

Incorporating Machine Learning Tools into Image Analysis Workflows using Fiji, Ilastik and StarDist

Ofra Golani

Ehud Sivan, Dean Ranmar, Reinat Nevo

May 2023

MICC Cell Observatory,
Weizmann Institute of Science

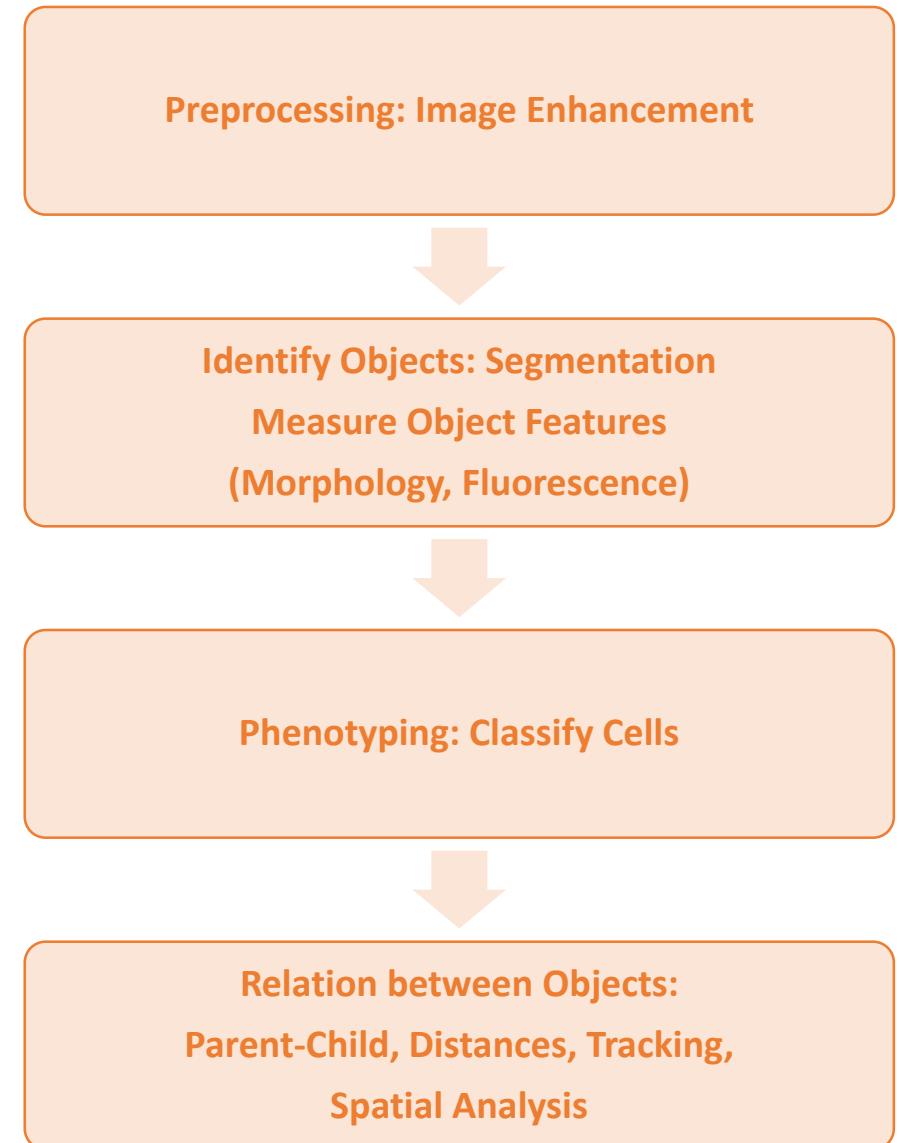
Image Analysis Workflows

- Break the analysis into **sequential steps**
- In each step: choose from **modules / components** of the corresponding category
- Interactive **optimization** of the analysis steps and **tuning** components and user-defined **parameters**
- **Automation**: Create pipe-line, looping on many images
- **Statistics**: Group results and Compare
- Visualization and **Inspection** (sometimes Manual correction)

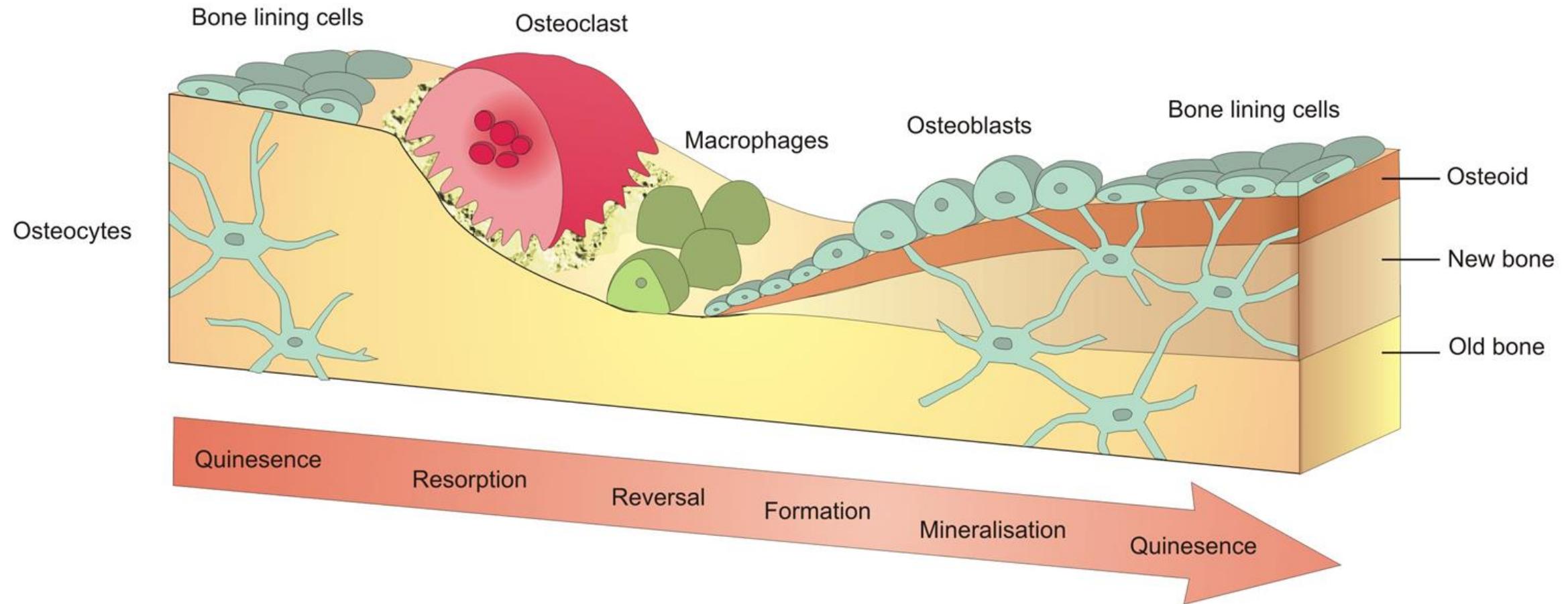
Which objects ?

What to Measure ?

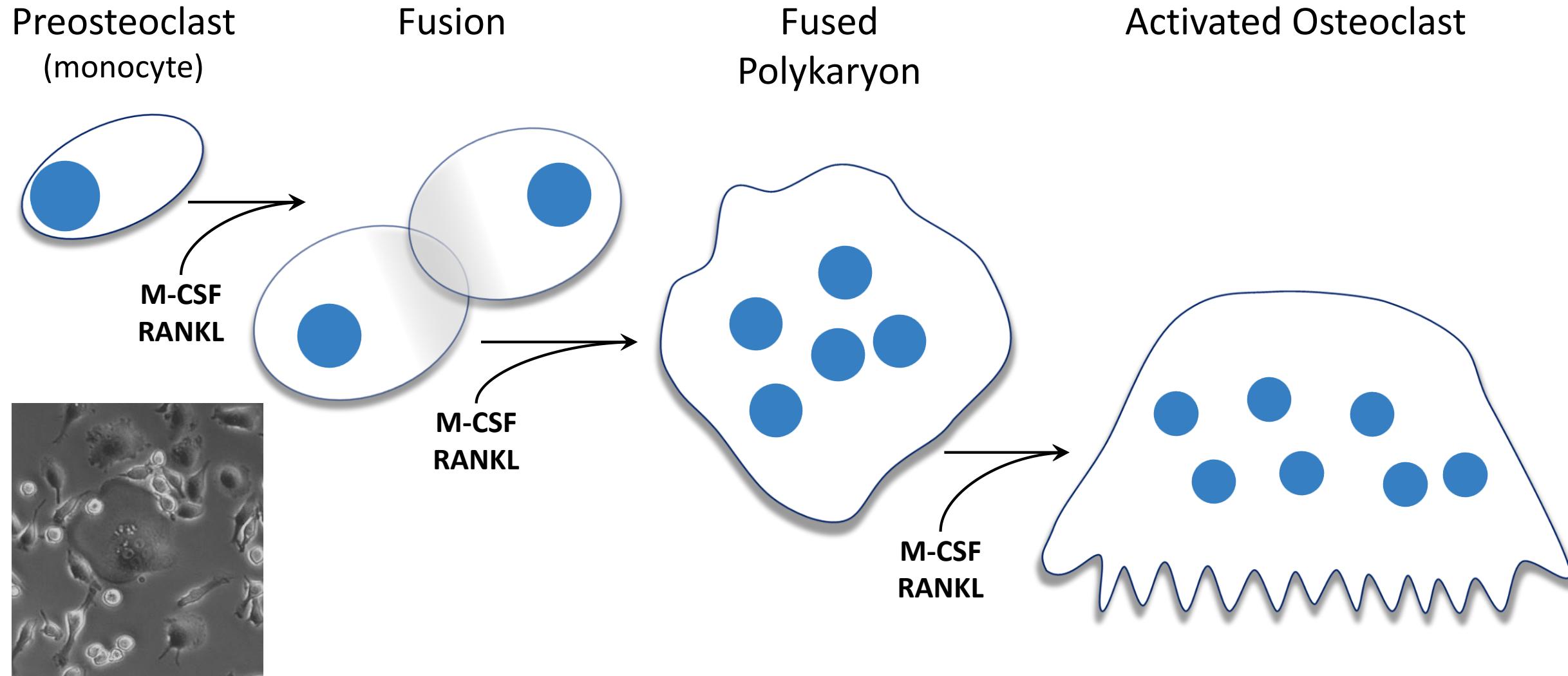
Which module suites for each step ?



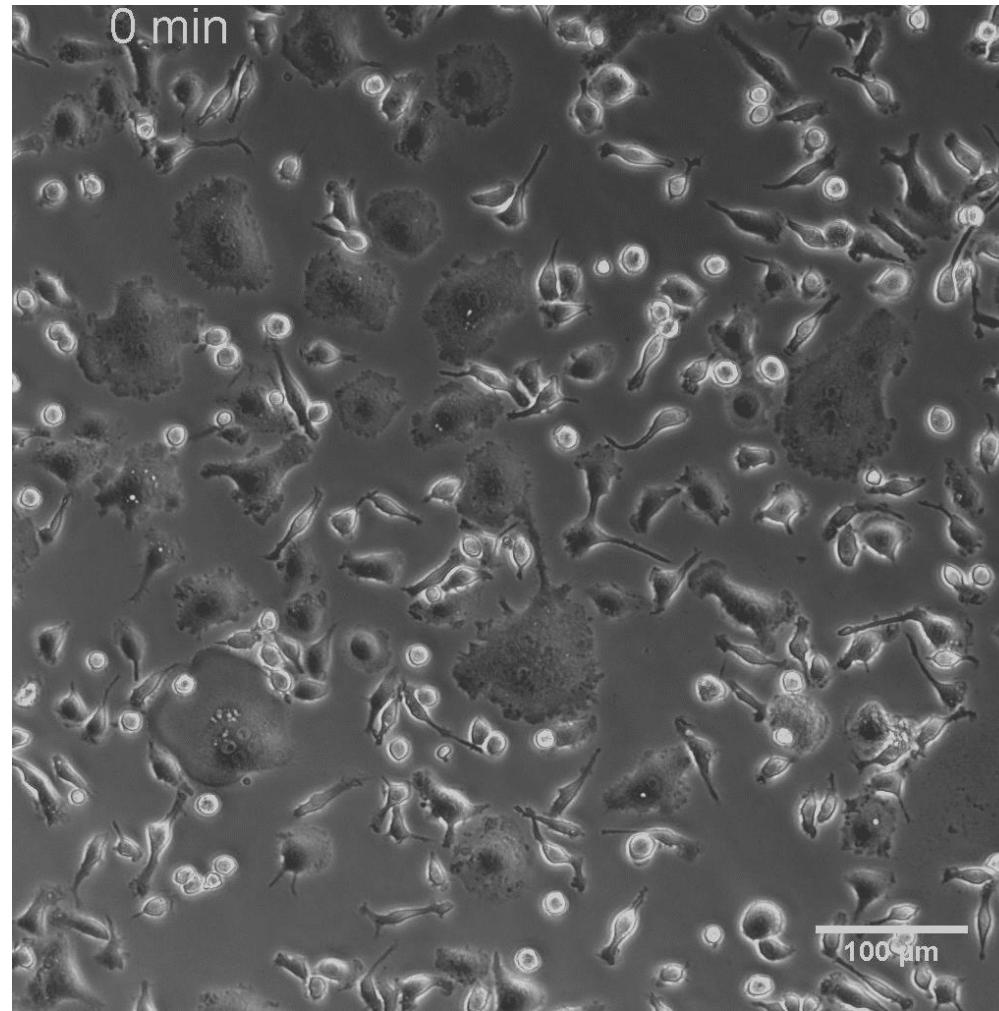
The Bone remodeling Process



Osteoclastogenesis



Fusion of osteoclasts precursors



Time lapse: 5 min

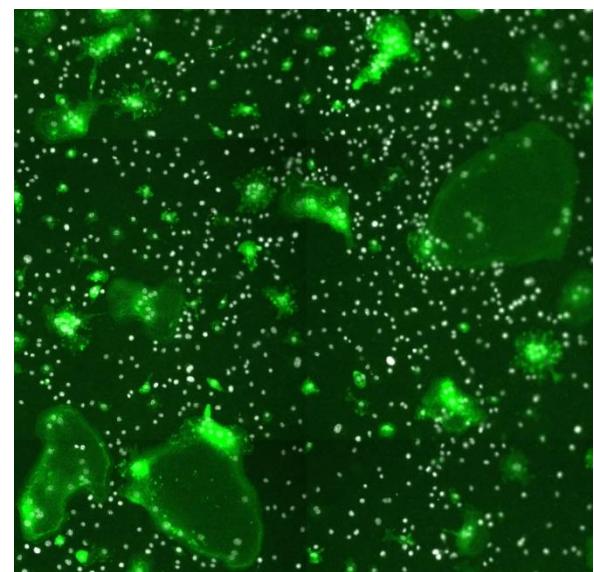
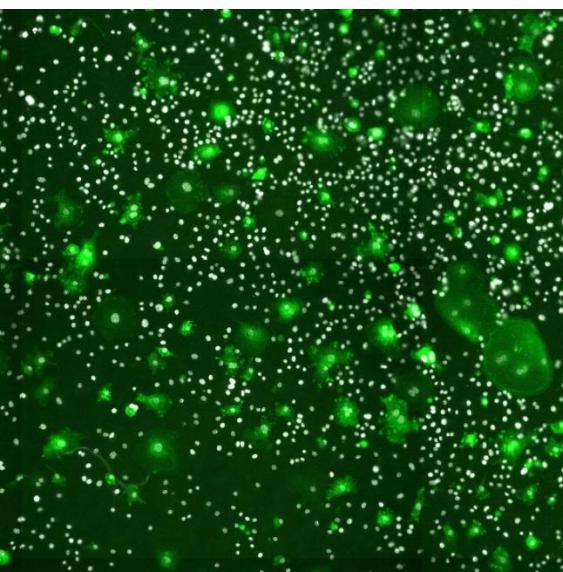
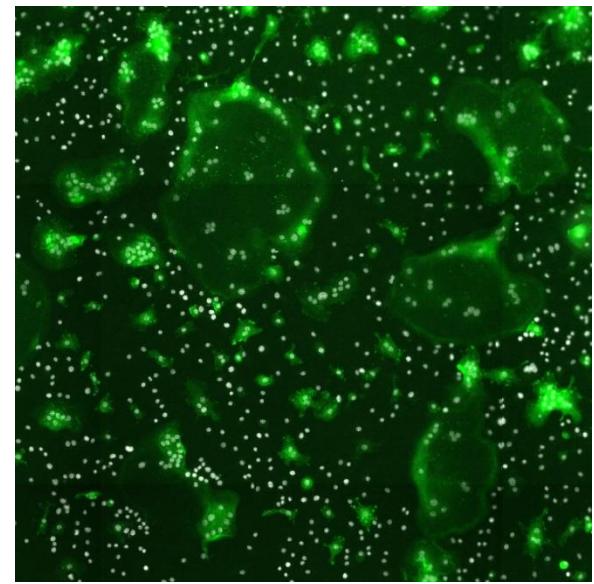
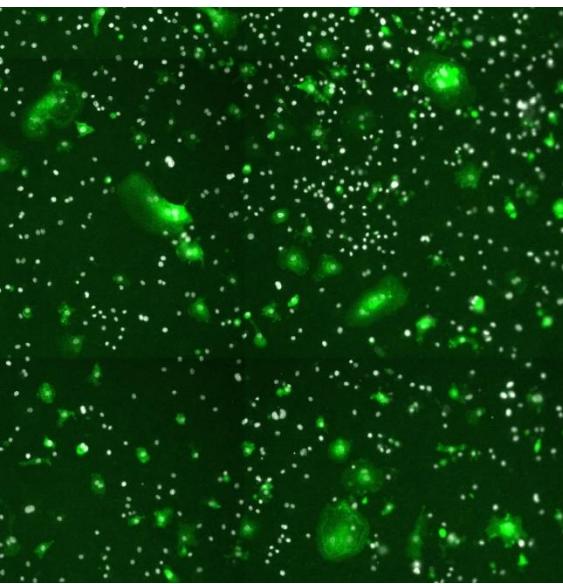
10x phase contrast

Osteoclasts from mouse spleen prepared by Maayan Barnea-Zohar from Prof. Ari Elson's lab

Dataset: Osteoclasts

Imaged with Hermes automated widefield microscope,
stitched, cropped for the workshop

Ch1: Osteoclast green
Ch2: Nuclei white



Building Analysis Workflows in Fiji

Which Objects ?

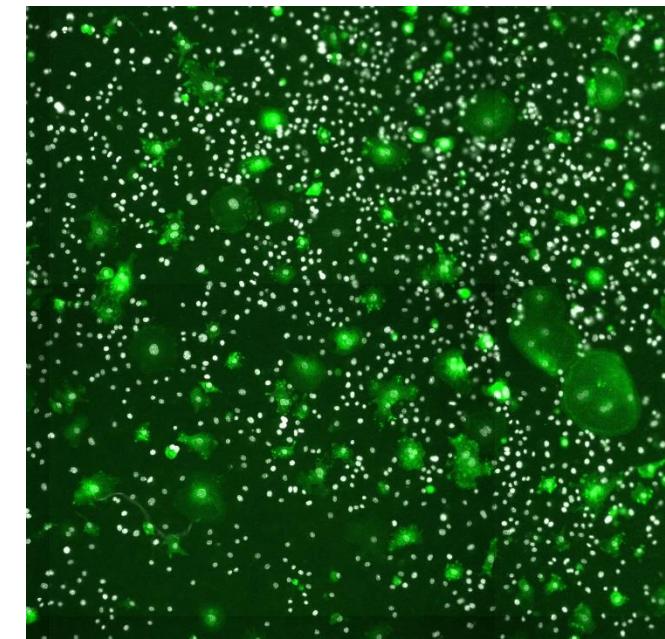
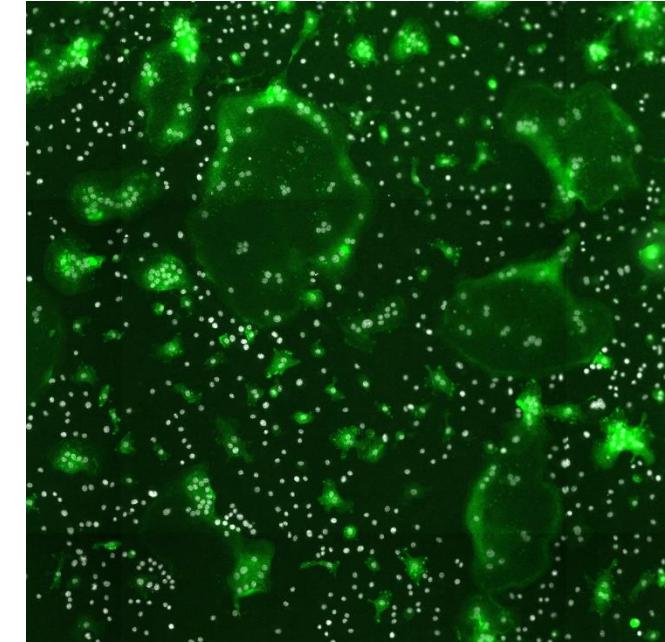
Osteoclasts (OCL): variable size, variable intensity
Nuclei: bright, ellipsoid, may be clustered

What to Measure ?

OCL size,
Number of nuclei per OCL
Sub-cellular distribution of the nuclei within the cells:
random vs clustered; Central vs peripheral

Which Component to use for each step ?

We'll see through the workshop...



Selecting suitable segmentation approach

Nuclei

Bright, ellipsoid, may be clustered

→ Let's try Threshold-based approach

Osteoclasts

Variable size, variable intensity

→ Let's try Threshold-based approach

Selecting suitable segmentation approach

Nuclei

Bright, ellipsoid, may be clustered

→ Classical threshold based method may work

→ Separation of touching objects by watershed / seeded watershed may work

→ Matches Stardist Deep learning instance segmentation scenario. This is our first go-to-tool for nuclei segmentation at the moment

→ Out-of-the-box Stardist works perfectly well

→ If needed, retraining on own data is rather simple with ZeroCostDL4Mic Notebook

Osteoclasts

Variable size, variable intensity

→ Threshold based not suitable

→ Filtering required

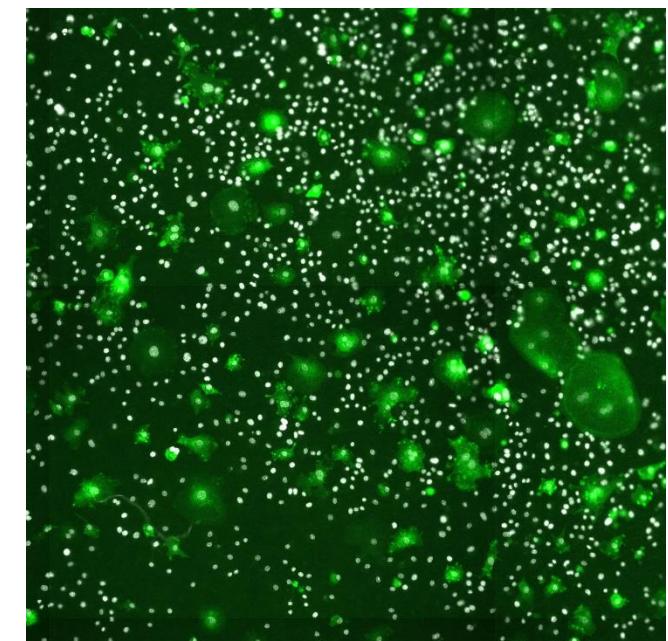
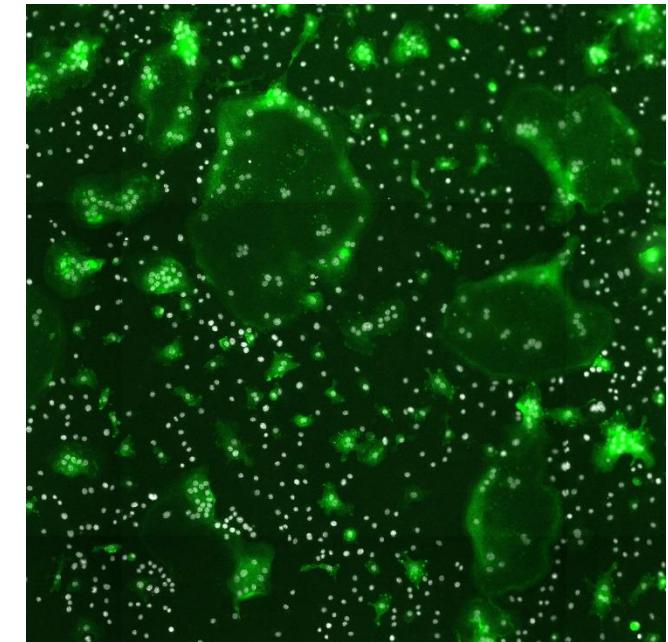
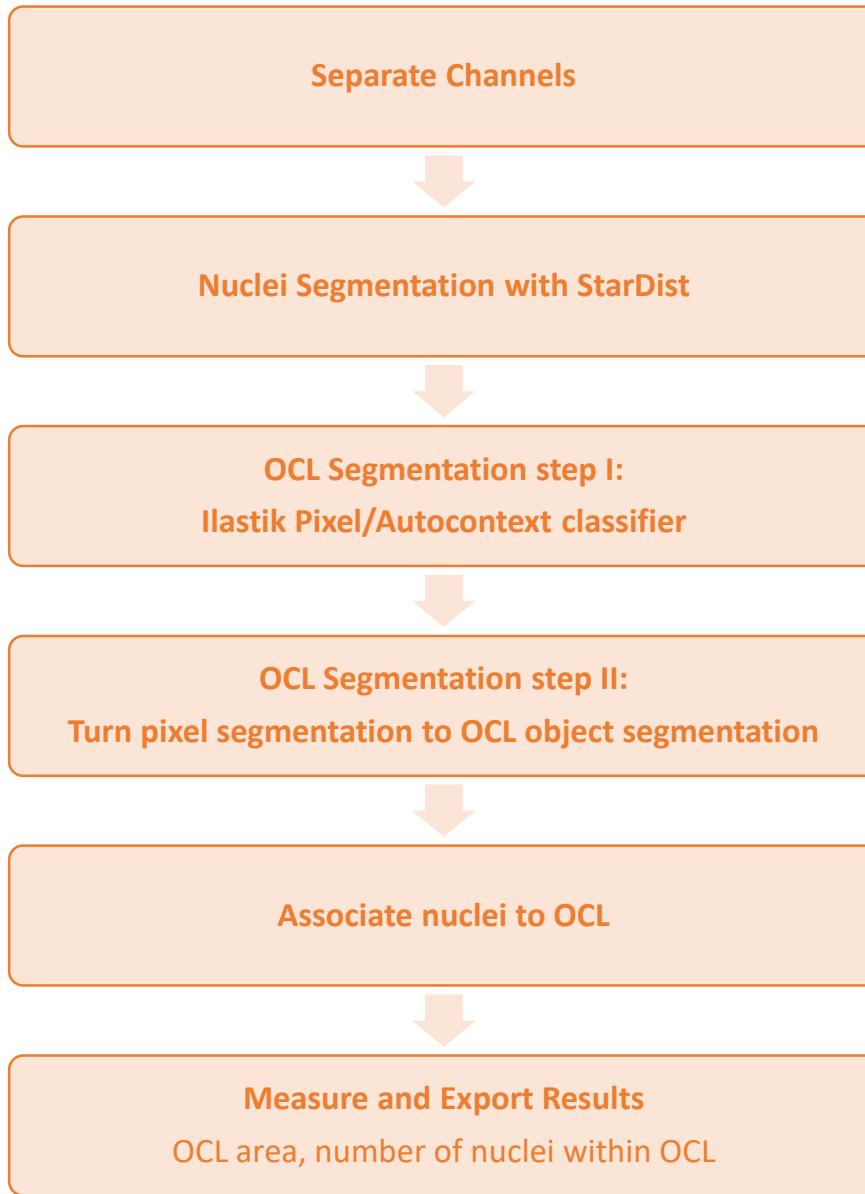
→ Multiple filter size / type needed

→ Instance segmentation Deep Learning (eg StarDist / Cellpose) assumes similar object size → will not work

→ Machine learning or Deep Learning Semantic segmentation approach

→ ML (lلاstik) chosen as it allows for rather easy sparse interactive training

Building Analysis Workflows in Fiji

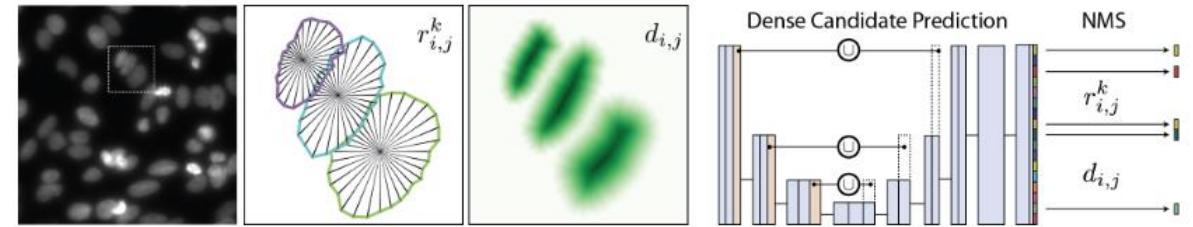
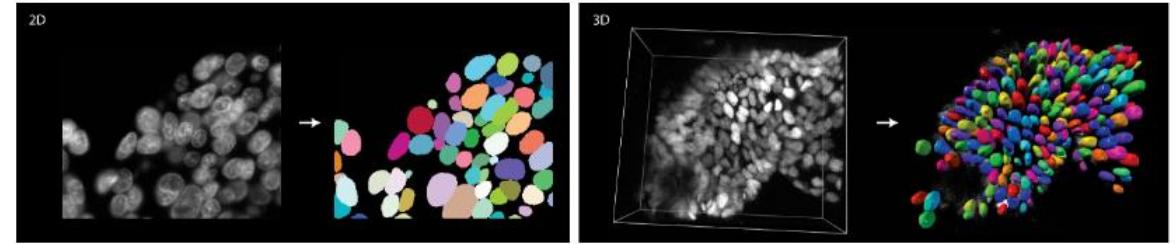


Nuclei Segmentation with StarDist

By: Uwe Schmidt, Martin Weigert

- Open source
- Fiji Plugin
- Out-of-the-box models for
2D Fluorescent and H&E images
- Training Notebooks available via StarDist and ZeroCostDL4Mic
- QuPath implementation: Extension, Running through script, Cell Expansion

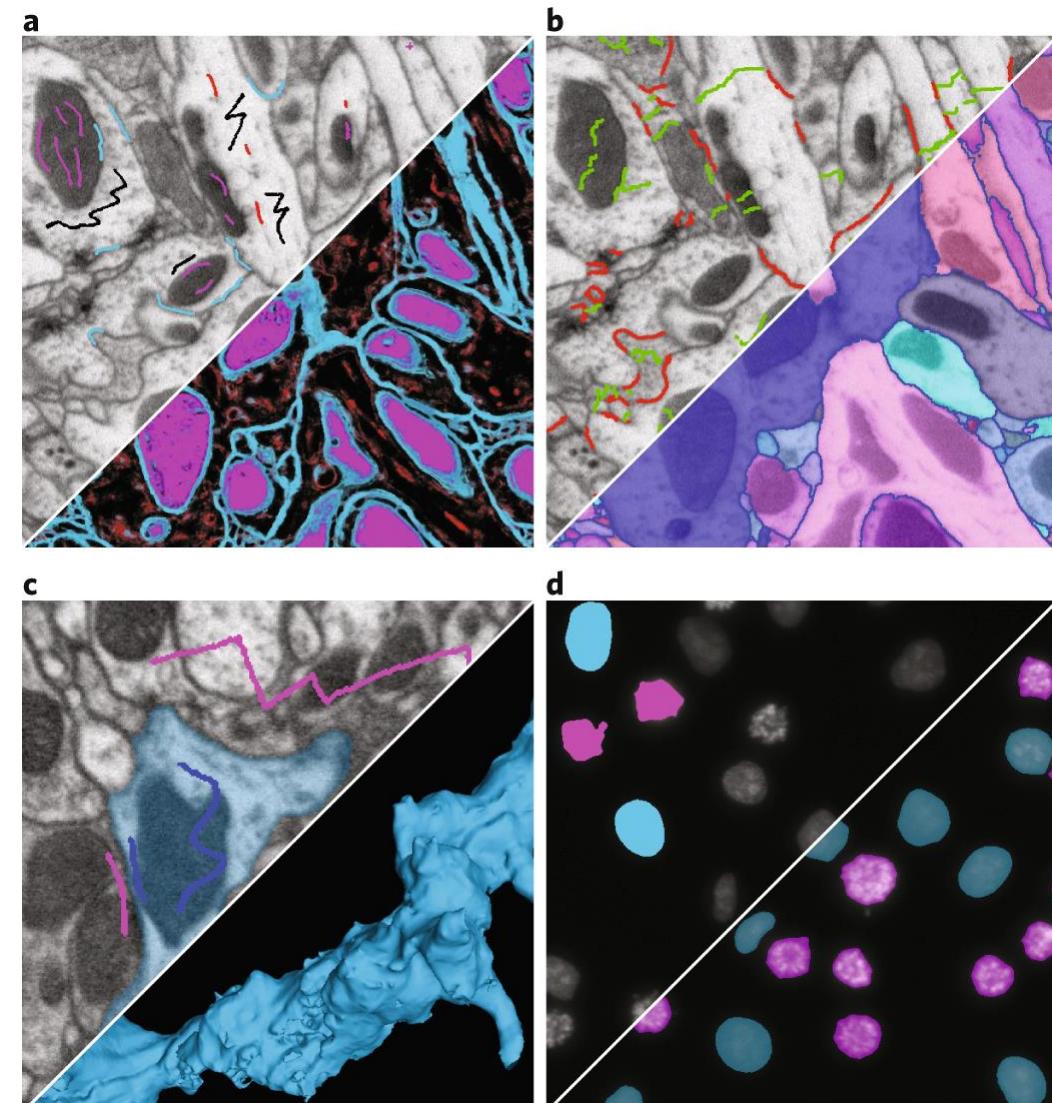
StarDist - Object Detection with Star-convex Shapes



Ilastik: Interactive machine learning for (bio)image analysis

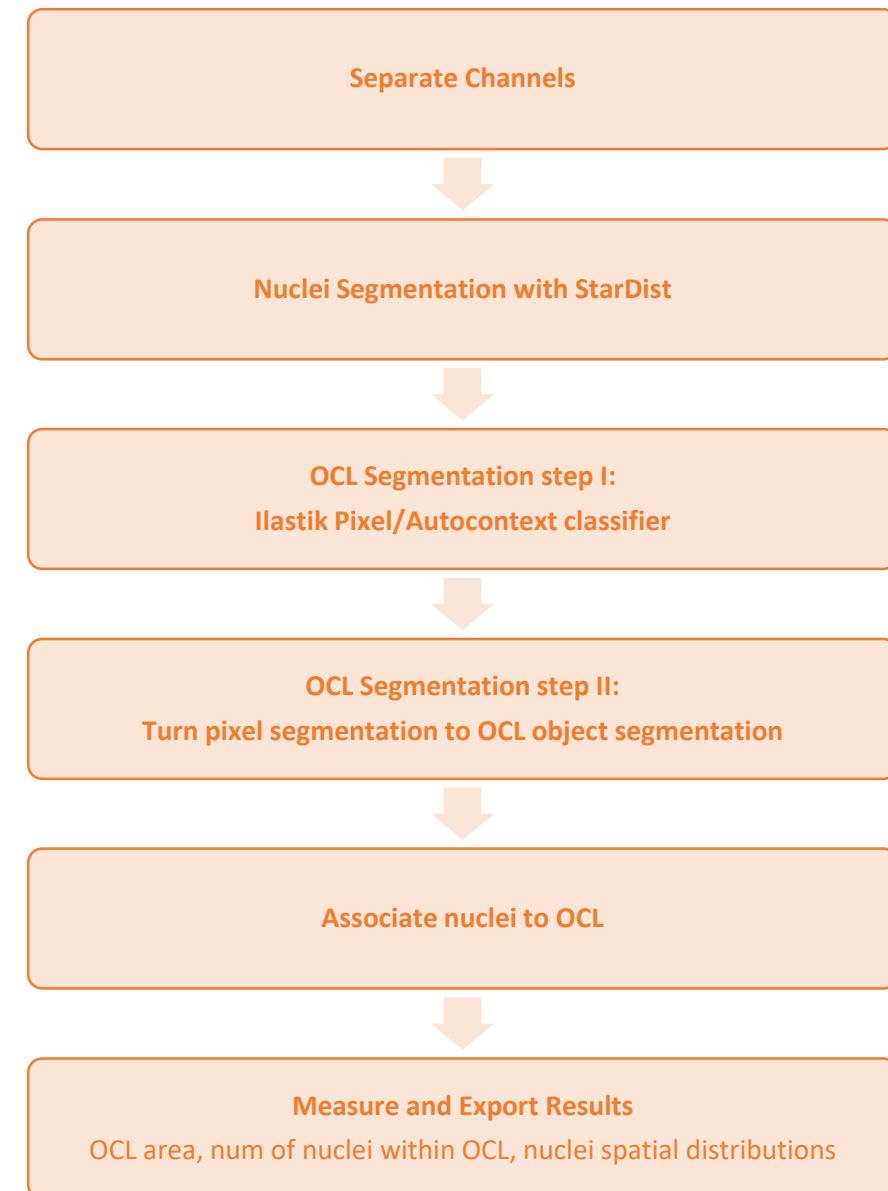
Developed in the group of Fred Hamprecht,
currently in the group of Anna Kreshuk

- Applications:
 - Pixel classification / Autocontext
 - Boundary based object segmentation
 - Object classification
 - Carving
 - Tracking, Run Neural network
- Open source
- Fiji Plugin



ISM Workshop - Outline

- Nuclei segmentation (+ recording)
- Learn Ilastik, train Pixel Classifier for OCL
- Use Ilastik from Fiji
- Pixels → Objects (+ recording)
- Parent-Child relation: BioVoxcel plugin
- Measurements & Quality control
- Automation 1: Recording → Single-file macro
- Automation 2: Processing a whole folder



ISM Workshop - Get Prepared

1. Install Fiji

- a. Download from <https://imagej.net/software/fiji/downloads> and unzip

Caution: “Program Files” not recommended! on Windows. We strongly recommend that you store your directory somewhere in your user space eg C:\Fiji.app rather than C:\Program Files

- b. Create desktop shortcut

2. Install Fiji plugins

- a. Start Fiji
- b. Help > Update...
- c. Click Manage update sites...
- d. Select the following update sites (keep the default ones)
 - I. 3D ImageJ Suite
 - II. BioVoxel
 - III. BioVoxel 3D Box
 - IV. Clij
 - V. Clij2
 - VI. Clijx-assistant
 - VII. Clijx-assistant-extensions
 - VIII. CSBDeep
 - IX. DeepImageJ
 - X. IJPB-plugins
 - XI. Ilastik
 - XII. PTBIOP
 - XIII. StarDist
- e. Click Close
- f. Click Apply Changes
- g. Restart Fiji

3. Update ImageJ

- a. Help>Update ImageJ...
- b. Upgrade to v1.54d
- c. Restart Fiji

4. Install Ilastik

- a. Download Ilastik version 1.4.0
from <https://www.ilastik.org/download.html>
- b. Install the downloaded exe
- c. Create a desktop shortcut

5. Download Workshop Material

to your computer from
The provided link

Image Inspection and Comparison

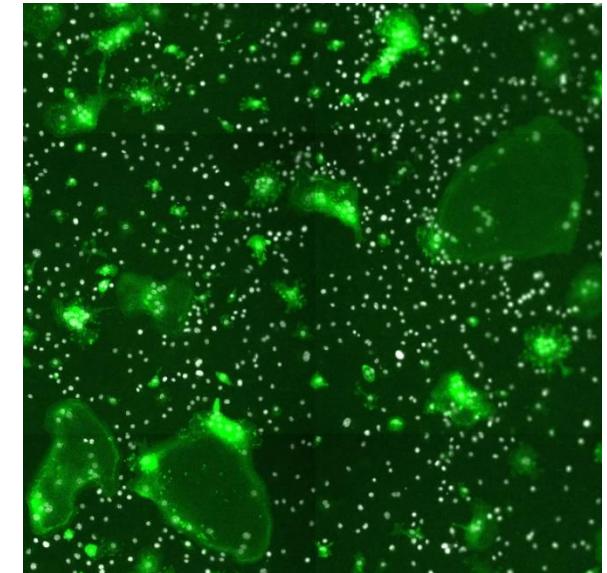
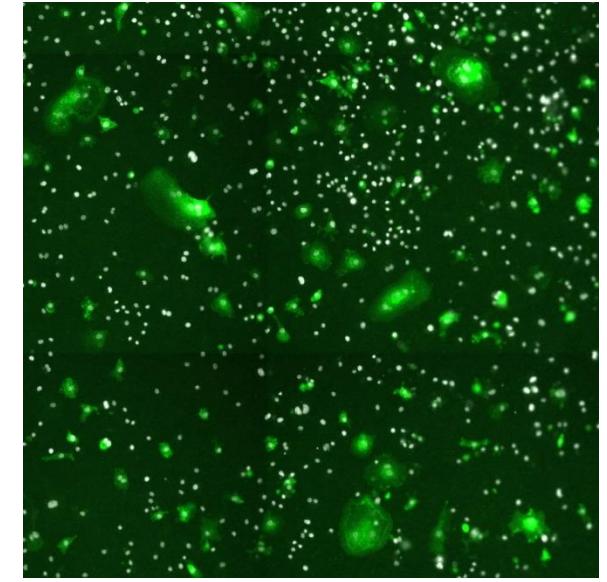
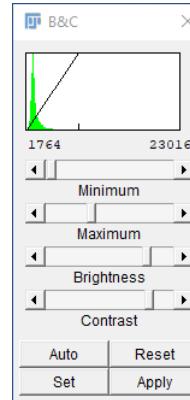
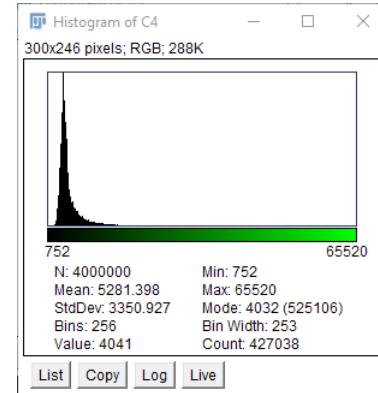
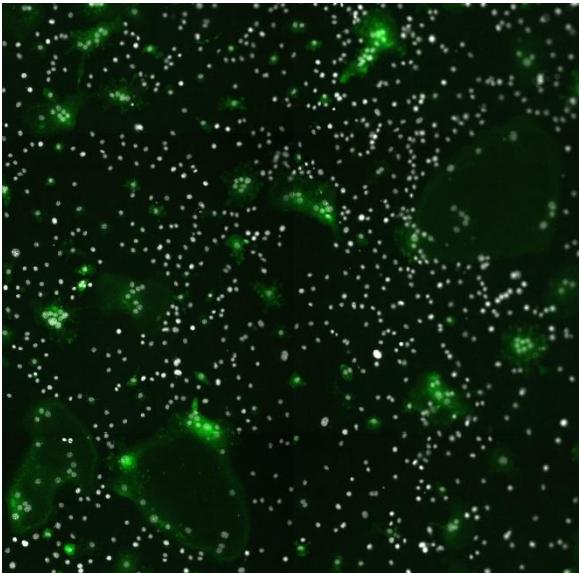
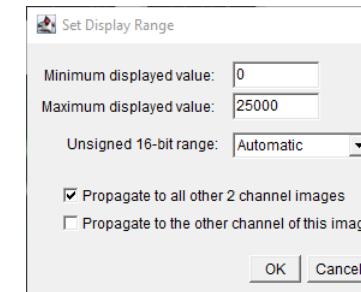
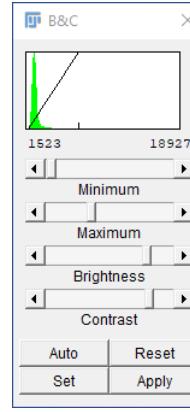
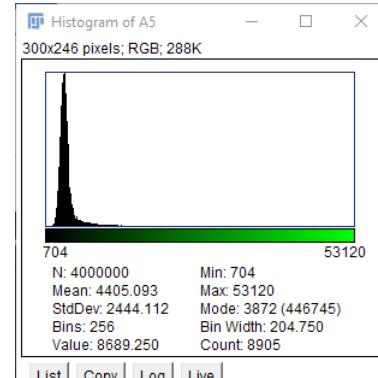
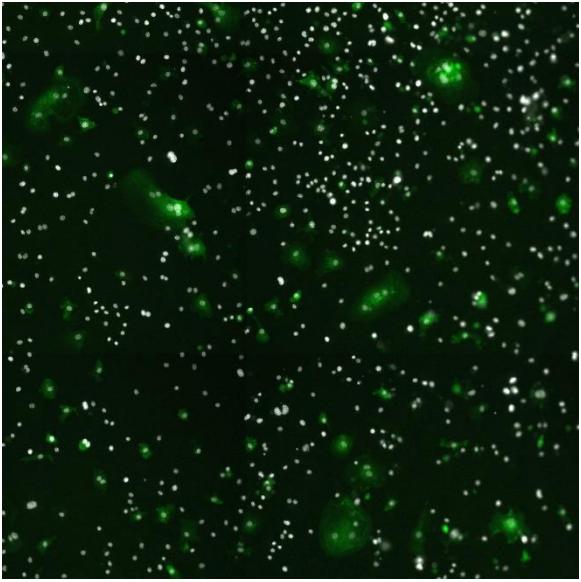


Image: OCL-Origin\{A5, C4}.tif

Ex 1: Segment Nuclei using StarDist

1. Open image in Fiji
2. Start recorder: Plugins>Macros>Record...
3. Duplicate ch2 and rename to Nuc Image>Duplicate or Ctrl+Shift+D
4. Run StarDist: Plugins>StarDist>StarDist 2D
5. Select Label Image, Image>Rename... to Nuclabel
6. Select Nuc image, Show All Rois
7. Flatten
8. Save flatten image A9_Nuc_Overlay.jpg
9. Create draft macro: Create Save it as MySegmentNuc.ijm
10. Inspect Results (on top of original image)
 - Place Nuc and Label Image side-by side,
 Use Analyze>Tools>Synchronize windows
 - Do we miss any nucleus
 - Do touching nuclei merge into a single one
 - Is there a nucleus that get over segmented
 - Inspection of results is extremely important
 - To assist inspection create quality control image
 - (if needed) How can we improve ?

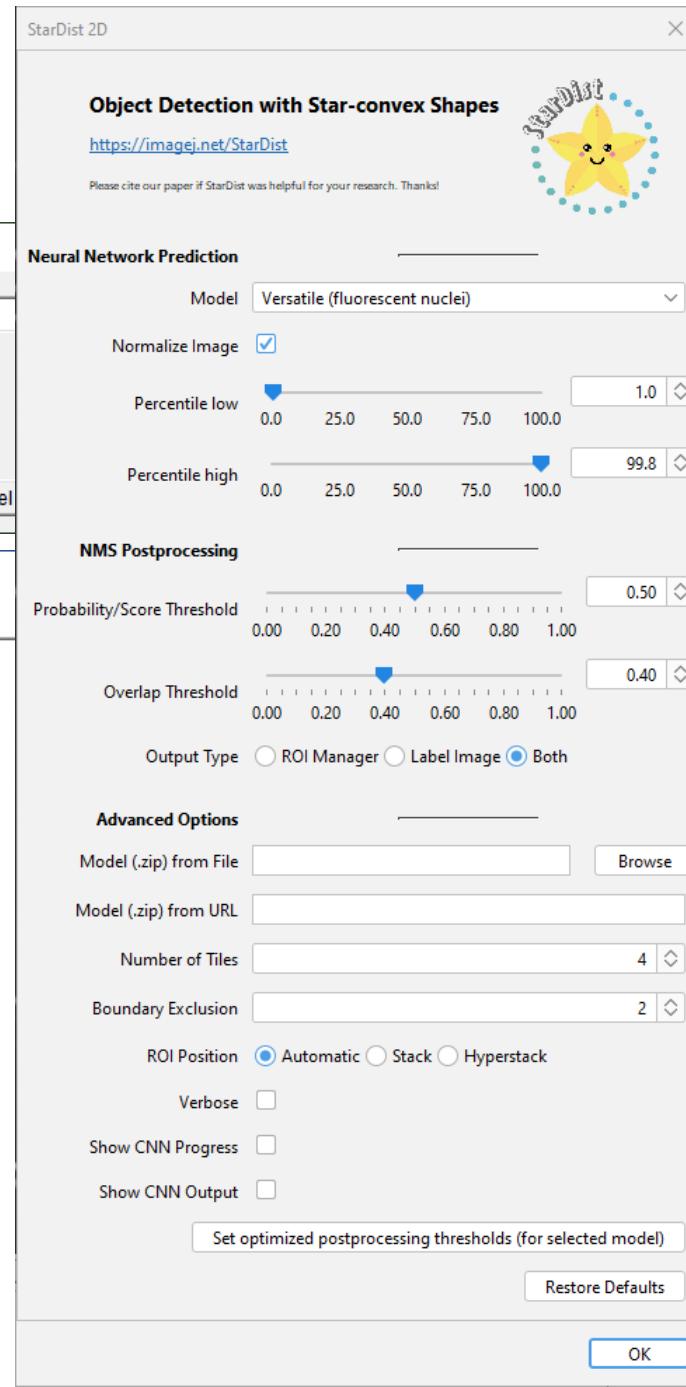
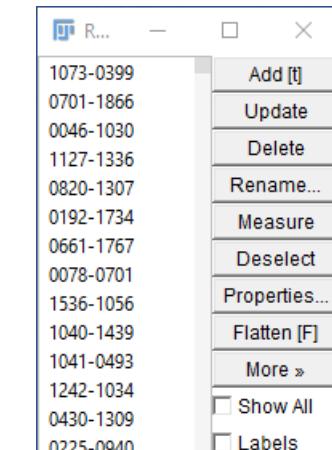
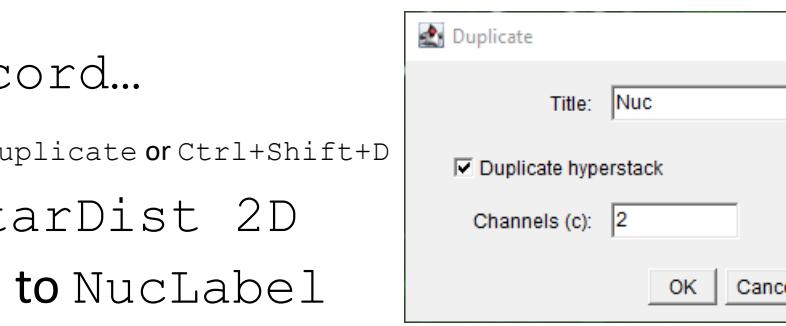


Image: OCL\{A5, A9, B5, C4}.tif

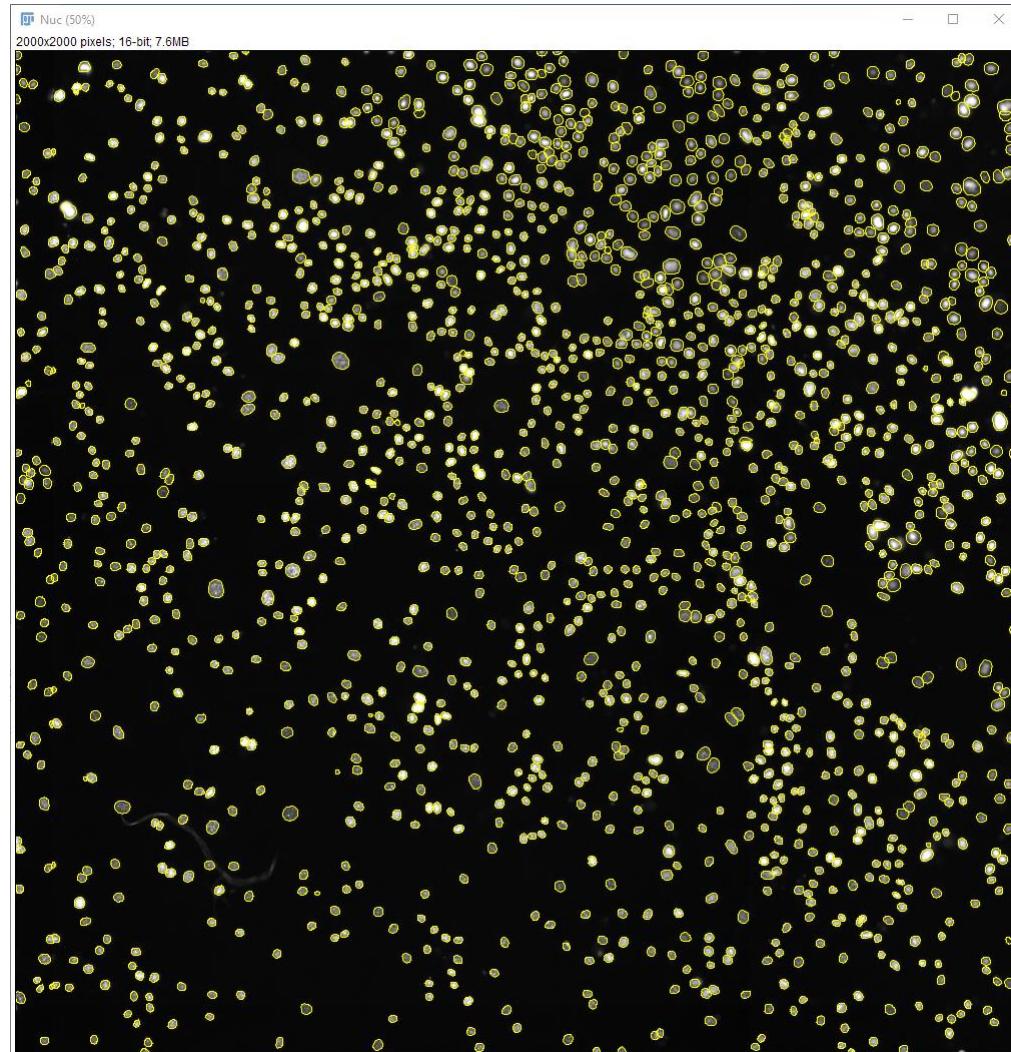
Nuclei Segmentation: Inspect Results

- Select the original image, Show/Hide ROIs
(or place Nuc and Label Image side-by side,
Use Analyze>Tools>Synchronize windows)
- Do we miss any nucleus
- Do touching nuclei merge into a single one
- Is there a nucleus that get over segmented

Inspection of results is extremely important
Do it for all images (at least all conditions)

To assist inspection create quality control image

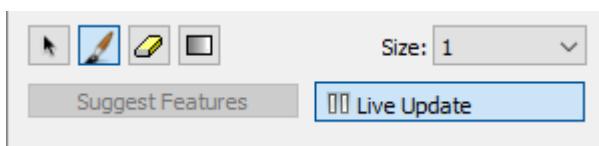
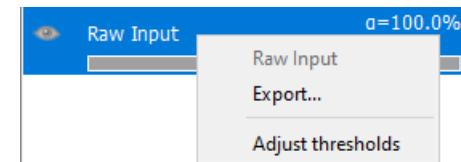
(if needed) How can we improve ?



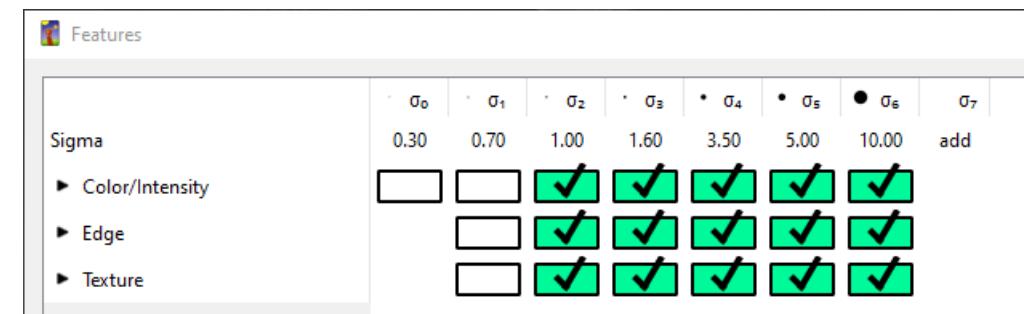
Ex 2: Train Ilastik classifier for OCL

1. Start Ilastik
2. Create New Project: Pixel Classification
3. Add New... Select C1 Images
4. Feature Selection: $\sigma=1-10$
5. Set Training Classes: OCL, BG
6. Interactively, Iteratively:
 - a. Draw / Correct annotations for each class
 - b. Live Update
 - c. Switch images
7. Save Project

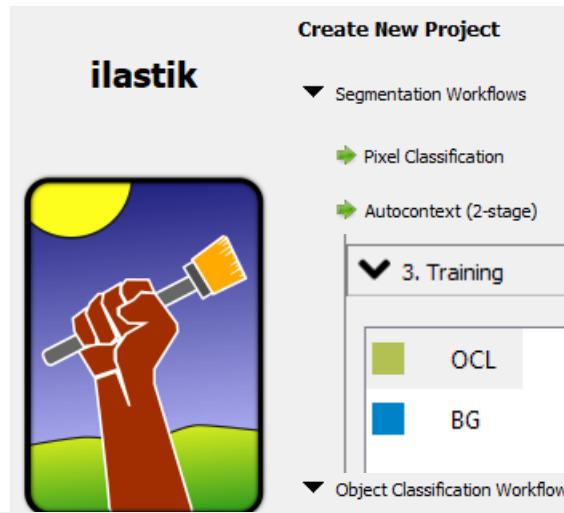
Tips: (de)activate Live Update
choose your drawing budget wisely
adjust contrast
keyboard shortcuts



Raw Data		Prediction Mask	Summary
Nickname	Location		
1 A5_C1	Relative Link: Data\OCL-SingleChannel\A5_C1.tif		
2 A9_C1	Relative Link: Data\OCL-SingleChannel\A9_C1.tif		
3 B5_C1	Relative Link: Data\OCL-SingleChannel\B5_C1.tif		
4 C4_C1	Relative Link: Data\OCL-SingleChannel\C4_C1.tif		
		Add New..	



Images: OCL-SingleChannel\{A5, A9, B5, C4}_C1.tif



Train Ilastik classifier - Good Practices

1. Use multiple images
2. Representative of all conditions
3. Choose where to put your drawing budget:
 - a. Start from few drawings and correct where it makes mistakes
 - b. No need to draw on regions that are correctly recognized
 - c. Put emphasis on the border of objects and gaps between objects, holes can be filled afterward
 - d. The more you draw, the more time it takes to compute (to train the classifier)
 - e. Small brush size
4. Generalization
 - a. Try to get reasonable results on single image (not perfect)
 - b. Switch to other training image(s)
 - c. Go back to the first image(s)
 - d. Test on images not used for training
5. Visualization
 - a. Segmentation / Prediction view / Uncertainty
 - b. Contrast
6. Run-time
 - a. (de)activate Live update
 - b. Zoom-in
 - c. .ilastikrc file
7. How can I improve
 - a. Less / more features (remove small σ / add larger σ - neighborhood)
 - b. Autocontext classifier

It takes time and patience to train a good classifier

Ex 3: Apply Ilastik Classifier from Fiji

1. Open image in Fiji
2. Start recorder: Plugins>Macros>Record...
3. Duplicate ch1 and rename to OCL (Image>Duplicate or Ctrl+Shift+D)
4. Initialize Ilastik: Plugins>Ilastik>configure Ilastik executable location
5. Apply Ilastik classifier: Plugins>Ilastik>Run Pixel Classification Prediction
6. Select Ilastik output image, Rename to ilastikLabelImage
7. Change LUT: Image>Lookup Tables...>glasbey on dark

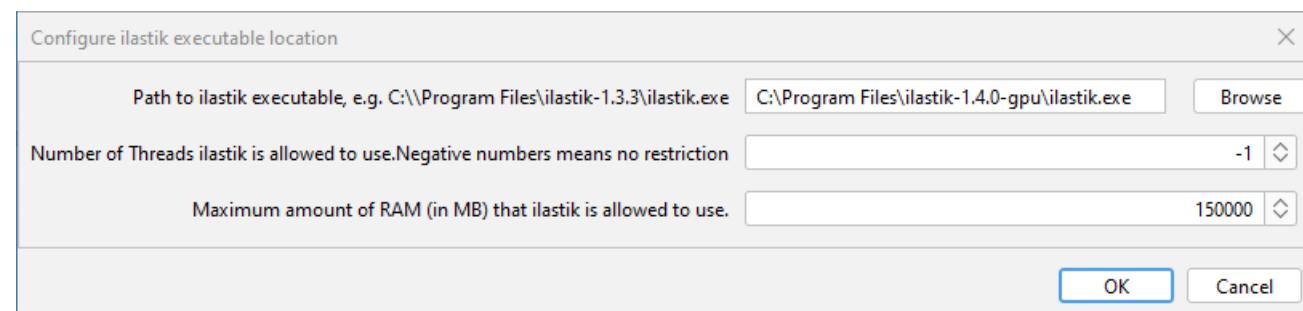
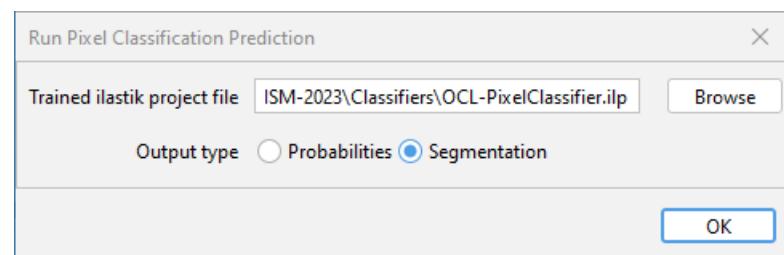
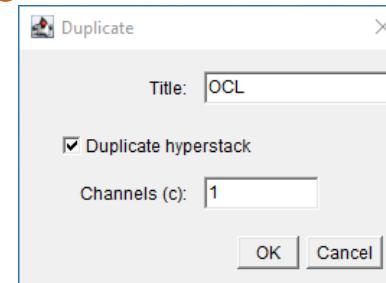


Image: OCL\{A5, A9, B5, C4}.tif

Use Ilastik output to Segment Osteoclasts

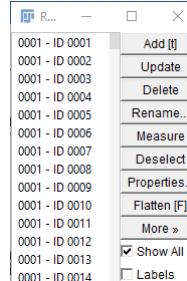
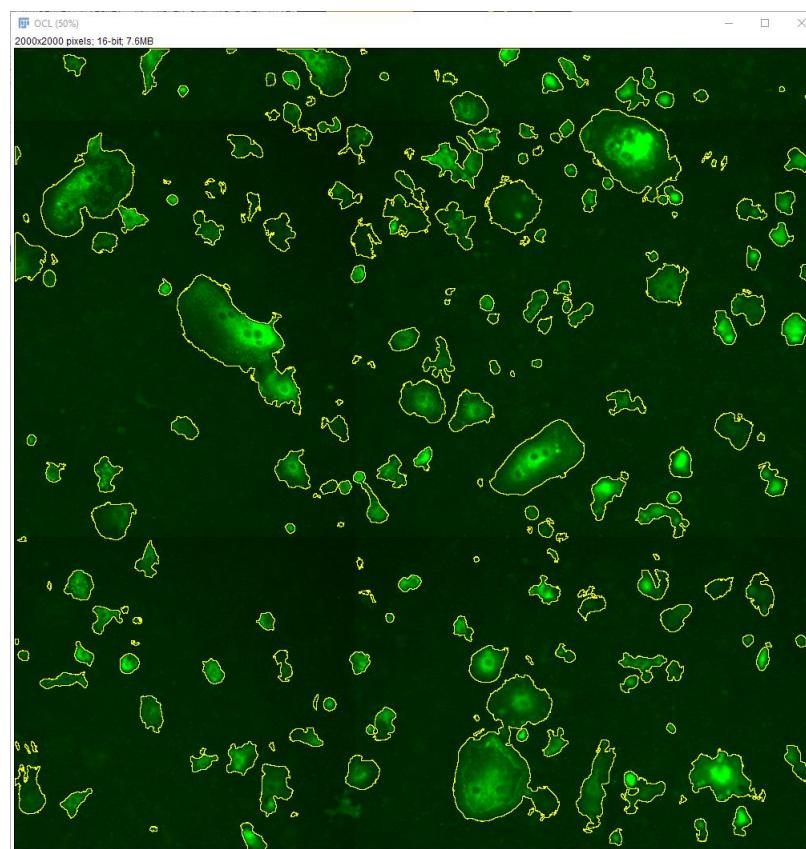
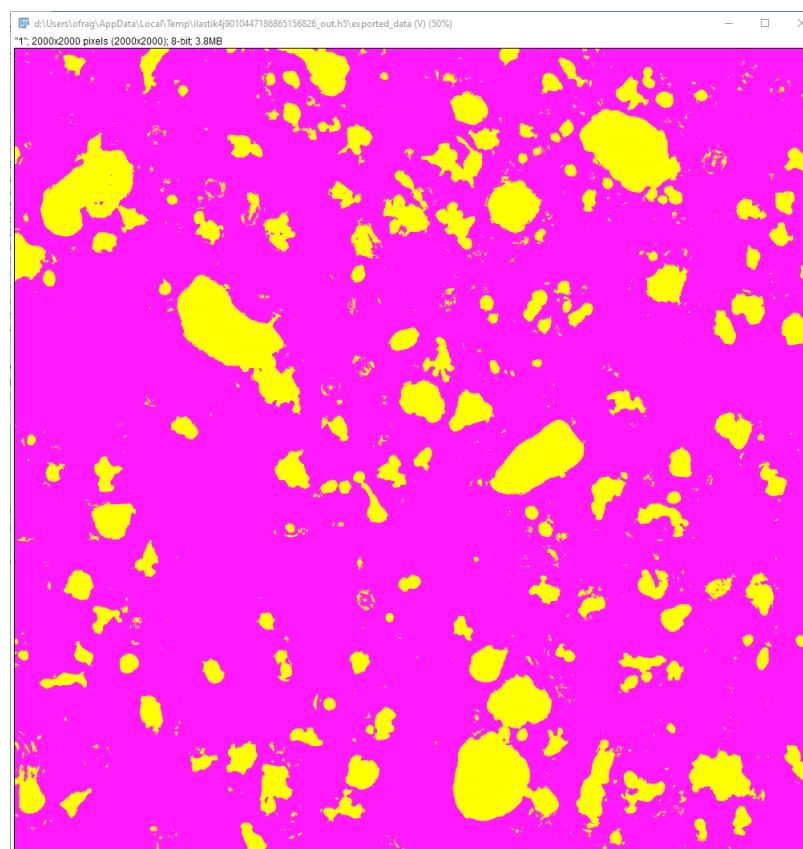
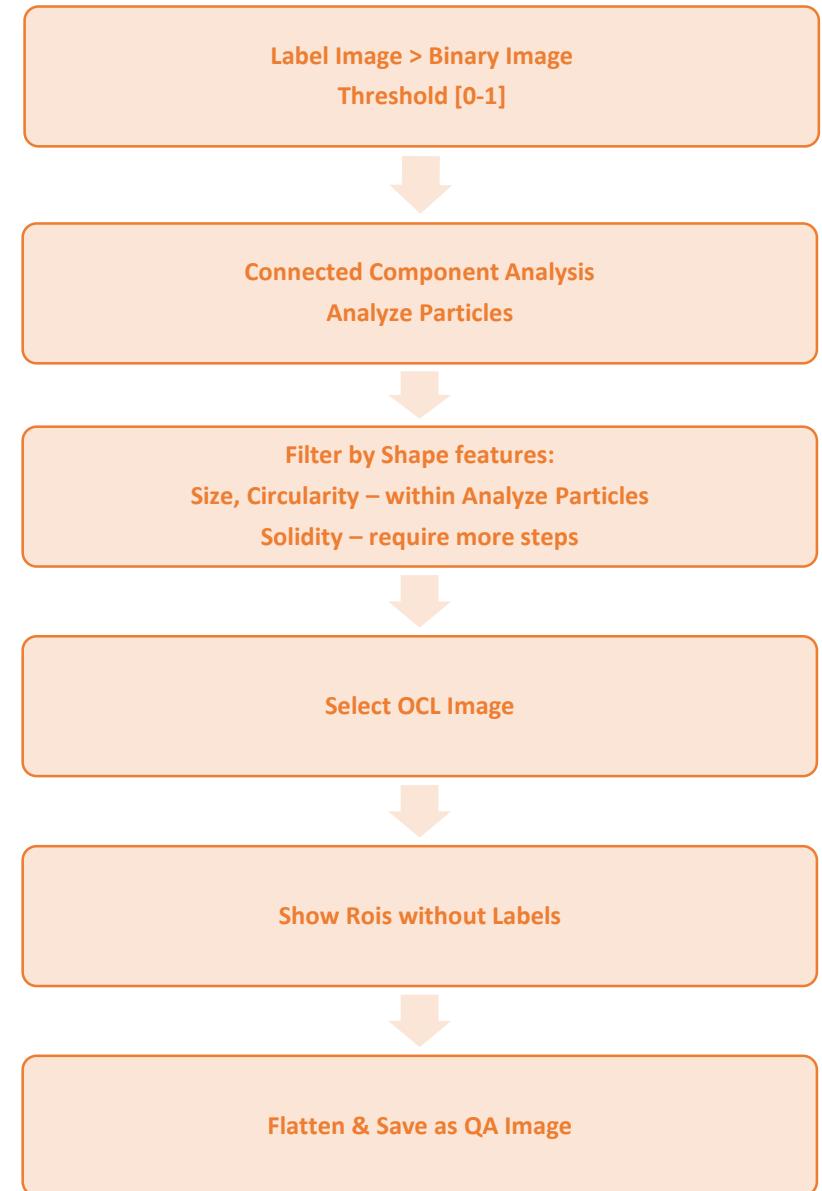
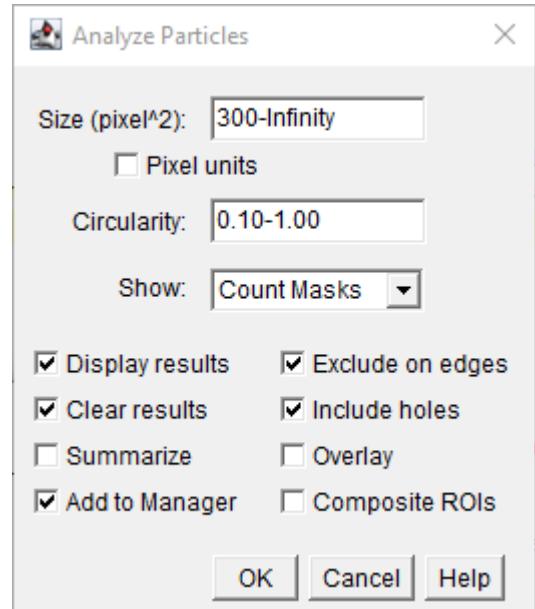
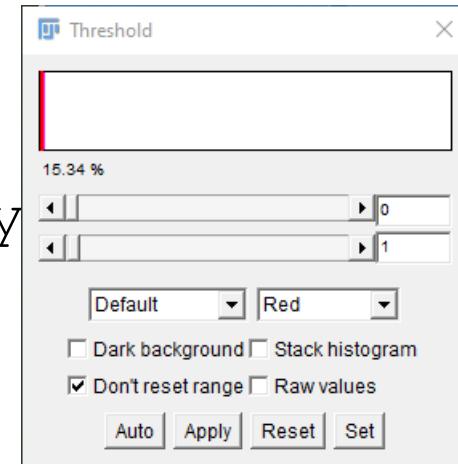


Image: OCL\{A5, A9, B5, C4}.tif



Ex 4: Use Ilastik output to Segment Osteoclasts

1. Convert to Binary mask: Image>Adjust>Threshold...: [0,1] Apply
2. Connected Component Analysis: Analyze>Analyze Particles...
Filter by size and circularity
3. Select the Count Mask image, Image>Rename... to OCL_Label
4. Switch to OCL window
5. From RoiManager : Show All
6. Show Rois
7. Flatten
8. Save flatten image A9_OCL_Overlay.jpg
9. Create draft macro, Save it as MySegmentOCL.ijm
10. Inspect Results (on top of original OCL image)



Inspect Results

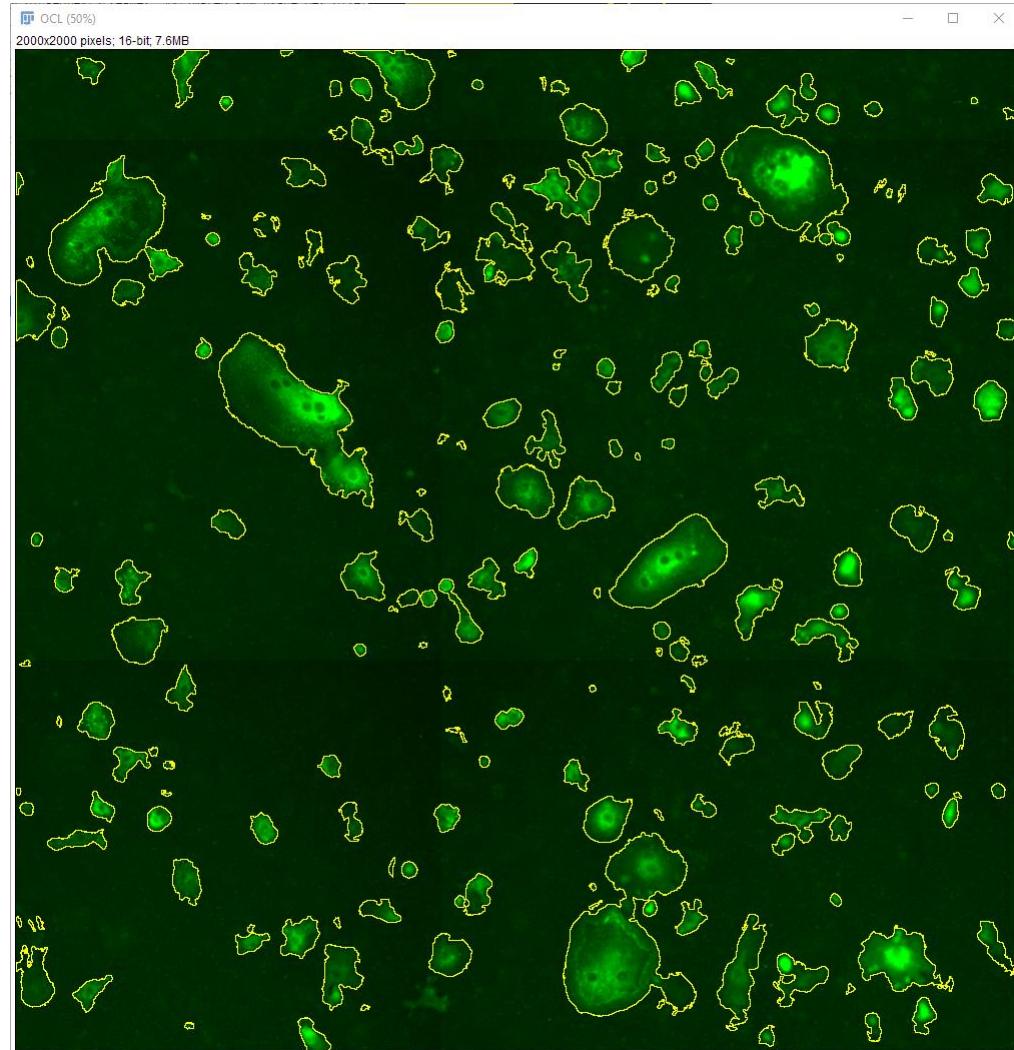
- Do we miss any OCL
- Do touching OCL merge into a single one
- Is there an OCL that get over segmented

- How can we improve ?

Inspection of results is extremely important

Do it for all images (at least all conditions)

To assist inspection create quality control image



Handling Labels and Objects in Fiji

What do we want to do with Objects ?

- Measurements
 - Eg. Intensity, shape
 - Filter objects based on some features
 - Color code display of object measurements
- Selecting objects within a region
- Morphological operations
- Object Classification ? (only through Ilastik ☹)
- Relation between objects:
 - Single type Objects: distances, densities, clustering,
 - Parent Child relations
 - Neighborhood
 - Overlap

Useful Plugins

(beyond Logical operations and scripting):

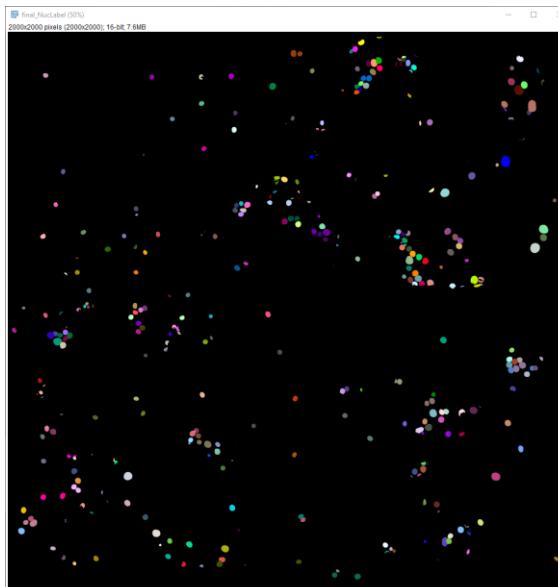
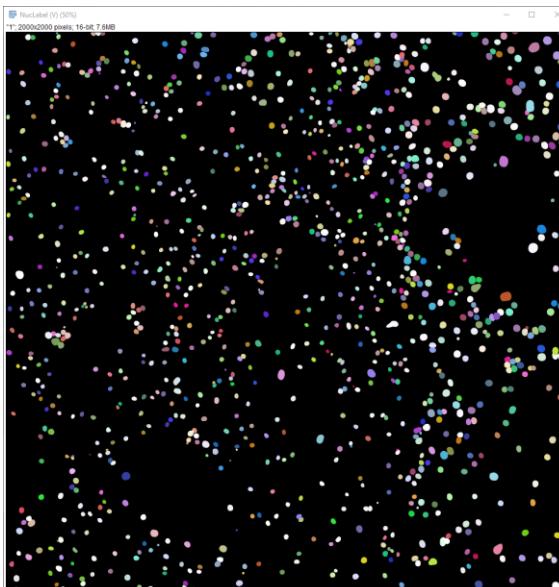
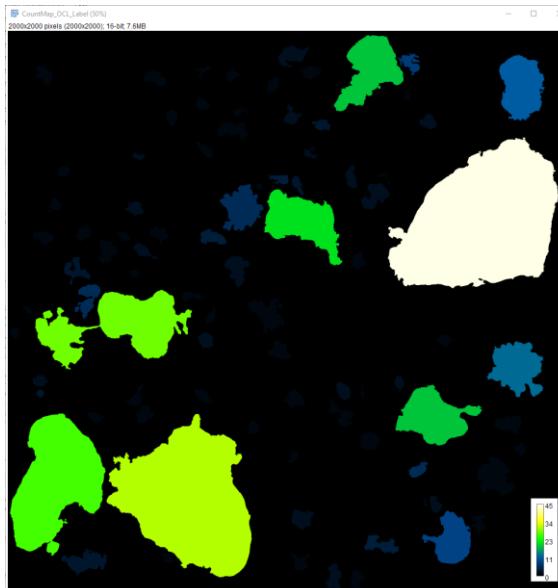
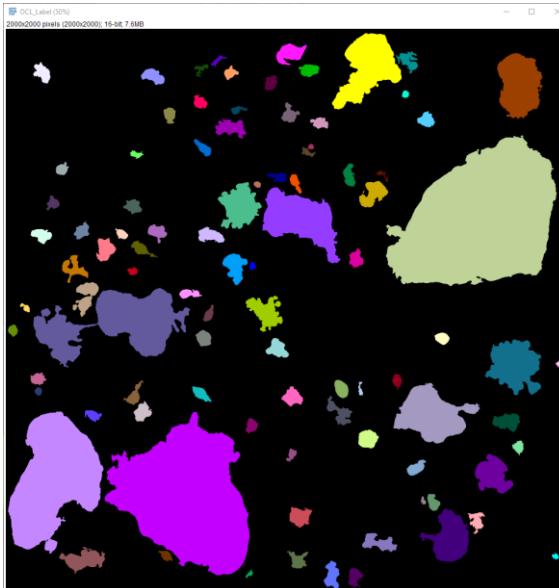
MorphoLibJ

[3D ImageJ Suite](#)

Clij2, ClijX, ClijX-assistant

[BioVoxel](#), [BioVoxels 3D Box](#),

Ex 5: Associate Nuclei to Osteoclasts



Primary_Results				
PRIM_OBJ_ID	SEC_OBJECT_COUNT	VOLUME (pixel ³)	PIXEL_COUNT	MEAN_INTEN
1	19	NaN	34765	15857.540
2	2	NaN	5031	6836.837
3	8	NaN	3052	5007.974
4	12	NaN	27817	6151.456
5	0	NaN	731	6252.498
6	1	NaN	2736	7147.965
7	1	NaN	963	7939.904
8	1	NaN	2392	11340.294
9	1	NaN	1435	8639.398
10	1	NaN	3023	12131.869
11	1	NaN	1853	10011.001
12	1	NaN	527	6009.321
13	1	NaN	1664	6522.346
14	2	NaN	2466	7726.547
15	1	NaN	964	4695.751
16	1	NaN	1848	6773.922
17	2	NaN	2306	6340.850
		NaN	1047	11700.070

One line for
each OCL

Secondary_Results				
PRIMARY_LABEL	IDENTIFIER	VOLUME (pixel ³)	PIXEL_COUNT	MEAN_INTENSITY
1.000	1	NaN	57	6832.842
1.000	2	NaN	336	21216.381
1.000	3	NaN	327	22962.153
1.000	4	NaN	155	28597.058
1.000	11	NaN	424	19587.321
1.000	13	NaN	504	23221.270
1.000	15	NaN	348	17872.782
1.000	18	NaN	372	18568.645
1.000	20	NaN	252	24383.619
1.000	23	NaN	405	17238.993
1.000	29	NaN	362	19426.652
1.000	30	NaN	291	20541.966
1.000	33	NaN	254	22843.969
1.000	35	NaN	392	31182.898
1.000	37	NaN	214	24872.972
1.000	39	NaN	198	32424.808
1.000	40	NaN	48	8903.333
		NaN	211	24800.050

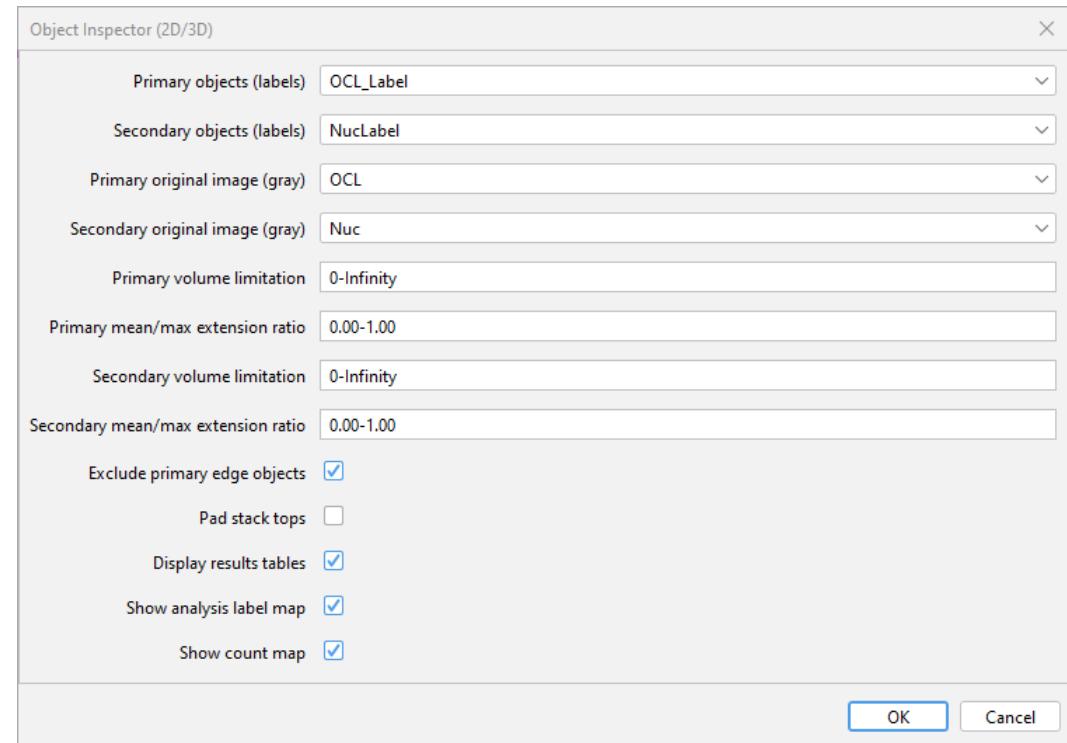
One line for
each Nucleus

Ex 5: Associate Nuclei to Osteoclasts

1. Run BioVoxcel 3D Box>Analysis>Object Inspector (2D/3D)
2. Select CountMap_OCL_Label
3. Show calibration bar:
Analyze>Tools>Calibration Bar...

Catch-up: If Nuclabel, OCL_Label are missing

1. Open image (A9.tif)
2. Run NucSegmentation.ijm
3. Select original image (A9.tif),
4. Tune OCLSegmentation.ijm, Run



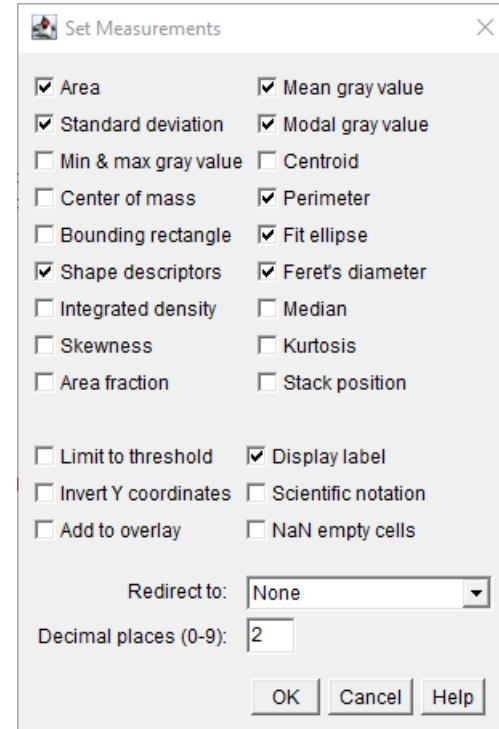
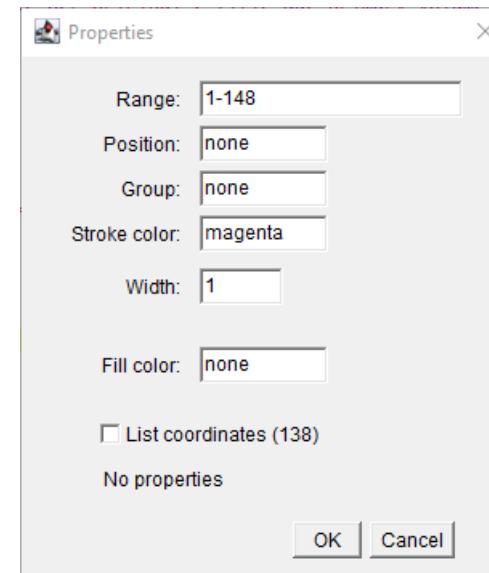
```
var IlastikExeLocation = "C:\\Program Files\\ilastik-1.4.0\\ilastik.exe";
var maxRAM = 24000;          // MB , set to 50-80% of available RAM
var pixelClassifierLocation = "D:\\YourFolder\\YourClassifier.ilp";
```

Save Results

- What do we want to save ?
 - QA images : overlay on top of original images
 - Color-code count image
 - Tables
 - Rois
- Saving Image:
 - On top of original images (not label image) to verify segmentation
 - Pay attention to intensity settings to enable fair image comparison
 - Flatten
 - Choose colors
- Use Image name as prefix to group results together

Ex 6: Save Results

1. Make sure Recorder is activated
2. Select CountMap_OCL_Label
3. Display calibration bar: Analyze>Tools>Calibration Bar..., Flatten, Save
4. Select CountMap_OCL_Label
5. Rename to A9_CountMap_OCL_Label
6. Analyze>Set Measurements...
7. Close Results table if it is open
8. From the RoiManager: click Measure
9. Save Results to A9_Results.csv
10. From the RoiManager: More>Save... A9_FinalOCLRois.zip
11. Select original image (A9.tif), from RoiManager Show All, Flatten
12. Close RoiManager
13. Select final_NucLabel, Plugin>BIOP>Image Analysis>ROIs>Label Image to ROIs
13. From RoiManager: Deselect, Properties...
14. Select the output flatten image of step 10, From the RoiManager: Show All, Flatten
15. Save new flatten image to A9_FinalOverlay.jpg
16. From the RoiManager: More>Save... A9_FinalNucRois.zip
17. From the Recorder: Create macro



Automate Your Work: Guidelines

- While recording rename temp images to fixed names: eg Nuc, IlastikLabelImage
- Record everything, including results saving
- Start with recorded code : from the recorder click Create
- Replace hard-coded values by variables (var): Ilastik path, classifier path and name, all numbers (min/max values)
- Store original image name and path for further image saving : getTitle
- Remove redundant lines
- Try to run on the same image and debug
- Try to run on another image
- Optionally pack in a function (ProcessFile)
- Once you have working code for single image, Use template code example for going over all images
- Add documentation

Automate Your Work: SegmentNuc

```
1 run("Duplicate...", "title=Nuc duplicate channels=2");
2 selectWindow("Nuc");
3 run("Command From Macro", "command=[de.csbdresden.stardist.StarDist2D], args=['input':'Nuc', 'modelChoice':'Versatile (fluorescent nuclei)', 'normalizeInput':'true', 'percentile'];
4 selectWindow("Label Image");
5 rename("NucLabel");
6 selectWindow("Nuc");
7 roiManager("Show All without labels");
8 run("Flatten");
9 saveAs("Jpeg", "A9_NucOverlay.jpg");
10
```

hard-coded values > variables
Store image name

```
1 // Get Image name
2 origName = getTitle();
3 origNameNoExt = File.getNameWithoutExtension(origName);
4
5 run("Duplicate...", "title=Nuc duplicate channels=2");
6 selectWindow("Nuc");
7 run("Command From Macro", "command=[de.csbdresden.stardist.StarDist2D], args=['input':'Nuc', 'modelChoice':'Versatile (fluorescent nuclei)', 'normalizeInput':'true', 'percentile'];
8 selectWindow("Label Image");
9 rename("NucLabel");
10 selectWindow("Nuc");
11 roiManager("Show All without labels");
12 run("Flatten");
13 saveAs("Jpeg", origNameNoExt+"NucOverlay.jpg");
14
```

```

1 run("Duplicate...", "title=OCL duplicate channels=1");
2 run("Configure ilastik executable location", "executablefile=[C:\\Program Files\\ilastik-1.4.0-gpu\\ilastik.exe] numthreads=-1 maxrammb=150000");
3 run("Run Pixel Classification Prediction", "projectfilename=[D:\\Ofra_Sync\\Docs\\BioImaging Training\\ISM-2023\\Classifiers\\OCL-PixelClassifier.ilp] inputimage=OCL pixelclassificationtype=Segmentation");
4 selectWindow("d:\\Users\\ofrag\\AppData\\Local\\Temp\\ilastik4j9010447186865156826_out.h5\\exported_data");
5 rename("ilastikLabelImage");
6 run("glasbey on dark");

7
8 setAutoThreshold("Default no-reset");
9 //run("Threshold...");
10 //setThreshold(0, 1);
11 run("Convert to Mask");
12 selectWindow("OCL");

13
14 run("Set Measurements...", "area mean standard modal perimeter fit shape feret's display redirect=None decimal=2");
15 run("Analyze Particles...", "size=300-Infinity circularity=0.1-1.00 show=[Count Masks] display exclude clear include add");
16 selectWindow("Count Masks of ilastikLabelImage");
17 rename("OCL_Label");
18 run("Fire");

19
20 selectWindow("OCL");
21 roiManager("Show All without labels");
22 run("Flatten");
23 saveAs("Jpeg", "A9_OCLOverlay.jpg");

```

hard-coded values > variables
Store image name

```

1 var IlastikExeLocation = "C:\\Program Files\\ilastik-1.4.0-gpu\\ilastik.exe";
2 var maxRAM = 24000; // MB , set to 50-80% of available RAM
3 var pixelClassifierLocation = "D:\\Ofra_Sync\\Docs\\BioImaging Training\\ISM-2023\\Classifiers\\OCL-PixelClassifier.ilp";
4
5 var minOCLSize = 300;
6 var minOCLCircularity = 0.1;

7
8 // Get Image name
9 origName = getTitle();
10 origNameNoExt = File.getNameWithoutExtension(origName);
11
12 run("Duplicate...", "title=OCL duplicate channels=1");
13 run("Configure ilastik executable location", "executablefile=[+IlastikExeLocation+] numthreads=-1 maxrammb="+maxRAM);
14 run("Run Pixel Classification Prediction", "projectfilename=[+pixelClassifierLocation+] inputimage=OCL pixelclassificationtype=Segmentation");
15 rename("ilastikLabelImage");
16 run("glasbey on dark");

17
18 setAutoThreshold("Default no-reset");
19 //setThreshold(0, 1);
20 run("Convert to Mask");
21 roiManager("reset");
22 run("Set Measurements...", "area mean standard modal perimeter fit shape feret's display redirect=None decimal=2");
23 run("Analyze Particles...", "size="+minOCLSize+"-Infinity circularity="+minOCLCircularity+"-1.00 show=[Count Masks] display exclude clear include add");
24 selectWindow("Count Masks of ilastikLabelImage");
25 rename("OCL_Label");
26 run("Fire");

27
28 selectWindow("OCL");
29 roiManager("Show All without labels");
30 run("Flatten");
31 saveAs("Jpeg", origNameNoExt+"_OCLOverlay.jpg");

```

Macro: SegmentOCL-recorded.ijm > SegmentOCL.ijm

Automate Your Work: Combine into full workflow

```
1 /*  
2  * Macro for Quantification of Osteoclasts (OCL) and the number of Nuclei within each of them  
3  * for ISM-2023 workshop on "Incorporating machine learning tools into image analysis workflows using Fiji, Ilastik and StarDist"  
4  *  
5  * By: Ofra Golani, MICC-Cell Observatory, Weizmann Institute of Science  
6  *  
7  * This macro process single composite image, ch1 for OCL, ch2 for Nuc  
8  * Related images are courtesy of Sabina Winograd-Katz and Benny Geiger  
9  */  
10  
11 var IlastikExeLocation = "C:\\Program Files\\ilastik-1.4.0\\ilastik.exe";  
12 var maxRAM = 24000; // MB , set to 50-80% of available RAM  
13 var pixelClassifierLocation = "C:\\ISM-2023\\Classifiers\\OCL-PixelClassifier.ilp";  
14  
15 var minOCLSize = 300;  
16 var minOCLCircularity = 0.1;  
17  
18 var outSubDir = "Results";  
19  
20 var setFixCountRange = 0; // Use 1 to set fixed range of count color map  
21 var maxCountDisplay = 45;  
22 var cleanupFlag = 1;  
23  
24 // Get Image name  
25 origName = getTitle();  
26 origNameNoExt = File.getNameWithoutExtension(origName);  
27  
28 // Create output folder  
29 inDir = File.directory;  
30 outDir = inDir + File.separator + outSubDir + File.separator;  
31 File.makeDirectory(outDir);  
32  
33 // Segment Nuc  
34 run("Duplicate...", "title=Nuc duplicate channels=2");  
35 selectWindow("Nuc");  
36 run("Command From Macro", "command=[de.csbdresden.stardist.StarDist2D], args=['input':'Nuc', 'modelChoice':'Versatile (fluorescent nuclei)', 'normalizeInput':'true', 'percentileBottom':'1.0', 'percentileTop:  
37 selectWindow("Label Image");  
38 rename("NucLabel");  
39 selectWindow("Nuc");  
40 //roiManager("Show All");  
41 roiManager("Show All without labels");  
42 run("Flatten");  
43 saveAs("Jpeg", outDir+origNameNoExt+"_AllNucOverlay.jpg");  
44  
45  
46 // Segment OCL  
47 selectWindow(origName);  
48 run("Duplicate...", "title=OCL duplicate channels=1");
```

Combine all pieces together

Store image name

Create output folder

Save QA images in output folder

Add documentation

Macro: SegmentOCL-SingleFile.ijm

Automate Your Work: Combine into full workflow

```
46 // Segment OCL
47 selectWindow(origName);
48 run("Duplicate...", "title=OCL duplicate channels=1");
49 run("Configure ilastik executable location", "executablefile=[+IlastikExeLocation+] numthreads=-1 maxrammb="+maxRAM);
50 run("Run Pixel Classification Prediction", "projectfilename=[+pixelClassifierLocation+] inputimage=OCL pixelclassificationtype=Segmentation");
51 rename("ilastikLabelImage");
52 run("glasbey on dark");
53
54 setAutoThreshold("Default no-reset");
55 //setThreshold(0, 1);
56 run("Convert to Mask");
57 roiManager("reset");
58 run("Set Measurements...", "area mean standard modal perimeter fit shape feret's display redirect=None decimal=2");
59 run("Analyze Particles...", "size="+minOCLSize+"-Infinity circularity="+minOCLCircularity+"-1.00 show=[Count Masks] display exclude clear include add");
60 selectWindow("Count Masks of ilastikLabelImage");
61 rename("OCL_Label");
62 run("Fire");
63
64 selectWindow("OCL");
65 roiManager("Show All without labels");
66 run("Flatten");
67 saveAs("Jpeg", outDir+origNameNoExt+"_AllOCLOverlay.jpg");
68
69 // Associate Nuc with OCL
70 run("Object Inspector (2D/3D)", "primary_imageplus=OCL_Label secondary_imageplus=NucLabel original_1_title=OCL original_2_title=Nuc primary_volume_range=0-Infinity primary_mmer_range=0.00-1.00 secondary_vo
71
72 // Save Count Map with overlay (to show OCL with zero count)
73 selectWindow("CountMap_OCL_Label");
74 //selectWindow(origNameNoExt+"_CountMap_OCL");
75 roiManager("Show None");
76 roiManager("Show All");
77 if (setFixCountRange) setMinAndMax(0, maxCountDisplay);
78 run("Calibration Bar...", "location=[Lower Right] fill=White label=Black number=5 decimal=0 font=12 zoom=2 overlay");
79 run("Flatten");
80 saveAs("Jpeg", outDir+origNameNoExt+"_CountMap_OCL.jpg");
81
82 // Save Table of shape features & count (Mode)
83 selectWindow("CountMap_OCL_Label");
84 rename(origNameNoExt+"_CountMap_OCL"); // rename to have the image name in the table
85 run("Clear Results");
86 run("Set Measurements...", "area modal perimeter fit shape feret's display redirect=None decimal=2");
87 roiManager("Measure");
88 selectWindow("Results");
89 saveAs("Results", outDir+origNameNoExt+"_Results.csv");
90
91 // Save OCL Rois
92 roiManager("Save", outDir+origNameNoExt+"_OCLRoiSet.zip");
93
```

Combine all pieces together
Store file name
Create output folder
Save QA images in output folder
Add documentation
Save Tables in output folder
Save Rois in output folder

Macro: SegmentOCL-SingleFile.ijm

Automate Your Work: Combine into full workflow

```
93 // Final QA Image
94 selectWindow(origName);
95 roiManager("Show All");
96 run("Flatten");
97 rename("tmpFlatten");
98
99 // replace OCL Rois with Final Nuc Rois
100 roiManager("reset");
101 selectWindow("final_NucLabel");
102 run("Label image to ROIs");
103 selectWindow("tmpFlatten");
104 roiManager("Set Color", "magenta");
105 roiManager("Set Line Width", 0);
106 roiManager("Show All");
107 run("Flatten");
108 saveAs("Jpeg", outDir+origNameNoExt+"_FinalOverlay.jpg");
109
110 // Save final Nuc Rois
111 roiManager("Save", outDir+origNameNoExt+"_FinalNucRoiSet.zip");
112
113 // Cleanup
114 if (cleanupFlag)
115 {
116     // close all image windows
117     run("Close All");
118
119     // reset RoiManager
120     roiManager("reset");
121
122     // close tables
123     selectWindow("Primary_Results");
124     run("Close");
125     selectWindow("Secondary_Results");
126     run("Close");
127     //selectWindow(outDir+origNameNoExt+"_Results.csv");
128     selectWindow("Results");
129     run("Close");
130 }
131 }
```

Combine all pieces together
Store file name
Create output folder
Save QA images in output folder
Add documentation
Save Tables in output folder
Save Rois in output folder
Create Final QA image
Cleanup

Create a macro for whole folder: Guidelines

- From the script editor: Templates>ImageJ1.x>Batch>Process Folder (ImageJ Macro)
- Open the single file macro (QuantifyOCL-singleFile.ijm)
- Copy the variables sections to just above the processFolder call
- Copy the rest of the code into the processFile function
- Fix indentation (select all and use Tab to indent inside)
- Add command for opening the file

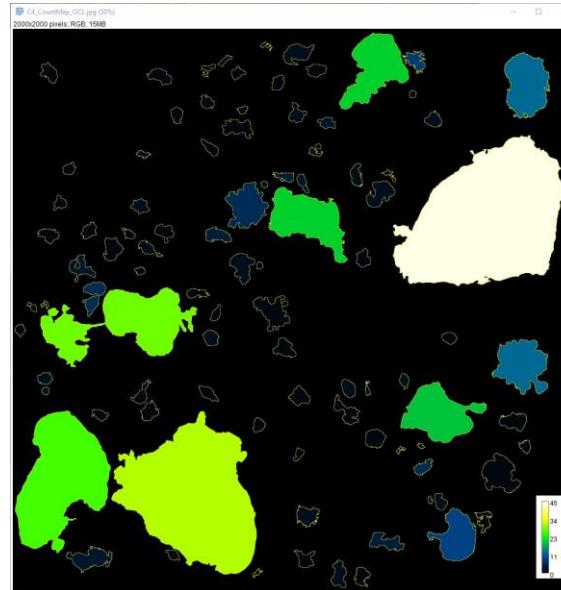
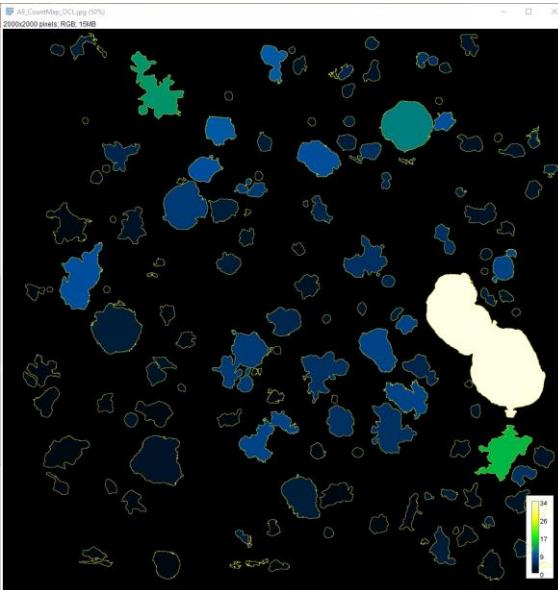
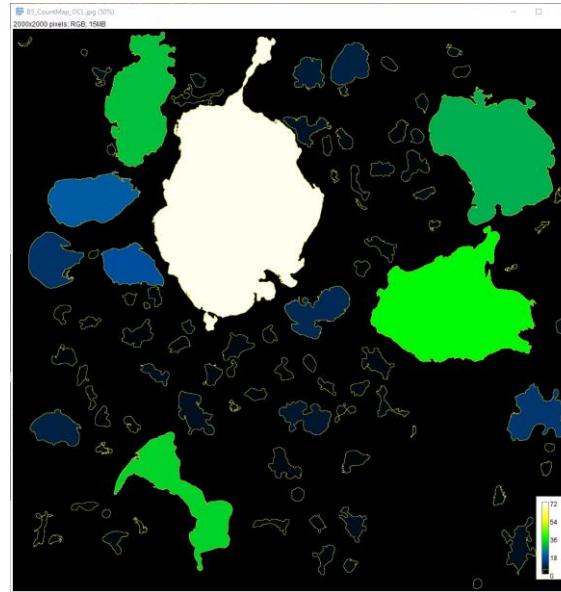
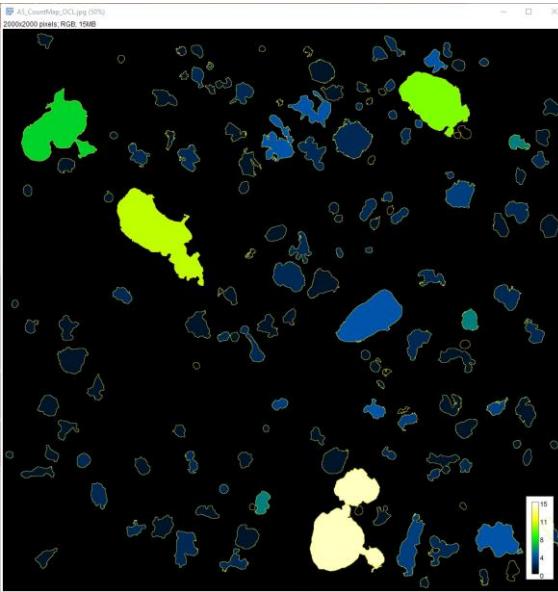
```
open(input + File.separator + file);
```

Ex 7: Apply to all images

1. Drag and drop the macro QuantifyOCL-WholeFolder.ijm into Fiji
2. Click Run
3. When prompted, select the folder in which all the images are
4. Results will appear in Results subfolder

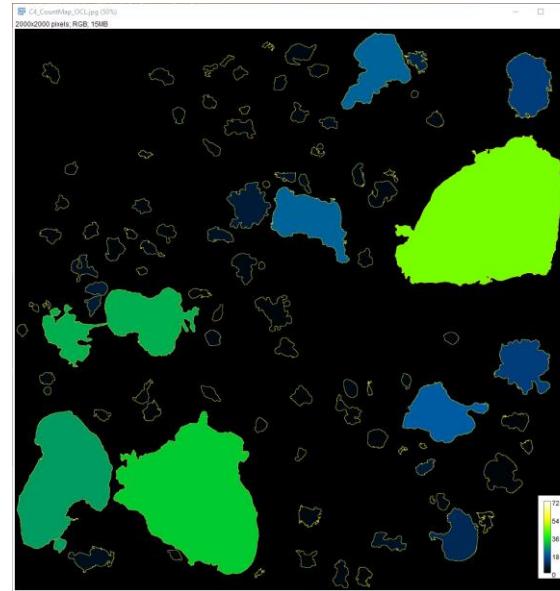
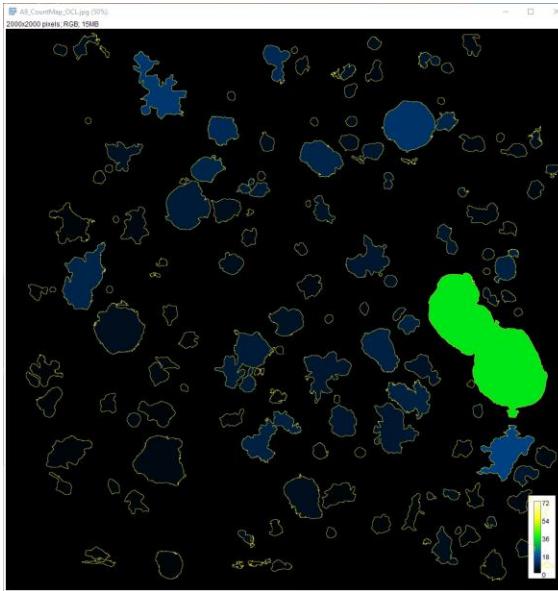
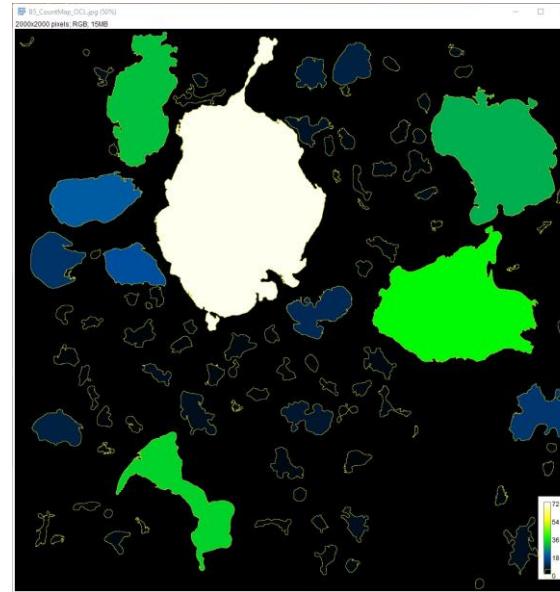
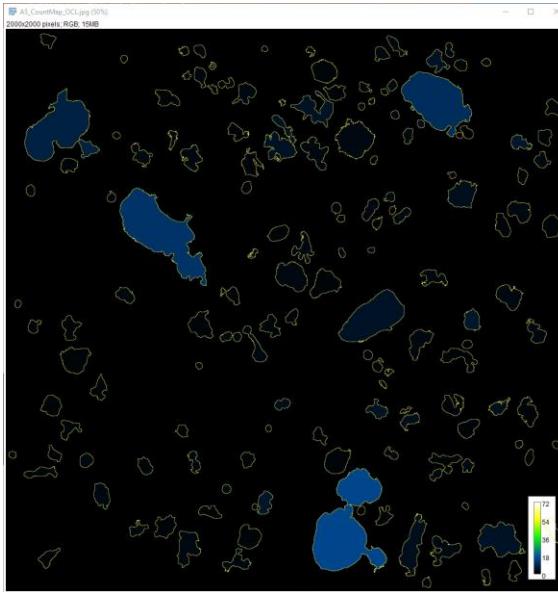
5. Compare color coded count maps.
How can we make fair comparison?

Comparing Results



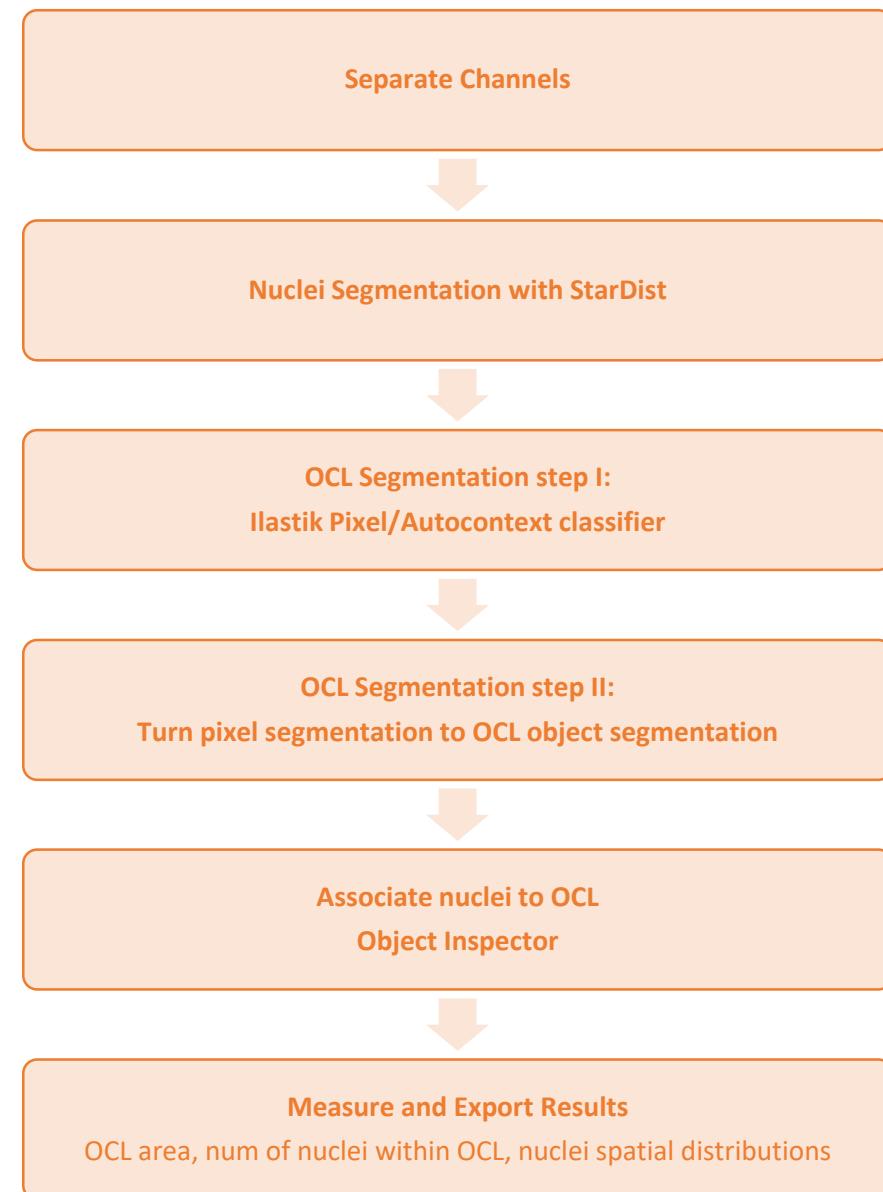
What is wrong ?

Comparing Results



Recap

- Identify your objects
 - Choose proper segmentation tool
 - Many segmentation tools available
- Why did we choose StarDist and Ilastik ?
- Can we manage only with StarDist and Ilastik ?
- Quality control images & Inspection of results !!!
- Tune parameters on multiple images / conditions
- Automation ...
- How can I apply this ?
 - Where do I start with my own images ?



Reproducible Science: Document and Share

Describe the workflow in details

“Image analysis done with Fiji” is not acceptable

Share code with sample images and classifiers

Github is a good option

Cite all the tools you used

In this case: Fiji, StarDist, Ilastik, BioVoxcel, Clij

Consider sharing your data,

Zenodo for small dataset or

IDR / BioImage Archive for larger datasets

The screenshot shows a GitHub repository page for 'Intestinal Barrier Function'. The repository has 35 commits from 'ofrag' on Nov 8, 2020. It contains files like README.md, LICENSE, README.jm, and TightJunctionAnalysis.jm. The 'About' section describes the project as 'Morphological analysis of Intestinal Barrier Function based on images of epithelial tight junction (TJ)'. The 'Overview' section includes a list of metrics: Shape of single cells: Area, Perimeter, Solidity, Roughness = $\text{Perim}^2 / (4 * \pi * \text{Area})$; Shape of edges between adjacent cells: Length, Euclidean distance and Straightness of the independent TJ elements. A 'Workflow' section shows two rows of images: the top row shows grayscale images of intestinal tissue with overlaid black outlines; the bottom row shows colored segmentation maps (blue, purple, yellow, orange) of the same tissue regions.

Acknowledgement and Citation

Fiji:

Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., ... Cardona, A. (2012). Fiji: an open-source platform for biological-image analysis. *Nature Methods*, 9(7), 676–682. [doi:10.1038/nmeth.2019](https://doi.org/10.1038/nmeth.2019)

Ilastik:

interactive machine learning for (bio)image analysis

Stuart Berg, Dominik Kutra, Thorben Kroeger, Christoph N. Straehle, Bernhard X. Kausler, Carsten Haubold, Martin Schiegg, Janez Ales, Thorsten Beier, Markus Rudy, Kemal Eren, Jaime I Cervantes, Buote Xu, Fynn Beuttenmueller, Adrian Wolny, Chong Zhang, Ullrich Koethe, Fred A. Hamprecht & Anna Kreshuk
in: Nature Methods, (2019) [Link at publisher](#), [BibTex file](#)

StarDist:

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.

BioVoxel

<https://biovoxxel.github.io/bv3dbox/>, doi: 10.5281/zenodo.7691609

MorphoLibJ

Legland, D., Arganda-Carreras, I., & Andrey, P. (2016). MorphoLibJ: integrated library and plugins for mathematical morphology with ImageJ. *Bioinformatics*, 32(22), 3532–3534. [doi:10.1093/bioinformatics/btw413](https://doi.org/10.1093/bioinformatics/btw413)

[MorphoLibJ](#)'s code repository has its own [DOI](#).

Clij2

Robert Haase, Loic Alain Royer, Peter Steinbach, Deborah Schmidt, Alexandr Dibrov, Uwe Schmidt, Martin Weigert, Nicola Maghelli, Pavel Tomancak, Florian Jug, Eugene W Myers. *CLIJ: GPU-accelerated image processing for everyone*. [Nat Methods 17, 5-6 \(2020\) doi:10.1038/s41592-019-0650-1](https://doi.org/10.1038/s41592-019-0650-1)