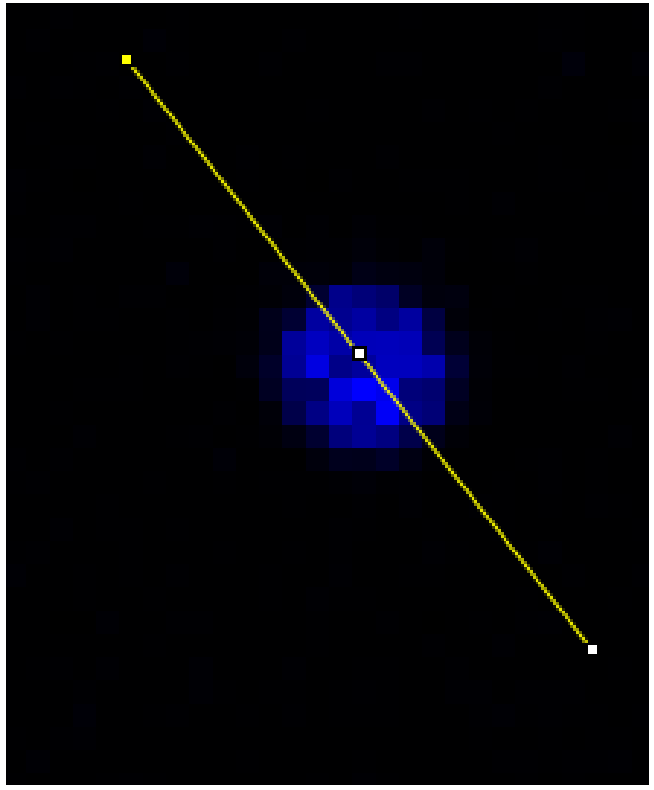
Edit > Selection > Make Band : we will measure only in the band



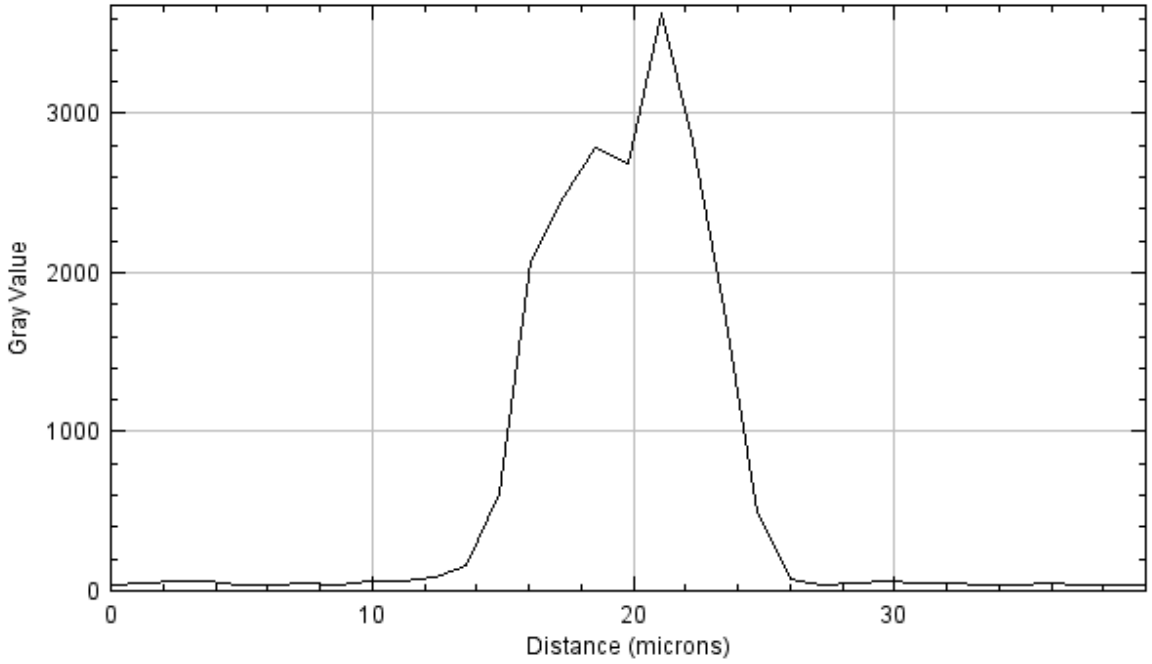


V/ Quantification: Plot profile

Draw a line

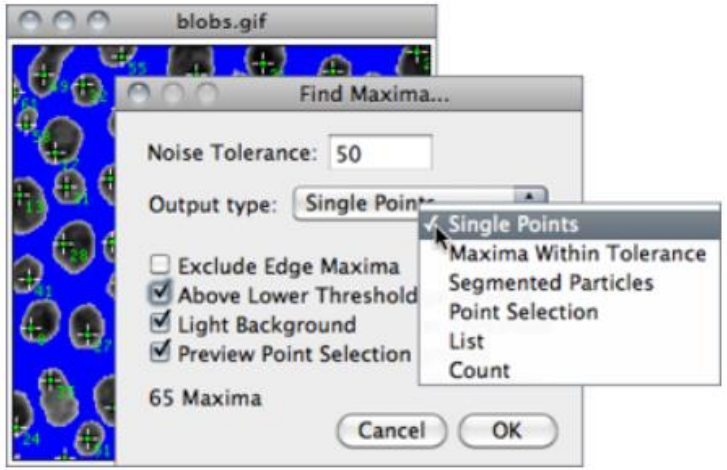


Analyze > Plot Profile (**k**)

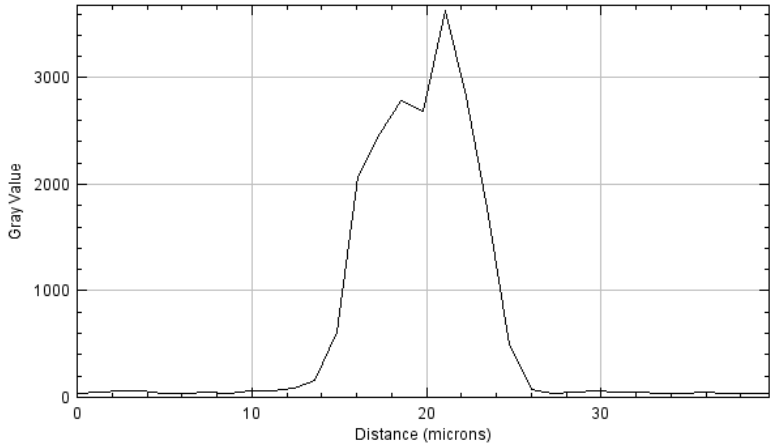


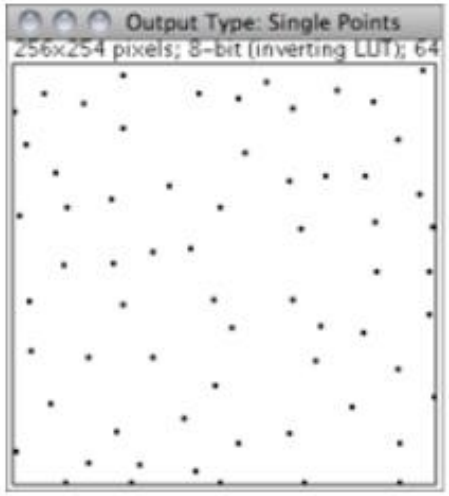
V/ Quantification: Find maxima

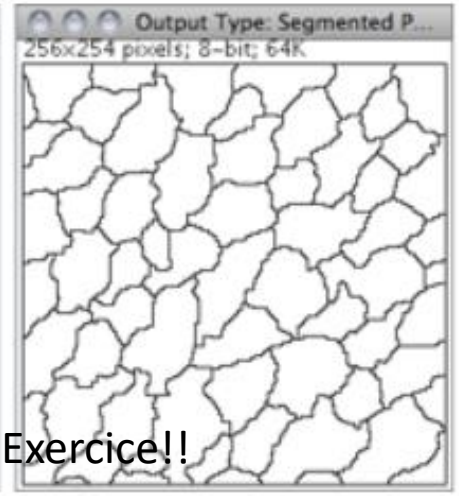
Process > Find Maxima…



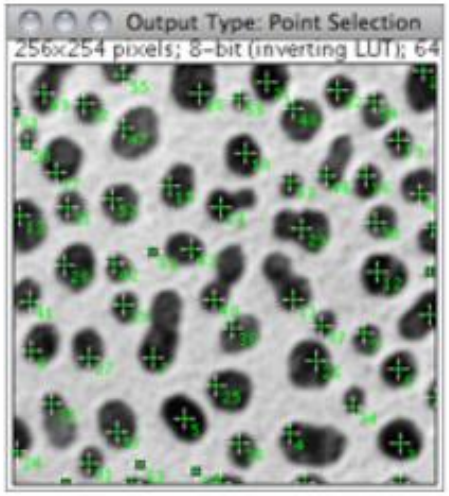
… detects the bumps!

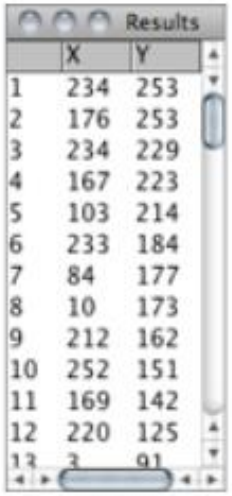


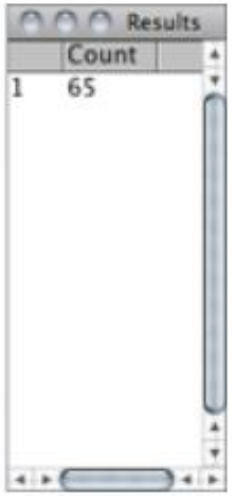




Exercice!!





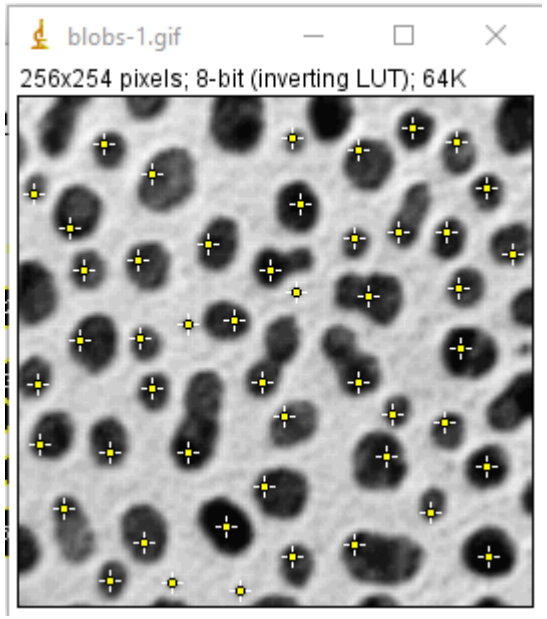


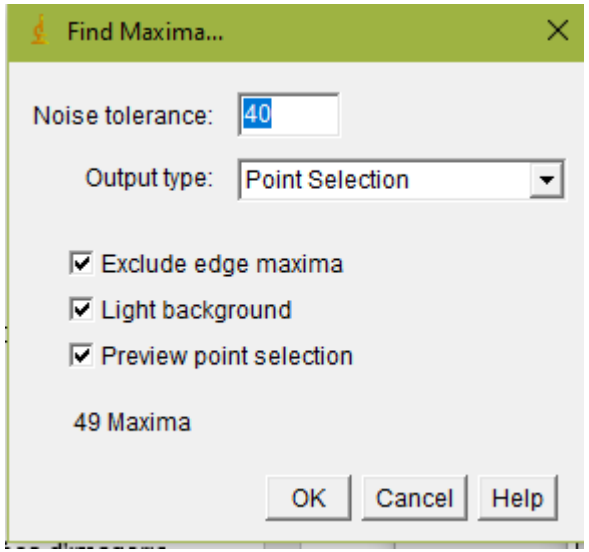
V/ Quantification: Find maxima: *Hands-on!*

Exercice 2a

Step 1 Open the blobs

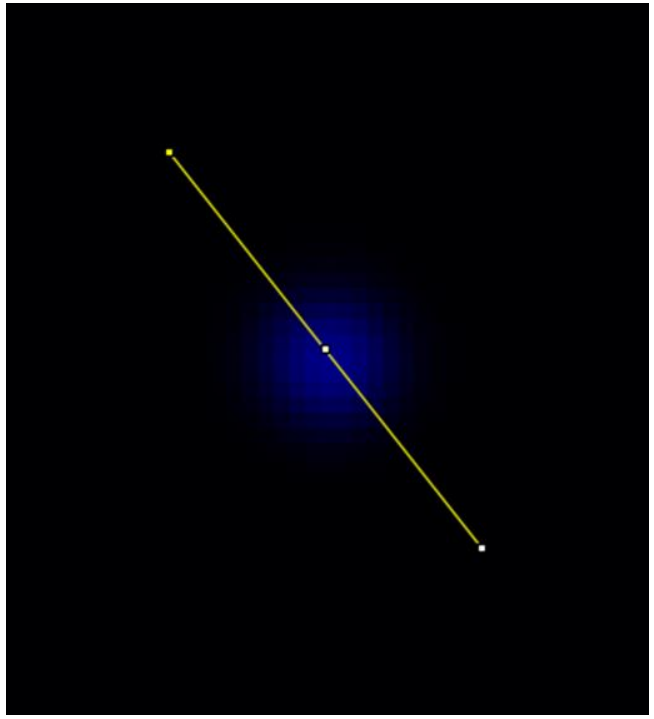
Step 2 Process > Find Maxima… Try different parameters!



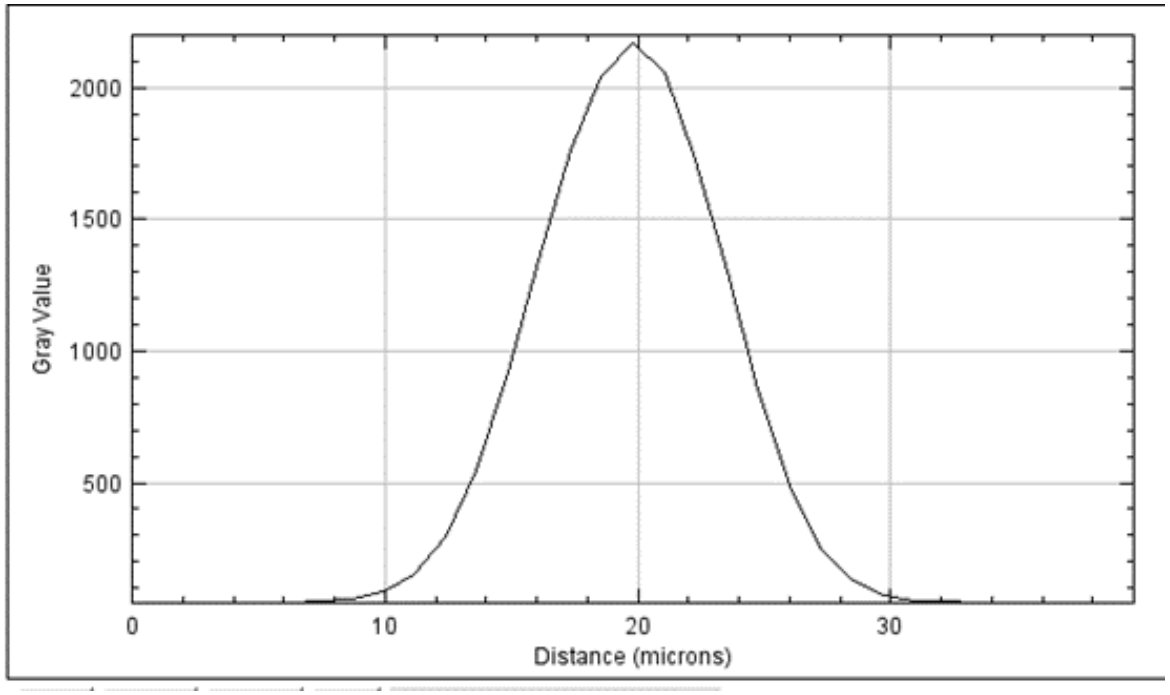


V/ Quantification: Find Maxima

Even more efficient
after a gaussian blur

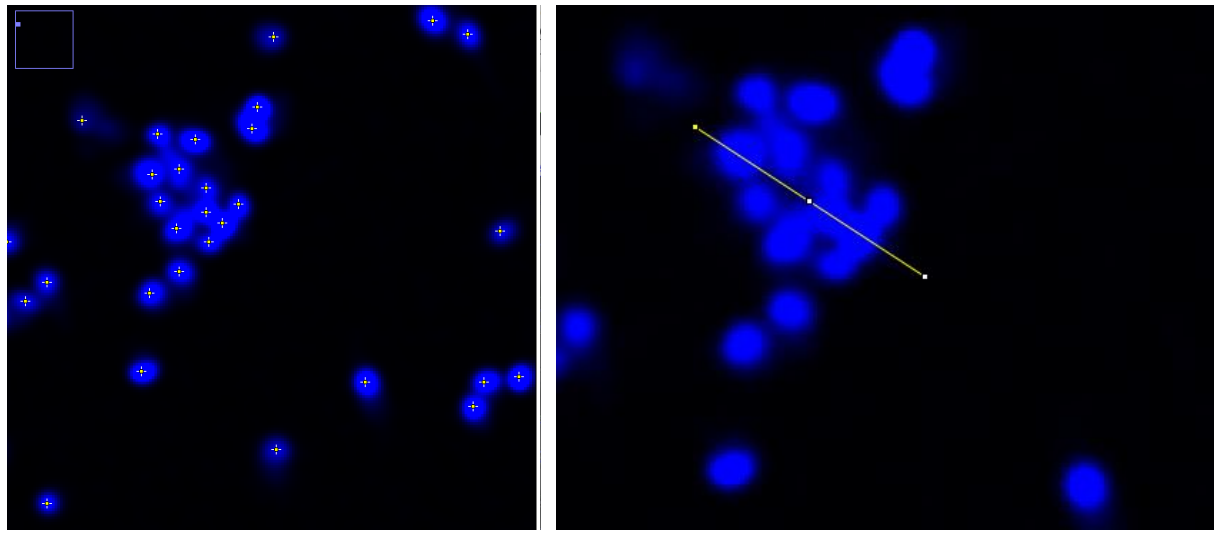


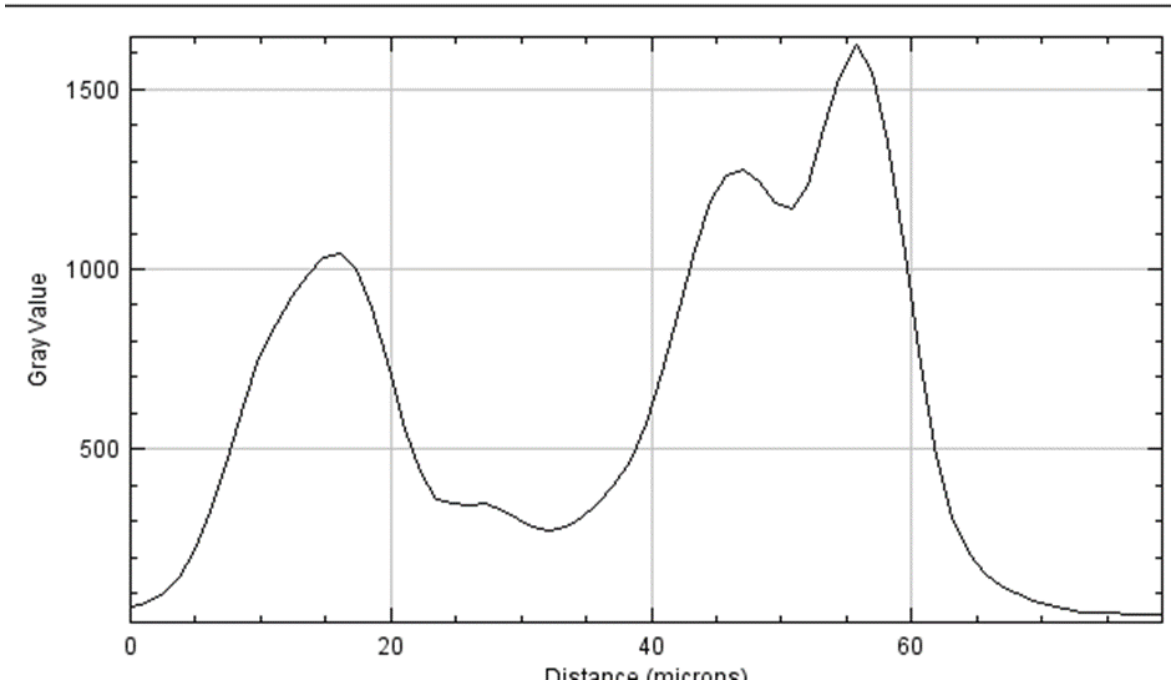
Analyze > Plot Profile (**k**)



V/ Quantification: Find Maxima

Even more efficient
after a gaussian blur



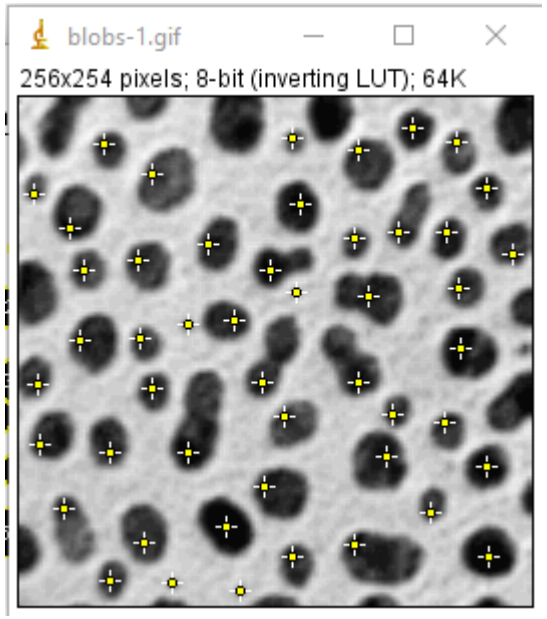


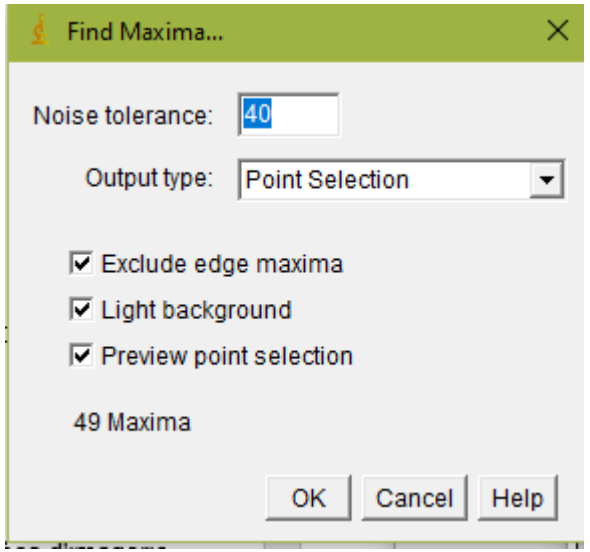
V/ Quantification: Find maxima: *Hands-on!*

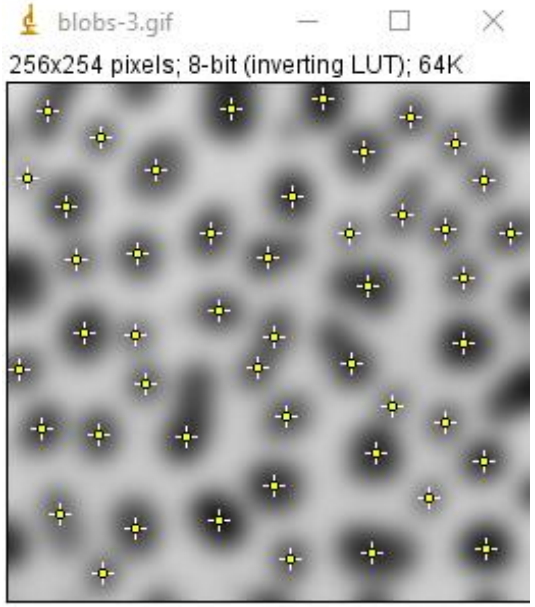
Exercice 2b

Step 3 Process > Filters > Gaussian Blur…
sigma = to radius of objects

Step 4 Process > Find Maxima… Try different parameters!







Hands-on 2: Quantify the number of red spots per nucleus

Hands-on 2: Quantify the number of red spots per nucleus

Image: PLA.tif

Deciding: 3D or 2D? If 2D max intensity projection?

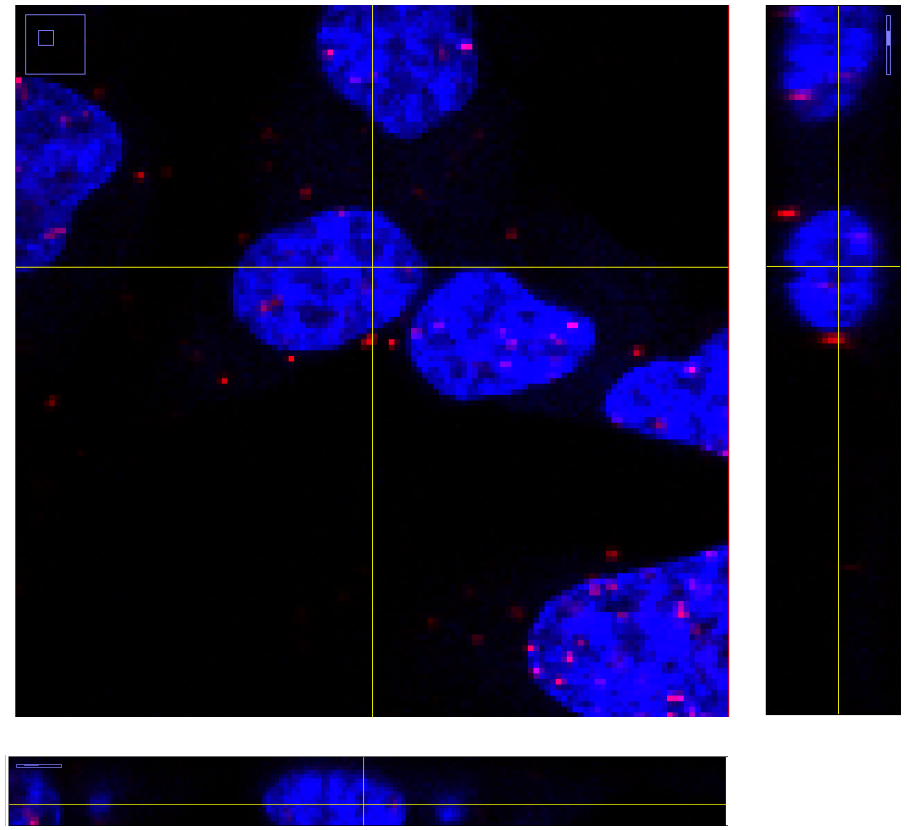


Image › Stacks › Orthogonal Views

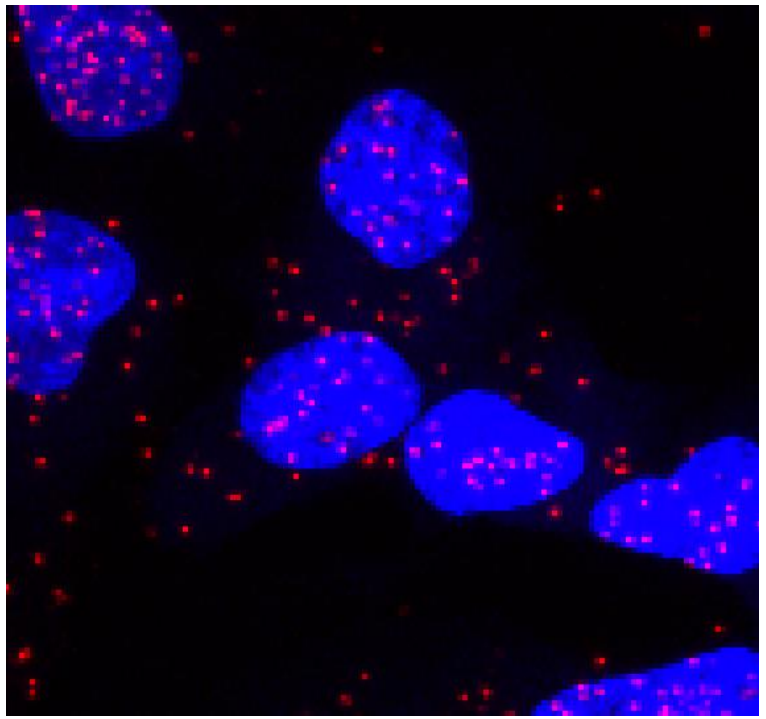


Image › Stacks › Z Project...

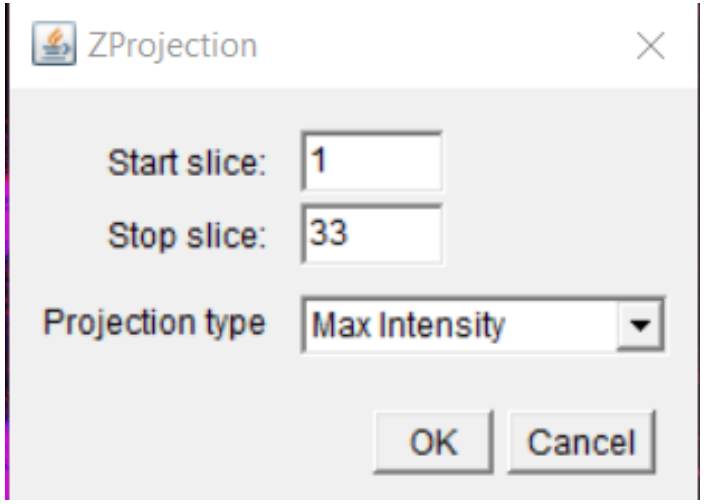
(max projection)

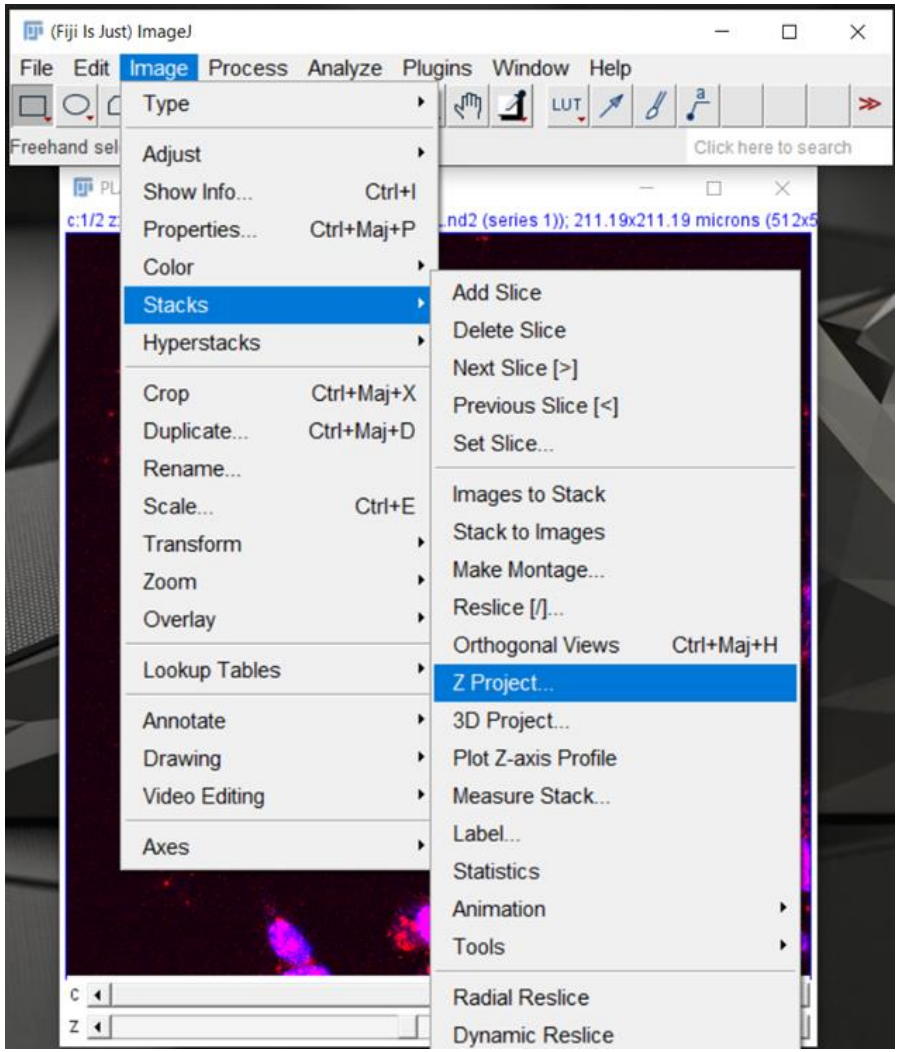
Hands-on 2: Quantify the number of red spots per nucleus

Image: PLA.tif

Step 1 Z projection in maximum intensity

Step 2 Close original image

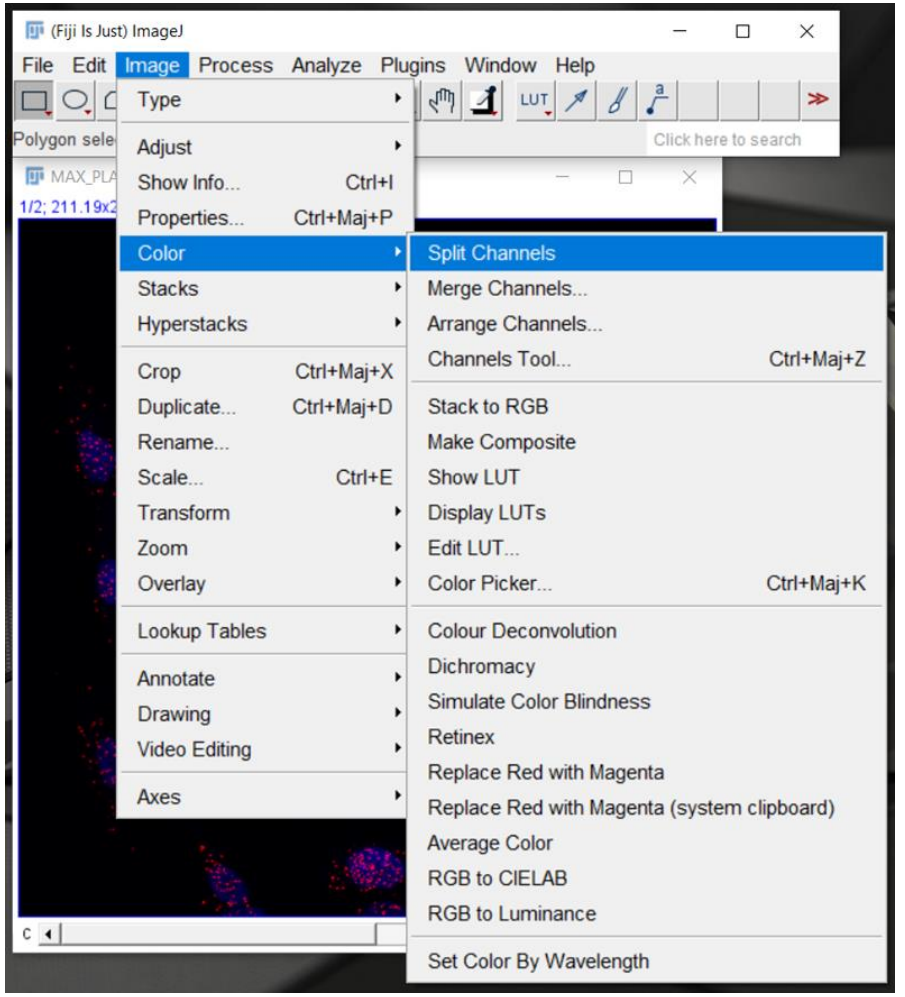




Hands-on 2: Quantify the number of red spots per nucleus

Image: PLA.tif

Step 3 Split Channels

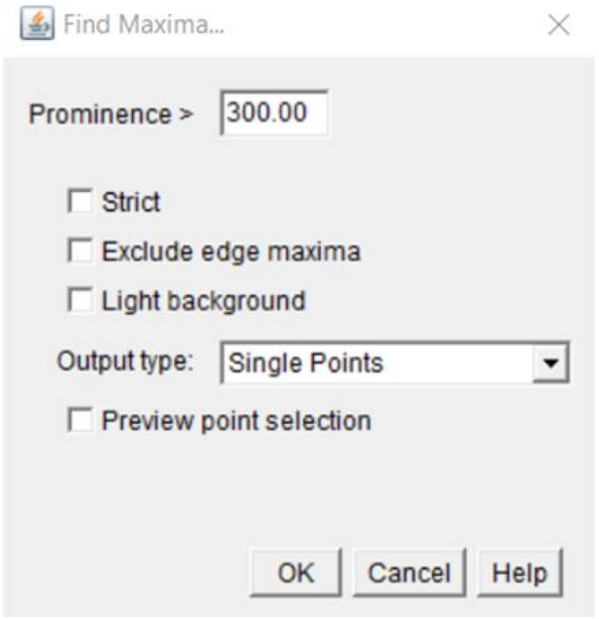


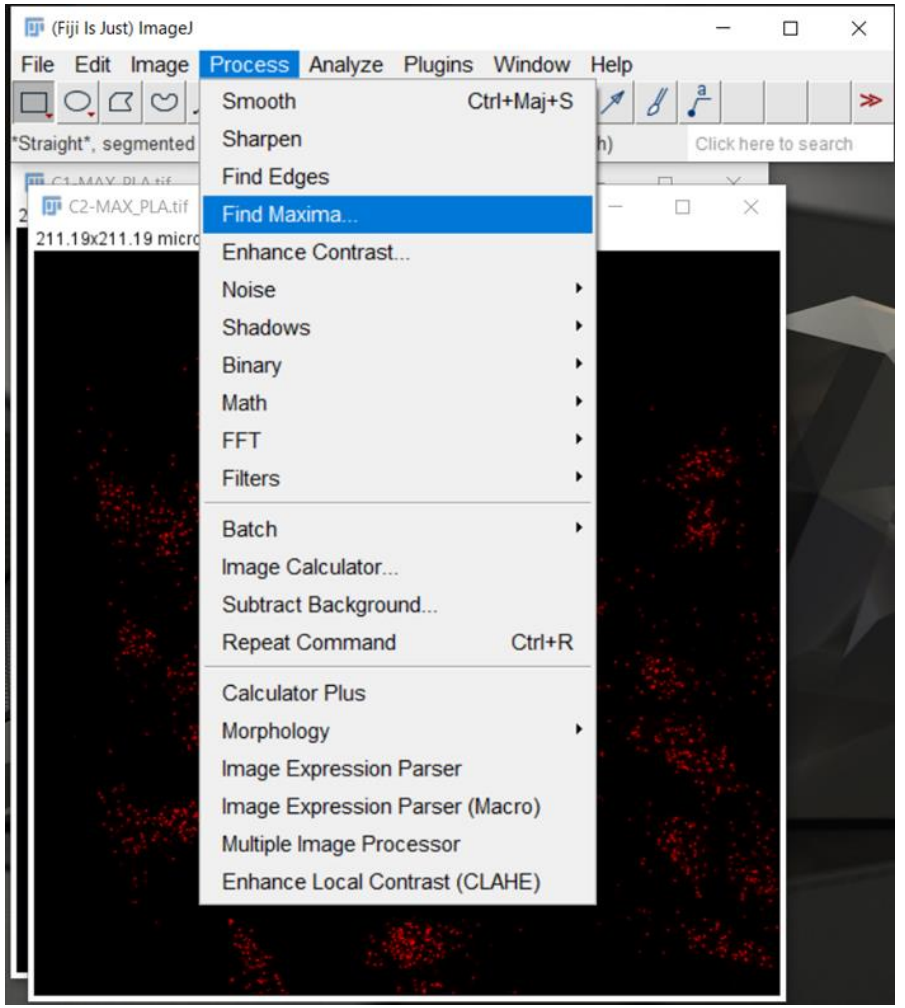
Hands-on 2: Quantify the number of red spots per nucleus

Image: PLA.tif

Step 4 Find Maxima on C2, with Single Points image as output

Step 5 Close C2

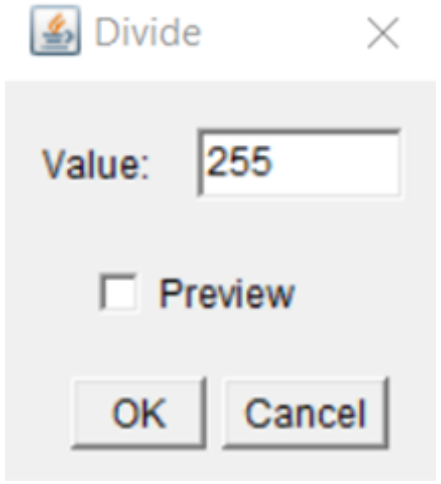


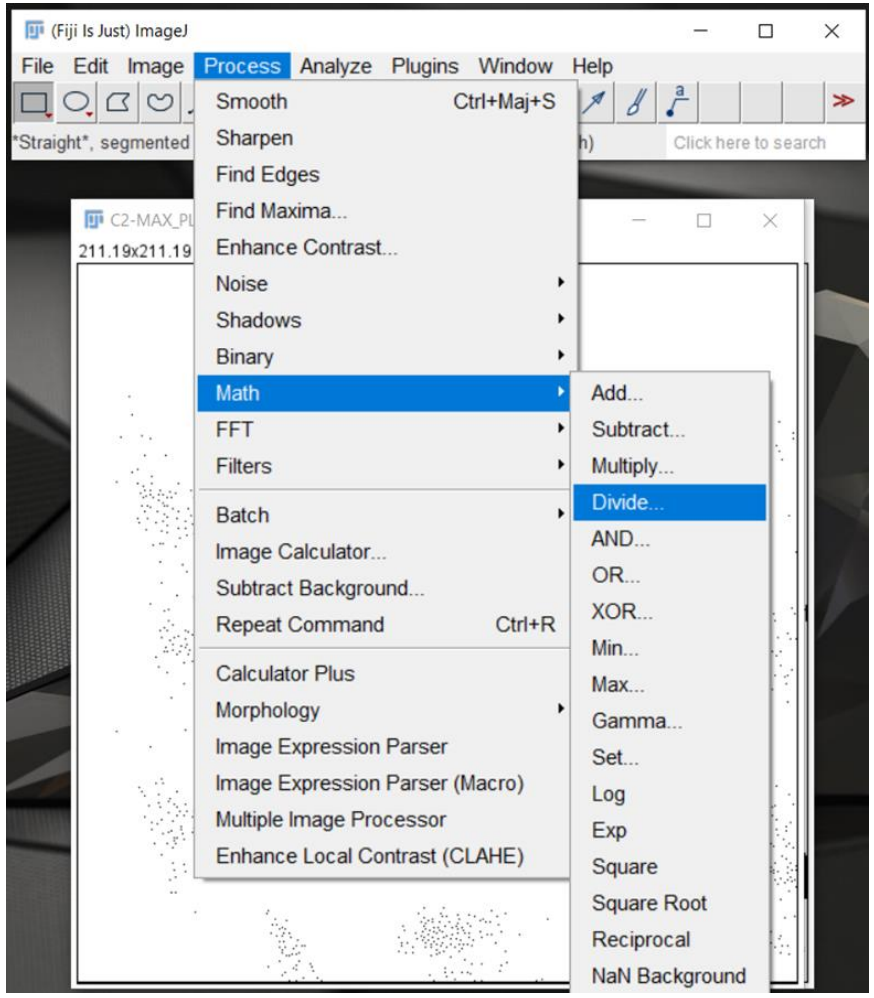


Hands-on 2: Quantify the number of red spots per nucleus

Image: PLA.tif

Step 6 Divide values by 255



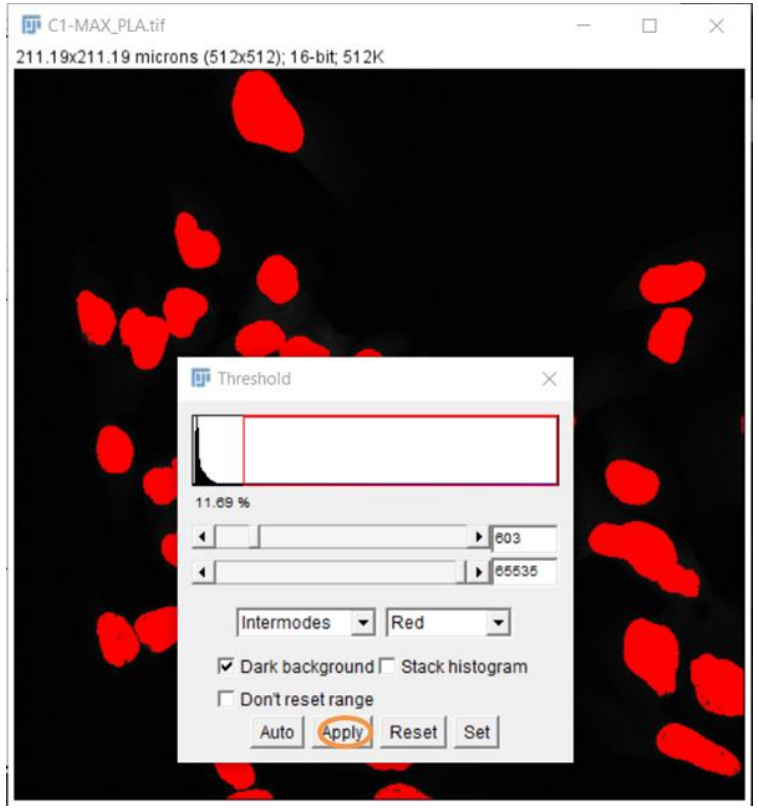


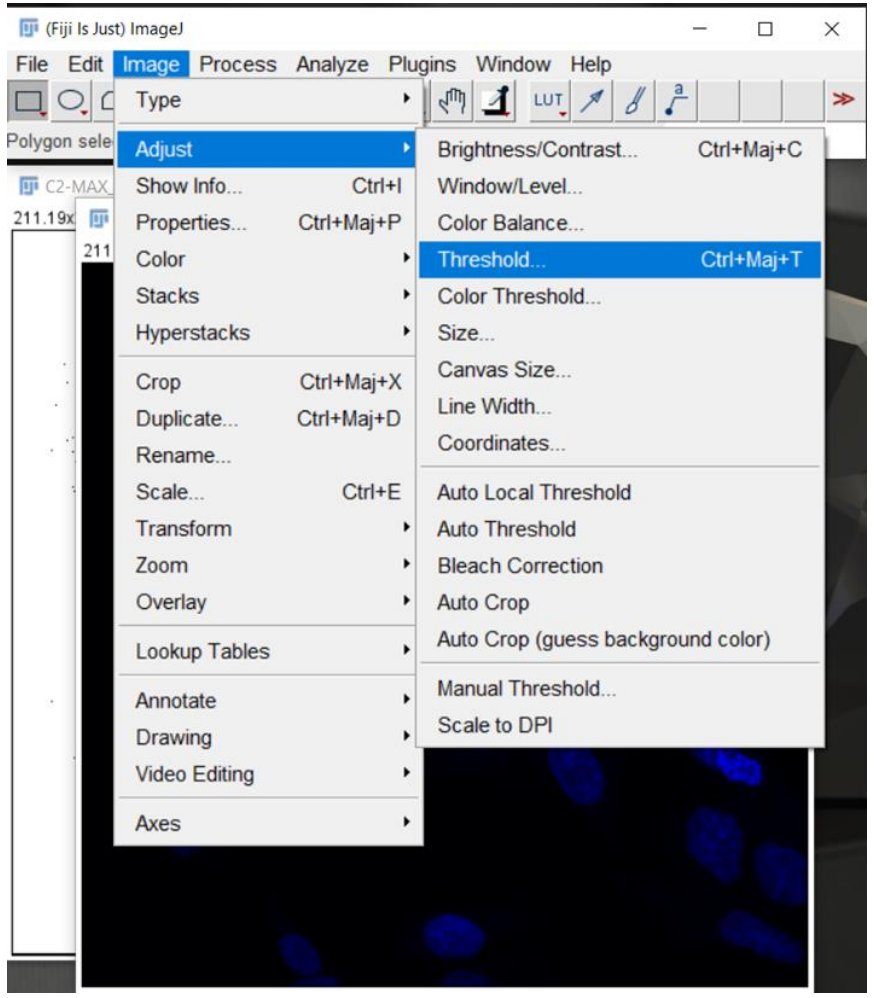
Hands-on 2: Quantify the number of red spots per nucleus

Image: PLA.tif

Step 7

Segment C1

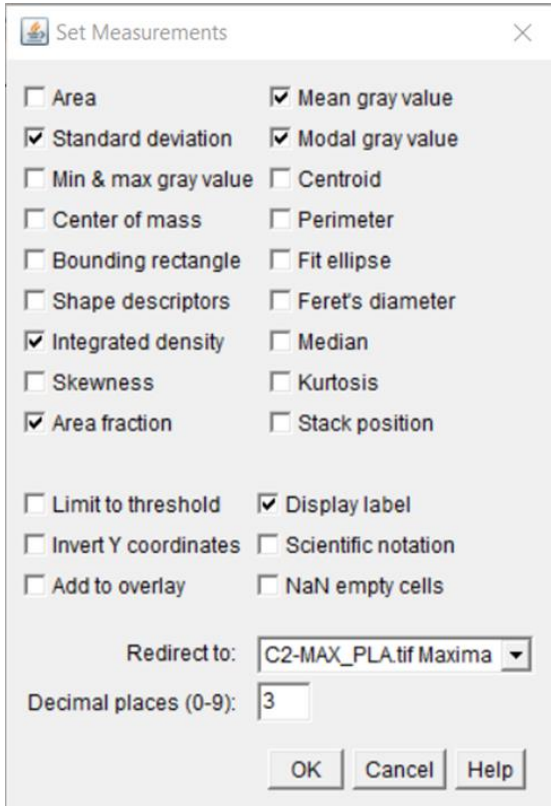


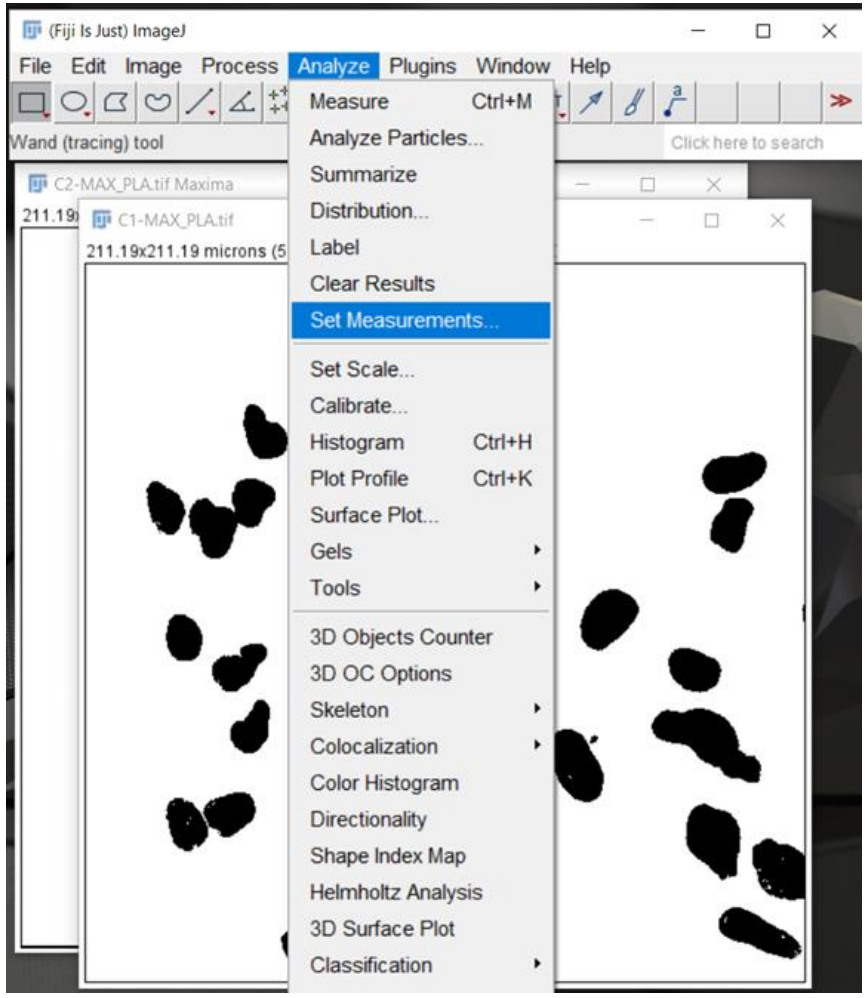


Hands-on 2: Quantify the number of red spots per nucleus

Image: PLA.tif

Step 8 Set Measurements with Integrated density and redirect towards C2

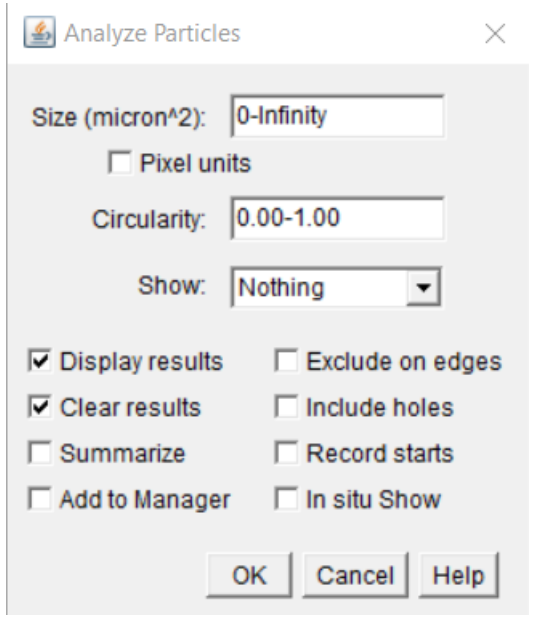


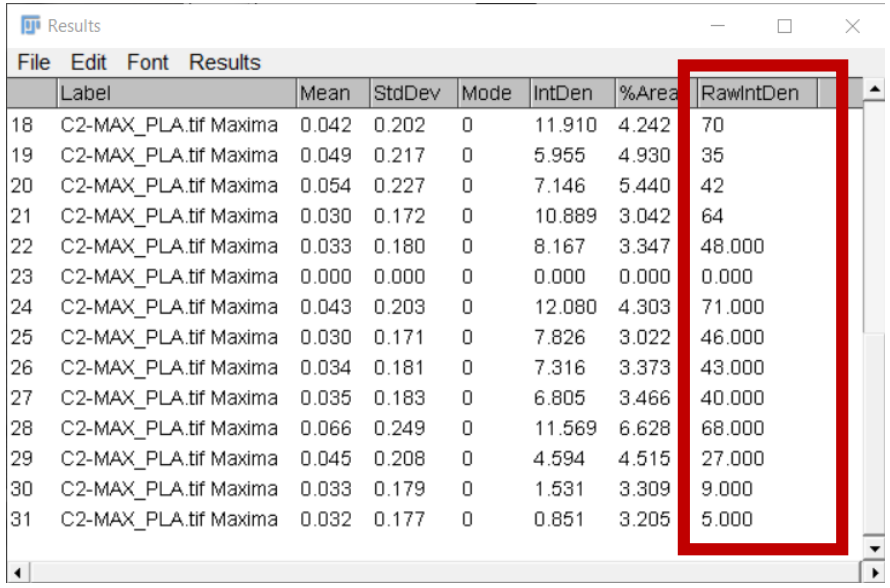


Hands-on 2: Quantify the number of red spots per nucleus

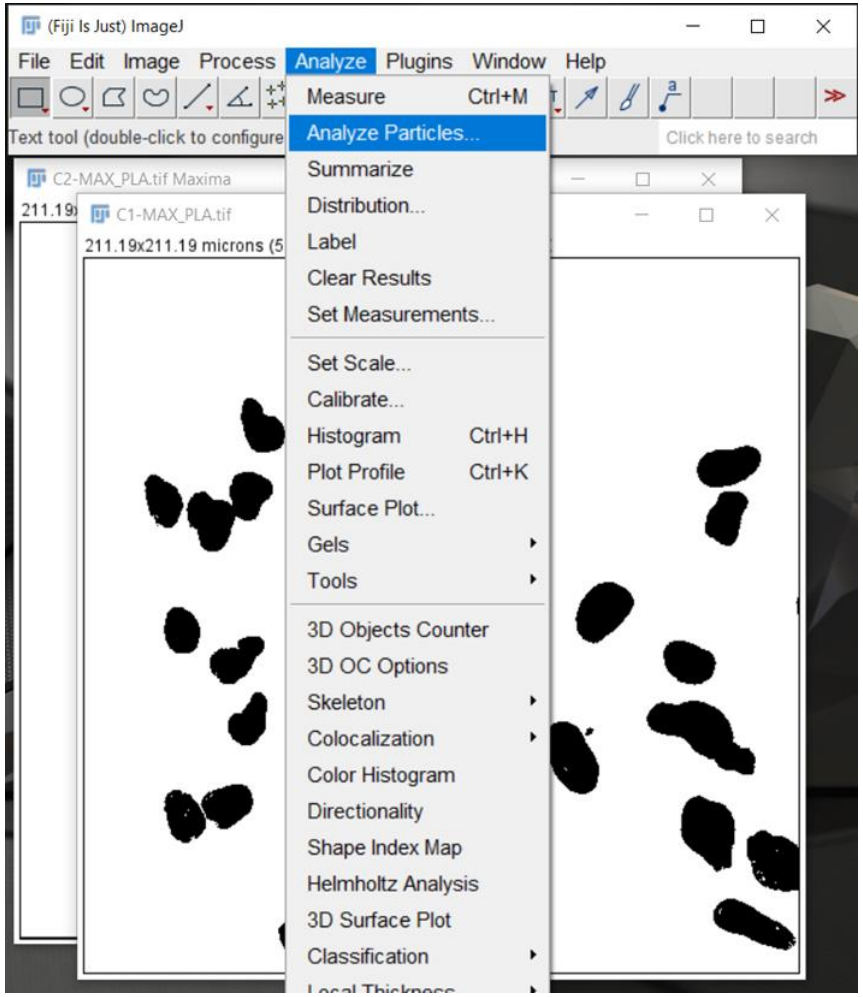
Image: PLA.tif

Step 9 — Analyze Particles





	Label	Mean	StdDev	Mode	IntDen	%Area	RawIntDen
18	C2-MAX_PLA.tif Maxima	0.042	0.202	0	11.910	4.242	70
19	C2-MAX_PLA.tif Maxima	0.049	0.217	0	5.955	4.930	35
20	C2-MAX_PLA.tif Maxima	0.054	0.227	0	7.146	5.440	42
21	C2-MAX_PLA.tif Maxima	0.030	0.172	0	10.889	3.042	64
22	C2-MAX_PLA.tif Maxima	0.033	0.180	0	8.167	3.347	48.000
23	C2-MAX_PLA.tif Maxima	0.000	0.000	0	0.000	0.000	0.000
24	C2-MAX_PLA.tif Maxima	0.043	0.203	0	12.080	4.303	71.000
25	C2-MAX_PLA.tif Maxima	0.030	0.171	0	7.826	3.022	46.000
26	C2-MAX_PLA.tif Maxima	0.034	0.181	0	7.316	3.373	43.000
27	C2-MAX_PLA.tif Maxima	0.035	0.183	0	6.805	3.466	40.000
28	C2-MAX_PLA.tif Maxima	0.066	0.249	0	11.569	6.628	68.000
29	C2-MAX_PLA.tif Maxima	0.045	0.208	0	4.594	4.515	27.000
30	C2-MAX_PLA.tif Maxima	0.033	0.179	0	1.531	3.309	9.000
31	C2-MAX_PLA.tif Maxima	0.032	0.177	0	0.851	3.205	5.000



V/ Initiation to image analysis automation

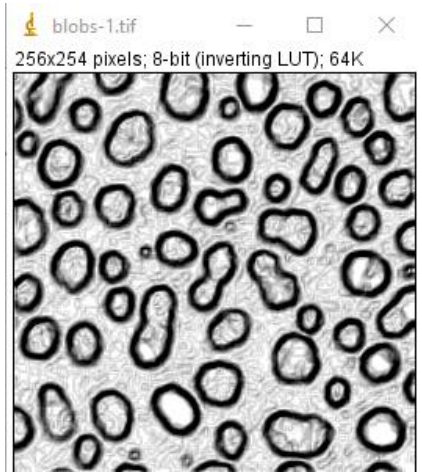
VI/ Automation: Macro Recorder: *Hands-on!*

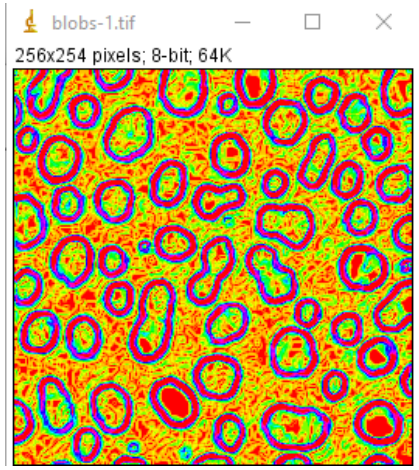
Step 1 Open an image

Step 2 Plugins>Macros>Record…

Step 3 Apply a filter, e.g. Process > Find Edges

Step 4 Change the LUT, e.g. « Spectrum »





VI/ Automation: Macro Recorder: *Hands-on!*

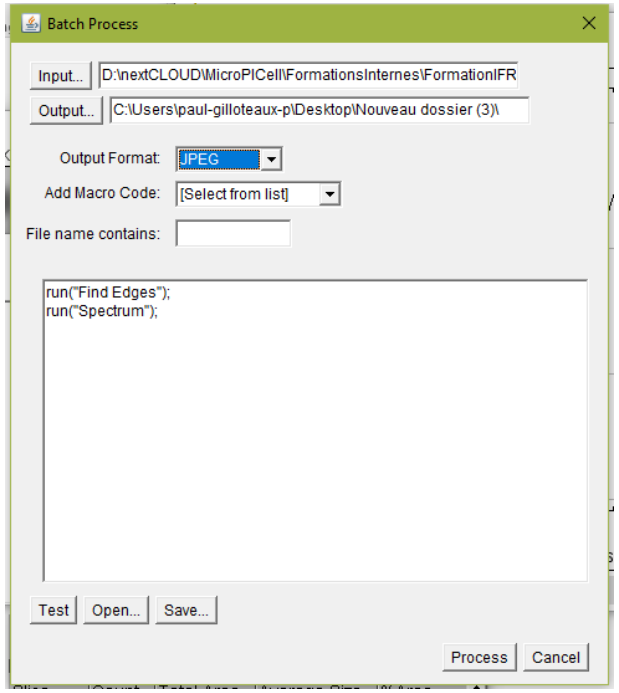
Imagine that you want to apply the same processing to all the images in a folder…

Step 1 Copy the lines from the Recorder

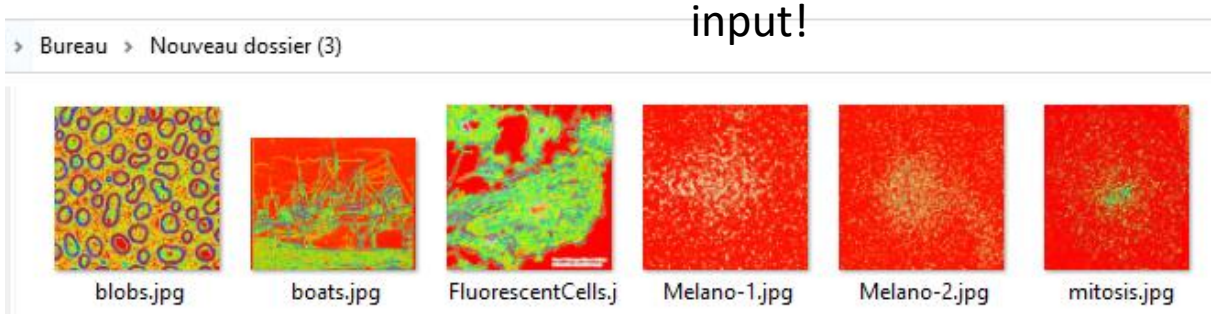
Step 2 Process > Batch > Macro

Step 3 Paste the lines and select parameters

Step 4 Click on Process and close message if any pops up



Batch Process opens all the images containing the specified expression or extension, applies the commands listed, and saves the last active image in its final state, with the specified extension.
Careful to not overwrite your OG files by selecting the same output folder and extensions as your input!



VI/ Automation: Macro Recorder: *Hands-on!*

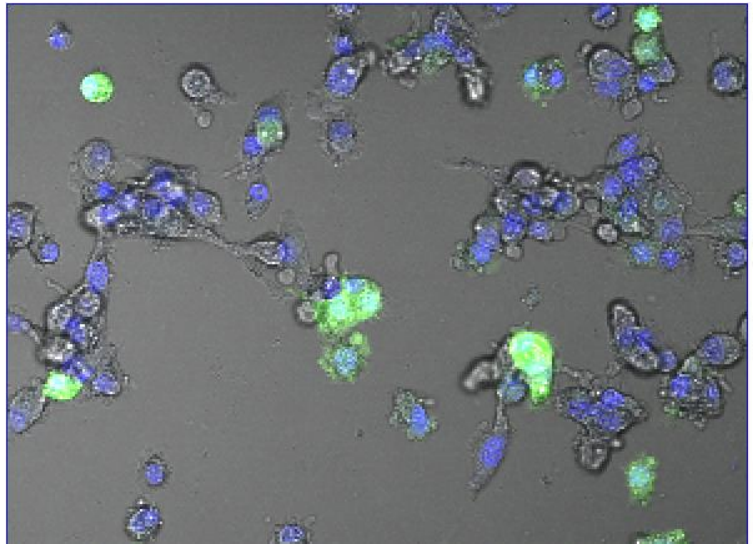
Now redo all the steps of Example 1 with the Recorder on!

Step 1 Create a directory with both melano images

Step 2 Record all steps of Exemple 1

Step 3 Create a macro, clean it

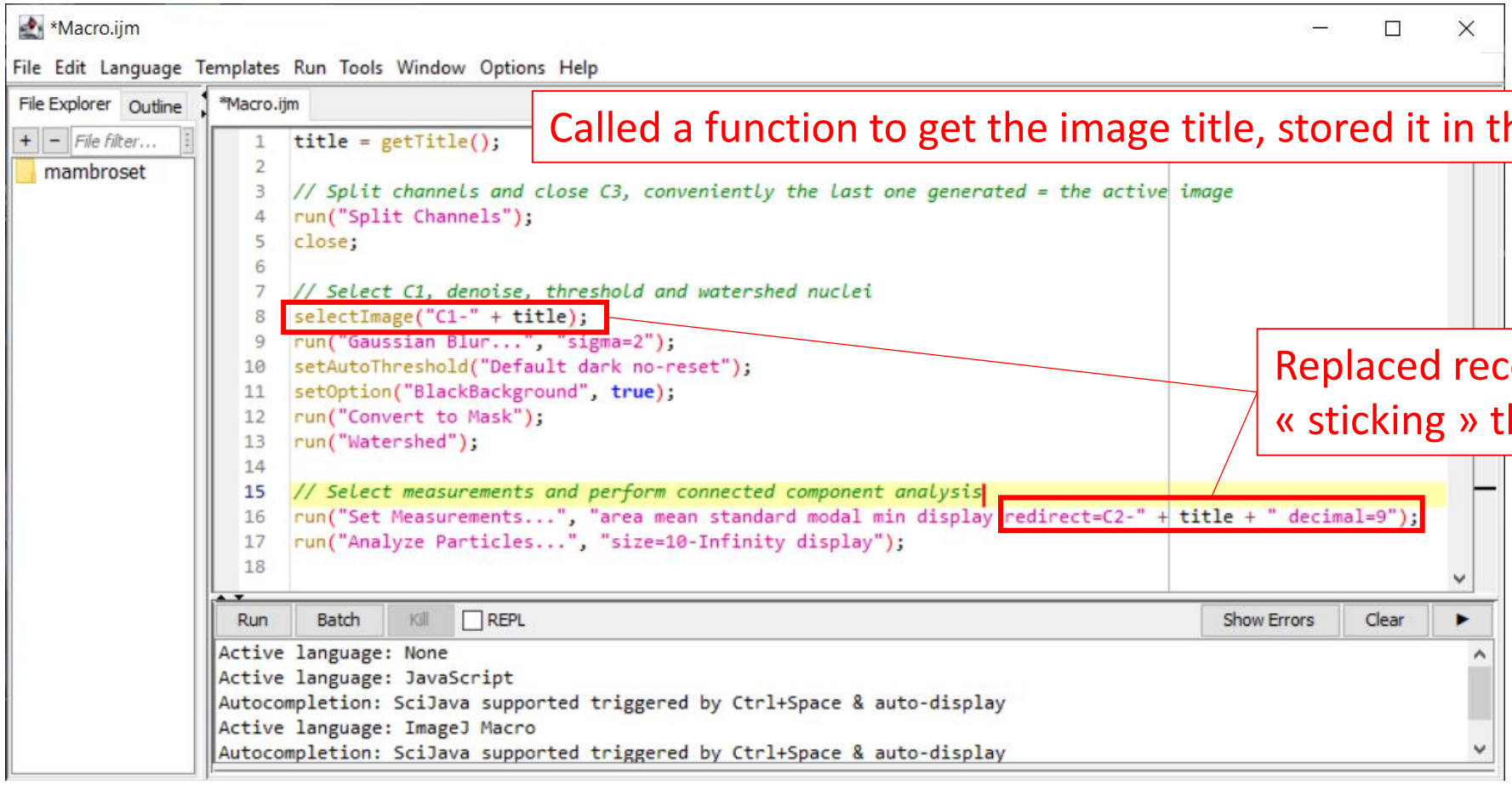
Step 4 What do you think will pose a problem?



VI/ Automation: Comments and variables

Comments (//) make the code more readable.

Variables (“boxes” where to store information) make the code more general



Called a function to get the image title, stored it in the « title » variable

Replaced recorded image name by « sticking » the variable with + signs

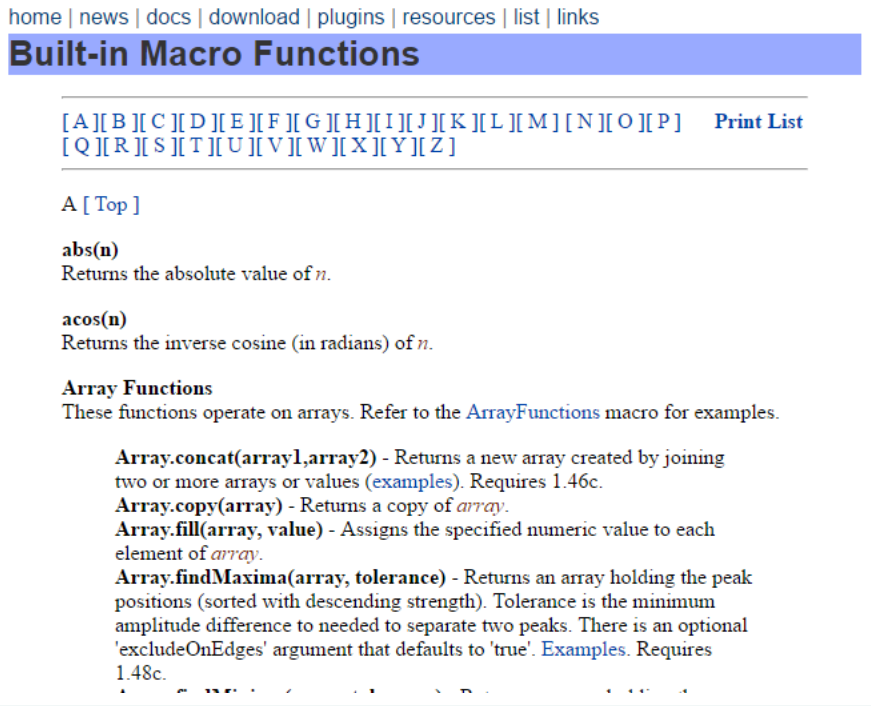
VI/ Automation: Macro fubnctions reference

Info about coding macros:

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

Macro Functions:

- https://imagej.nih.gov/ij/developer/macro/functions.html
- Help > Macro Functions…



home | news | docs | download | plugins | resources | list | links

Built-in Macro Functions

[A][B][C][D][E][F][G][H][I][J][K][L][M] [N][O][P] **Print List**
[Q][R][S][T][U][V][W][X][Y][Z]

A [Top]

abs(n)
Returns the absolute value of *n*.

acos(n)
Returns the inverse cosine (in radians) of *n*.

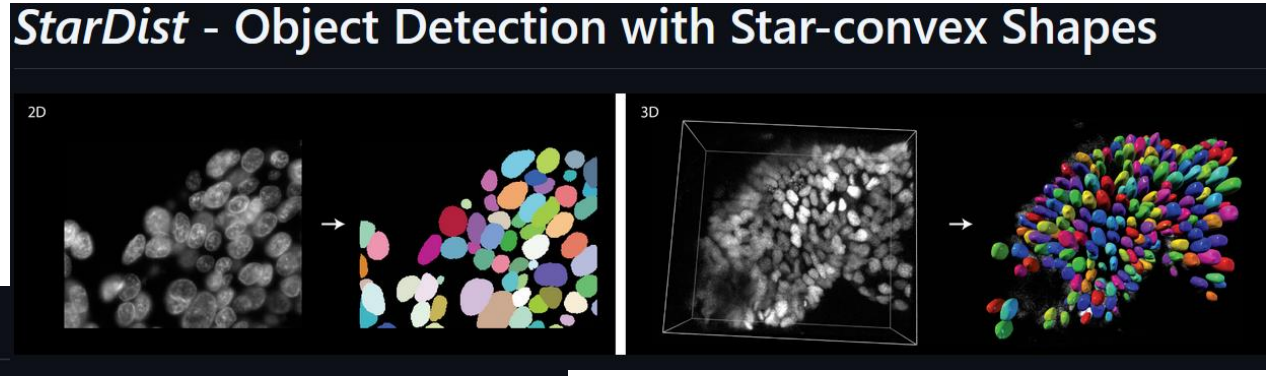
Array Functions
These functions operate on arrays. Refer to the ArrayFunctions macro for examples.

Array.concat(array1,array2) - Returns a new array created by joining two or more arrays or values (examples). Requires 1.46c.
Array.copy(array) - Returns a copy of *array*.
Array.fill(array, value) - Assigns the specified numeric value to each element of *array*.
Array.findMaxima(array, tolerance) - Returns an array holding the peak positions (sorted with descending strength). Tolerance is the minimum amplitude difference to needed to separate two peaks. There is an optional 'excludeOnEdges' argument that defaults to 'true'. Examples. Requires 1.48c.

Bonus: Intro to Machine Learning: Segmentation: StarDist

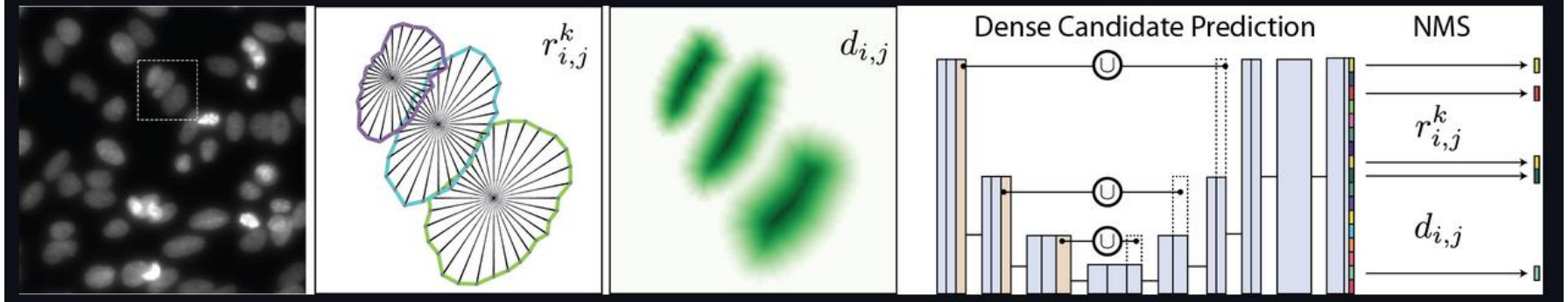
https://github.com/qupath/qupath-extension-stardist

https://github.com/stardist/stardist



Overview

The following figure illustrates the general approach for 2D images. The training data consists of corresponding pairs of input (i.e. raw) images and fully annotated label images (i.e. every pixel is labeled with a unique object id or 0 for background). A model is trained to densely predict the distances (r) to the object boundary along a fixed set of rays and object probabilities (d), which together produce an overcomplete set of candidate polygons for a given input image. The final result is obtained via non-maximum suppression (NMS) of these candidates.

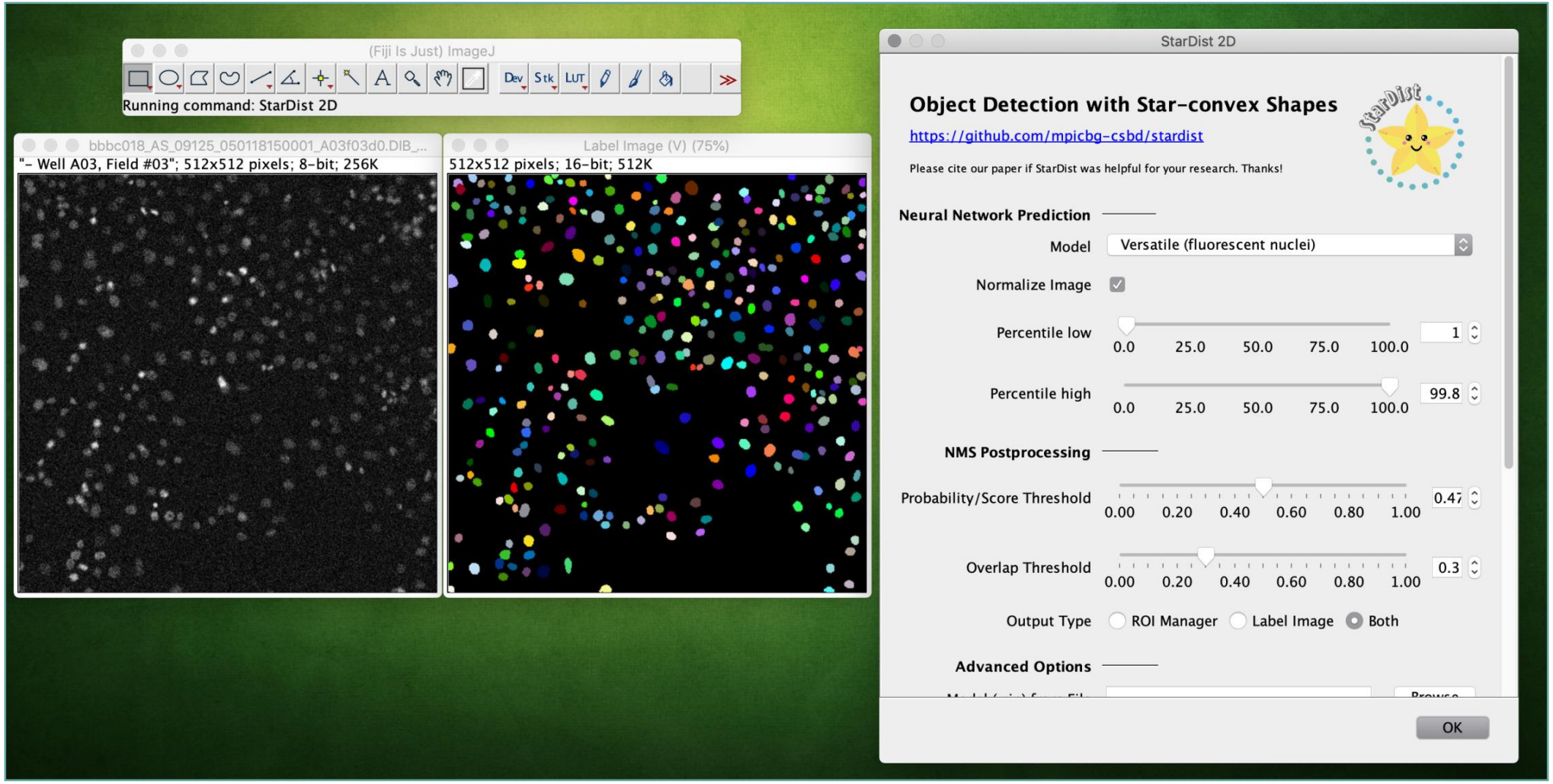


If you use this extension, you should cite the original StarDist publication

•Uwe Schmidt, Martin Weigert, Coleman Broaddus, and Gene Myers. *Cell Detection with Star-convex Polygons*.
International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), Granada, Spain, September 2018.

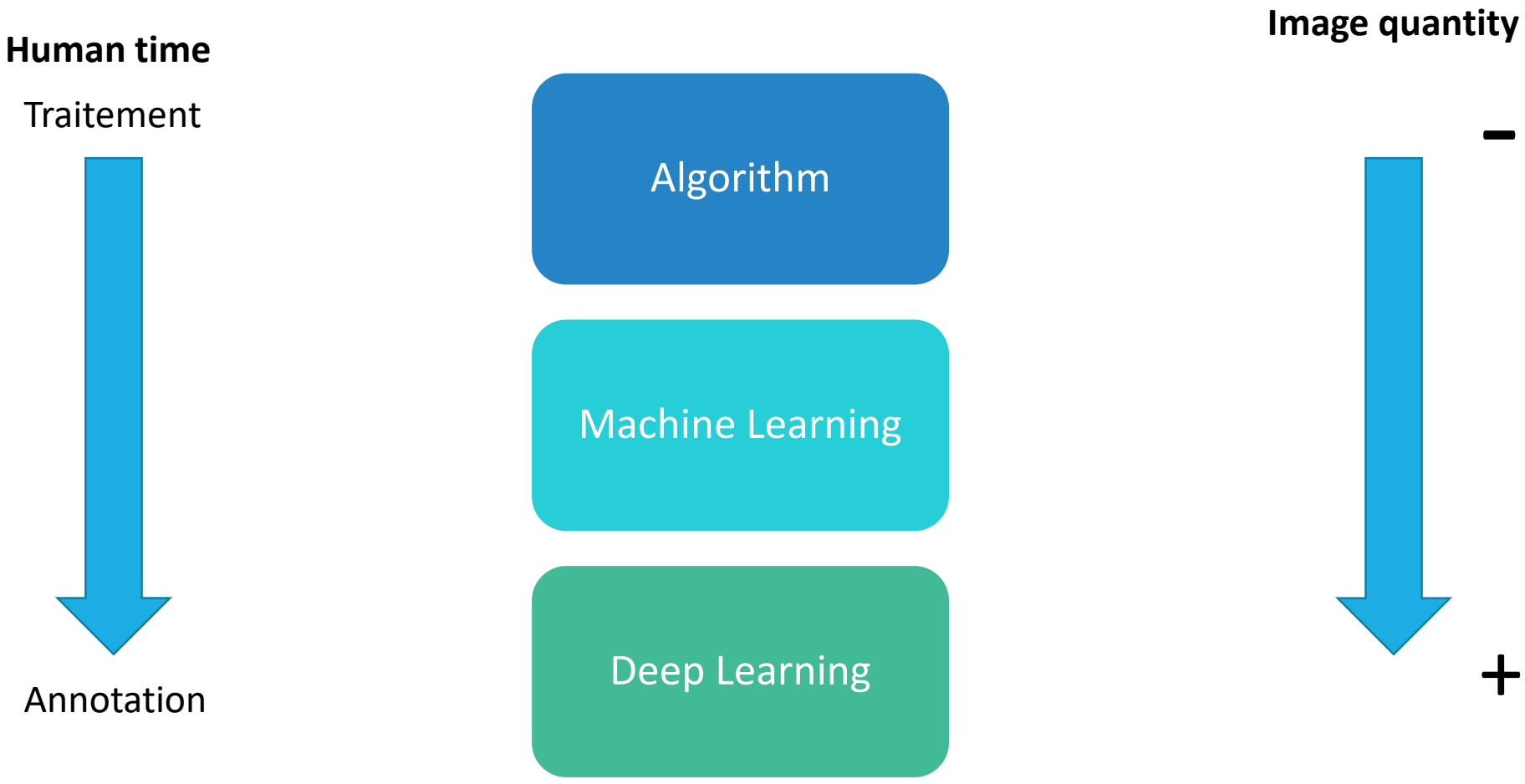
Bonus: Intro to Machine Learning: Segmentation: StarDist

https://imagej.net/plugins/stardist

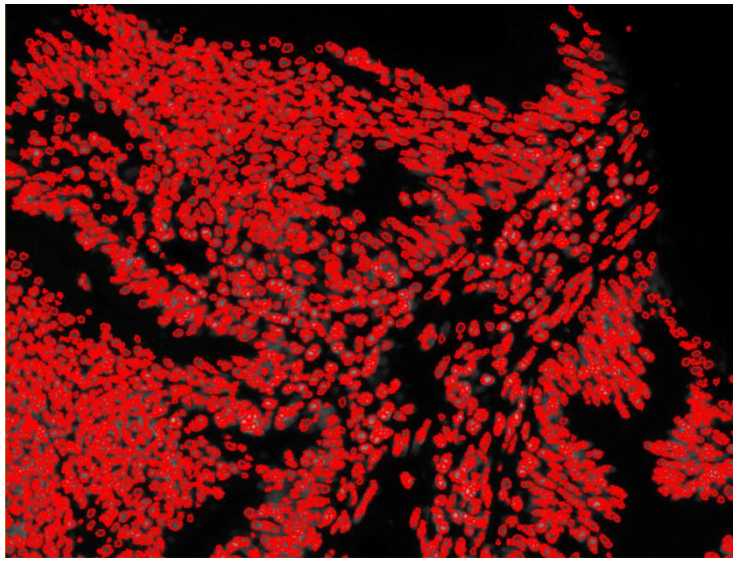


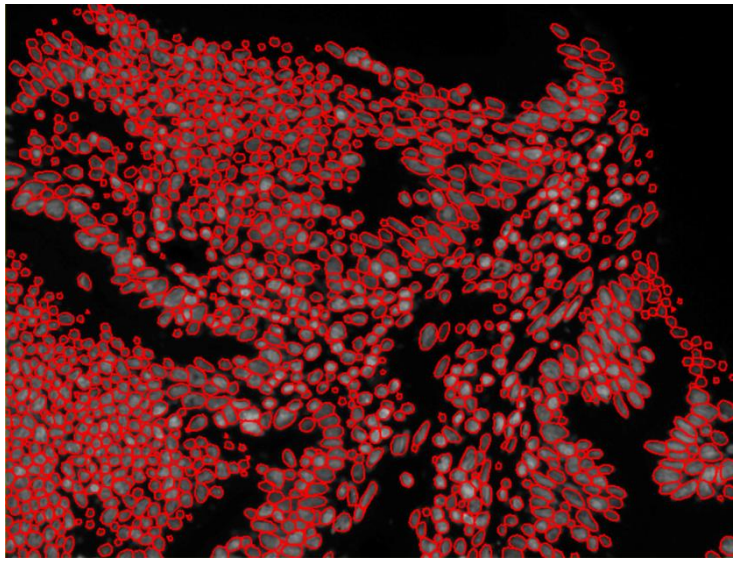
Bonus: Intro to Machine Learning: Comparison

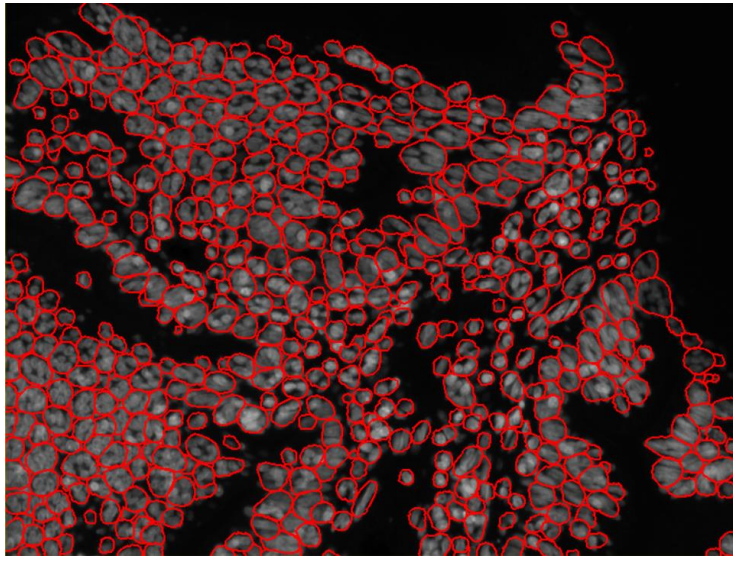
Differences: Paradigm shift

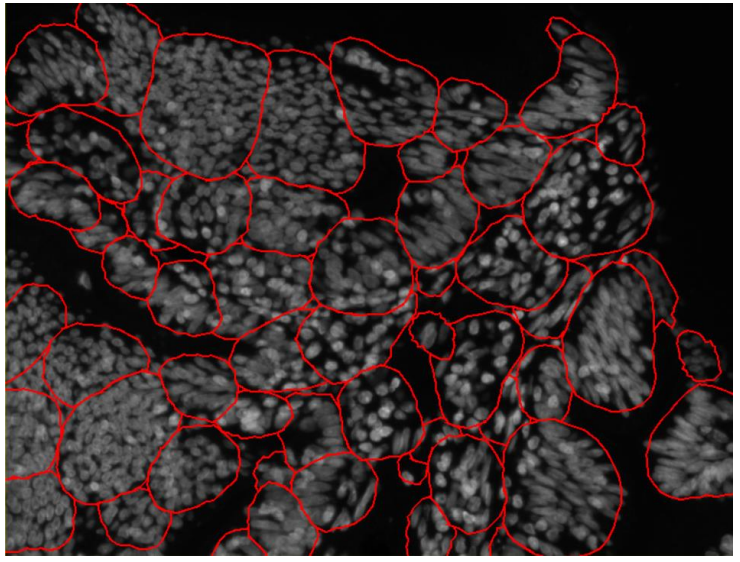


Bonus: Intro to Machine Learning: Warning









Bonus: Intro to Machine Learning: Warning