



Illustration reprinted from Pete Bankhead.

Introduction to Bioimage Analysis using QuPath

Antoine A. Ruzette & Simon F. Nørrelykke

Image Analysis Collaboratory



Get the course materials

<https://hms-iac.github.io/qupath-workshop>

One-stop resource for everything we'll cover today

Let's download an example image

1. Browse to the workshop website >
2. Download the .vsi whole-slide image
3. Once done, unzip it
4. Save it
4. Right-click on the installer file > Open > Confirm Open

Workshop plan

1. Introduction to digital image analysis
2. Installing QuPath and your first project
3. GUI layout and toolbars
4. Introducing objects: annotations and detections
5. Saving, sharing and receiving QuPath projects
6. Nuclei detection and measurements (incl. StarDist)
7. Cell classification
8. Automating tissue annotations (pixel classifier)
9. Advance topic: scripting and workflows

Acknowledgments

- **Pete Bankhead** et al.
 - QuPath and its amazing documentation
- **Peter Sobolewski**
 - *Introduction to QuPath* workshop at the The Jackson Laboratory
- **Nina Kozlova**
 - Whole-slide image used in this workshop

Self-introductions

1. My **name** is *Antoine*
2. My **position** is as an *Associate in Systems Biology*
3. My **lab** is *the Image Analysis Collaboratory and the Megason Lab*
4. I have *confocal microscopy* **images** of *cancer tissues, embryos, ...*
5. A **fun fact** about me is *I used to be a brewer*

Goals

1. **Motivate** the use of algorithms in image analysis
2. **Introduce** some image-analysis nomenclature
3. **Learn** to use QuPath effectively and reproducibly

Reasons to learn image processing

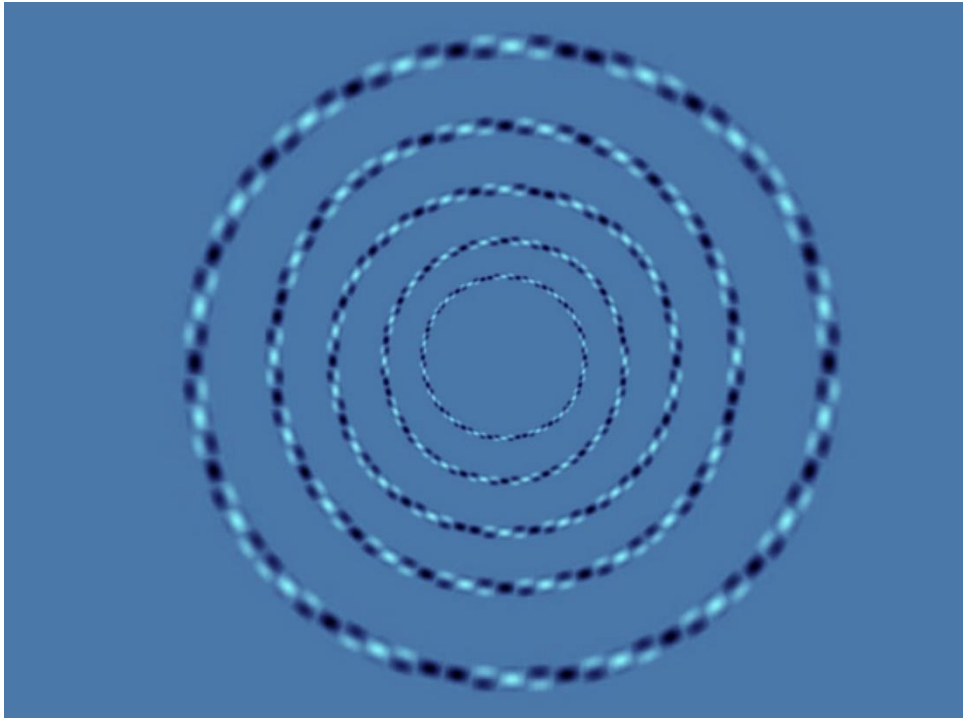
- Make pretty pictures (processing)
 - publications, talks, websites, ...
- Get numbers out of pictures (analysis)
 - cell sizes, vessel lengths, GPF expression level, ...
- Make experiment possible (automation)
 - whole-genome screen: millions of images
- Objectivity and Reproducibility
 - in science, it's your duty!

Reasons not to learn image processing

none

Why should we analyze images
with computers at all?

Color perception and pattern recognition is individual – science less so

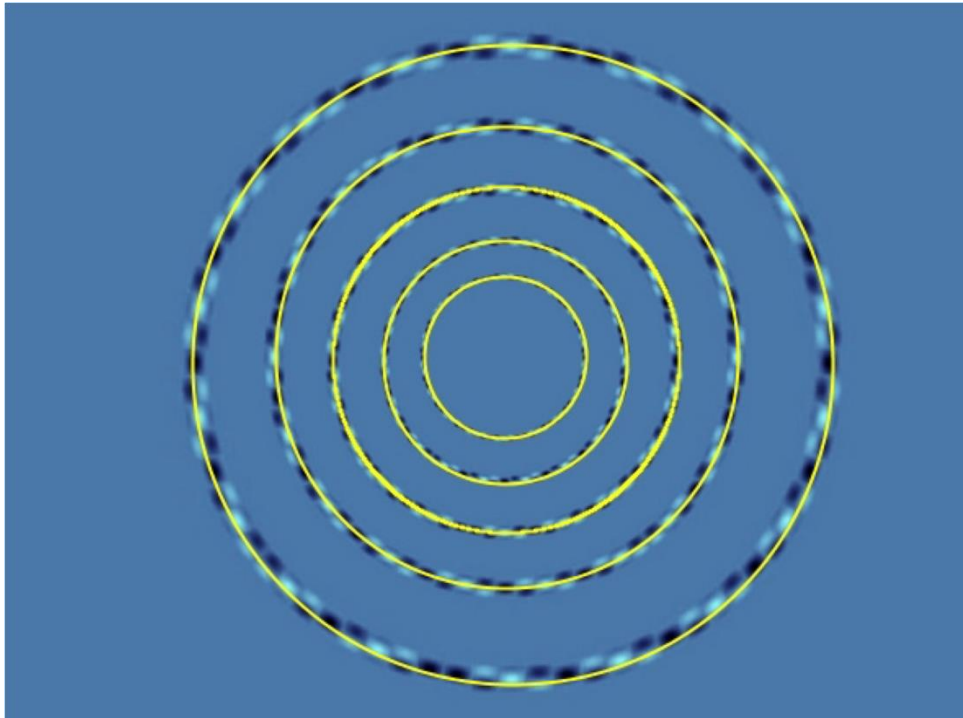


<https://www.moillusions.com/perfect-circles-optical-illusion/>

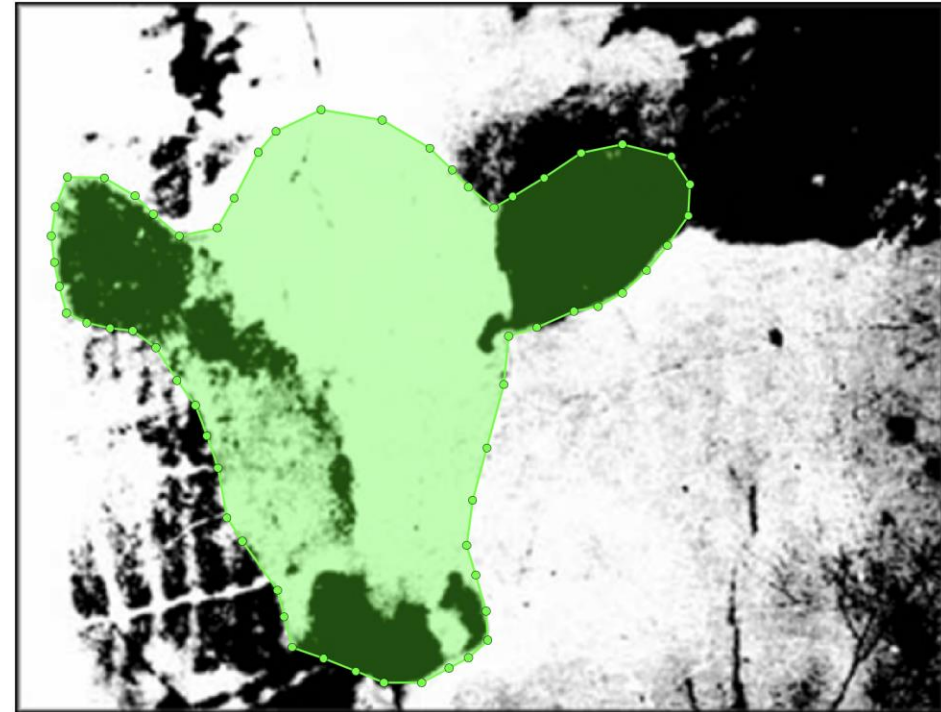


<http://www.brainbashers.com>

Color perception and pattern recognition is individual – science less so



<https://www.moillusions.com/perfect-circles-optical-illusion/>



<http://www.brainbashers.com>

In other words,

“Each human brain is a very complex neural network trained on different data – predictions will vary”

Antoine

A typical image analysis workflow

- There are typically *five* steps in an image analysis
- Often a good idea to structure work along these lines before starting



Think of this even ***before*** you acquire the images!

otherwise image analysis may become only a *post-mortem* on your experiment

Image processing vs analysis

Image Formation
object in → image out

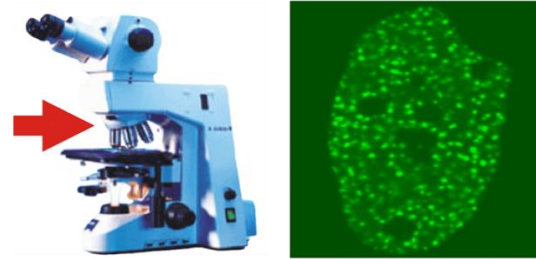


Image Processing
image in → image out

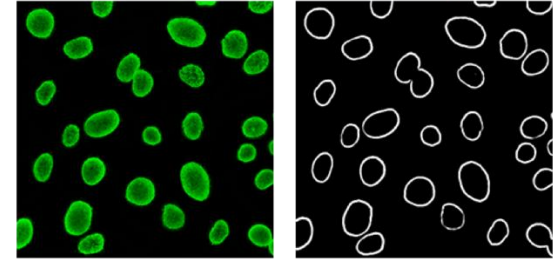
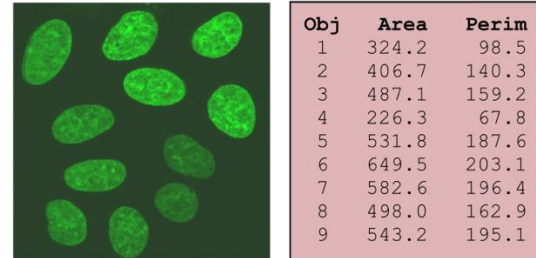
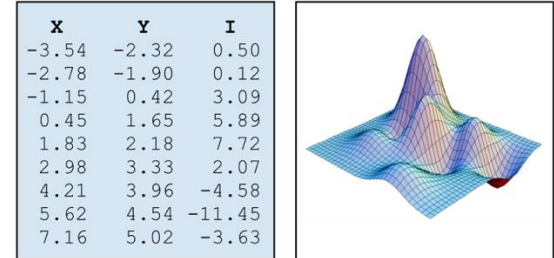


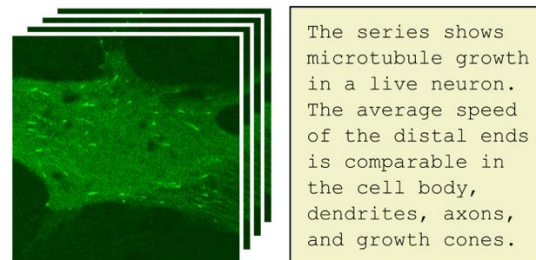
Image Analysis
image in → features out



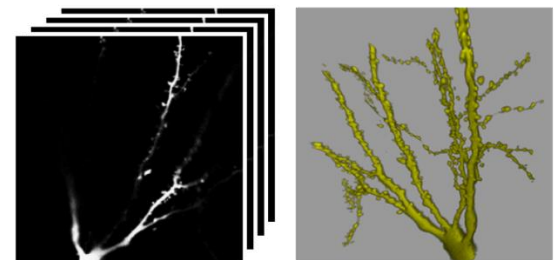
Computer Graphics
numbers in → image out



Computer Vision
image in → interpretation out

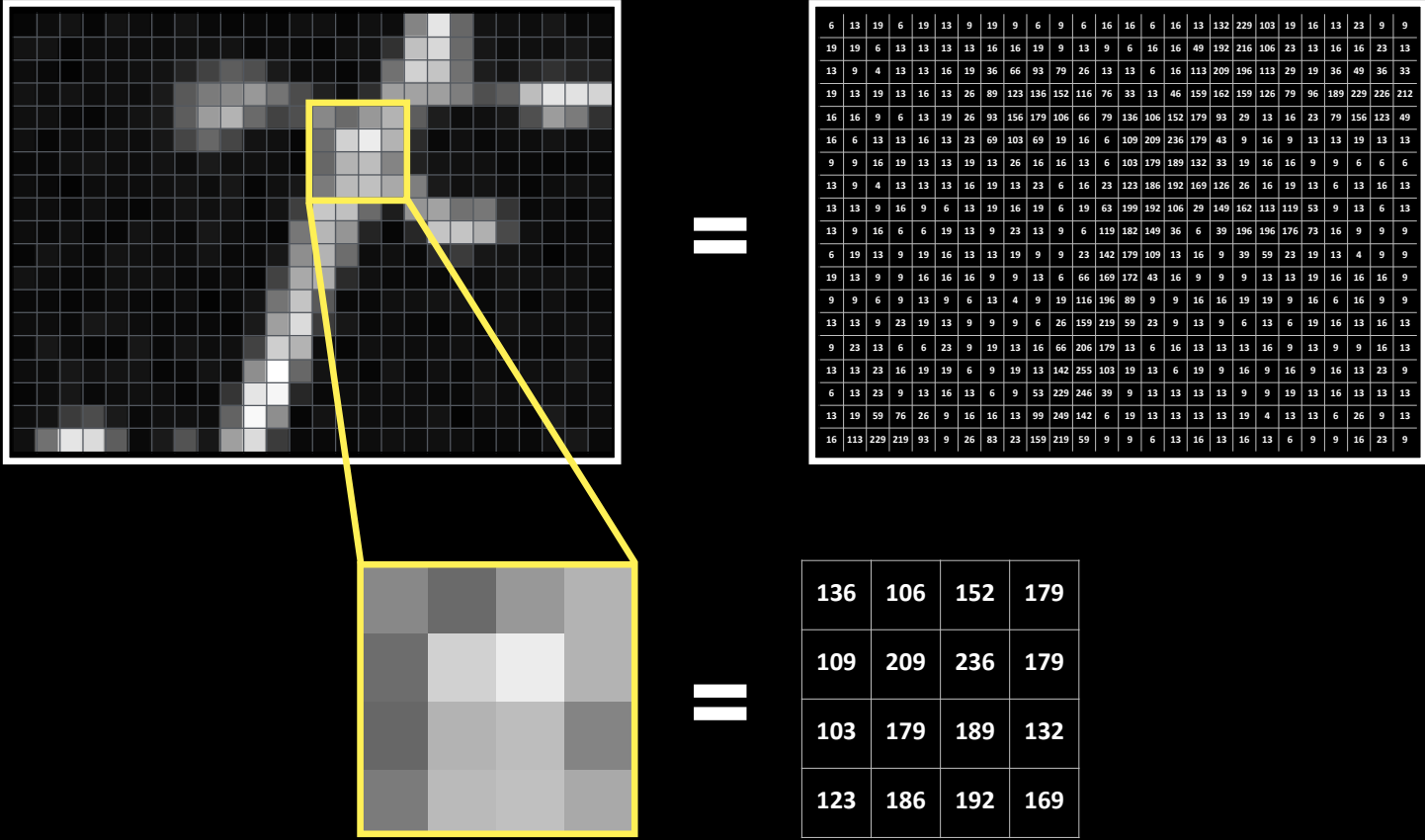


Visualization
image in → representation out



What is an image?

A digital image is a matrix of numbers!



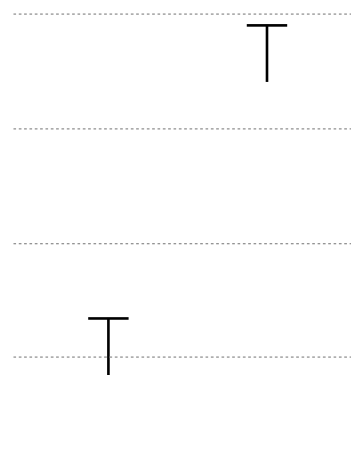
Pixel = Picture Element

Images in publications and presentations
should be used to **communicate** a finding...
not **be** the finding

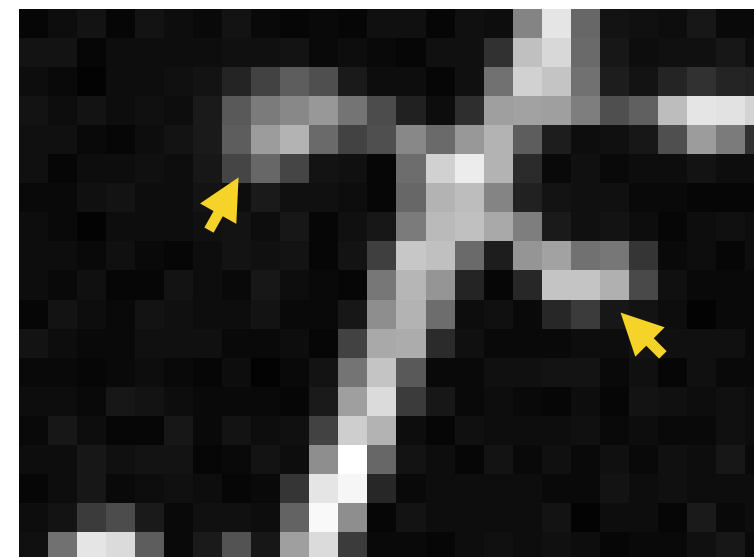
this is your data

6	13	19	6	19	13	9	19	9	6	9	6	16	16	6	16	13	132	229	103	19	16	13	23	9	9
19	19	6	13	13	13	13	16	16	19	9	13	9	6	16	16	49	192	216	106	23	13	16	16	23	13
13	9	4	13	13	16	19	36	66	93	79	26	13	13	6	16	113	209	196	113	29	19	36	49	36	33
19	13	19	13	16	13	26	89	123	136	152	116	76	33	13	46	159	162	159	126	79	96	189	229	226	212
16	16	9	6	13	19	26	93	156	179	106	66	79	136	106	152	179	93	29	13	16	23	79	156	123	49
16	6	13	13	16	13	23	69	103	69	19	16	6	109	209	236	179	43	9	16	9	13	13	19	13	13
9	9	16	19	13	13	19	13	26	16	16	13	6	103	179	189	132	33	19	16	16	9	9	6	6	6
13	9	4	13	13	13	16	19	13	23	6	16	23	123	186	192	169	126	26	16	19	13	6	13	16	13
13	13	9	16	9	6	13	19	16	19	6	19	63	199	192	106	29	149	162	113	119	53	9	13	6	13
13	9	16	6	6	19	13	9	23	13	9	6	119	182	149	36	6	39	196	196	176	73	16	9	9	9
6	19	13	9	19	16	13	13	19	9	9	23	142	179	109	13	16	9	39	59	23	19	13	4	9	9
19	13	9	9	16	16	16	9	9	13	6	66	169	172	43	16	9	9	9	13	13	19	16	16	16	9
9	9	6	9	13	9	6	13	4	9	19	116	196	89	9	9	16	16	19	19	9	16	6	16	9	9
13	13	9	23	19	13	9	9	9	6	26	159	219	59	23	9	13	9	6	13	6	19	16	13	16	13
9	23	13	6	6	23	9	19	13	16	66	206	179	13	6	16	13	13	13	16	9	13	9	9	16	13
13	13	23	16	19	19	6	9	19	13	142	255	103	19	13	6	19	9	16	9	16	9	16	13	23	9
6	13	23	9	13	16	13	6	9	53	229	246	39	9	13	13	13	13	9	9	19	13	16	13	13	13
13	19	59	76	26	9	16	16	13	99	249	142	6	19	13	13	13	13	19	4	13	13	6	26	9	13
16	113	229	219	93	9	26	83	23	159	219	59	9	9	6	13	16	13	16	13	6	9	9	16	23	9

this is your result



this just helps to
communicate the result

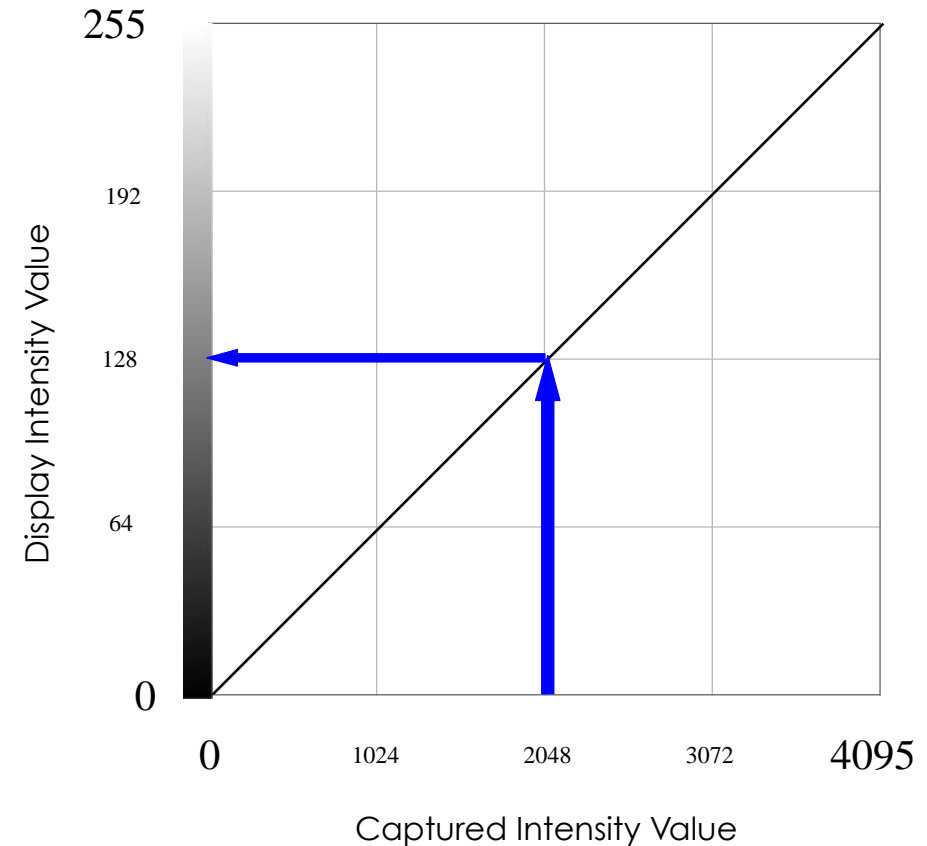
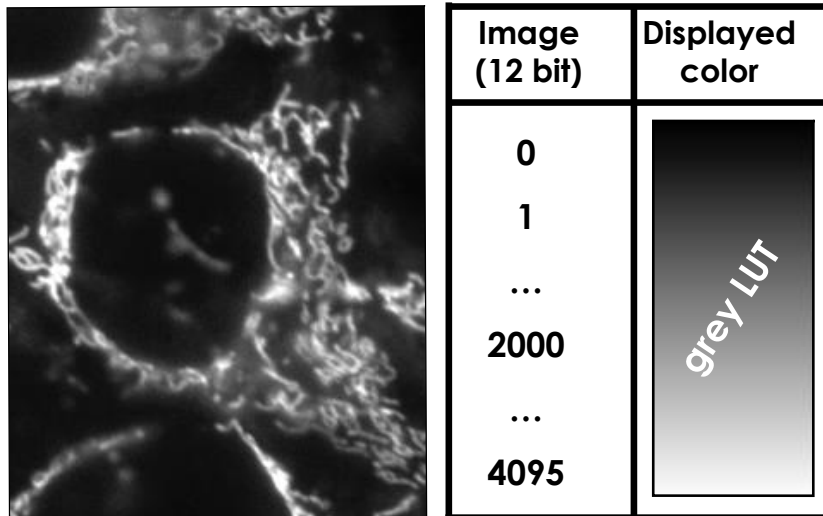
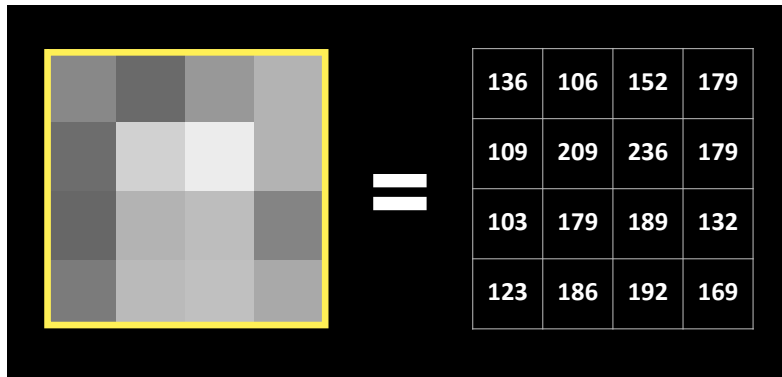


Display your images

Mapping Image Intensity to Monitor Intensity (LookUp Tables)

LUT = how the grey values are displayed

LUTs do not change the pixel values



Images and Colors

Lookup Tables (LUTs)

LUT = how the grey values are displayed

LUTs do not change the pixel values

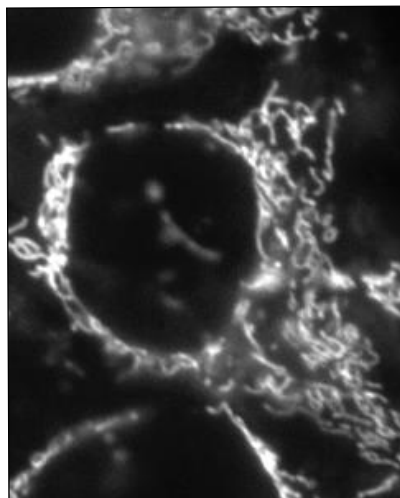


Image (8 bit)	Displayed color
0	
1	
...	
100	
255	

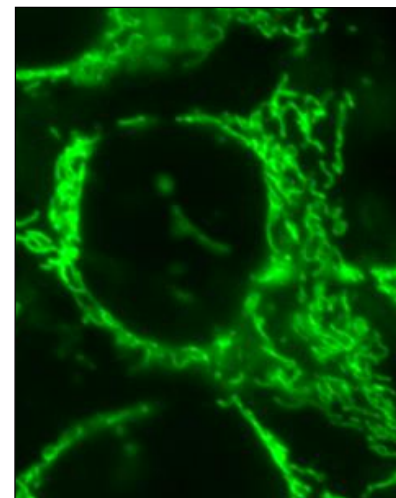
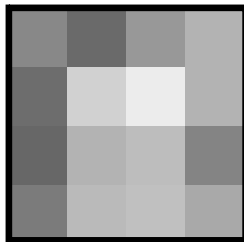
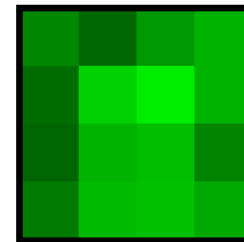


Image (8 bit)	Displayed color
0	
1	
...	
100	
255	



=

136	106	152	179
109	209	236	179
103	179	189	132
123	186	192	169



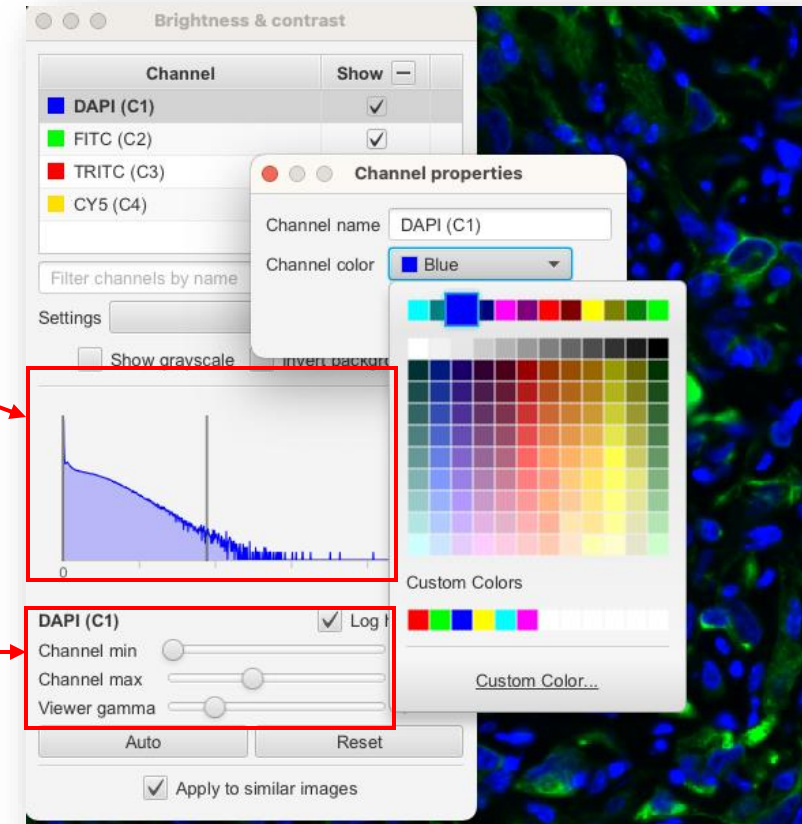
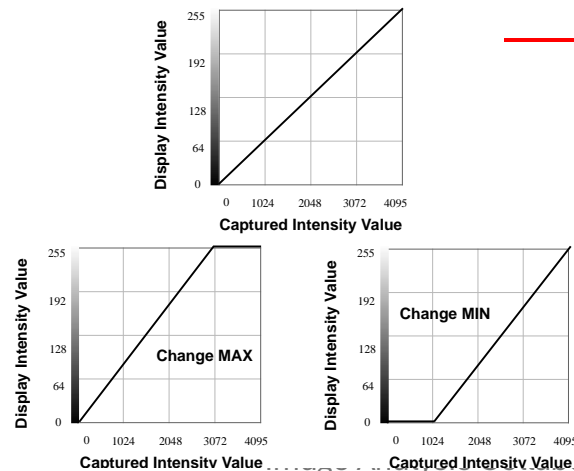
=

136	106	152	179
109	209	236	179
103	179	189	132
123	186	192	169

Display images: color, brightness & contrast

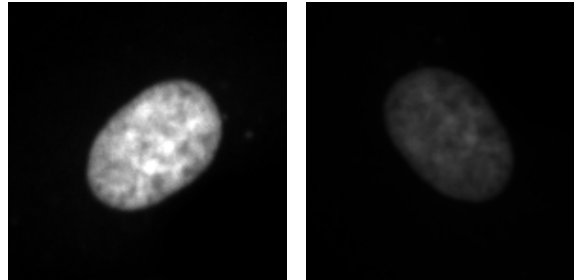
- If you are imaging a blue fluorophore, you are not forced to display it in blue!
- Pixel histogram represents the distribution of pixel values in the image
- LUT range

**You are NOT changing the pixels values, you are just changing how the image is displayed (unless you click on the "Apply" button).*

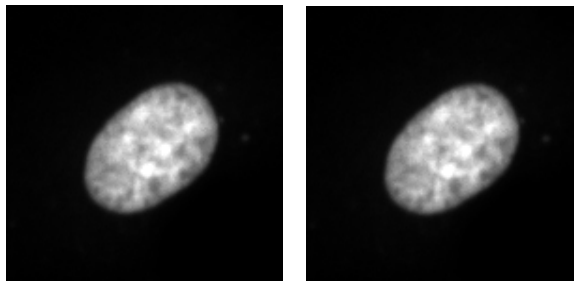


Display a file: Brightness & Contrast

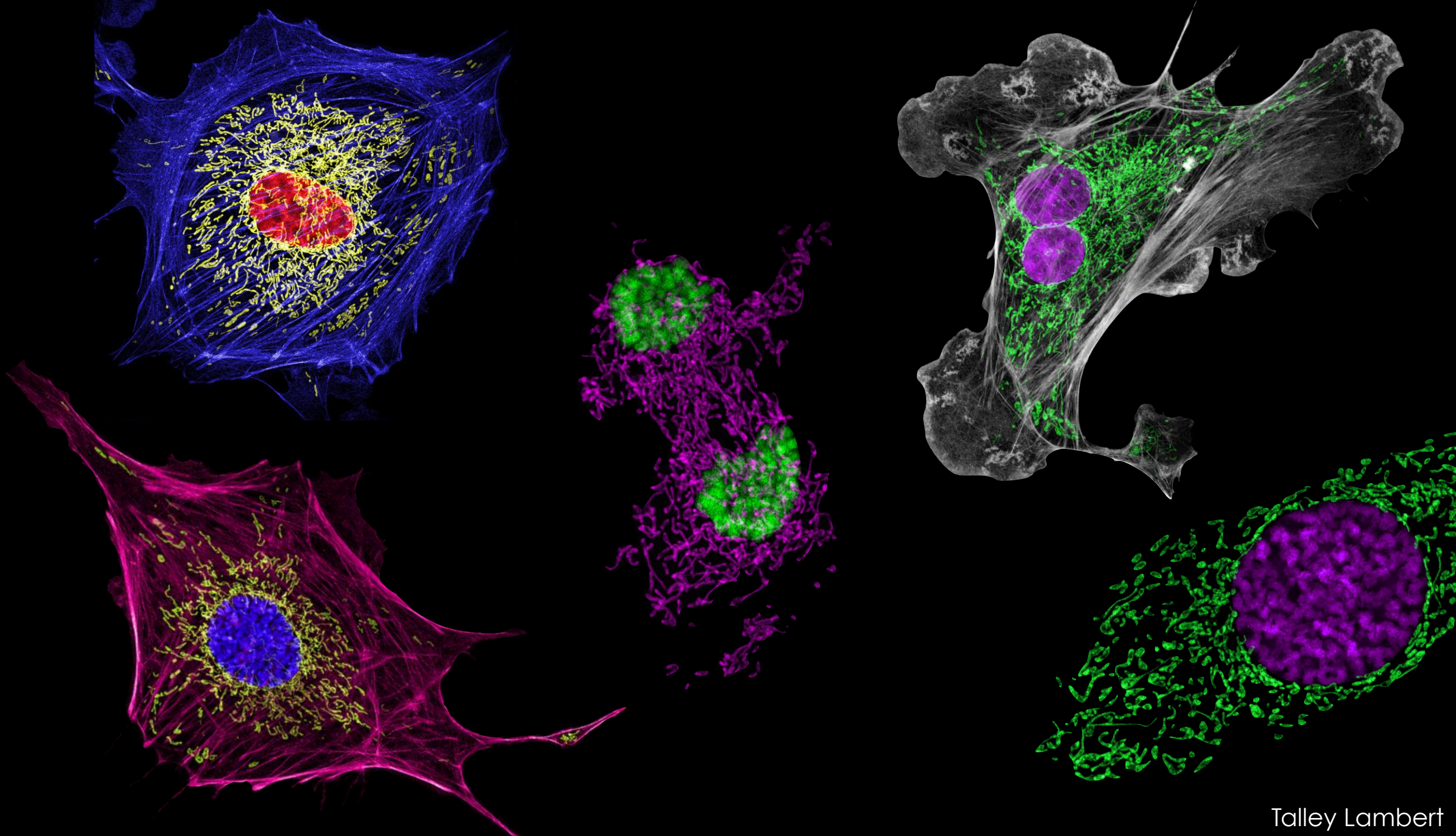
Which image has more fluorescence?



<i>Mean:</i>	4803	4803
<i>Display range:</i>	188- 16828	188- 45514



<i>Mean:</i>	4803	4803
<i>Display range:</i>	188- 16828	188- 16828



Save the downloaded example image (cont.)

1. Browse to the workshop website >
2. Download the .vsi whole-slide image (~2-5 min)
3. Create a folder named ***qupath_workshop*** (outside of your *downloads* folder)
4. Once the download is finished, unzip
5. Save the unzipped folder in the newly created ***qupath_workshop*** folder

Introduction to QuPath

You before this workshop



You after this workshop



Illustration reprinted from Pete Bankhead.

What is QuPath?

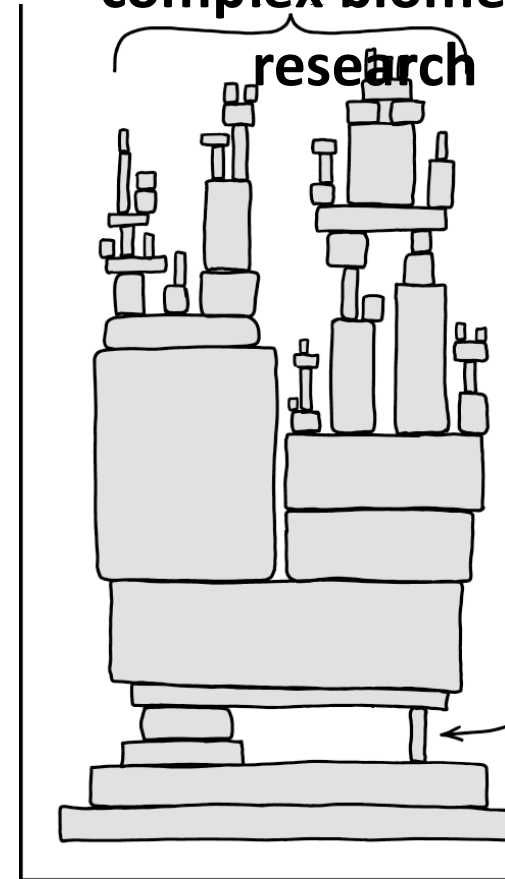
QuPath is an open-source software for bioimage analysis

- Developed and maintained by Pete Bankhead and his team at the University of Edinburgh

Key features:

1. Performant when working with very large 2D images, like those produced by slide scanner
2. Extremely well maintained

The world's most
complex biomedical



The open-source
image analysis
software **Pete
Bankhead** has
been thanklessly
developing in the
UK since 2016

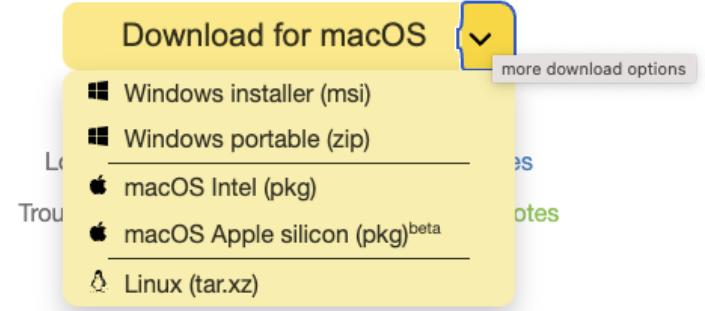
Adapted from <https://xkcd.com/2347/>

What is QuPath good at?

- It has a nice graphical user interface (GUI)
- It was designed to handle very large 2D images
- It supports common image analysis tasks: segmentation, annotation, feature analysis, and classification
- It support extensive visualization options
- It integrates with many other existing tools (Stardist, ImageJ, ...)
- It support scripting (Groovy, akin to Java)

What is QuPath not good at?

- Limited to 2D images
 - Only supports the visualization of single planes
- Does not support all file format (e.g. zarr/NGFF)



Installing Qupath

```
INSTALL.SH
#!/bin/bash

pip install "$1" &
easy_install "$1" &
brew install "$1" &
npm install "$1" &
yum install "$1" & dnf install "$1" &
docker run "$1" &
pkg install "$1" &
apt-get install "$1" &
sudo apt-get install "$1" &
steampcmd +app_update "$1" validate &
git clone https://github.com/"$1"/"$1" &
cd "$1"; ./configure; make; make install &
curl "$1" | bash &
```

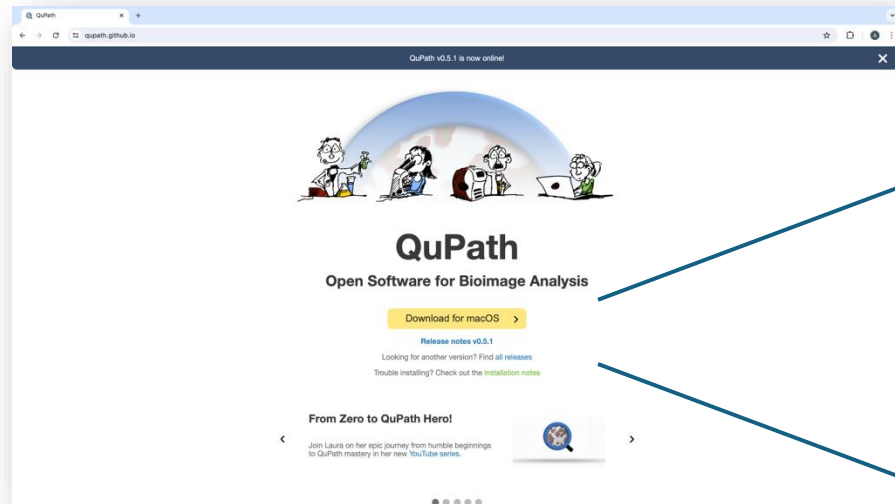
<https://xkcd.com/1654/>

Download QuPath

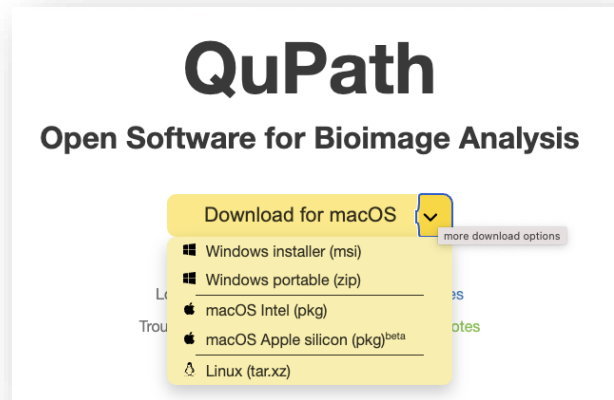
1. Go to <https://qupath.github.io/> (see *useful links on website*)

2. Download the installer for the latest version

NB: we recommend the .msi file for Windows users





3. Choose your OS

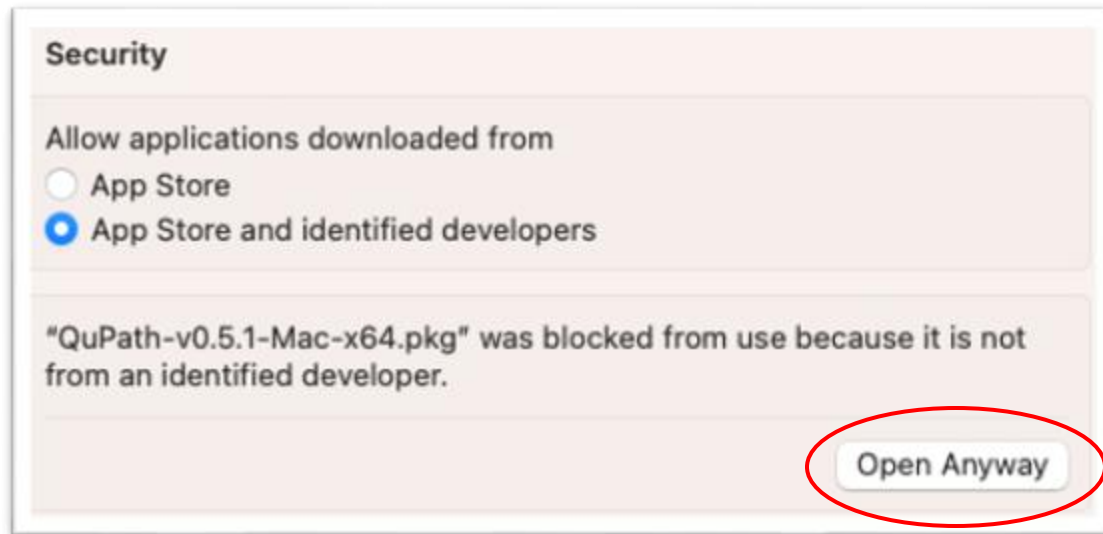


4. Right-click on the installer file > Open > Confirm Open

For macOS users

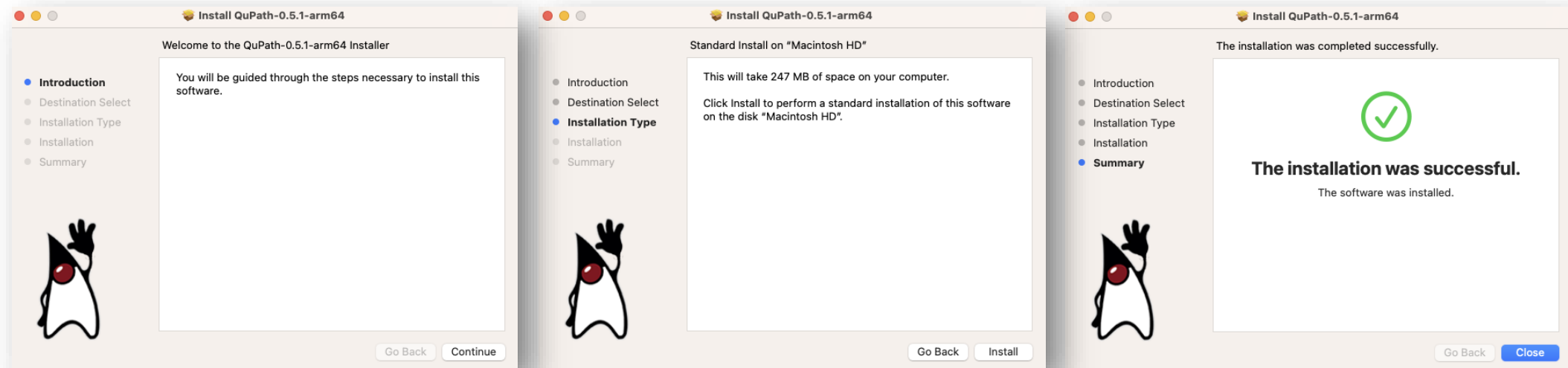
Confirm that QuPath installer is not a malware

On your Mac, choose Apple menu  > System Settings, then click Privacy & Security  in the sidebar. (You may need to scroll down.)



Download QuPath

Follow the steps of the installer



Manage different QuPath versions

macOS users:

- *Applications* > multiple versions of QuPath installed > Choose the latest one
- Cmd + space, then choose from the list of available versions

Windows users:

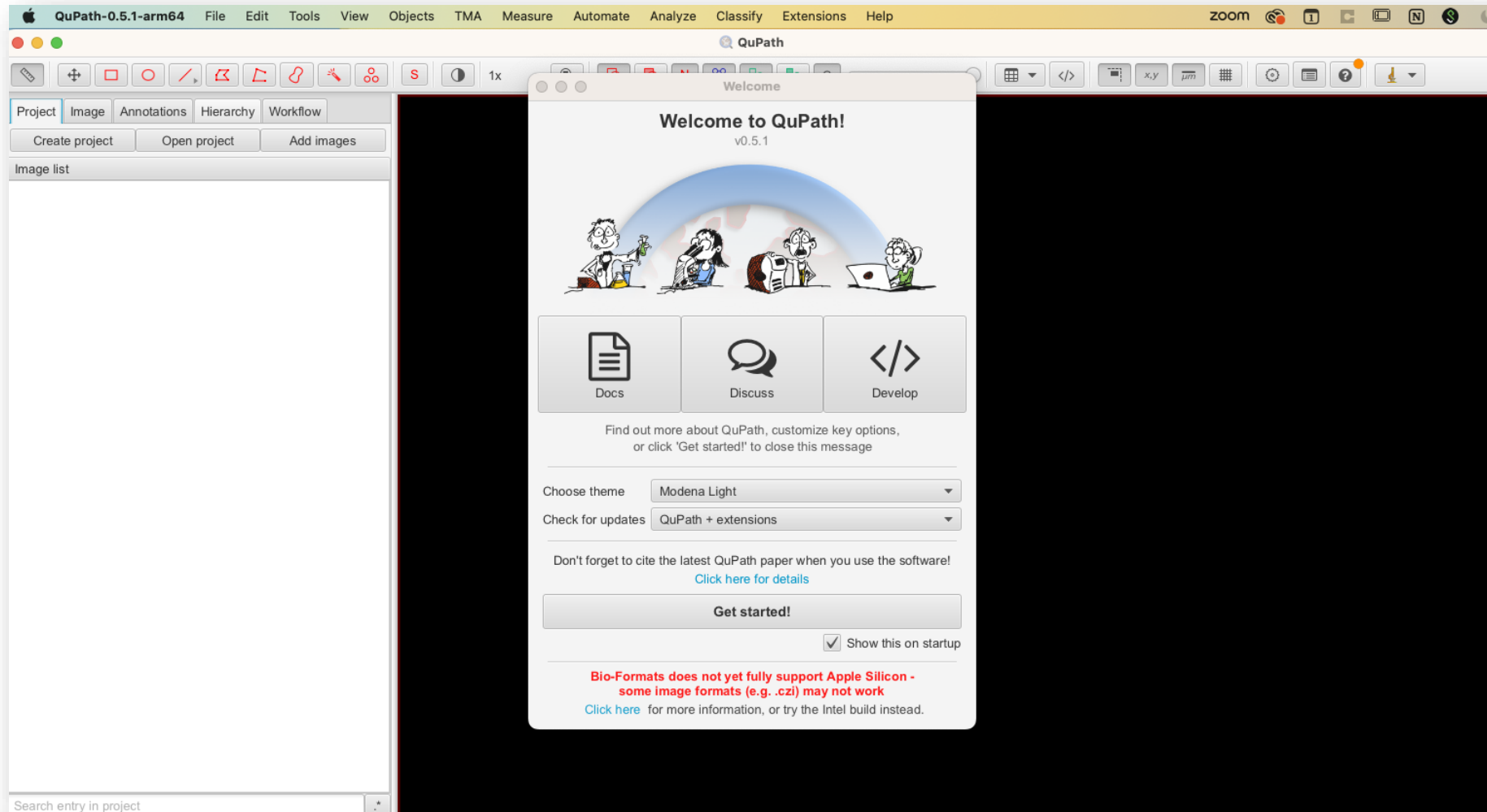
- C:\Program Files\QuPath
- Opening your application manager should prompt you the different version of QuPath that have been downloaded

Linux users:

- I don't know

Trick: *using multiple versions of QuPath allows to easily run more than one app at the same time on your laptop (i.e. doing so from a single app would require launching each instance from a separate terminal on macOS).*

Open the QuPath application



Welcome to QuPath!



Useful resources:

Documentation:

<https://qupath.readthedocs.io/en/0.5/>

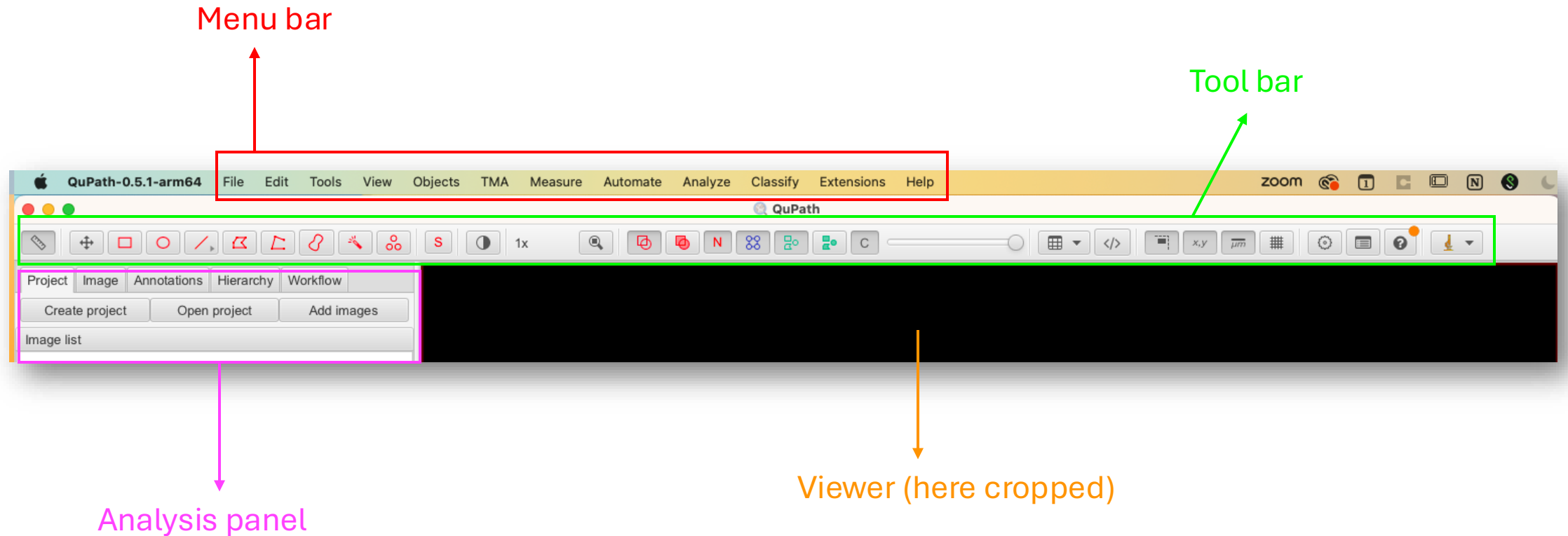
QuPath-specific place in 'The Forum':

<https://forum.image.sc/tag/qupath>

Updater

For now, let's get started

Graphic User Interface (GUI) – intro



Getting help

- In-app documentation: Help menu



- QuPath documentation: <https://qupath.github.io/>

- The Forum: <https://forum.image.sc/>  image.sc

- During this workshop, ask questions to your neighbors, the TA's and me!

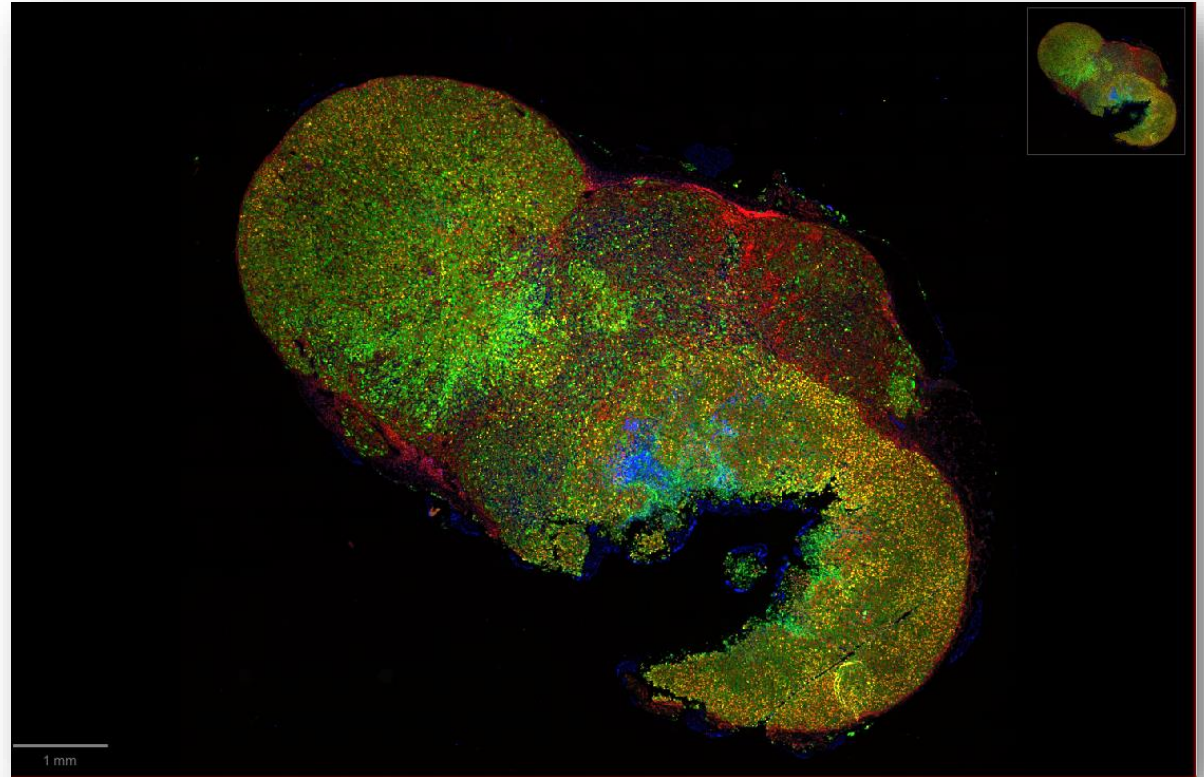
A fluorescence microscopy image of a tissue section. The image shows a complex, irregular shape with a dark background. The tissue is stained with multiple fluorescent markers, resulting in a mix of colors: a prominent blue channel, a green channel, and a red channel. The blue channel appears to be the most widespread, covering most of the tissue area. The green and red channels are more localized, with some green appearing in the lower right and red appearing in the upper left. The overall appearance is that of a histological section where different cell types or structures are being highlighted by different dyes.

Your first project in QuPath

Classification of proliferating cancer cells in solid tumors

- Whole-slide image
 - Already been stitched
- 4 channels
 - DAPI
 - Keratin (FITC)
 - Fibronectin (TRITC)
 - Ki67 (CY5)

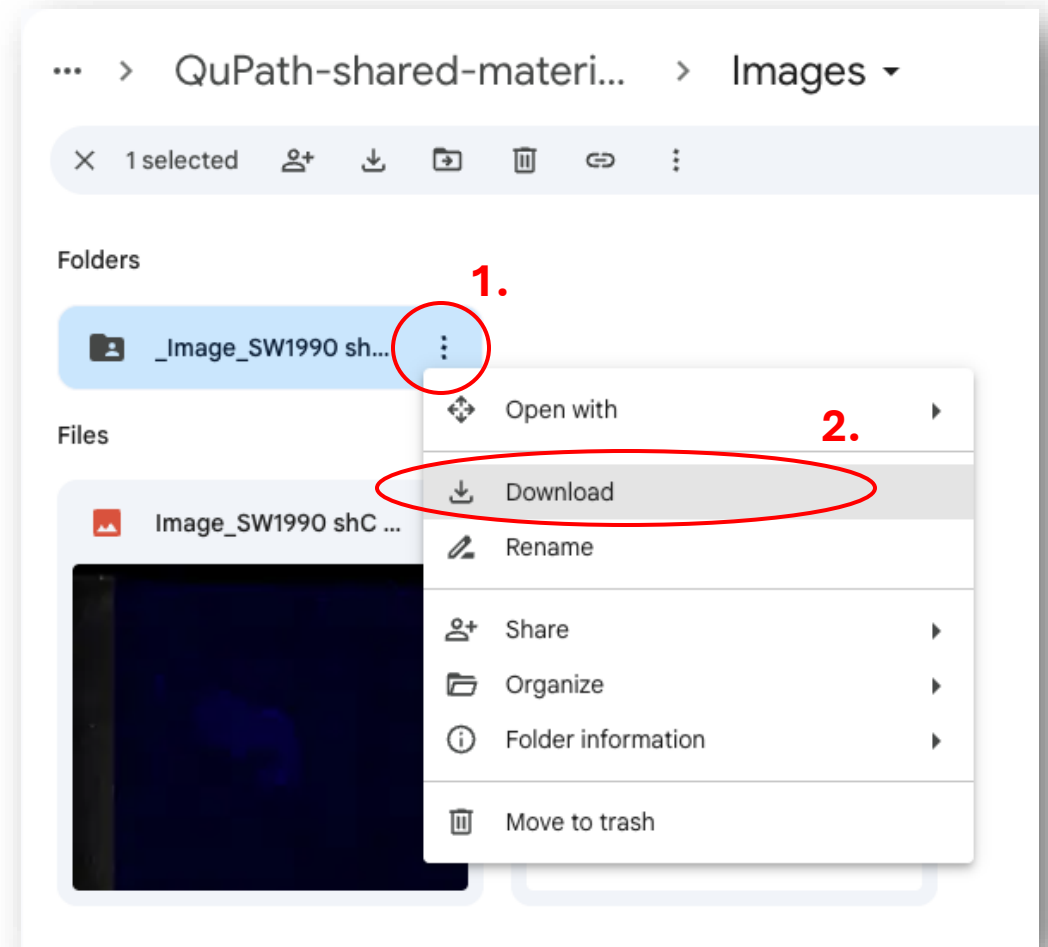
At the end of this course:
you will have classified
proliferating (Ki67) cancer
cells and reveal their spatial
distribution to regions with
high-fibronectin content



Courtesy of Nina Kozlova, PhD

Download the image from the shared folder

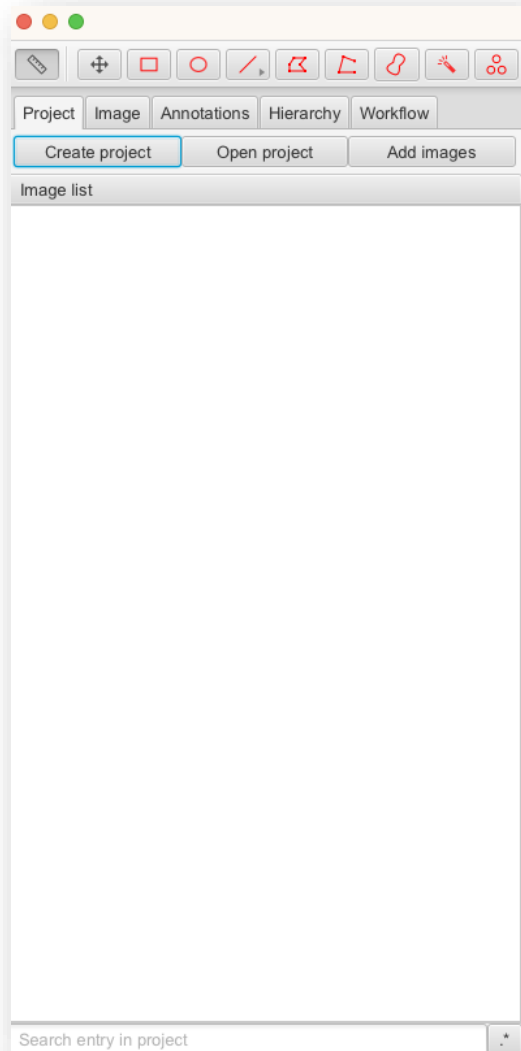
1. Download the whole folder from the Google Drive
 1. Image size: ~3GB; it will take a minute or two to download
2. Unzip it
3. Transfer the image in a new **Images** folder in your QuPath project folder



Key concept: QuPath project

- Projects are the way to organize your work in QuPath
- In other words, they are folders
 - Group together images
 - Organize data, scripts, classifiers, etc
 - They only save data, not the original images
- Allow you to share your work with other QuPath users
 - Always send the images along!

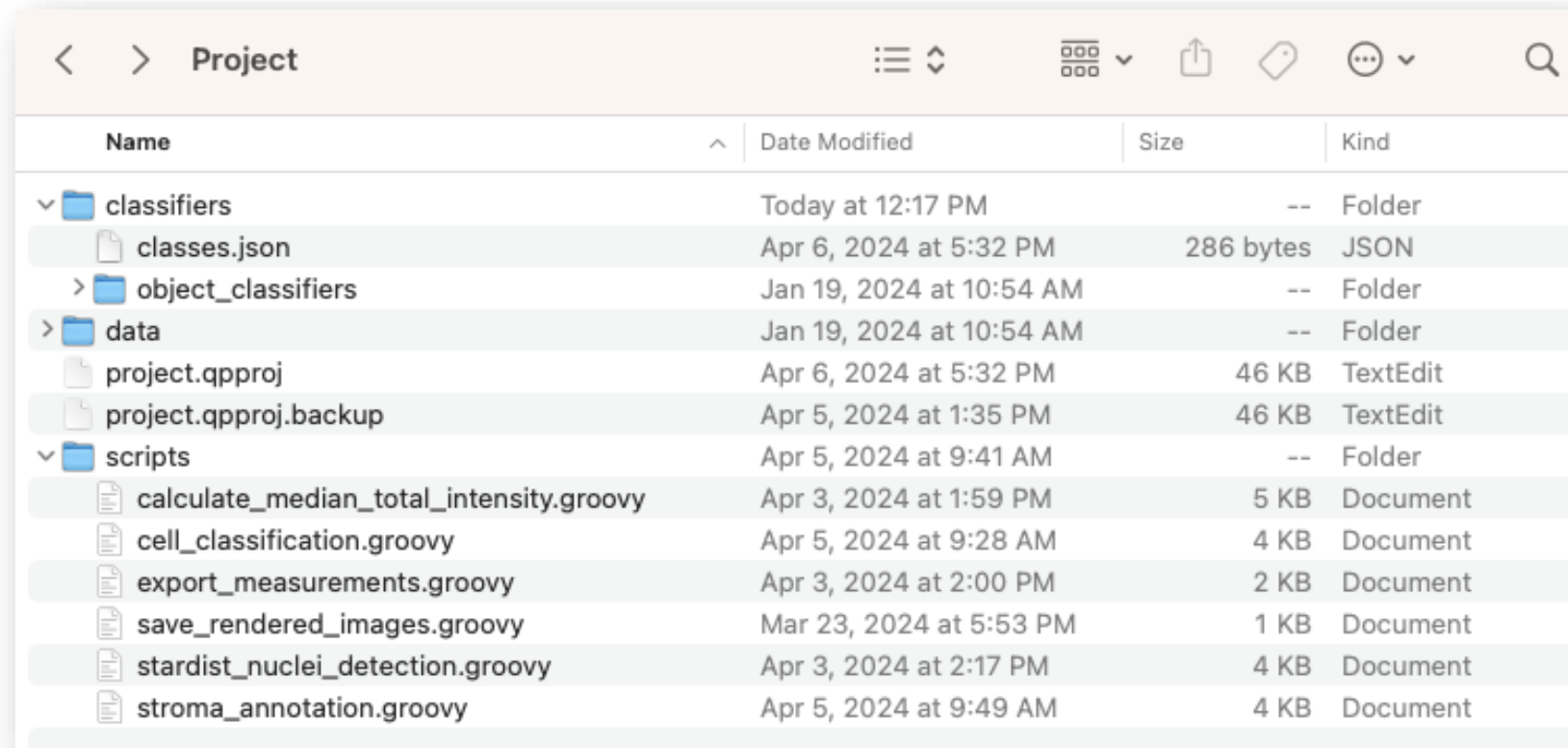
How to create a project?



- *Create project* button
or
- File > Project... > Create project
- ! Make sure to create an **empty** folder for your project
 - Sometimes, you have to do this twice in the empty folder

Anatomy of a QuPath project

After a bit of time working on it...

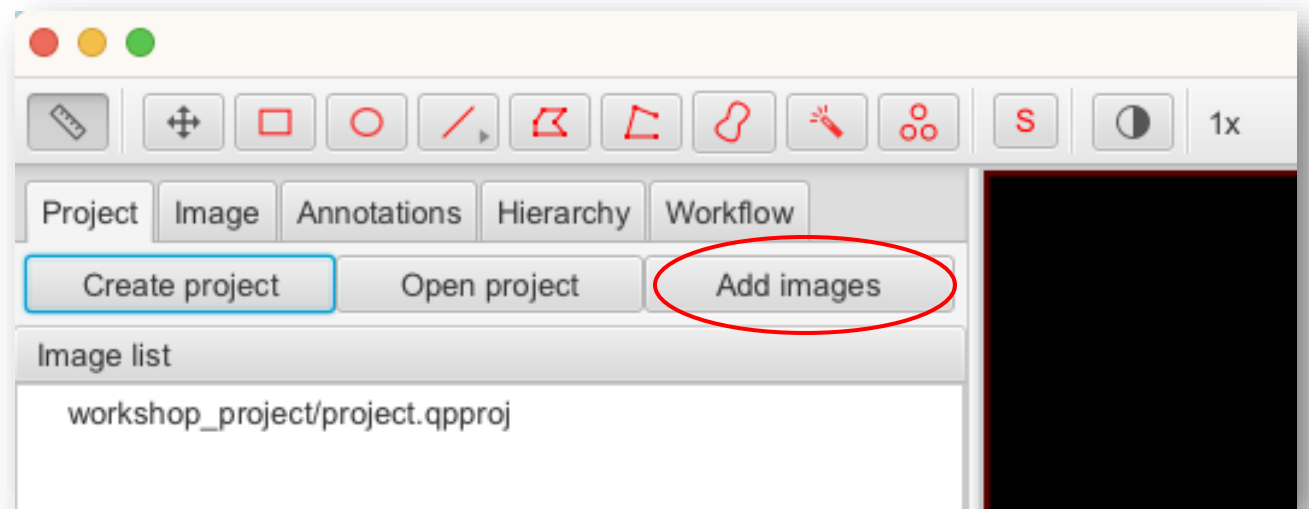


The screenshot shows a file explorer window titled "Project". The window displays a list of files and folders with columns for Name, Date Modified, Size, and Kind. The files and folders are as follows:

Name	Date Modified	Size	Kind
classifiers	Today at 12:17 PM	--	Folder
classes.json	Apr 6, 2024 at 5:32 PM	286 bytes	JSON
object_classifiers	Jan 19, 2024 at 10:54 AM	--	Folder
data	Jan 19, 2024 at 10:54 AM	--	Folder
project.qpproj	Apr 6, 2024 at 5:32 PM	46 KB	TextEdit
project.qpproj.backup	Apr 5, 2024 at 1:35 PM	46 KB	TextEdit
scripts	Apr 5, 2024 at 9:41 AM	--	Folder
calculate_median_total_intensity.groovy	Apr 3, 2024 at 1:59 PM	5 KB	Document
cell_classification.groovy	Apr 5, 2024 at 9:28 AM	4 KB	Document
export_measurements.groovy	Apr 3, 2024 at 2:00 PM	2 KB	Document
save_rendered_images.groovy	Mar 23, 2024 at 5:53 PM	1 KB	Document
stardist_nuclei_detection.groovy	Apr 3, 2024 at 2:17 PM	4 KB	Document
stroma_annotation.groovy	Apr 5, 2024 at 9:49 AM	4 KB	Document

Add an image to your project

1. Check your emails! Download this folder containing an example whole-slide image
2. Add an image
 - *Add images* button
 - Select the .vsi file

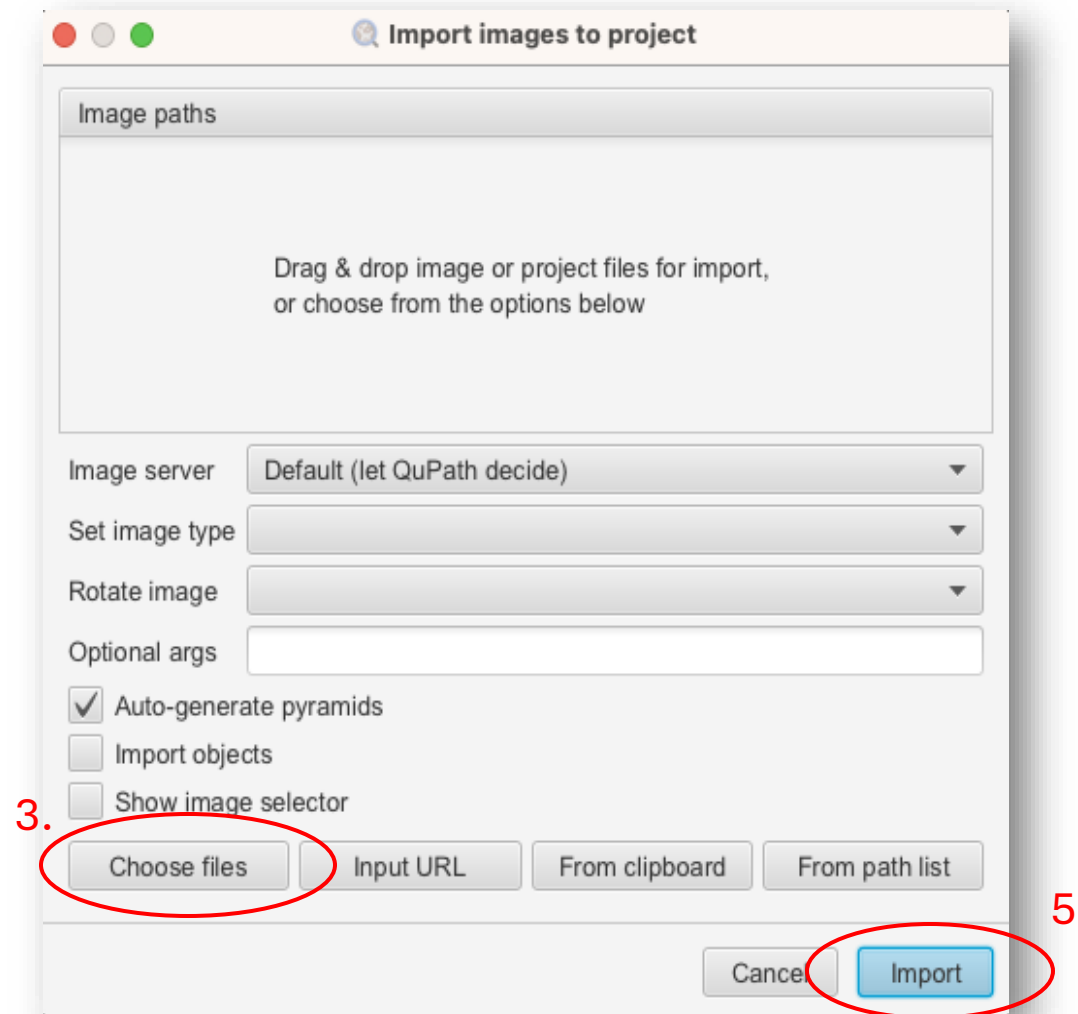


Add an image to your project

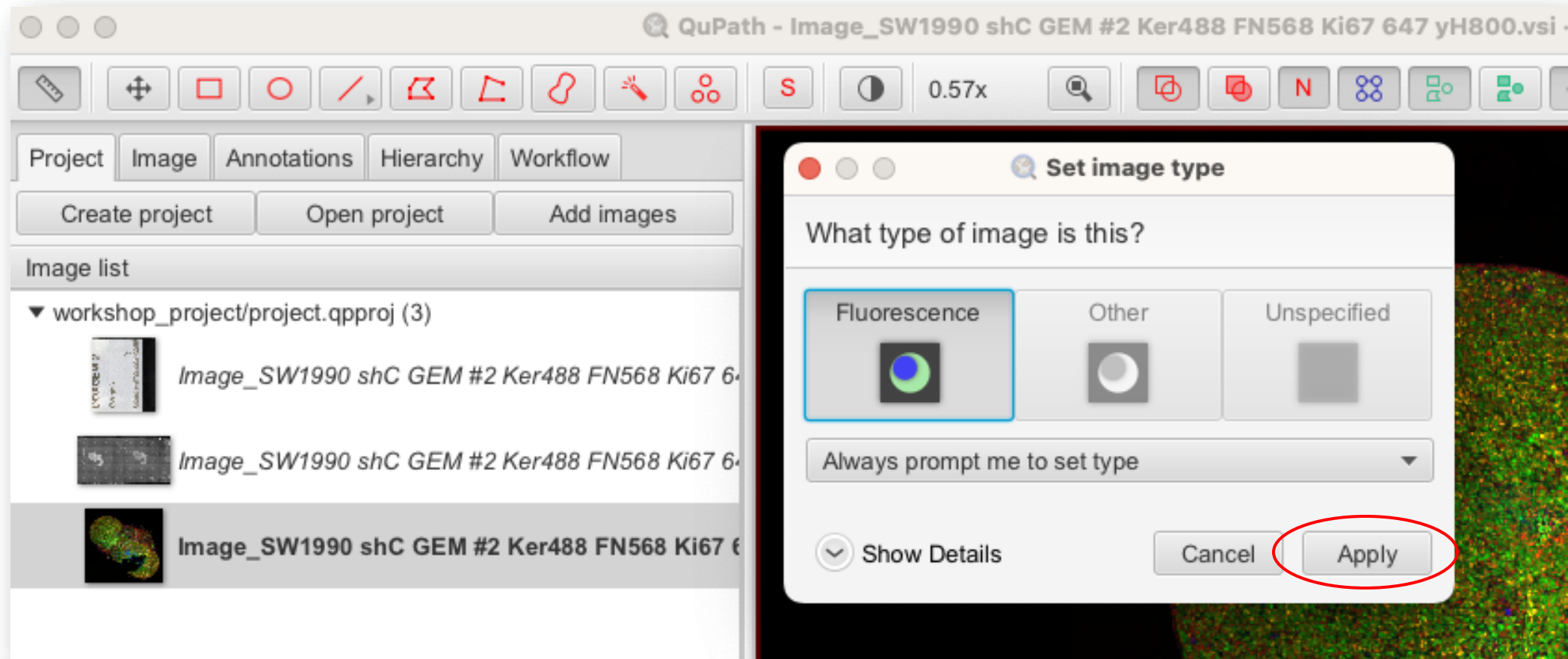
3. Select the .vsi image using *Choose files* or drag-and-drop

4. Use default settings

5. Click *import*



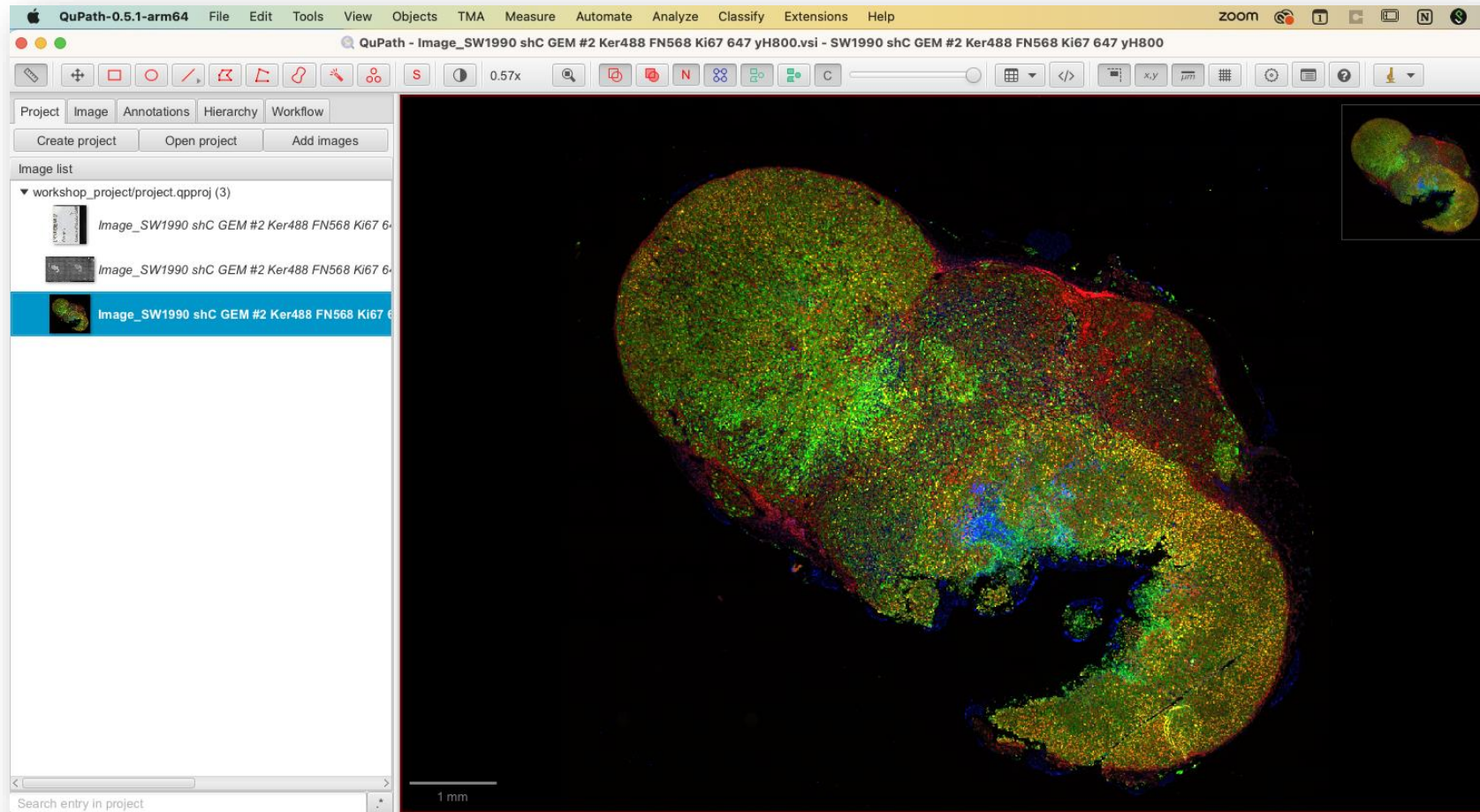
Set image type



- *Other image types are supported: Brightfield H&E, H-DAB, other brightfield*

Yay! We have a QuPath project with an image

Double-click an image to open it in the viewer



QuPath works on copies of your original files

- QuPath access the image pixels and metadata via an image server
 - Akin to a copy of the original file
- Manipulating files within a QuPath project will never modify the original files or pixels
 - Deleting, duplicating, processing, etc will not be reflected in your original files

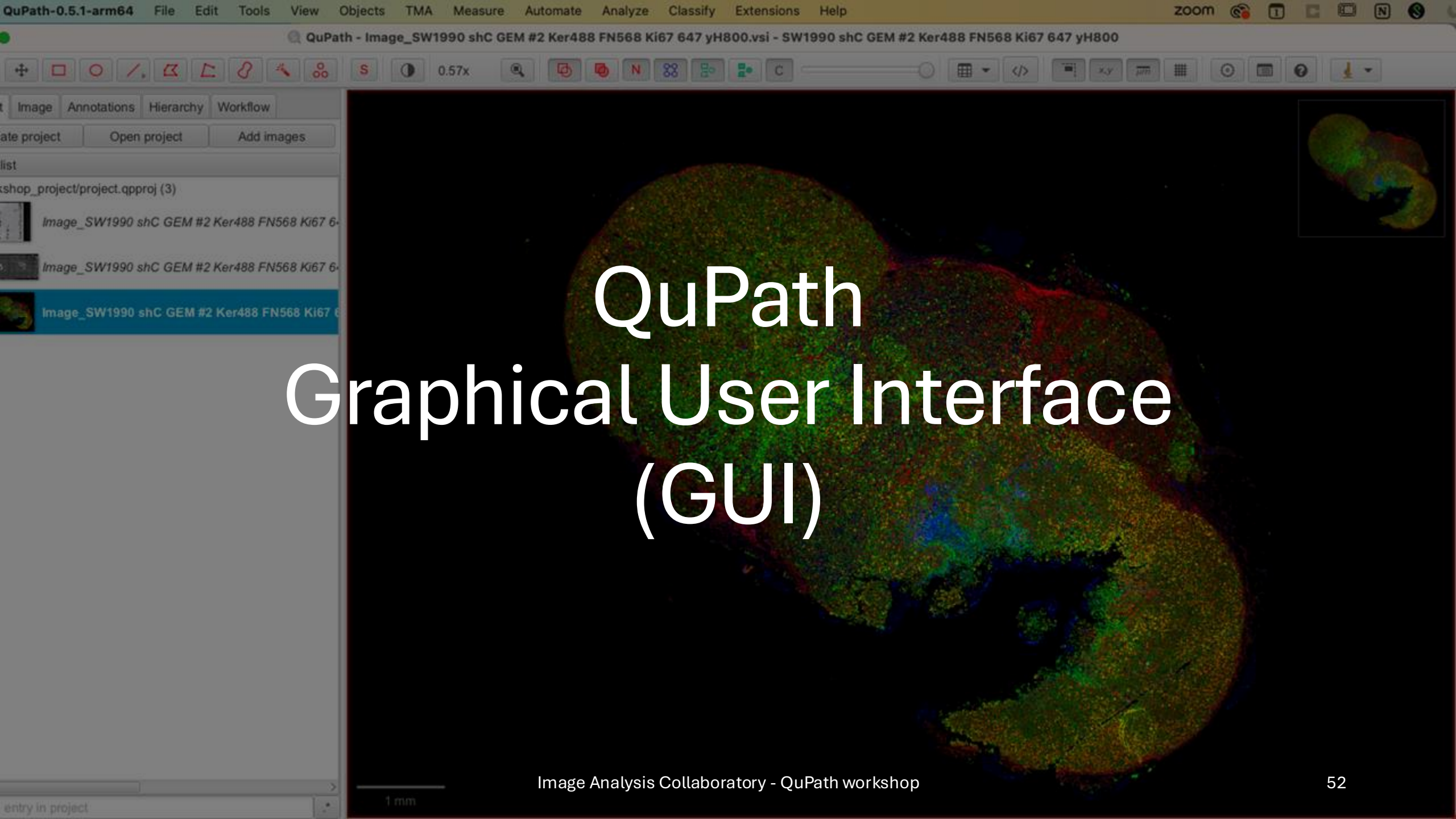
QuPath projects are portable

- Sharing a project:
 - Zip up the entire project directory
 - Email it to your collaborators

The project folder only contains QuPath objects and data, unless you had placed them there. Ensure that they can access the actual image files.

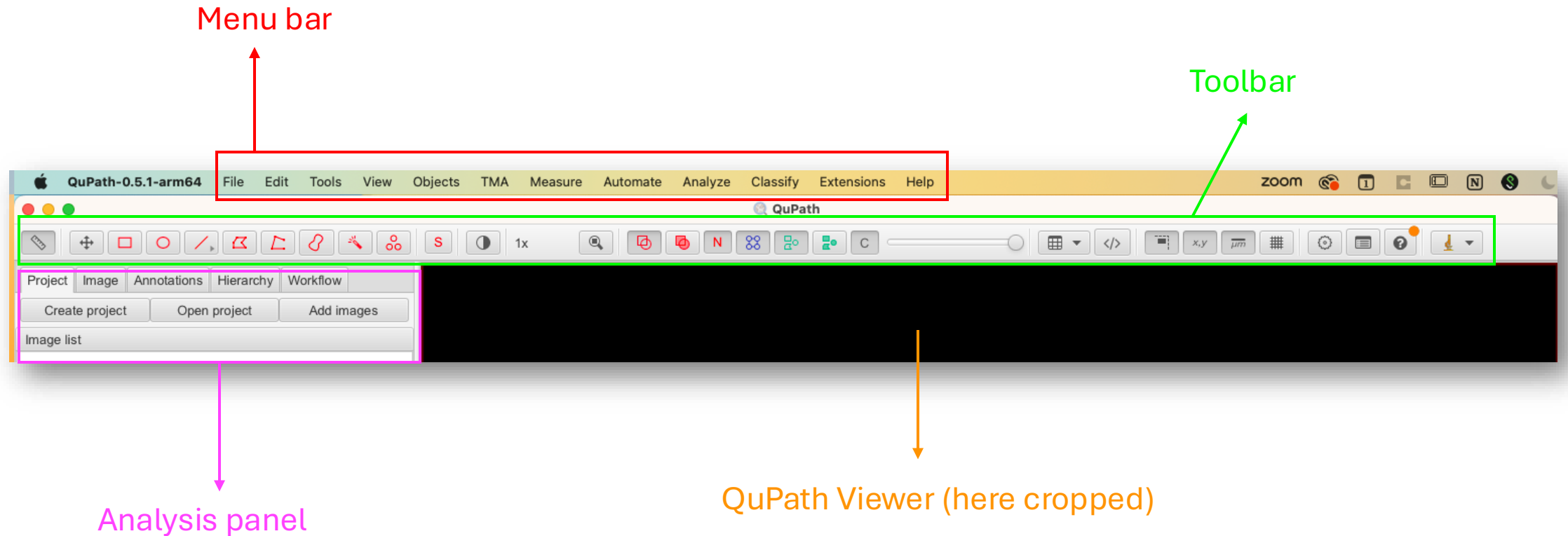
QuPath projects are portable

- Receiving a project:
 - The project still contains image paths specific to the local machine of the sender
 - If you move the image, you will be prompted to update the file path



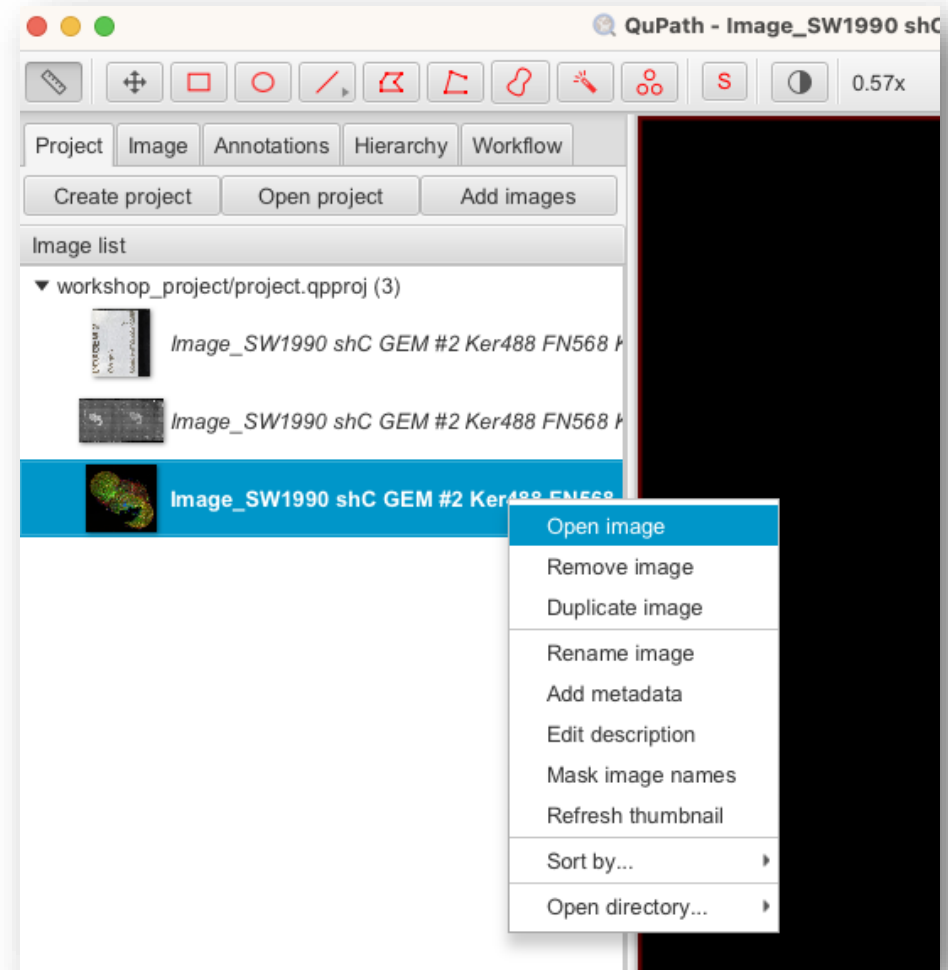
QuPath Graphical User Interface (GUI)

Graphic User Interface (GUI)



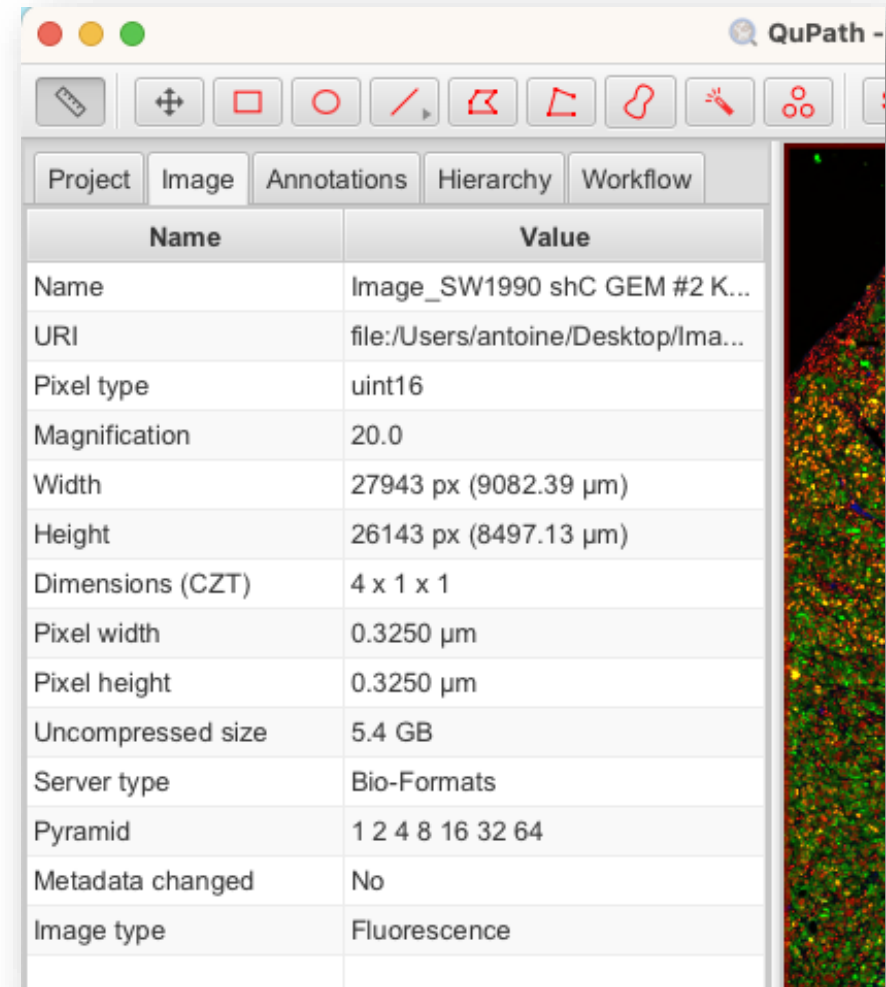
Analysis Panel

- *Project* tab > right-click on an image
 - *Open, remove, rename and duplicate images*
 - *Edit metadata*



Analysis Panel

- *Image* tab
 - Name and image file path
 - Magnification: 20x
 - Pixel type, width and height are crucial for scale calibration
 - Dimensions: 4 channels + 2D
 - Pyramid: level of downsampling in the viewer
 - Image type: previously set to fluorescence



QuPath viewer

Viewer

Mini-map: overview

Name	Value
Name	Image_SW1990 shC GEM #2 K...
URI	file:/Users/antoine/Desktop/Ima...
Pixel type	uint16
Magnification	20.0
Width	27943 px (9082.39 μm)
Height	26143 px (8497.13 μm)
Dimensions (CZT)	4 x 1 x 1
Pixel width	0.3250 μm
Pixel height	0.3250 μm
Uncompressed size	5.4 GB
Server type	Bio-Formats
Pyramid	1 2 4 8 16 32 64
Metadata changed	No
Image type	Fluorescence

1 mm

6547.99, 4758.85 μm
9779, 9166, 1002, 943

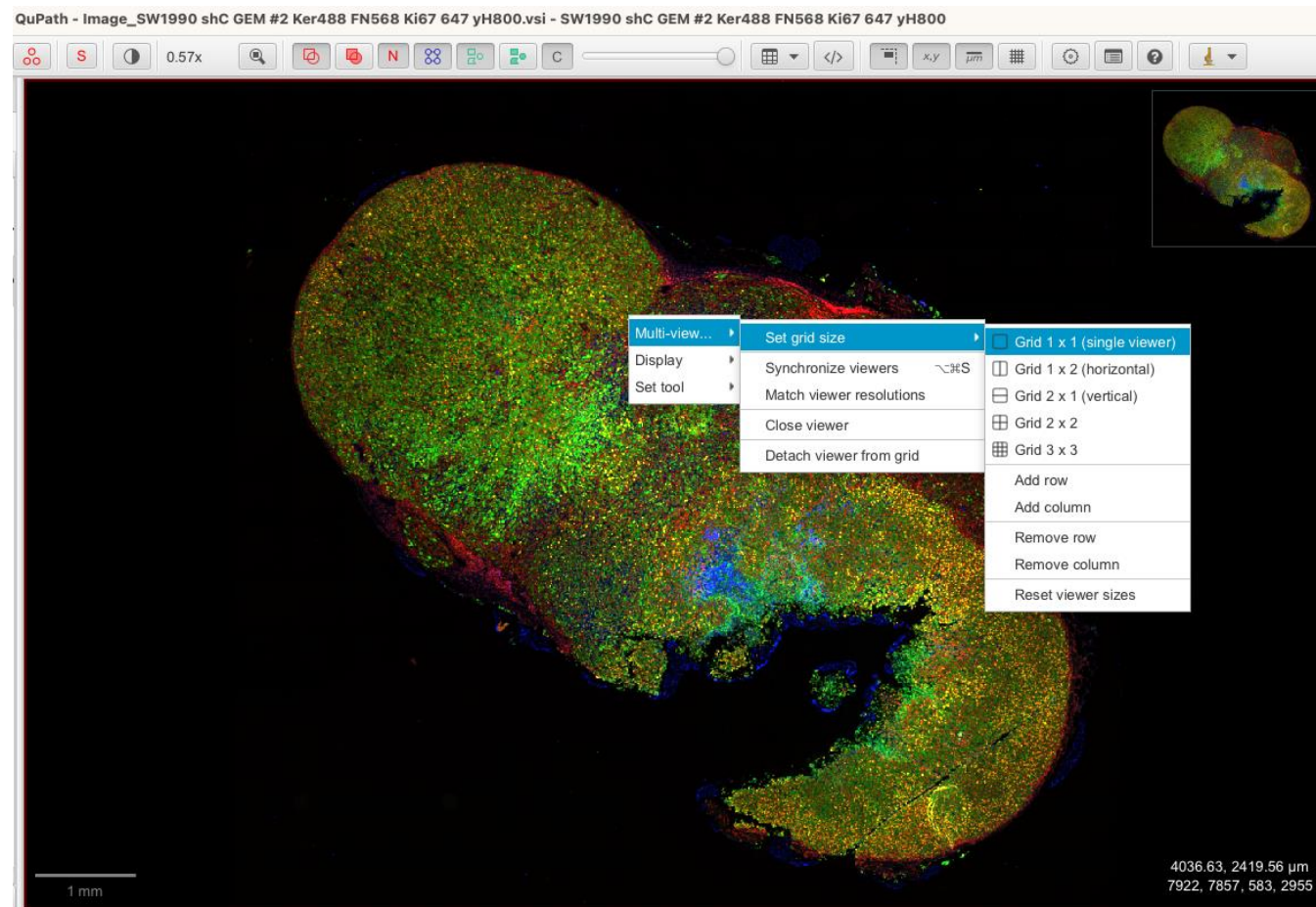
Info bar: pixel coord and value

Scale bar

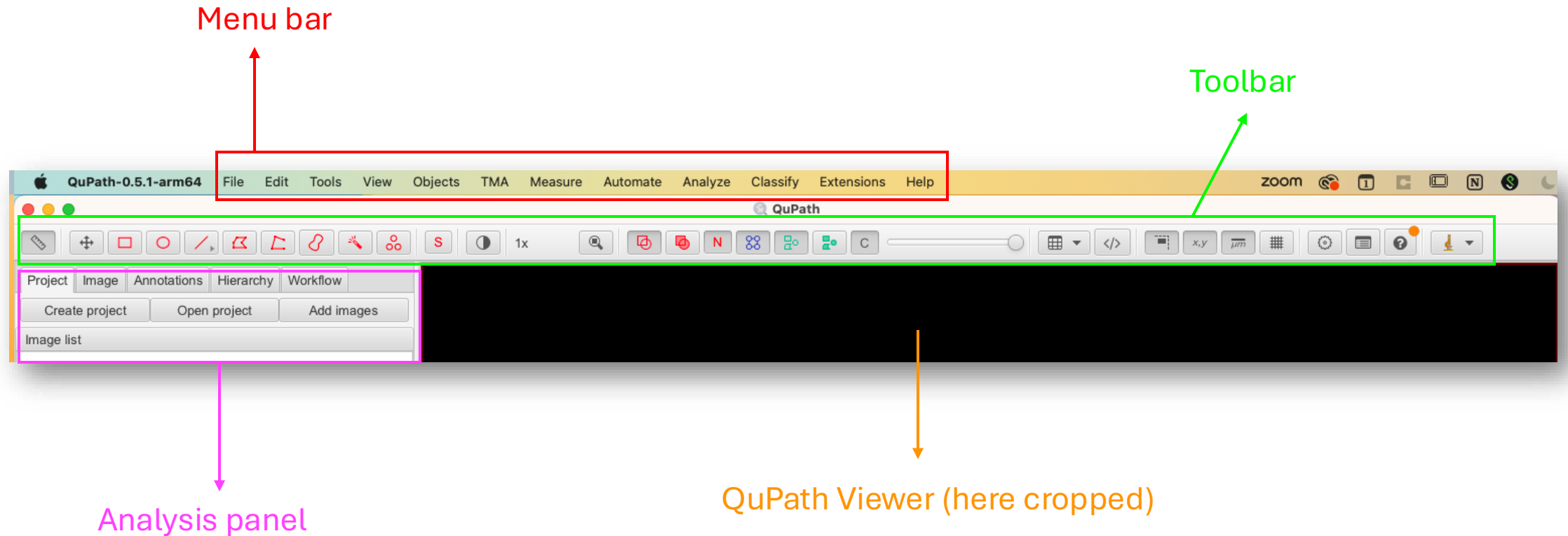
Image Analysis Collaboratory - QuPath workshop

Multi-viewer

- Right-click in the viewer

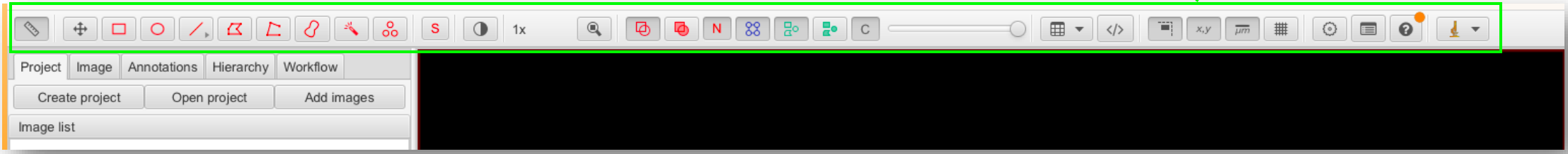
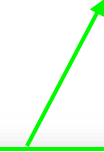


Graphic User Interface (GUI)

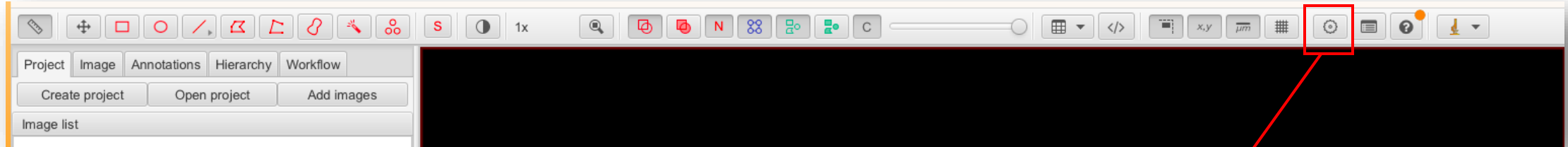


Toolbar

Toolbar

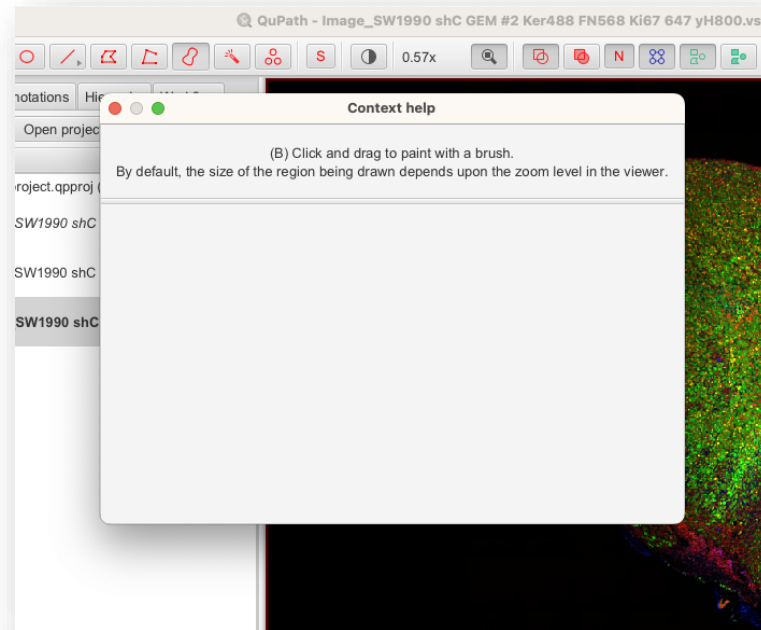
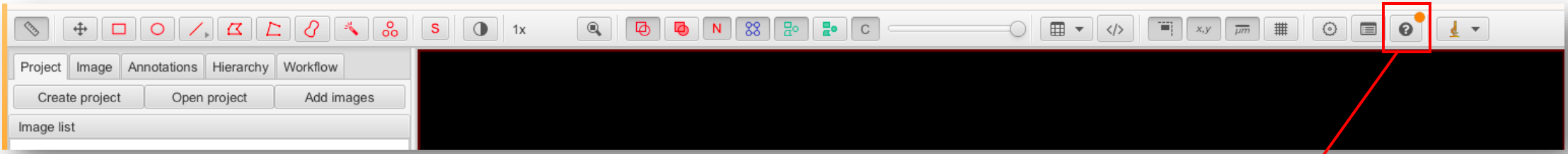


Toolbar



Preferences
Settings, GUI
customization,
extensions, ...

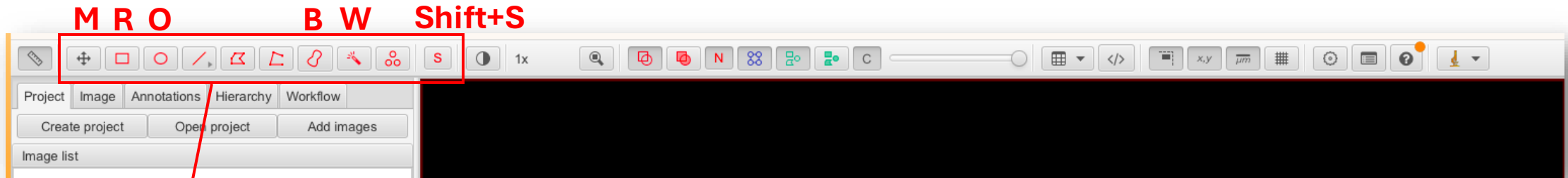
Toolbar



Interactive Help
Provides contextual help based on your cursor location

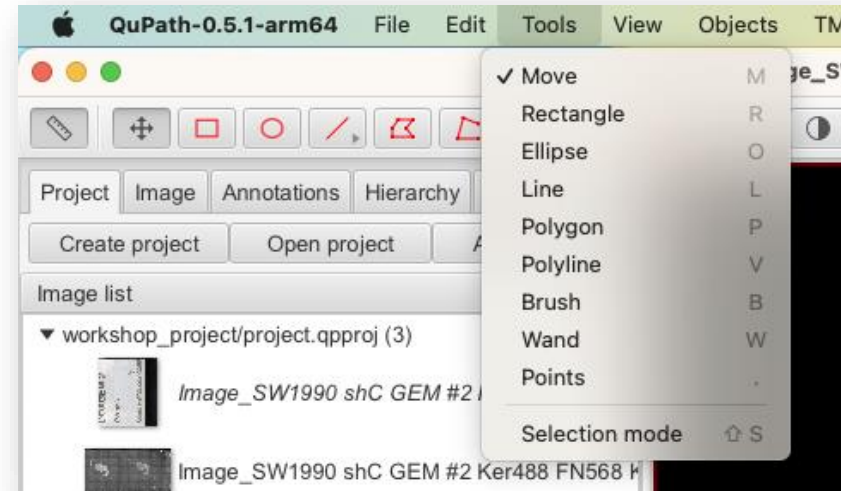
Example when my cursor is on the paint brush tool

Toolbar



Annotation tools

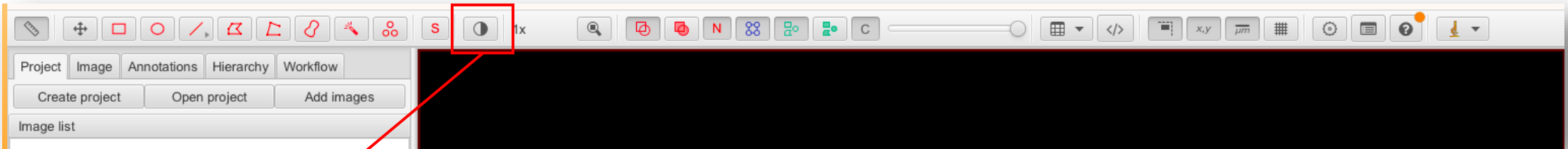
- **M** key: move tool
- **R** key: draw a rectangle annotation
- **O** key: draw an ellipse annotation
- **B** key: paint with a brush
- **W** key: draw with a wand tool
- And many more!



Annotation tools are also accessible in the *Tools* menu

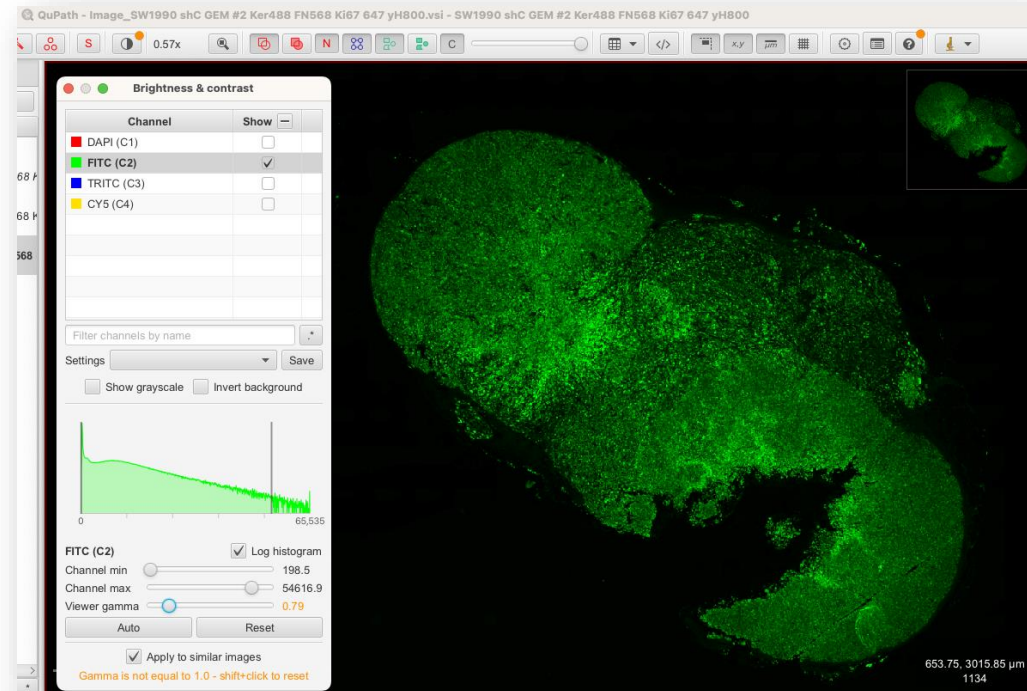
Toolbar

Shift+C



Brightness and contrast

- Toggle on/off channels
- Adjust LUT range
- Visualize intensity histogram



Example for FITC (Keratin) channel

Practice time

Exercises 1: QuPath projects and GUI

A fluorescence microscopy image of a cell, likely a yeast cell, showing a green fluorescent signal. The cell is outlined in yellow. Overlaid on the cell are several colored outlines: a blue 'U' shape, a purple 'P' shape, a red 'A' shape, a teal 'T' shape, and a green 'H' shape. The text 'Introducing objects: annotations and detections' is centered over the cell in white. In the top right corner, there is a small inset image showing a zoomed-in view of a specific region of the cell, with a red box indicating the area of interest.

Introducing objects: annotations and detections

Key concept: QuPath objects

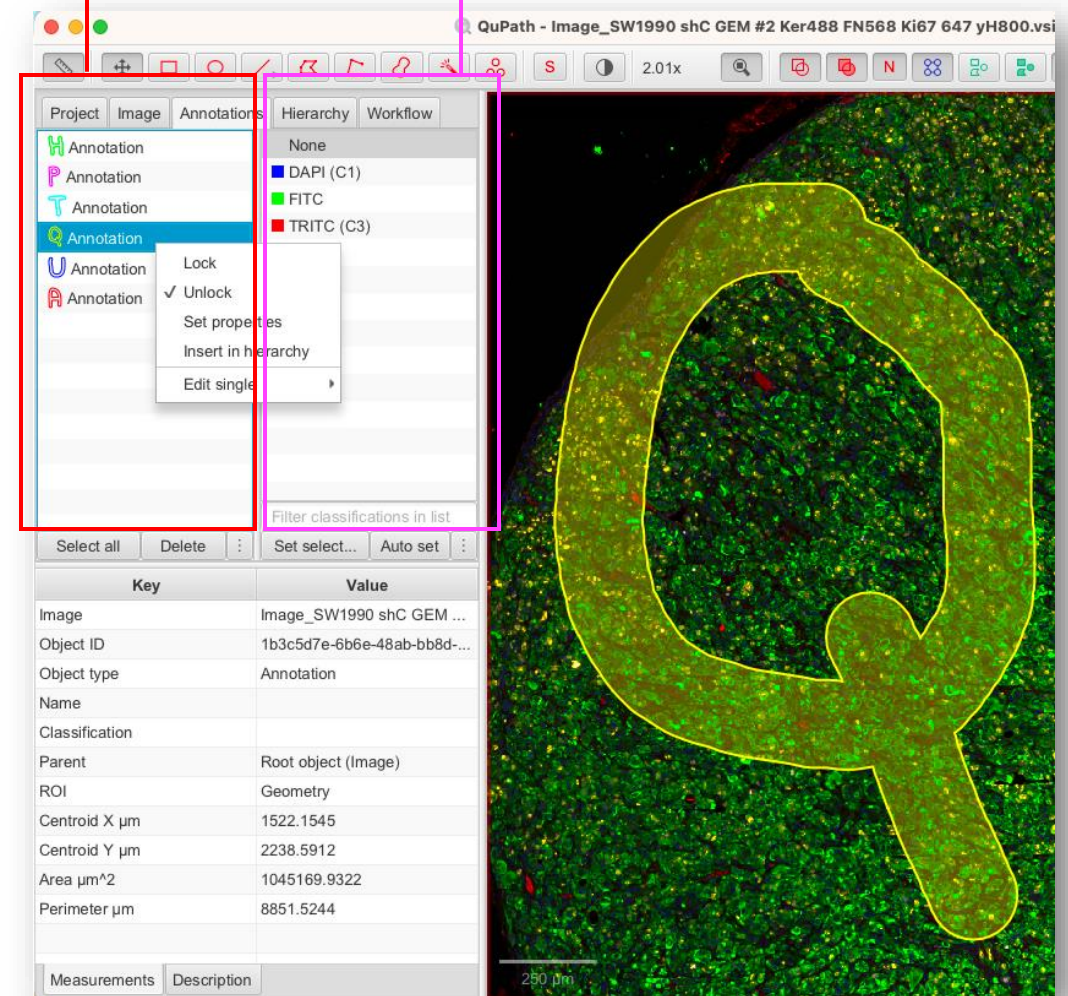
- **Objects** are a ‘thing’ in an image which encapsulates not only its shape but also some properties about it
 - **Annotations:** Objects that you usually create yourself, by drawing on the image
 - They are flexible, up to ~100 per image
 - Can be edited
 - Often used to define regions
 - **Detections:** Objects that QuPath usually creates for you
 - They are efficient, up to ~millions per image
 - Can be deleted but not edited
 - Often used to define cells

Analysis Panel

- *Annotations* tab
 - Annotation list lets you select, delete
 - Right-click to **lock** or edit properties (name, color)
 - Shift or Command/Control to multi-select

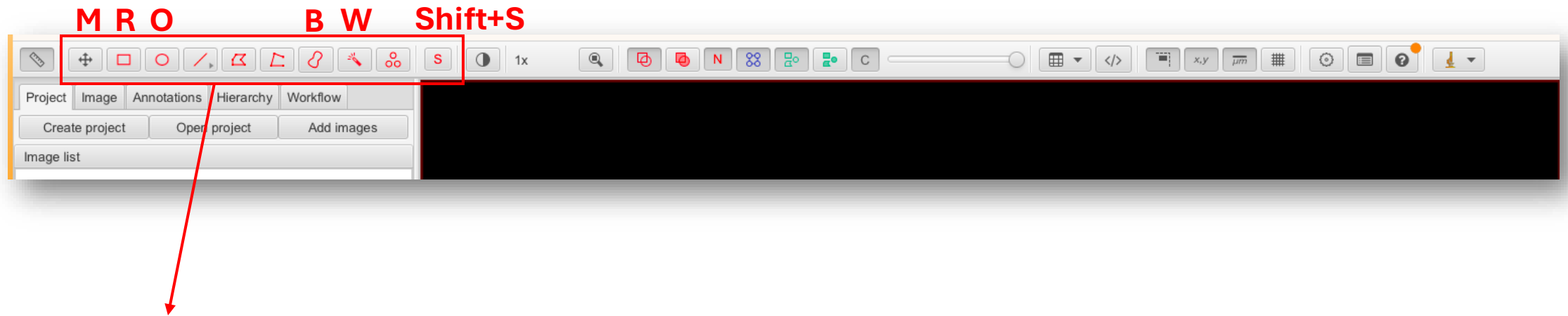
Annotation list

Classification list



How to create manual annotations?

Select one of the annotation tools from the toolbar then scribble on the image!



Annotation tools

- **M** key: move tool
- **R** key: draw a rectangle annotation
- **O** key: draw an ellipse annotation
- **B** key: paint with a brush
- **W** key: draw with a wand tool
- And many more!

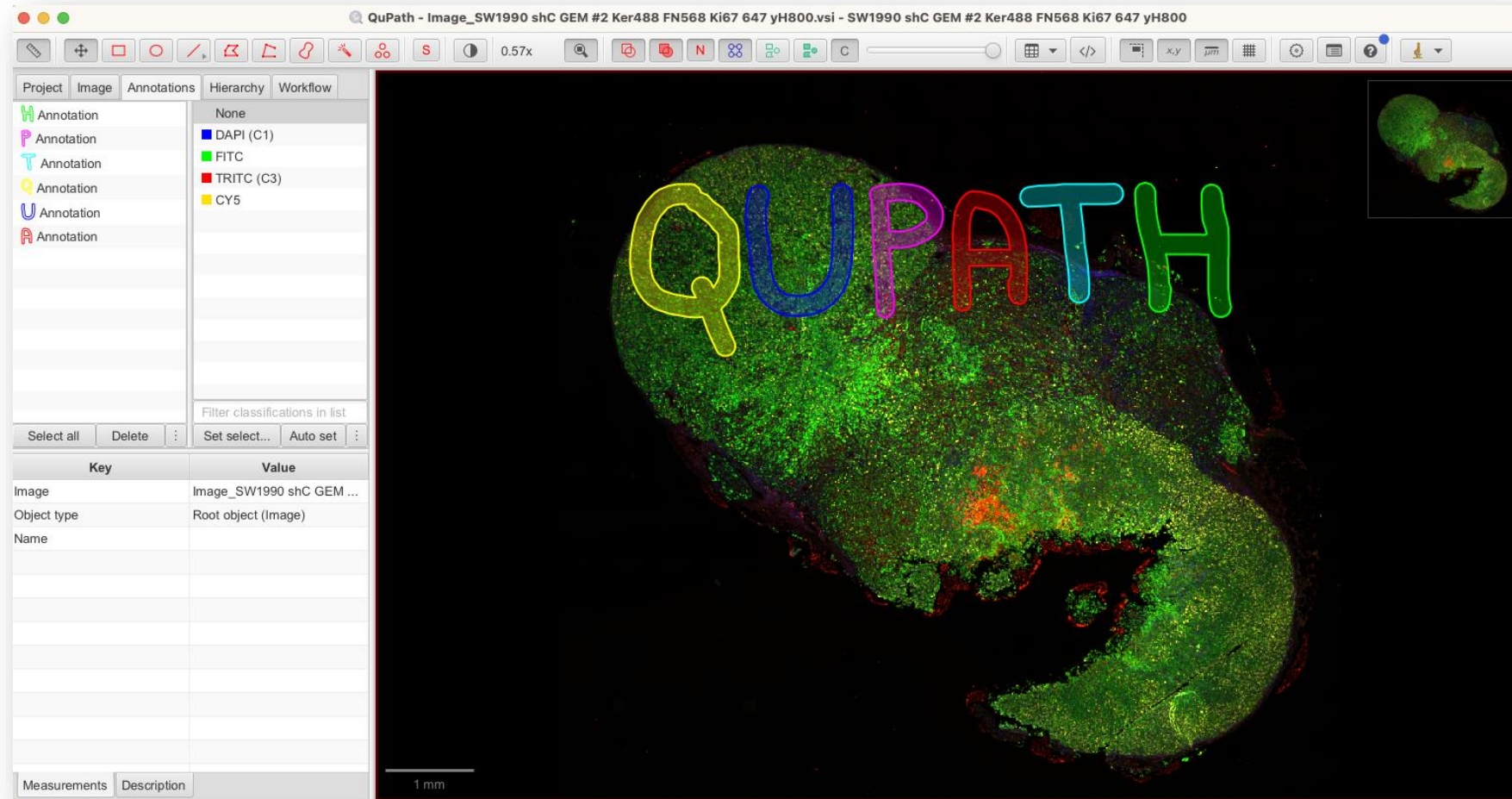
**Remember to always
lock your annotation
to prevent accidental
editing!**

Practice time

Exercises 2: QuPath manual annotations

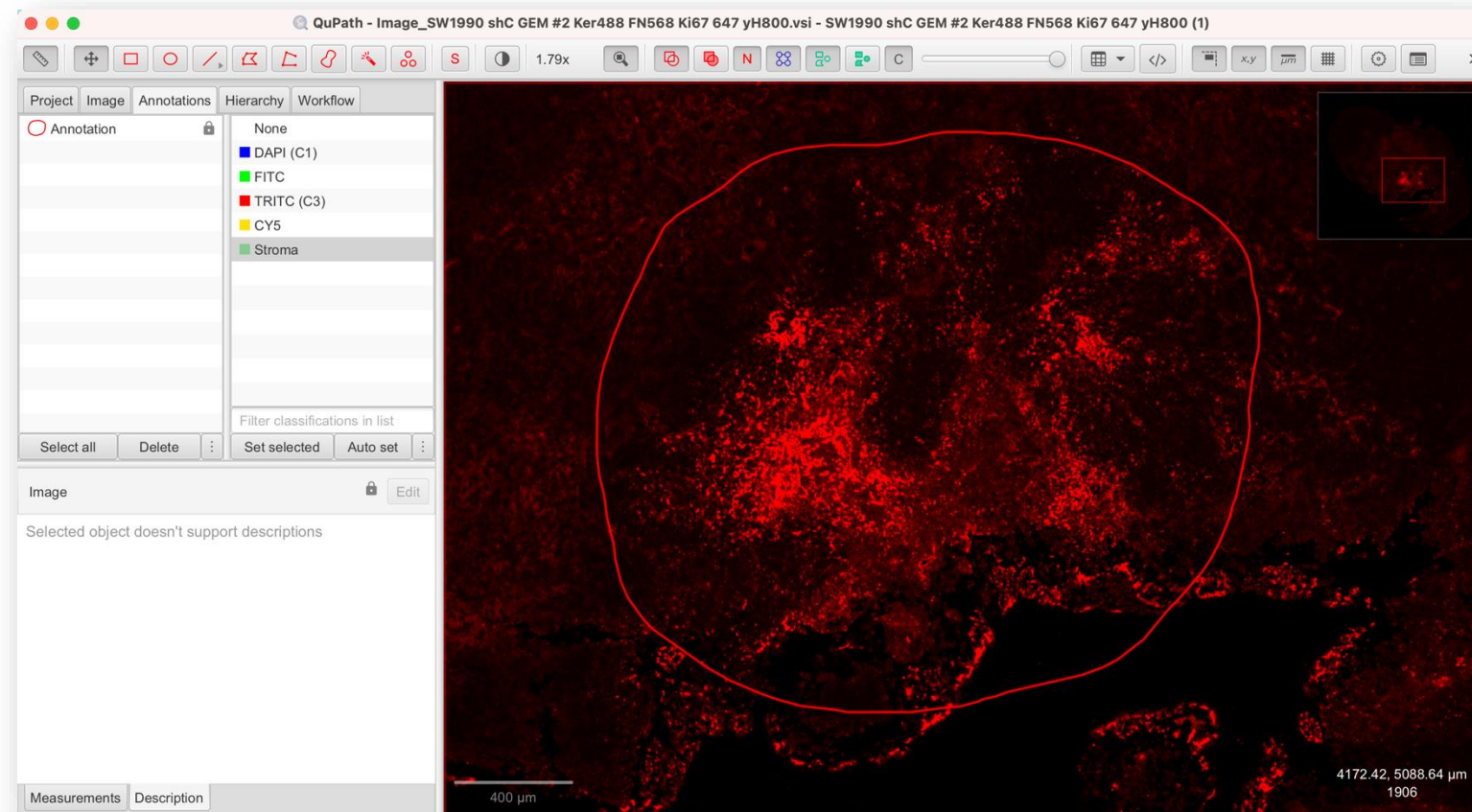
Recreate these annotations

Decide on which annotations tool from the toolbar is best to do so



Create a region of interest with the annotation tools

In the TRITC channel (fibronectin), create a region of interest that enclose high-fibronectin content regions



Once you have finished your annotation, **lock** it:

Right-click in the viewer
> *Annotations* > *Lock*

or

Right-click on the annotation in the analysis panel > *Lock*

A microscopic image of tissue, likely a histological section, showing a dense population of cells. The cell boundaries are highlighted with red outlines, and the nuclei are stained blue. The overall appearance is that of a cellular structure, possibly an epithelial layer or a tumor section. The text "Cell detection" is overlaid in the center in white.

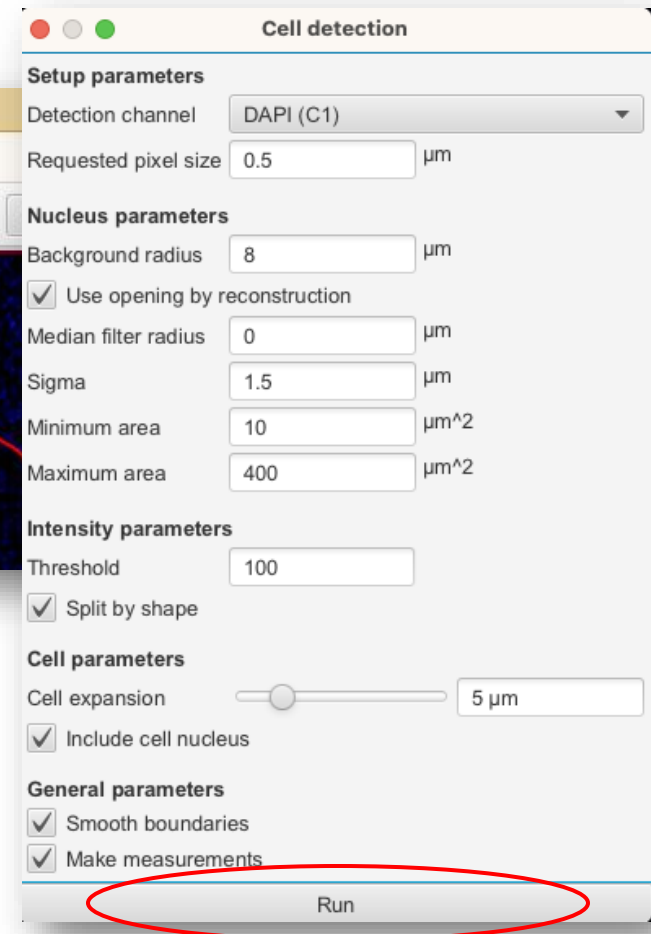
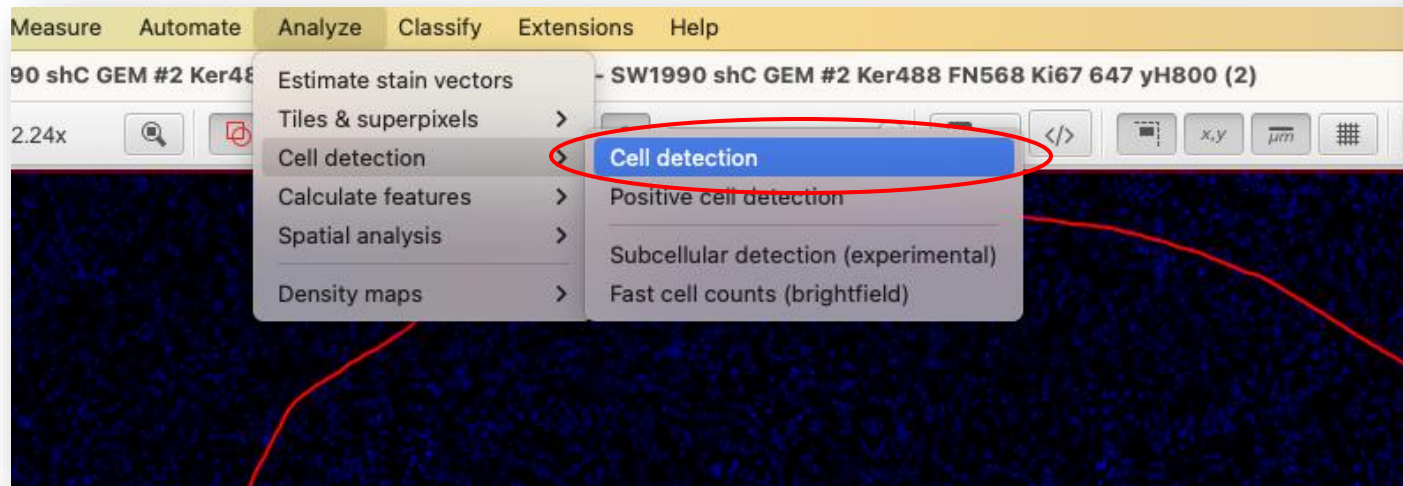
Cell detection

Cell detection

- QuPath offers three main options:
 1. Built-in cell segmentation algorithm, based on nucleus thresholding and cell body expansion
 2. StarDist as an extension (DL)
 3. Cellpose as an extension (DL) – not covered here
- All yield *Cell Detections* objects that will have shape and intensity measurements for nucleus, cell and membrane
- Detection can be computationally intensive so we will start from the region of interest

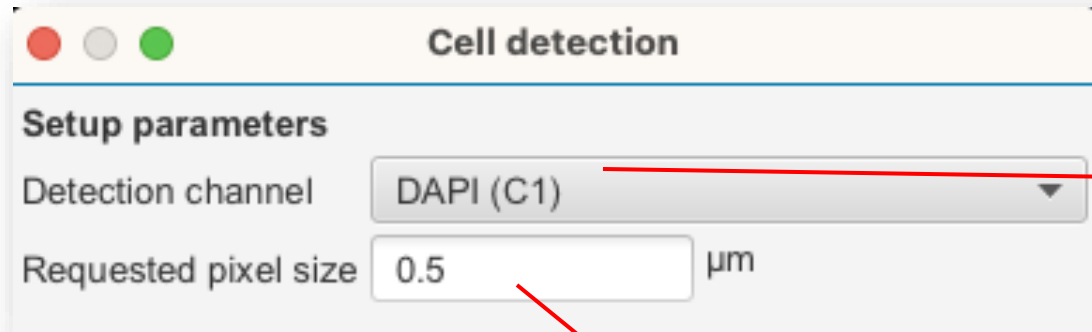
Cell detection

1. Built-in cell segmentation algorithm, based on nucleus thresholding and cell body expansion



Note that positive cell detection allows for multi-class segmentation on the fly

Cell detection parameters



Channel e.g. DAPI

The resolution of the image used in the segmentation algorithm

- Enter **0** for full resolution
- Default **0.5** typically good trade-off between cost and details

Cell detection parameters

Radius of median filter
(smoothing):

- Enter **0** to disable
- Higher values will smooth off details

Radius of gaussian filter
(smoothing):

- Enter **0** to disable
- Higher values will smooth off details

Nucleus parameters

Background radius μm

Use opening by reconstruction

Median filter radius μm

Sigma μm

Minimum area μm²

Maximum area μm²

Radius of area used for
background subtraction

Allowed area interval for
detections; nuclei detection is
removed if outside of the interval

Cell detection parameters

Uses roundness of detections shape to split clusters/clumps; keep it ticked for most usages

Intensity parameters

Threshold

Split by shape

Cell parameters

Cell expansion

Include cell nucleus

General parameters

Smooth boundaries

Make measurements

Run

Minimum signal intensity of nuclei relative to background

How much to expand nuclei to get cell boundaries

- Enter **0** to disable
- Enter small values **0 – 2** for peri-nuclear measurements
- Enter values **~5** for cytoplasm measurements, depending on tissues

If ticked, will generate measurements specific to each detected nuclei and inferred cytoplasm

Note on the hierarchy of objects in QuPath

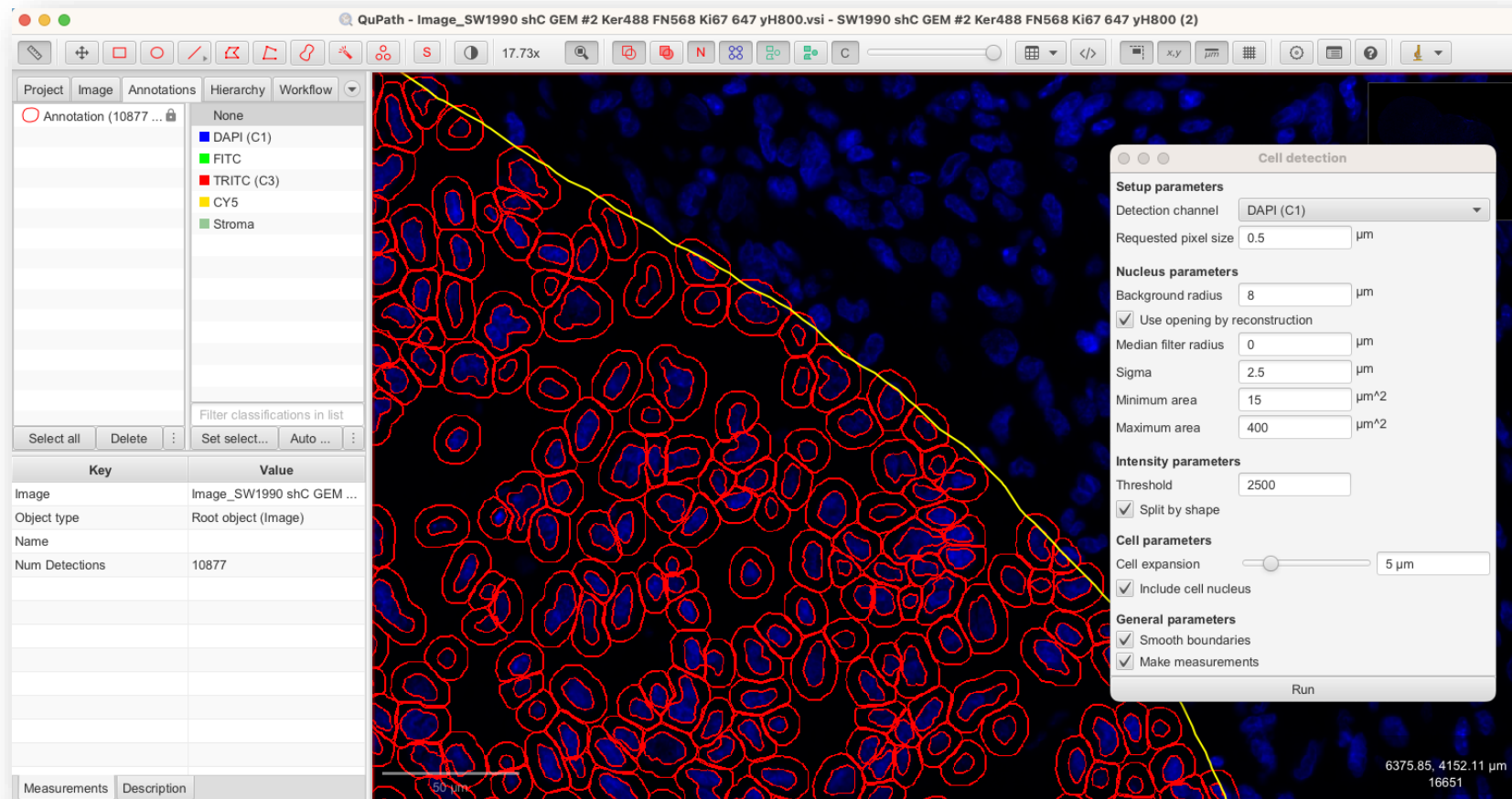
- QuPath allows to nest objects in one another to organize your projects
 - Child-parent link
 - Very useful to organize and restrict the analysis to parts of an image
 - Can be used to restrict image processing within a ROI or a detected tissue region

Practice time

Exercise 3.a: QuPath cell detection

Exercise: explore parameters

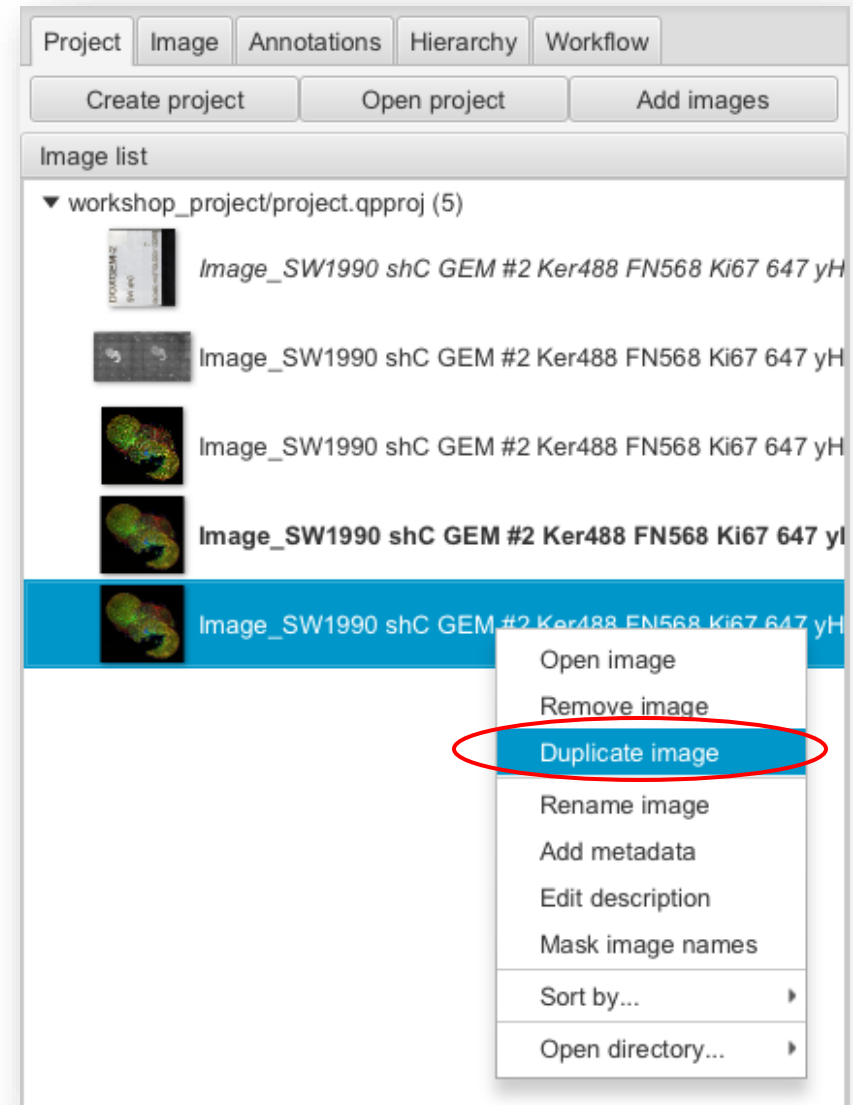
- I found that default parameters tend to over-segment nuclei so adapted the parameters to be slightly stricter (min area and threshold increased)



Duplicate your image

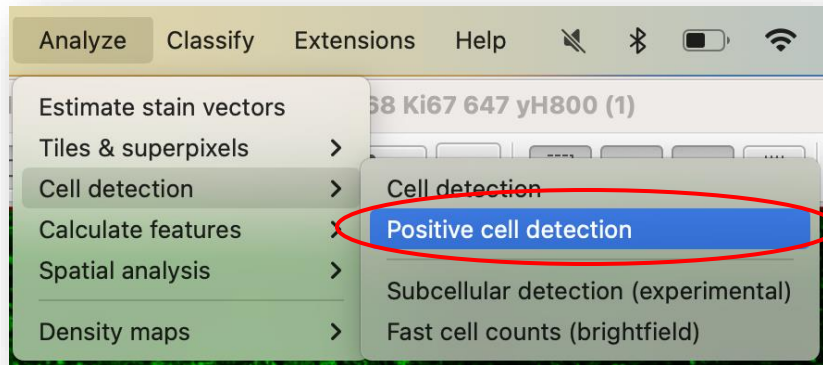
- Copy your cell detection results for future work on it
- *Project tab > Image list > Option+click or right-click on the image name > Duplicate image*

**It duplicates QuPath objects,
not the actual image**

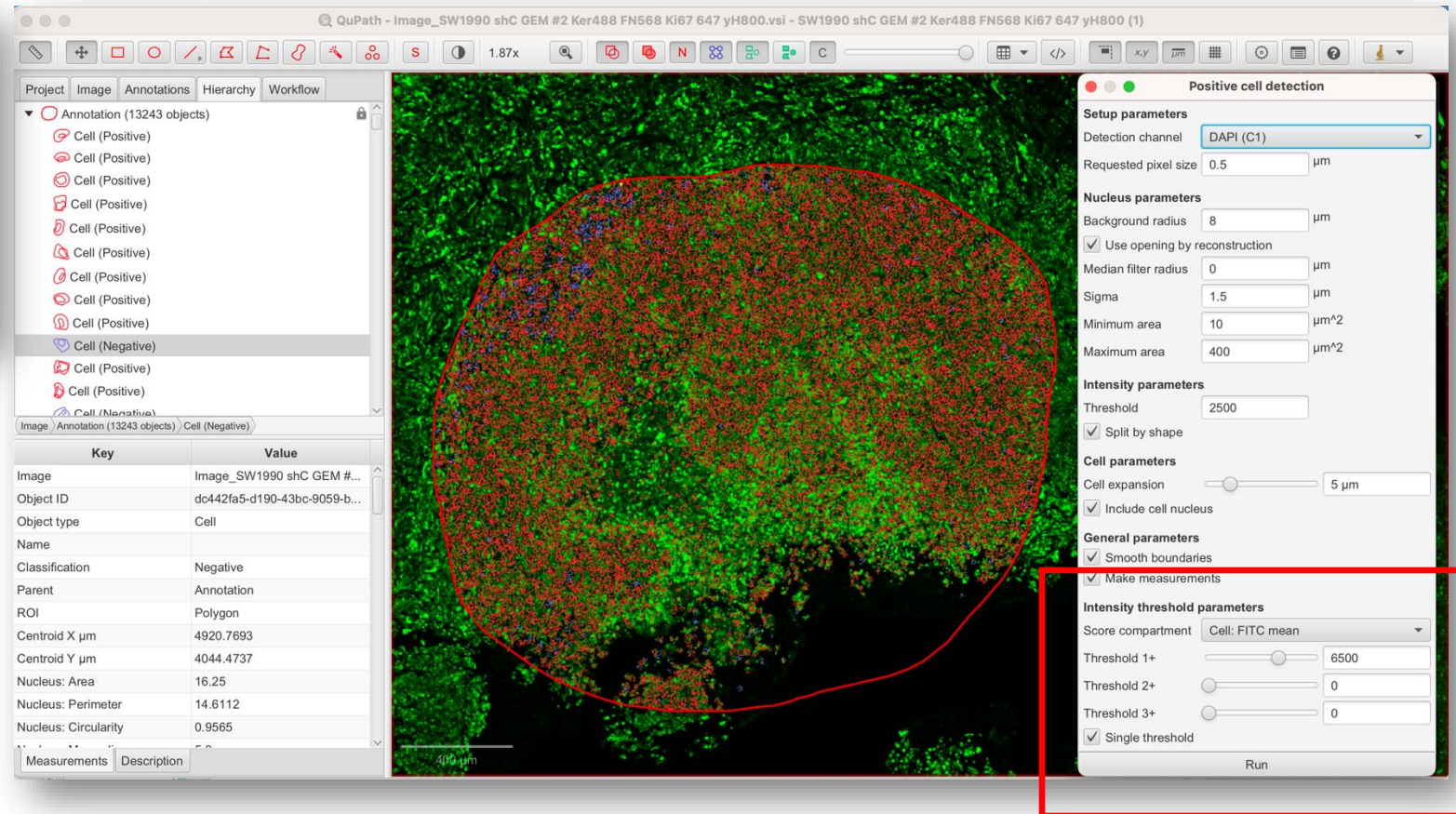


Detecting cells with an extra condition

- *Analyze > Cell detection > Positive cell detection*

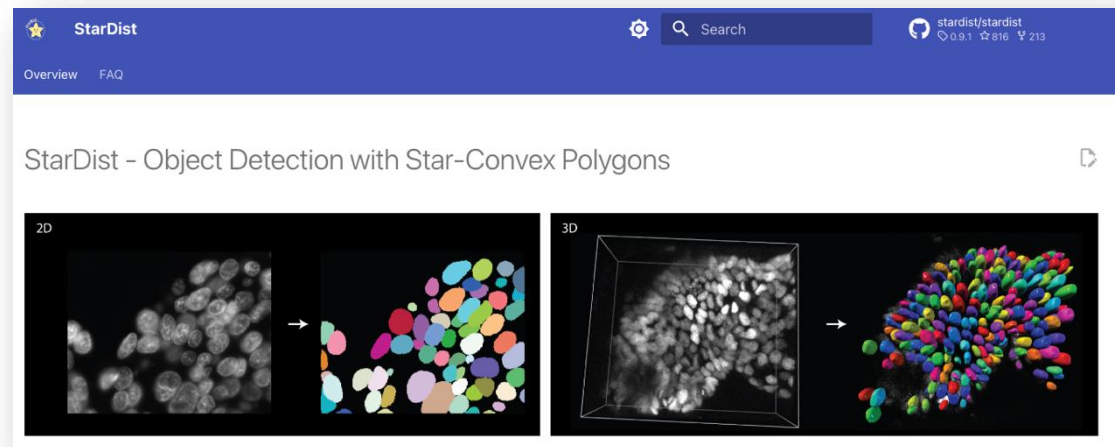


- Adds an extra step of classifying all cells as positive or negative immediately according to staining intensity



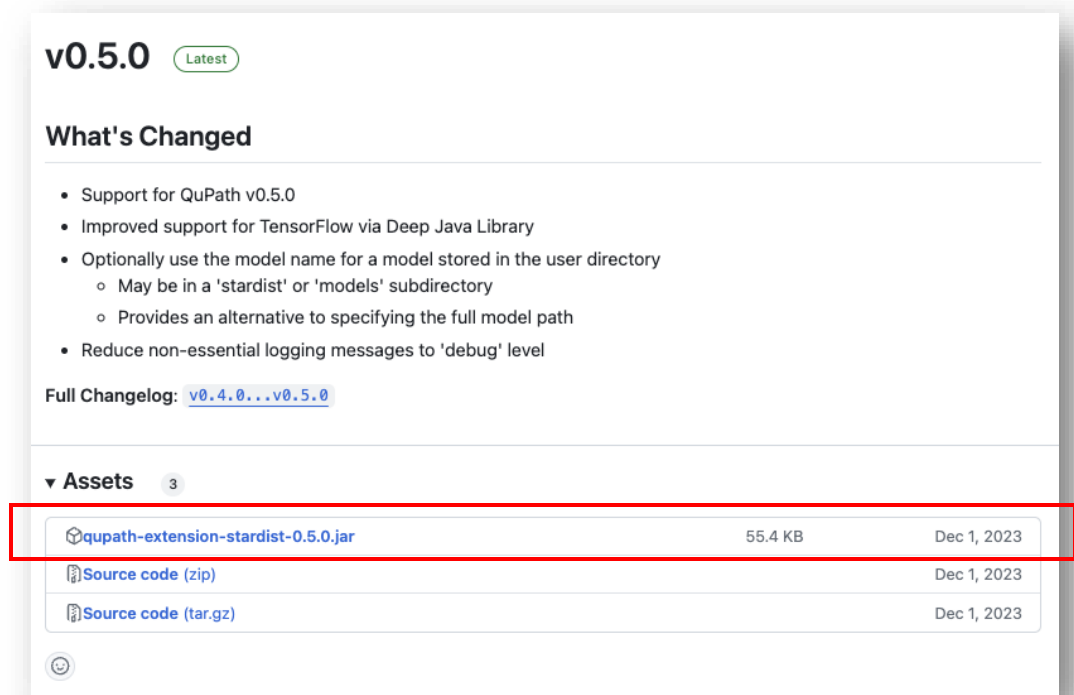
Deep learning-based cell segmentation

- DL-based methods can typically capture more complex patterns, tend to mitigate human bias such as threshold hand-picking
- **However**, they are more computationally expensive and often need fine-tuning or re-training for specific applications
- StarDist is a deep learning model trained to detect specific kinds of nuclei in different kinds of image



Installing StarDist extension in QuPath

- Browse to <https://github.com/qupath/qupath-extension-stardist/releases>
- Download the .jar file compatible with your QuPath version
 - For this workshop, get [qupath-extension-stardist-0.5.0.jar](#)
- Drag and drop the .jar file onto QuPath main window, and... that's it!



v0.5.0 Latest

What's Changed

- Support for QuPath v0.5.0
- Improved support for TensorFlow via Deep Java Library
- Optionally use the model name for a model stored in the user directory
 - May be in a 'stardist' or 'models' subdirectory
 - Provides an alternative to specifying the full model path
- Reduce non-essential logging messages to 'debug' level

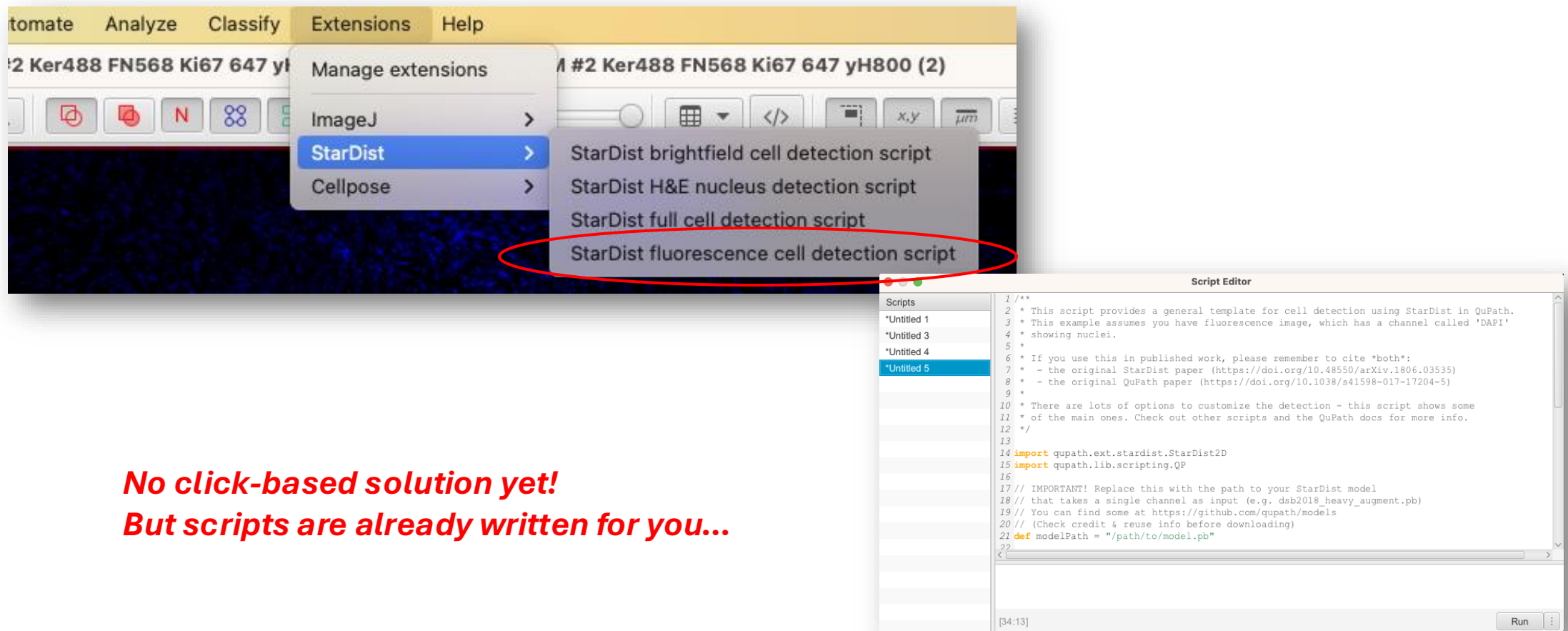
Full Changelog: [v0.4.0...v0.5.0](#)

▼ **Assets** 3

qupath-extension-stardist-0.5.0.jar	55.4 KB	Dec 1, 2023
Source code (zip)		Dec 1, 2023
Source code (tar.gz)		Dec 1, 2023

Using StarDist extension in QuPath

- Go to *Extensions* tab > *StarDist* > *StarDist fluorescence cell detection script*



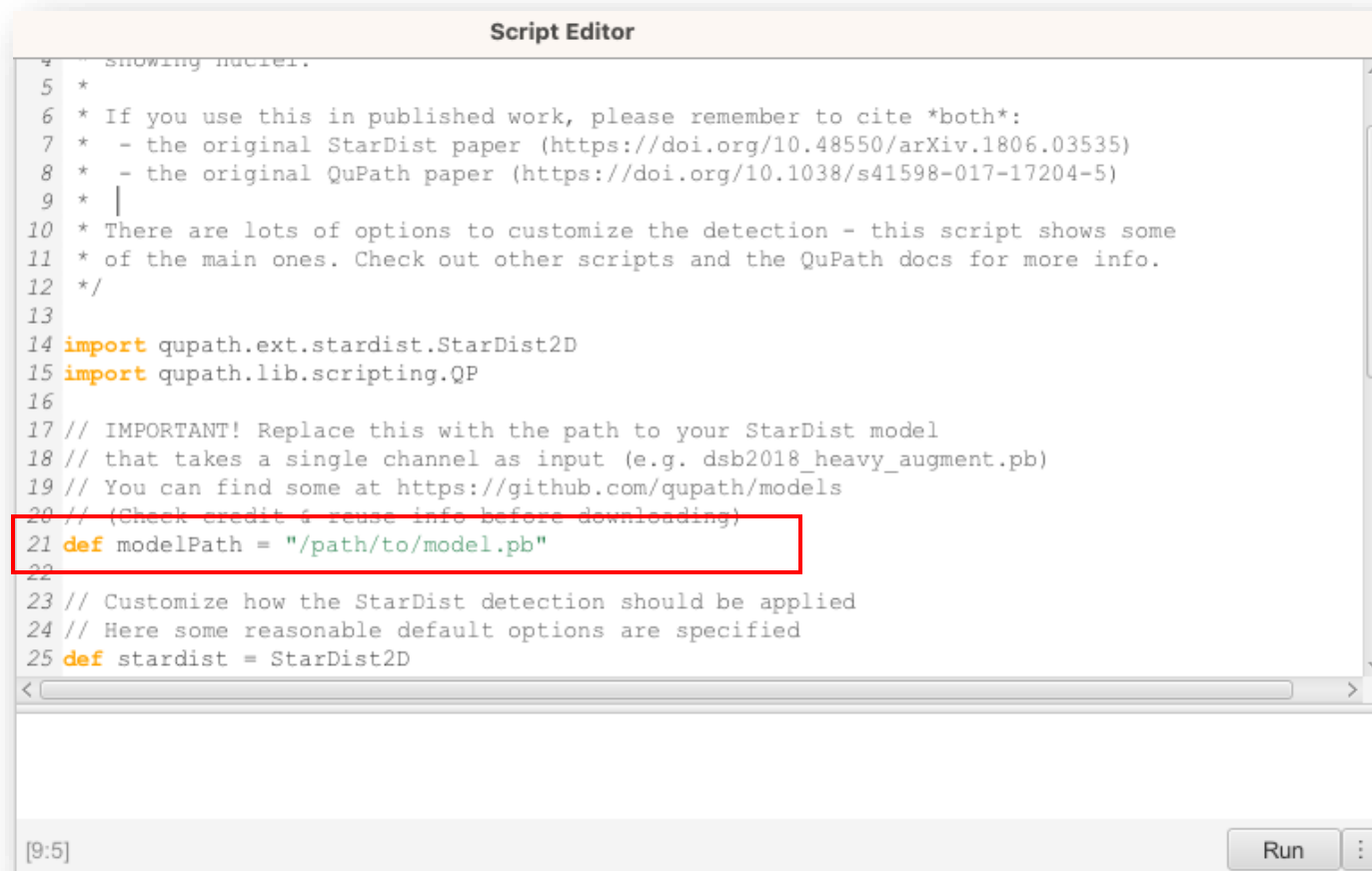
The screenshot shows the QuPath software interface. The 'Extensions' menu is open, and the 'StarDist' option is selected. A sub-menu is displayed, listing several StarDist scripts. The 'StarDist fluorescence cell detection script' is circled in red. Below the main interface, a 'Script Editor' window is open, displaying the code for the selected script. The code includes comments and imports for the StarDist extension.

```
1 /**
2  * This script provides a general template for cell detection using StarDist in QuPath.
3  * This example assumes you have fluorescence image, which has a channel called 'DAPI'
4  * showing nuclei.
5  *
6  * If you use this in published work, please remember to cite *both*:
7  * - the original StarDist paper (https://doi.org/10.48550/arXiv.1806.03535)
8  * - the original QuPath paper (https://doi.org/10.1038/s41598-017-17204-5)
9  *
10 * There are lots of options to customize the detection - this script shows some
11 * of the main ones. Check out other scripts and the QuPath docs for more info.
12 */
13
14 import qupath.ext.stardist.StarDist2D
15 import qupath.lib.scripting.QP
16
17 // IMPORTANT! Replace this with the path to your StarDist model
18 // that takes a single channel as input (e.g. dsb2018_heavy_augment.pb)
19 // You can find some at https://github.com/qupath/models
20 // (Check credit & reuse info before downloading)
21 def modelPath = "/path/to/model.pb"
22
23
```

No click-based solution yet!
But scripts are already written for you...

Using StarDist extension in QuPath

- Requires to load a pre-trained model (basically the weights)



```
4  * showing nuclei.
5  *
6  * If you use this in published work, please remember to cite *both*:
7  * - the original StarDist paper (https://doi.org/10.48550/arXiv.1806.03535)
8  * - the original QuPath paper (https://doi.org/10.1038/s41598-017-17204-5)
9  * |
10 * There are lots of options to customize the detection - this script shows some
11 * of the main ones. Check out other scripts and the QuPath docs for more info.
12 */
13
14 import qupath.ext.stardist.StarDist2D
15 import qupath.lib.scripting.QP
16
17 // IMPORTANT! Replace this with the path to your StarDist model
18 // that takes a single channel as input (e.g. dsb2018_heavy_augment.pb)
19 // You can find some at https://github.com/qupath/models
20 // (Check credit & reuse info before downloading)
21 def modelPath = "/path/to/model.pb"
22
23 // Customize how the StarDist detection should be applied
24 // Here some reasonable default options are specified
25 def stardist = StarDist2D
```

Note: StarDist is rather computationally expensive, typically can take ~ 5 min for 100k detections

StarDist for 2D segmentation of DAPI-stained nuclei

- Some pre-trained StarDist models are freely available as *.pb* files (frozen)
- Go to <https://github.com/qupath/models/raw/main/stardist> and download the `dsb2018_heavy_augment.pb` model

StarDist models

Here you can find pre-trained StarDist models as frozen *.pb* files that are compatible with OpenCV's DNN module.

This means they can be used in QuPath via the [QuPath StarDist extension](#) without any requirement to install TensorFlow.

Downloads

The converted model files are

- [dsb2018_heavy_augment.pb](#) - single channel
- [dsb2018_paper.pb](#) - single channel
- [he_heavy_augment.pb](#) - RGB images

***dsb2018_heavy_augment.pb* is pre-trained for 2D fluorescence images (one detection channel)**

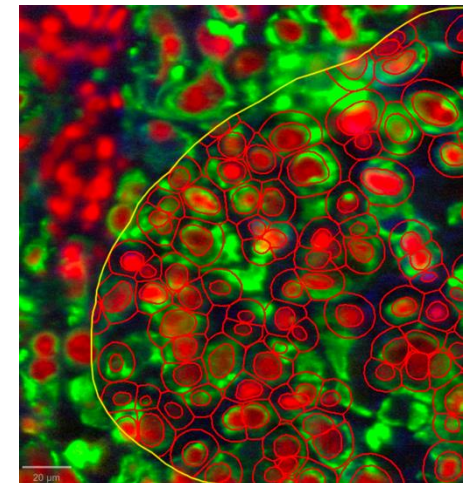
Using StarDist extension in QuPath

- Change the value of the **modelPath** variable to an actual StarDist model path in the script

Change the
channel
name

```
Script Editor
19 // You can find some at https://github.com/qupath/models
20 // (Check credit & reuse info before downloading)
21 def modelPath = "/Users/antoine/Desktop/test_qupath_workshop/models/dsb2018_heavy_augment.pb"
22
23 // Customize how the StarDist detection should be applied
24 // Here some reasonable default options are specified
25 def stardist = StarDist2D
26     builder(modelPath)
27     .channels('DAPI (C1)') // Extract channel called 'DAPI'
28     .normalizePercentiles(1, 99) // Percentile normalization
29     .threshold(0.5) // Probability (detection) threshold
30     .pixelSize(0.5) // Resolution for detection
31     .cellExpansion(5) // Expand nuclei to approximate cell boundaries
32     .measureShape() // Add shape measurements
33     .measureIntensity() // Add cell measurements (in all compartments)
34     .build()
35
INFO: Done!
[27:24] Stopped: 0:00:12
```

Make sure to
select the ROI in
QuPath before
running the script.

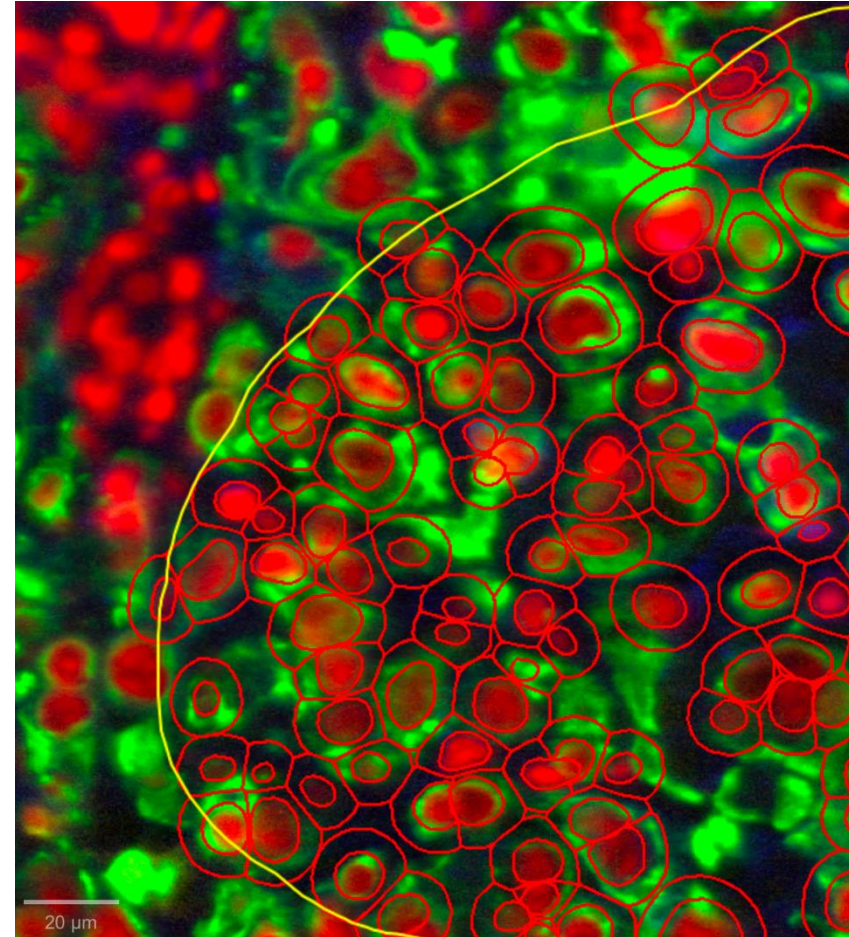


Allow cell boundaries to bleed over the ROI

- Add `.constrainToParent(false)`

```
def stardist = StarDist2D
    .builder(modelPath)
    .channels('DAPI')
    .normalizePercentiles(1, 99)
    .threshold(0.5)
    .pixelSize(0.5)
    .cellExpansion(5)
    .measureShape()
    .measureIntensity()
    .constrainToParent(false)
    .build()
```

Add this
line →

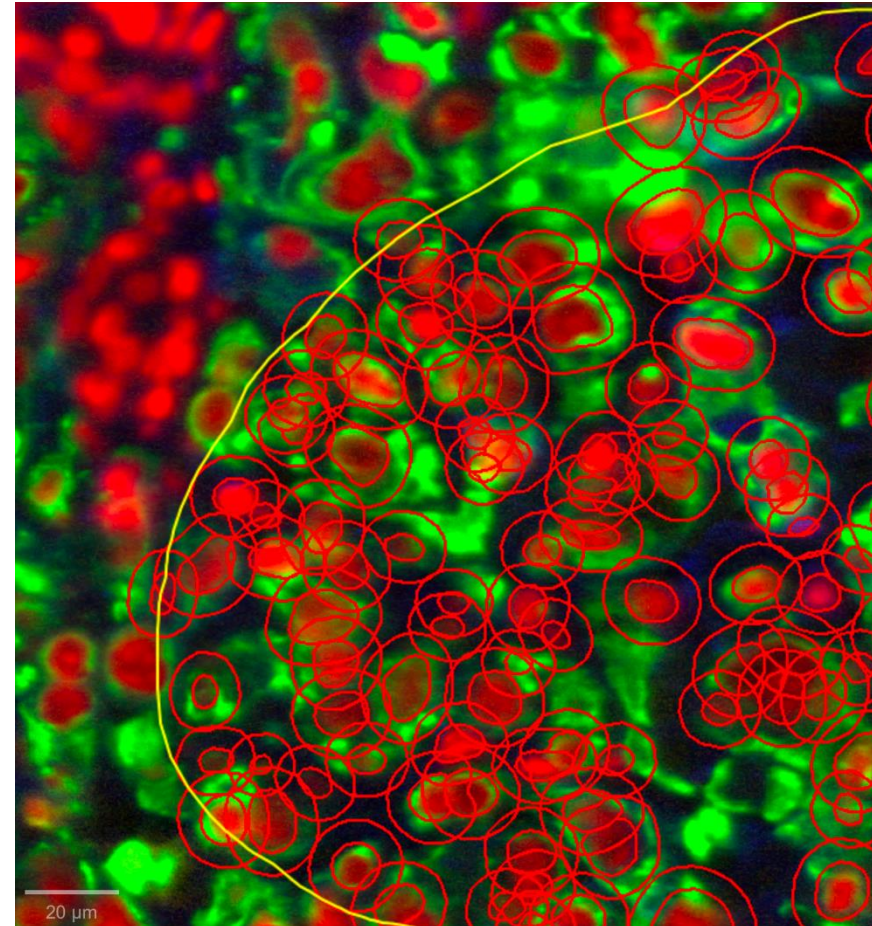


Do not constrain cell expansion with neighbors

- Add `.ignoreCellOverlaps(true)`

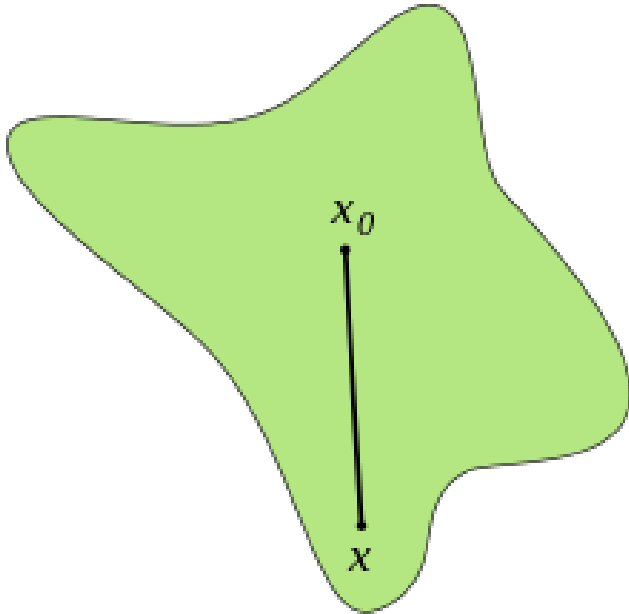
```
def stardist = StarDist2D
    .builder(modelPath)
    .channels('DAPI')
    .normalizePercentiles(1, 99)
    .threshold(0.5)
    .pixelSize(0.5)
    .cellExpansion(5)
    .measureShape()
    .measureIntensity()
    .ignoreCellOverlaps(true)
    .build()
```

Add this
line

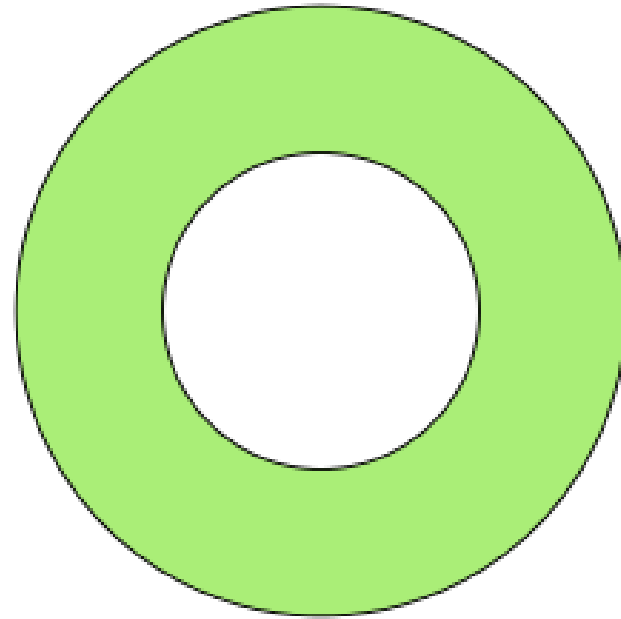


Exercise 3.b: QuPath cell detection with StarDist

Compare StarDist to threshold-based cell detection, what do you observe?



StarDist can segment



StarDist can **not** segment

A microscopic image of a tissue section, likely stained with hematoxylin and eosin (H&E). The image shows a dense population of cells. The cell nuclei are stained blue, and the cytoplasm and extracellular matrix are stained pink. The image has been processed to highlight the cell boundaries with yellow and blue outlines, indicating a segmentation or detection process. The text "Cell detection measurements" is overlaid in the center of the image.

Cell detection measurements

Detection measurements

- Each detection object (i.e. a cell) has its measurement list
 - Intensity features
 - Haralick (texture) features
 - Shape features
 - Smoothed features
- *Annotations* tab > select a cell in the viewer > inspect its measurements list

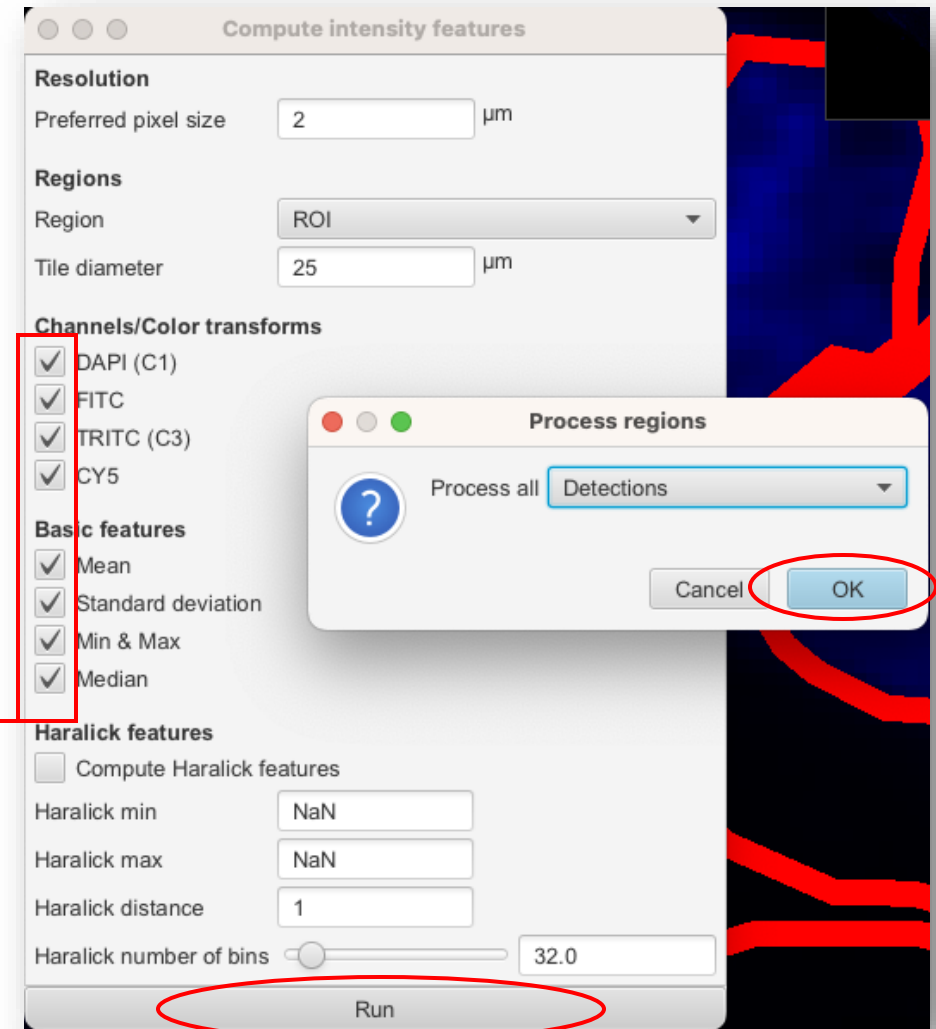
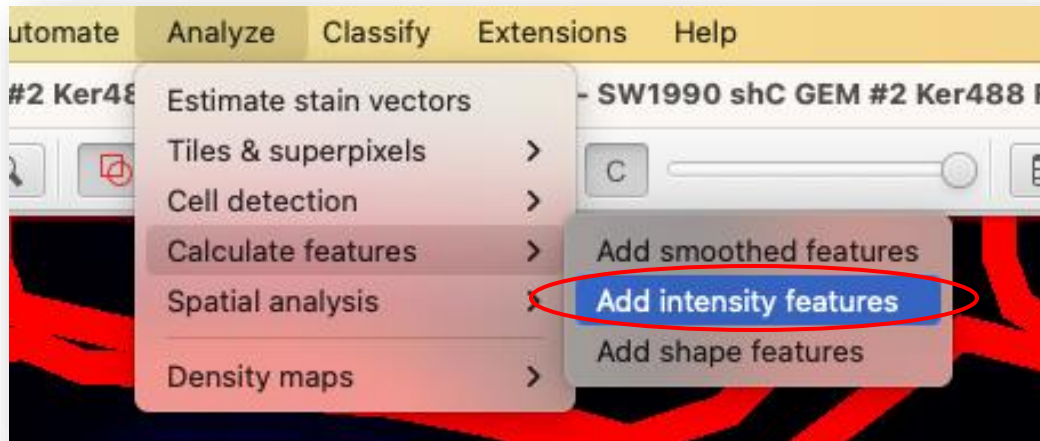
By default, basic intensity and shape features are calculated

The screenshot displays the QuPath software interface. On the left, the 'Annotations' tab is active, showing a list of detection objects. The 'Measurements' tab is selected, displaying a table of features for a specific cell. The table includes various intensity and shape features, with a red box highlighting the 'Nucleus' related measurements. On the right, a microscopy image shows several cells with their nuclei outlined in red and yellow. A 5 µm scale bar is visible in the bottom right corner of the image.

Key	Value
Image	Image_SW1990 shC GE...
Object ID	30b2c2ac-f5bb-4a97-8063...
Object type	Cell
Name	
Classification	
Parent	Annotation
ROI	Polygon
Centroid X µm	5196.1596
Centroid Y µm	4629.3682
Nucleus: Area	53
Nucleus: Perimeter	30.1735
Nucleus: Circularity	0.7315
Nucleus: Max caliper	12.4038
Nucleus: Min caliper	5.7647
Nucleus: Eccentricity	0.901
Nucleus: DAPI (C1) mean	12956.8555
Nucleus: DAPI (C1) sum	2785724

Calculating measurements

- *Analyze > Calculate features > Add intensity features*



Tick boxes of the channels and features of interest

Need a custom feature? Script it!

Visualizing measurements

- *Measure > Show detection measurements*

The screenshot shows the QuPath software interface. The 'Measure' menu is open, with 'Show detection measurements' highlighted. A red arrow points from this menu item to a table of detection measurements. Another red arrow points from the text 'Columns: measurements' to the table's columns. A third red arrow points from the text 'Rows: cells' to the table's rows.

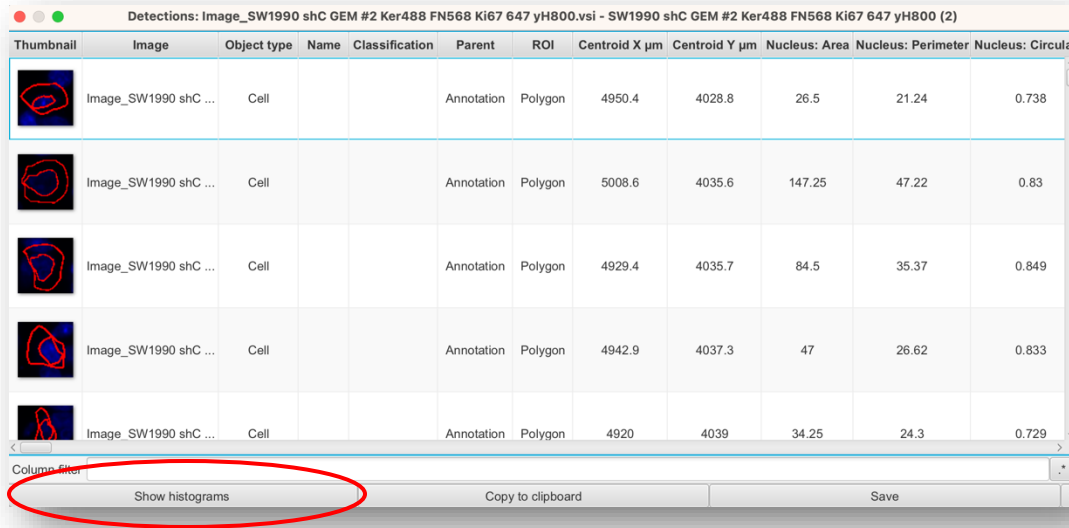
Columns: measurements

Rows: cells

Thumbnail	Image	Object type	Name	Classification	Parent	ROI	Centroid X μm	Centroid Y μm	Nucleus: Area	Nucleus: Perimeter	Nucleus: Circular.
	Image_SW1990 shC ...	Cell			Annotation	Polygon	4950.4	4028.8	26.5	21.24	0.738
	Image_SW1990 shC ...	Cell			Annotation	Polygon	5008.6	4035.6	147.25	47.22	0.83
	Image_SW1990 shC ...	Cell			Annotation	Polygon	4929.4	4035.7	84.5	35.37	0.849
	Image_SW1990 shC ...	Cell			Annotation	Polygon	4942.9	4037.3	47	26.62	0.833
	Image_SW1990 shC ...	Cell			Annotation	Polygon	4920	4039	34.25	24.3	0.729

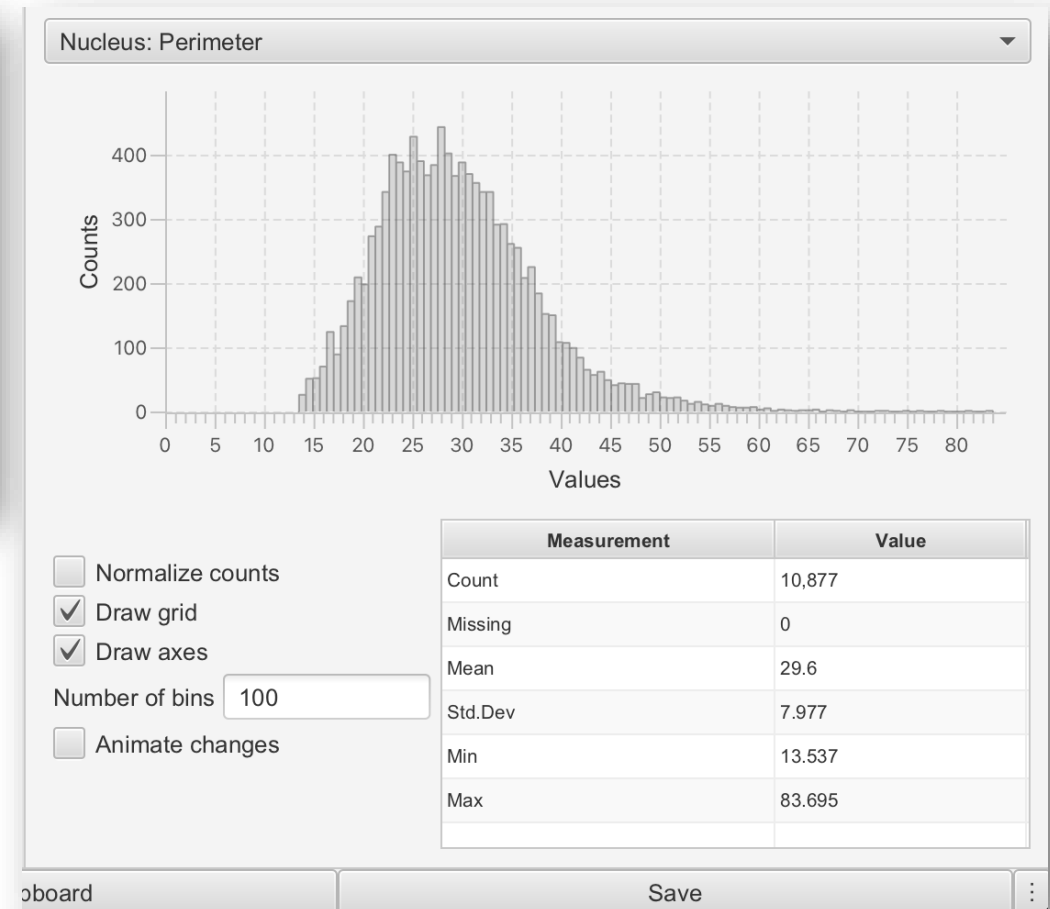
Visualizing measurement distributions

- *Measure > Show detection measurements*



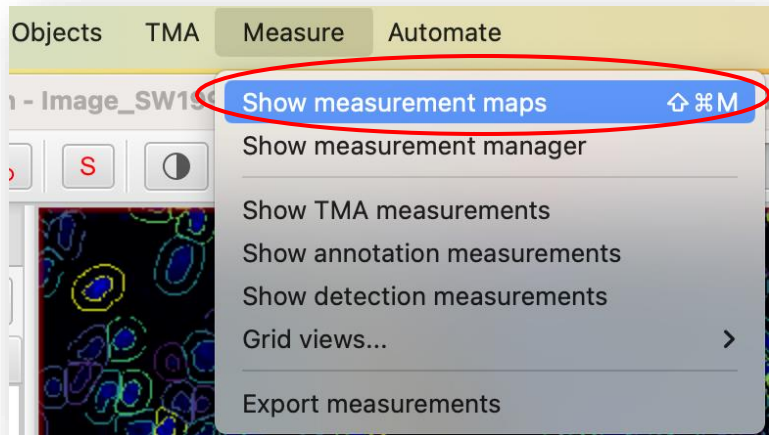
Thumbnail	Image	Object type	Name	Classification	Parent	ROI	Centroid X μm	Centroid Y μm	Nucleus: Area	Nucleus: Perimeter	Nucleus: Circular...
	Image_SW1990 shC ...	Cell			Annotation	Polygon	4950.4	4028.8	26.5	21.24	0.738
	Image_SW1990 shC ...	Cell			Annotation	Polygon	5008.6	4035.6	147.25	47.22	0.83
	Image_SW1990 shC ...	Cell			Annotation	Polygon	4929.4	4035.7	84.5	35.37	0.849
	Image_SW1990 shC ...	Cell			Annotation	Polygon	4942.9	4037.3	47	26.62	0.833
	Image_SW1990 shC ...	Cell			Annotation	Polygon	4920	4039	34.25	24.3	0.729

FYI, it is not possible to export distribution plots...

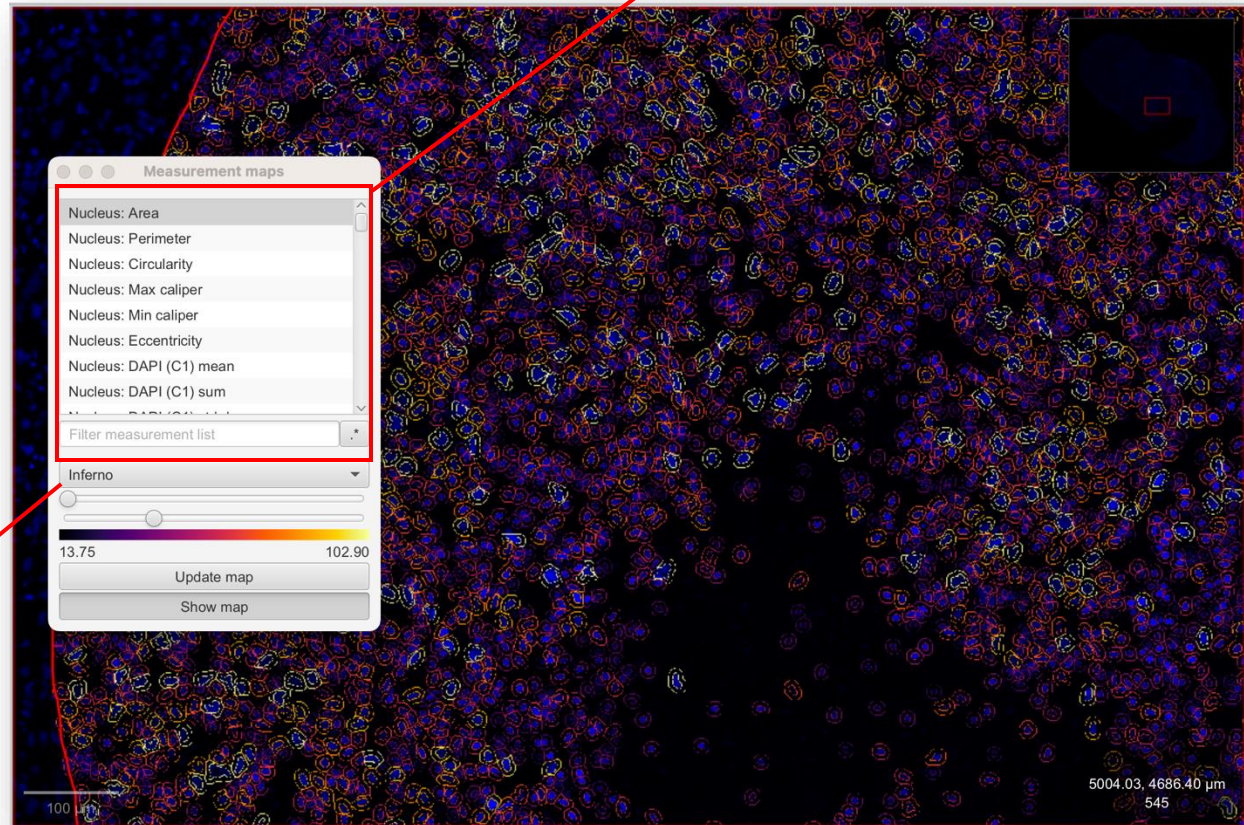


Visualizing measurements as heat maps

- *Measure > Show measurement maps*

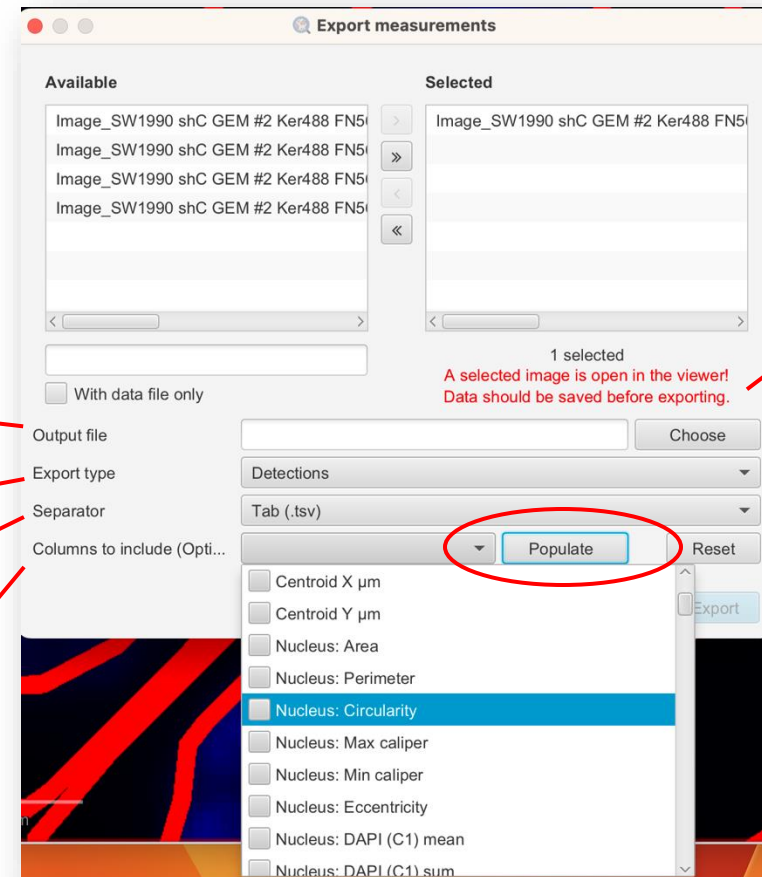
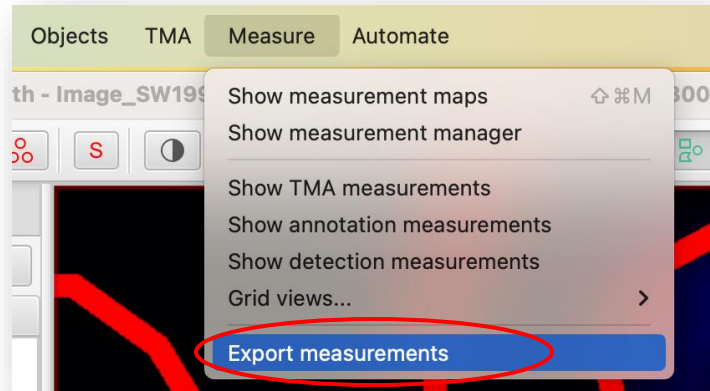


Your favorite color map,
and its range of value



Export measurements the right way

- *Measure > Export measurements*
- Drag an image from *Available* to *Selected*



Output file location

Measurement type to be exported

File type (.tsv, .csv)

List of measurements to include in the export

Save your results first!

A microscopic image of a tissue section, likely stained with hematoxylin and eosin (H&E). The image shows a dense population of cells. The cell nuclei are stained blue, and the cytoplasm and extracellular matrix are stained pink. The image has been processed with a segmentation algorithm. The cell boundaries are outlined in red, and the nuclei are outlined in purple. Some cells have a yellowish-green fill, possibly indicating a specific classification or feature. The text "Cell classification" is overlaid in the center of the image.

Cell classification

What is cell classification?

explain in one sentence to a toddler what is cell classification in the context of image analysis

< 5/5 >



Cell classification is like sorting different kinds of toys into separate boxes based on how they look.

Thank you, I'm sure students will love being compared to toddlers

< 3/3 >



It's all about finding the right analogy for understanding! If it helps simplify a complex concept, then it's worth it.



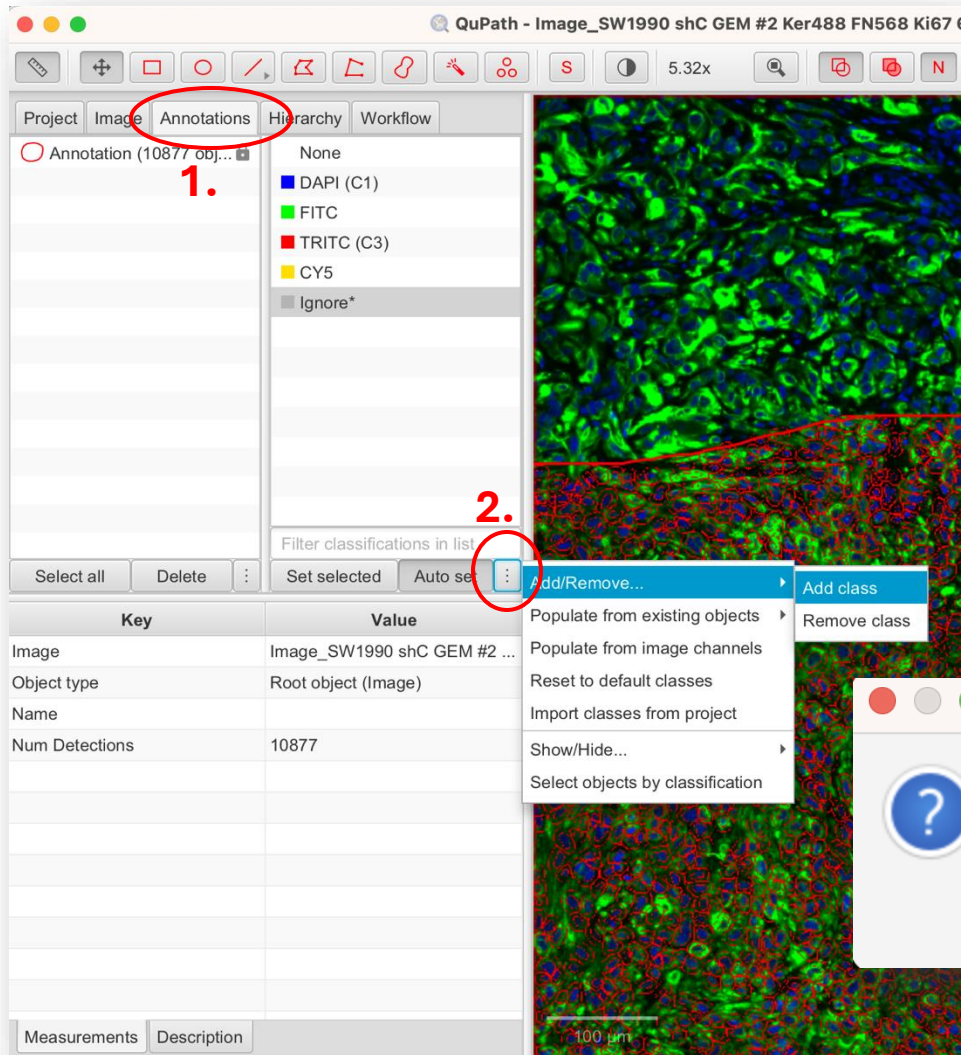
What is cell classification?


- Cell classification in image analysis is sorting different types of cells into groups based on their features or characteristics, such as shape, color, position, neighbors, etc.
- If visually you can't discriminate cells in your images, then your algorithm might be separating cell on very fine differences (careful of *overfitting*)

Cell classification in QuPath

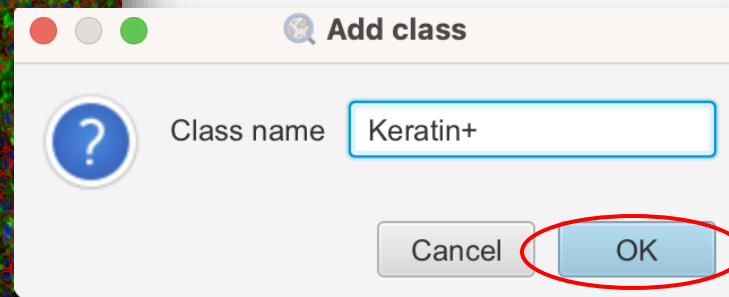
- Single measurement classifier
- Composite thresholder: combine single measurement thresholds together
- Train a machine learning classifier

Create a class named 'Keratin+'



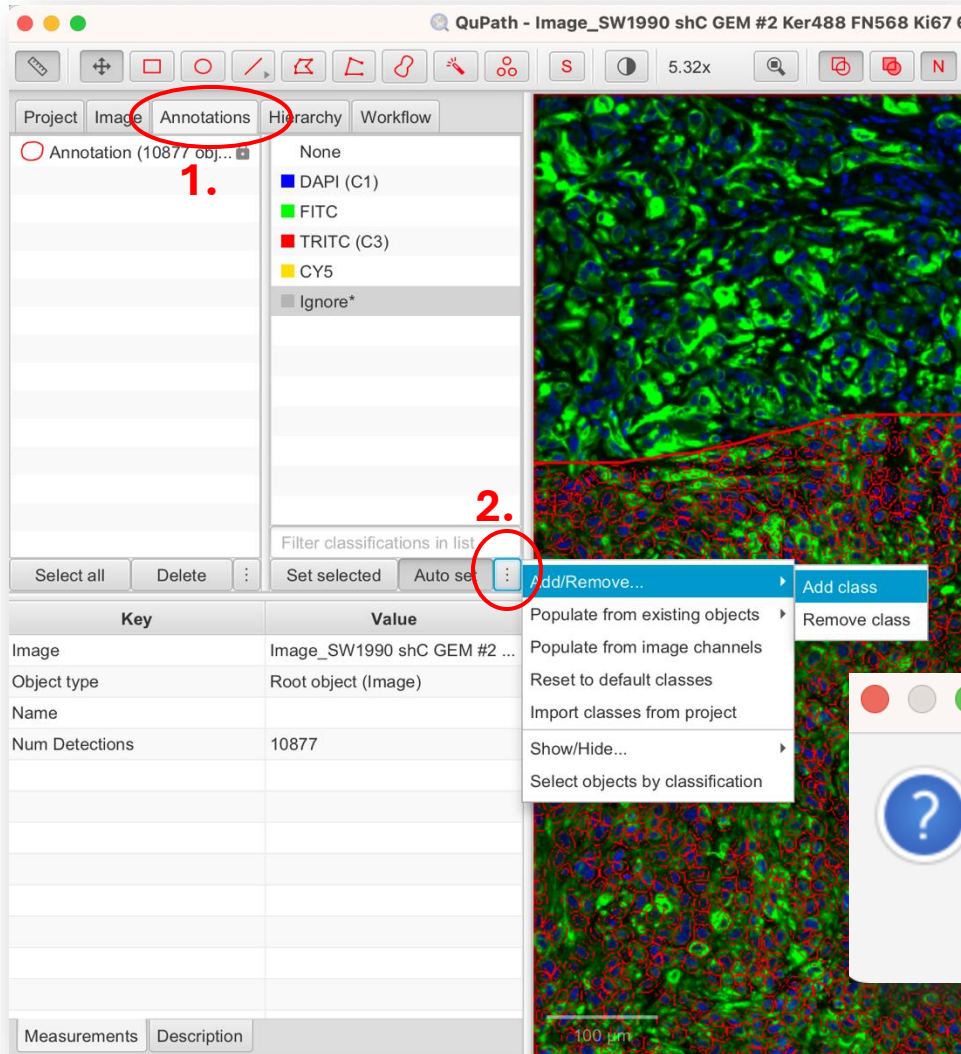
- Annotations tab > Classification list >  > Add/Remove... > Add class


**Default classes are channel names.
We'll change that.**



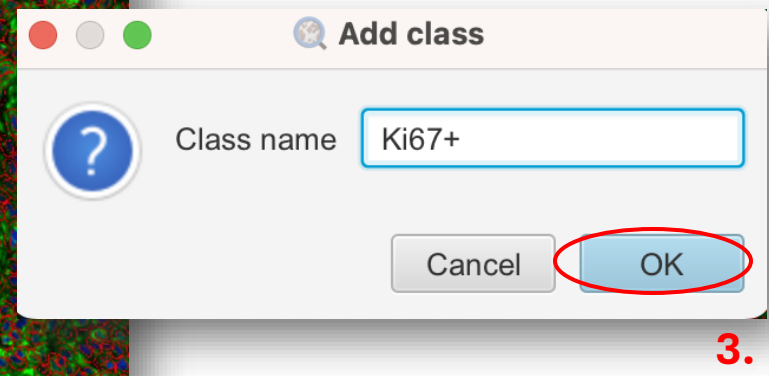
3.

Create a second class named 'Ki67+'



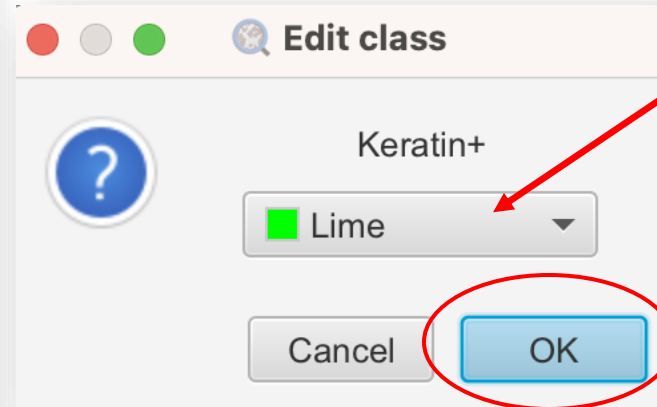
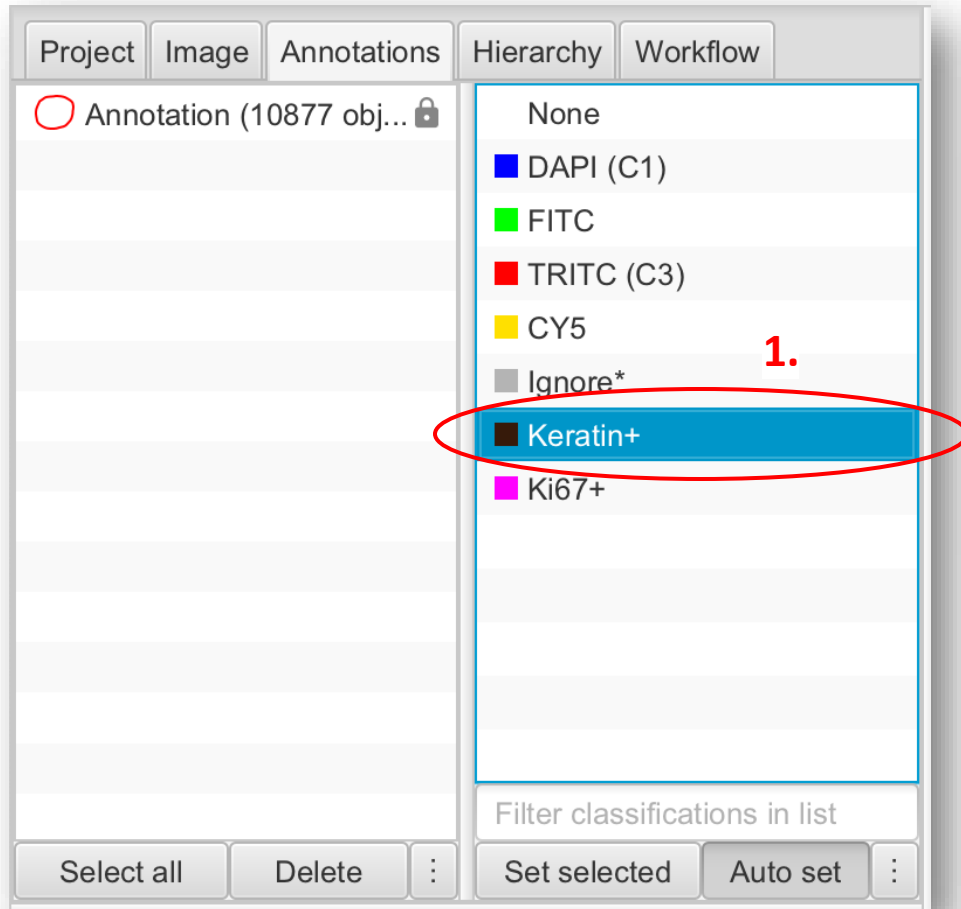
- Annotations tab > Classification list >  > Add/Remove... > Add class

Default classes are channel names.
We'll change that.



Change the color of a class

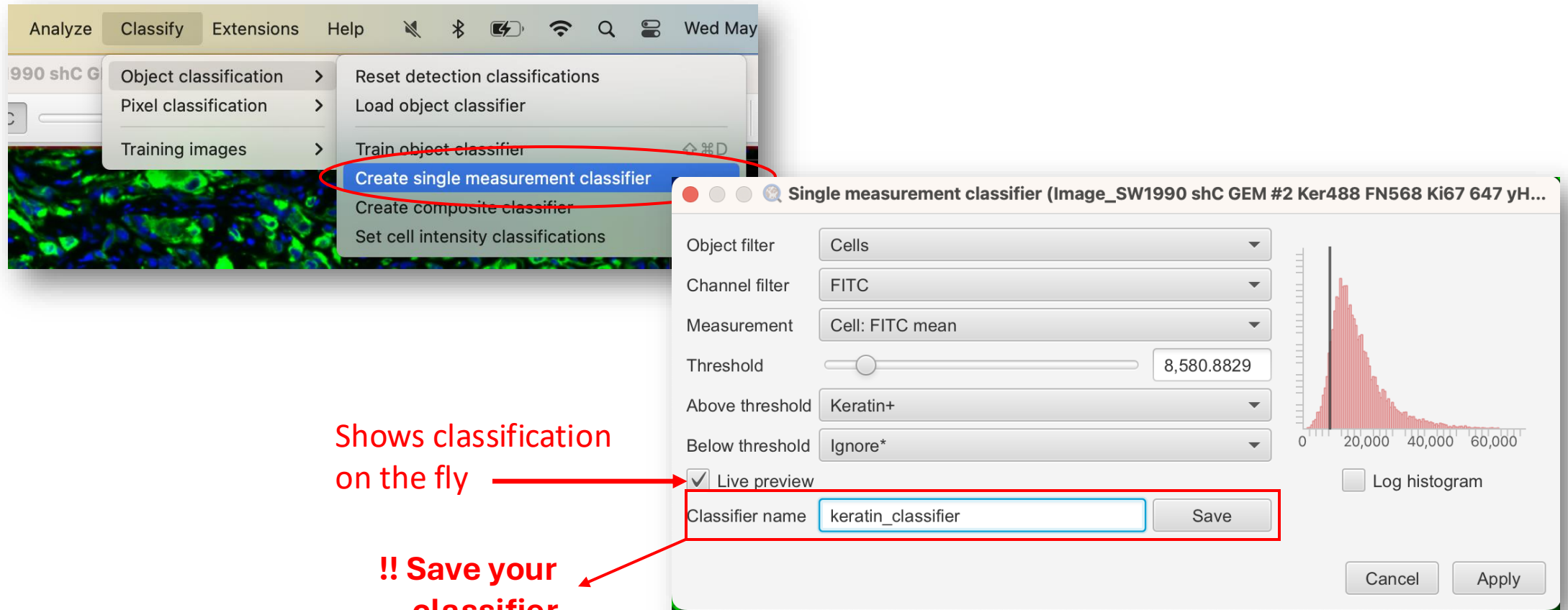
- Double click on the class > Edit class > Choose a new color > OK



By default, classes are populated by image channels.

Simple measurement classifier on Keratin signal intensity (FITC channel)

- *Classify > Object classification > Create single measurement classifier*



The image shows a screenshot of the Fiji software interface. The 'Classify' menu is open, and the 'Create single measurement classifier' option is highlighted with a red circle. The dialog box for 'Single measurement classifier' is open, showing the following settings:

- Object filter: Cells
- Channel filter: FITC
- Measurement: Cell: FITC mean
- Threshold: 8,580.8829
- Above threshold: Keratin+
- Below threshold: Ignore*
- Live preview
- Classifier name: keratin_classifier
- Buttons: Save, Cancel, Apply

A histogram on the right side of the dialog box shows the distribution of FITC mean values for the selected objects. The x-axis ranges from 0 to 60,000, and the y-axis represents frequency. A vertical line is drawn at the threshold value of 8,580.8829.

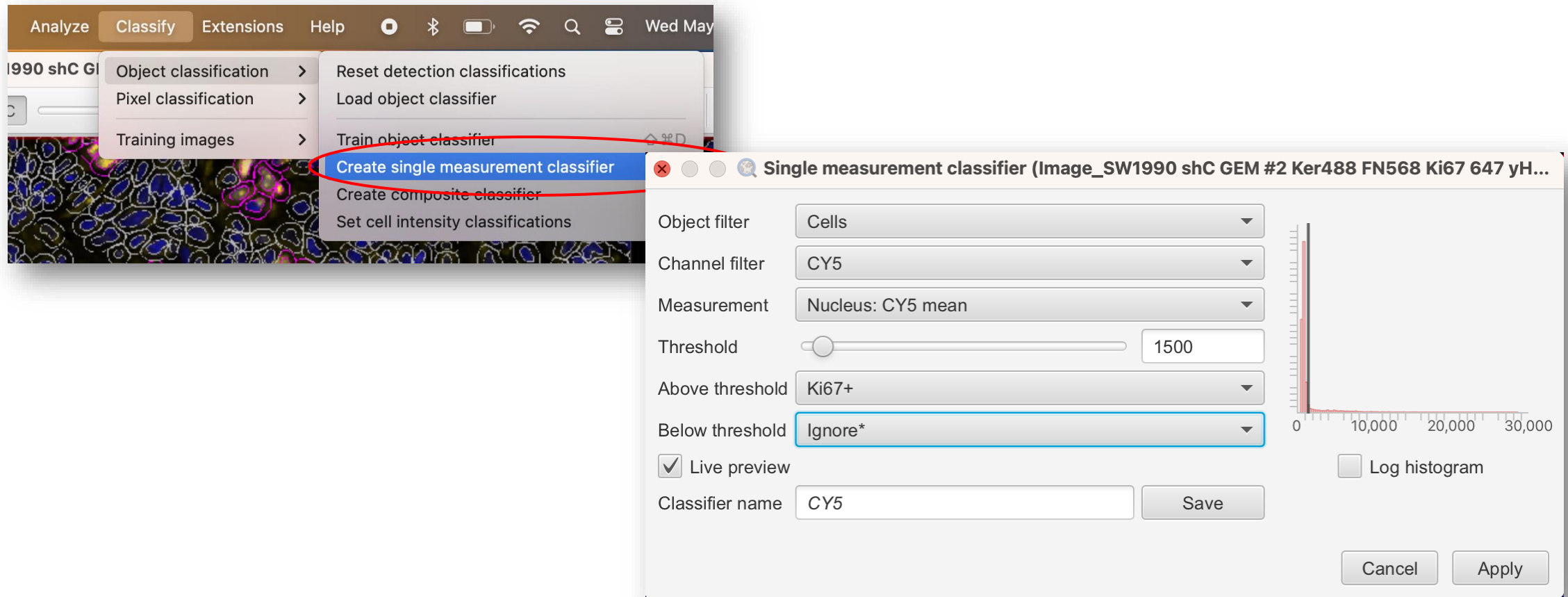
Red arrows point from the text annotations to the 'Live preview' checkbox and the 'Classifier name' field.

Shows classification on the fly

!! Save your classifier

Simple measurement classifier on Ki67 signal intensity (CY5 channel)

- *Classify > Object classification > Create single measurement classifier*



Practice time

Exercise 4.a: single-measurement classifier

Combine single measurement classifiers into a composite classifier

- *Classify > Object classification > Create composite classifier*

990 shC G

Analyze Classify Extensions Help

Object classification > Reset detection classifications

Pixel classification > Load object classifier

Training images > Train object classifier

Create single measurement classifier

Create composite classifier

Set cell intensity classifications

Create composite classifier

Move individual classifiers to the column on the right to be included in the composite classifier. Note that the order of classifiers in the list determines the order in which they will be applied.

Available

Selected

ki67_classifier

keratin_classifier

Classifier name: keratin_ki67_classifier

Save

Cancel

Save & apply

Select a classifier by moving it onto the 'Selected' list.

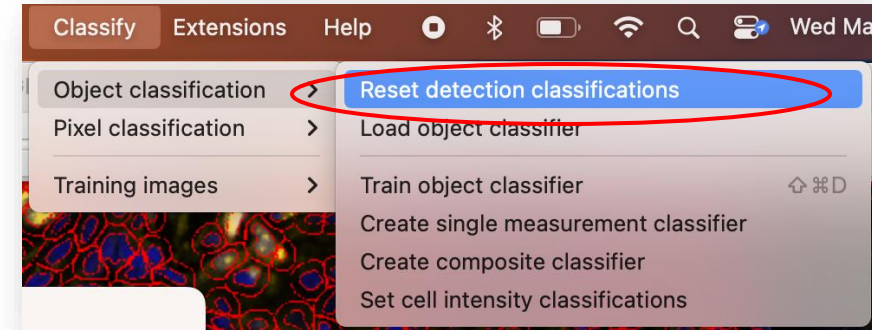
A name is required to 'Save & apply'

Practice time

Exercise 4.b: composite classifier

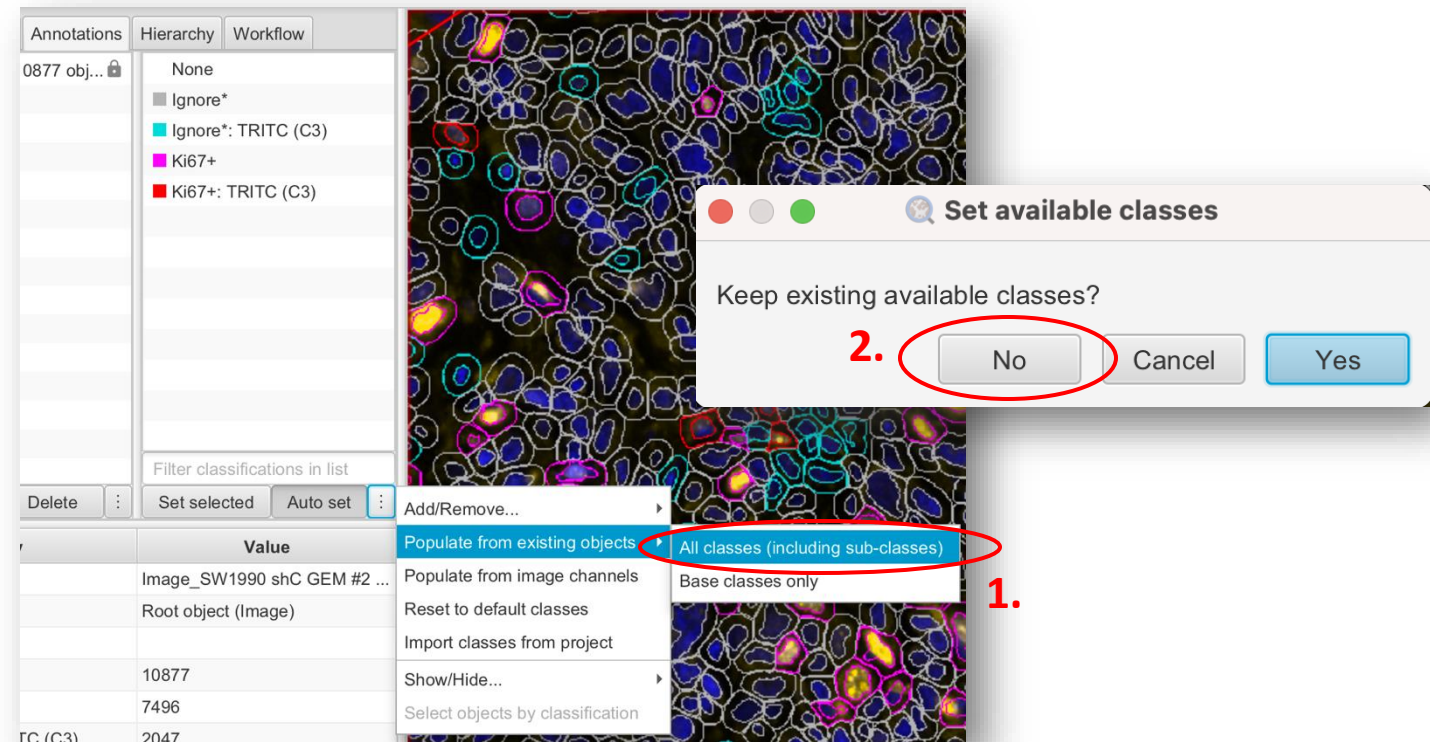
Reset detection classes

- *Classify > Object classification > Reset detection classifications*



Populate classes in the classification panel

- *Annotations tab > Classification list > [Menu Icon] > Populate from existing objects > All classes (including sub-classes)*



Object classification using machine learning

- Detections (and annotations) can be classified into classes using a ML classifier
- Classification requires measurements!
- Object classifiers are trained using manual annotations of 2 or more classes
 - Need to create some training data points
- **Live demo of object classification using ML**

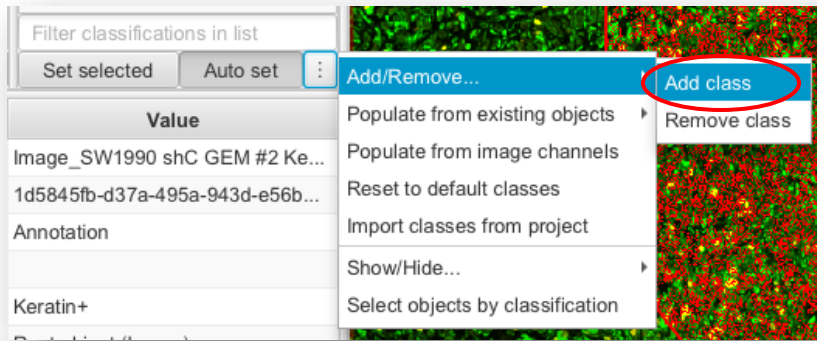
Reset your detection classes!



Adobe Stock | #32445303

Train an object classifier: create classes

- *Annotations* tab > Classification list >  > Add/Remove... > Add class



- Create 4 classes:

- Keratin+
- Keratin-
- Ki67+
- Ki67-

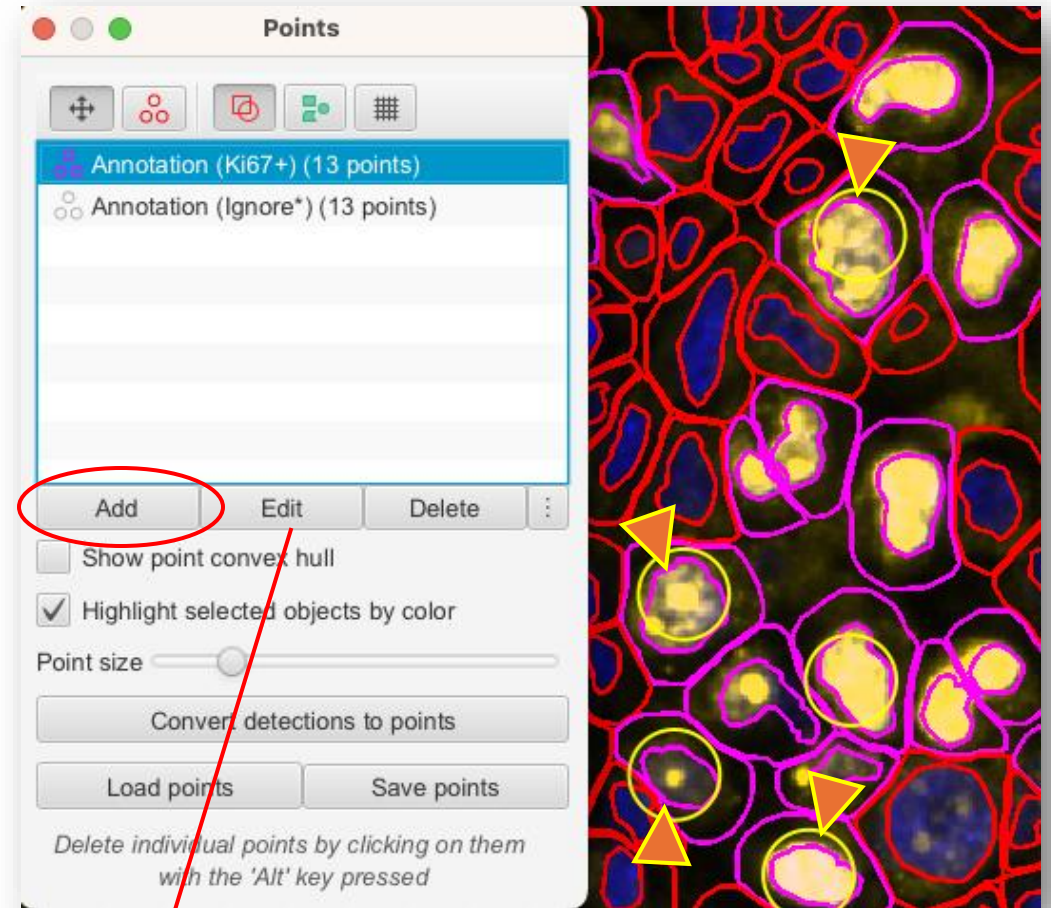


Train an object classifier: training data points

-  > Add > Label ~10 for each class

To remove a single point:
Option + click (Mac) or left-click

- Assign each training data set a class:
 - Select the annotation set
 - Select the class



Click edit to change color

 Training data

Train an object classifier: training data points

- Assign each training data set a class in the *Annotations* tab

The image shows two screenshots of a software interface, likely a microscopy image analysis tool, illustrating the process of assigning a class to an annotation. The interface has tabs for Project, Image, Annotations, Hierarchy, and Workflow.

Left Screenshot (Step 1): The 'Annotations' tab is active. A list of annotations is shown on the left, with the first one, 'Annotation (13 points)', circled in red and labeled '1.'. A dropdown menu is open for this annotation, showing classification options: None, Ki67+ (1), Ignore*, Keratin+ (1), Ki67-, and keratin-. The 'keratin-' option is highlighted in blue and circled in red, labeled '2.'. At the bottom of the dropdown, the 'Set selected' button is circled in red and labeled '3.'. Below the screenshot, the number '3.' is also written in red.

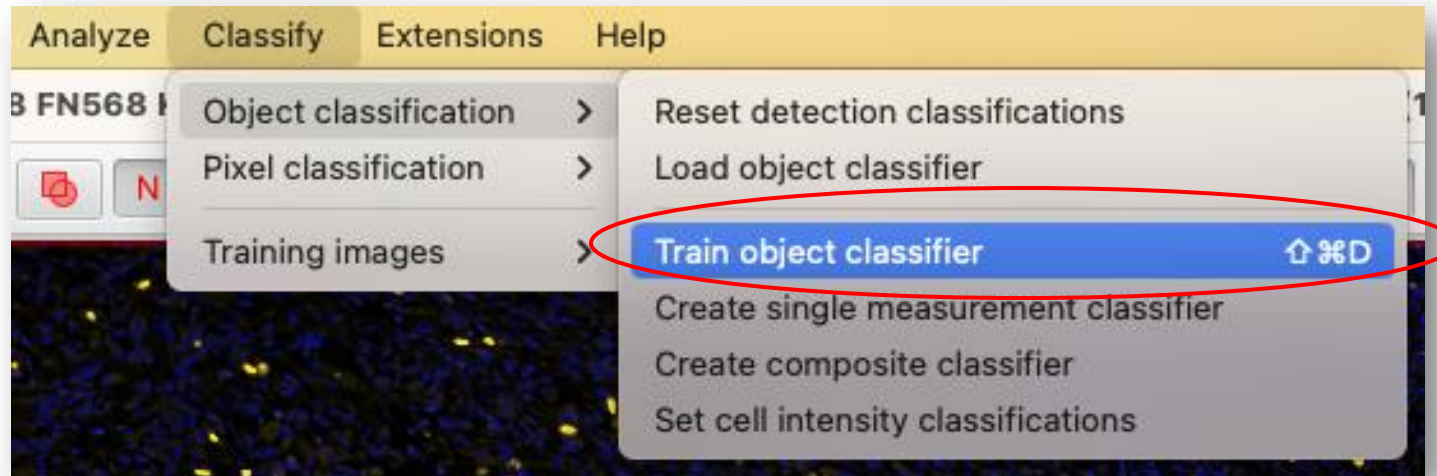
Right Screenshot: The same interface is shown after the class assignment. The 'keratin-' class is now assigned to the selected annotation, and the 'Set selected' button is no longer visible. The 'keratin-' class is also visible in the classification list on the right side of the interface.

A red arrow points from the left screenshot to the right screenshot, indicating the transition from selecting the class to applying it.

Make sure to lock your annotation: Ctrl+click > Lock

Train an object classifier

- *Classify > Object classification > Train object classifier*



Train an object classifier

- *Classify > Object classification > Train object classifier*

Model type (RT, ANN, k-NN)

Features: choose Selected measurements and click Select to restrict the feature space

Name	Selected
Keratin+	<input type="checkbox"/>
Ki67+	<input checked="" type="checkbox"/>
Ki67-	<input checked="" type="checkbox"/>
keratin-	<input type="checkbox"/>

Train object classifier

Object filter: Detections (all)

Classifier: Random trees (RTrees) [Edit]

Features: All measurements [Select]

Classes: Selected classes [Select]

Training: Points only

Load training | Advanced options

Live update

Training data

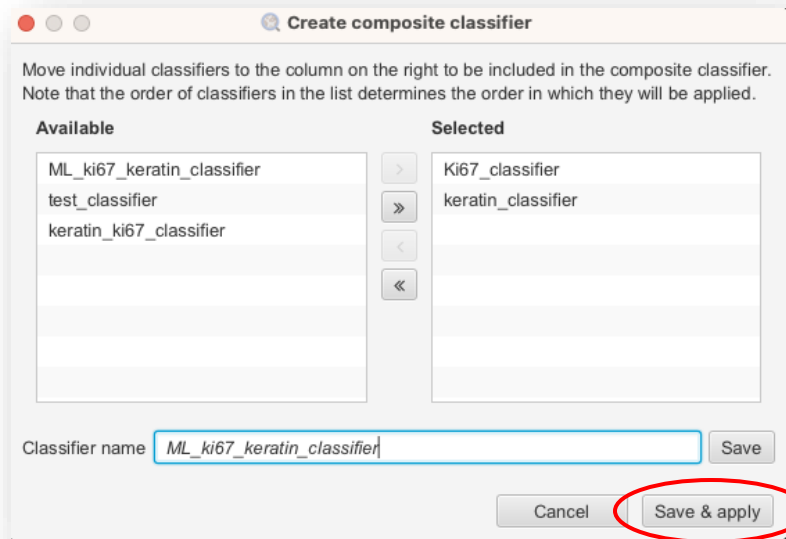
Ki67+ (magenta) | Ki67- (red)

Classifier name: Ki67_classifier [Save]

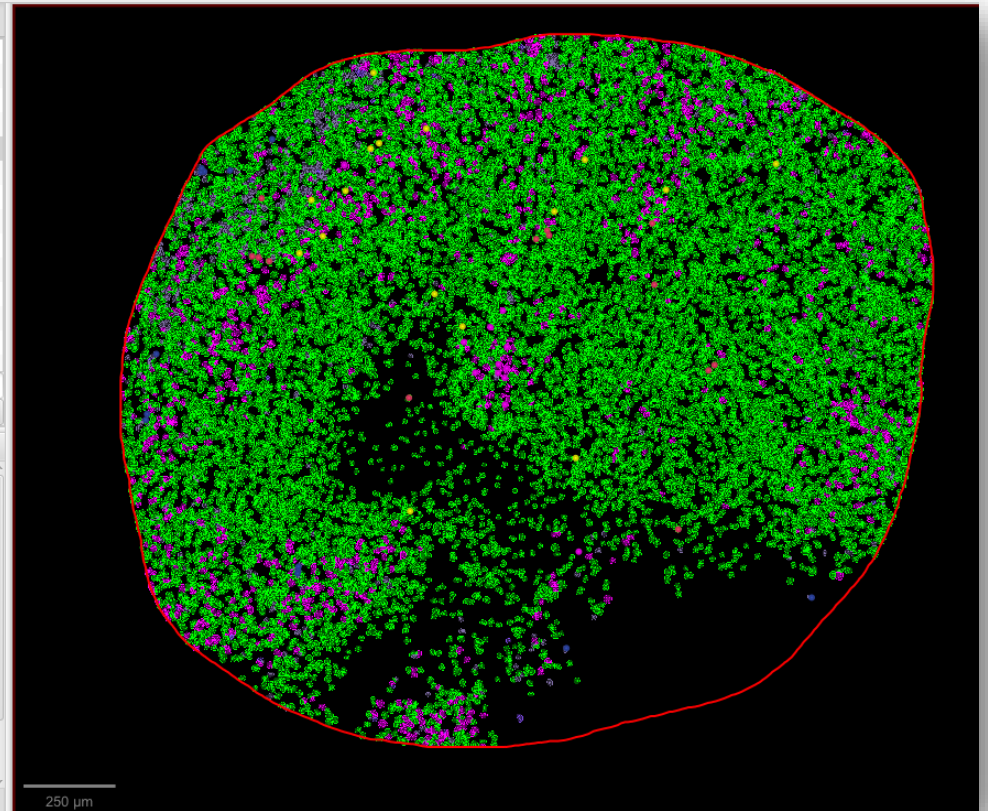
Name it to save it.

Combine multiple ML classifiers together

- *Classify > Object classification > Create composite classifier*



Key	Value
Image	Image_SW1990 shC GEM #2 K...
Object ID	1d5845fb-d37a-495a-943d-e56...
Object type	Annotation
Name	
Classification	Keratin+
Parent	Root object (Image)
ROI	Points
Centroid X μm	5173.7928
Centroid Y μm	4524.1271
Num Detections	0
Num Ki67+: Keratin+	0
Num Ki67+: keratin-	0



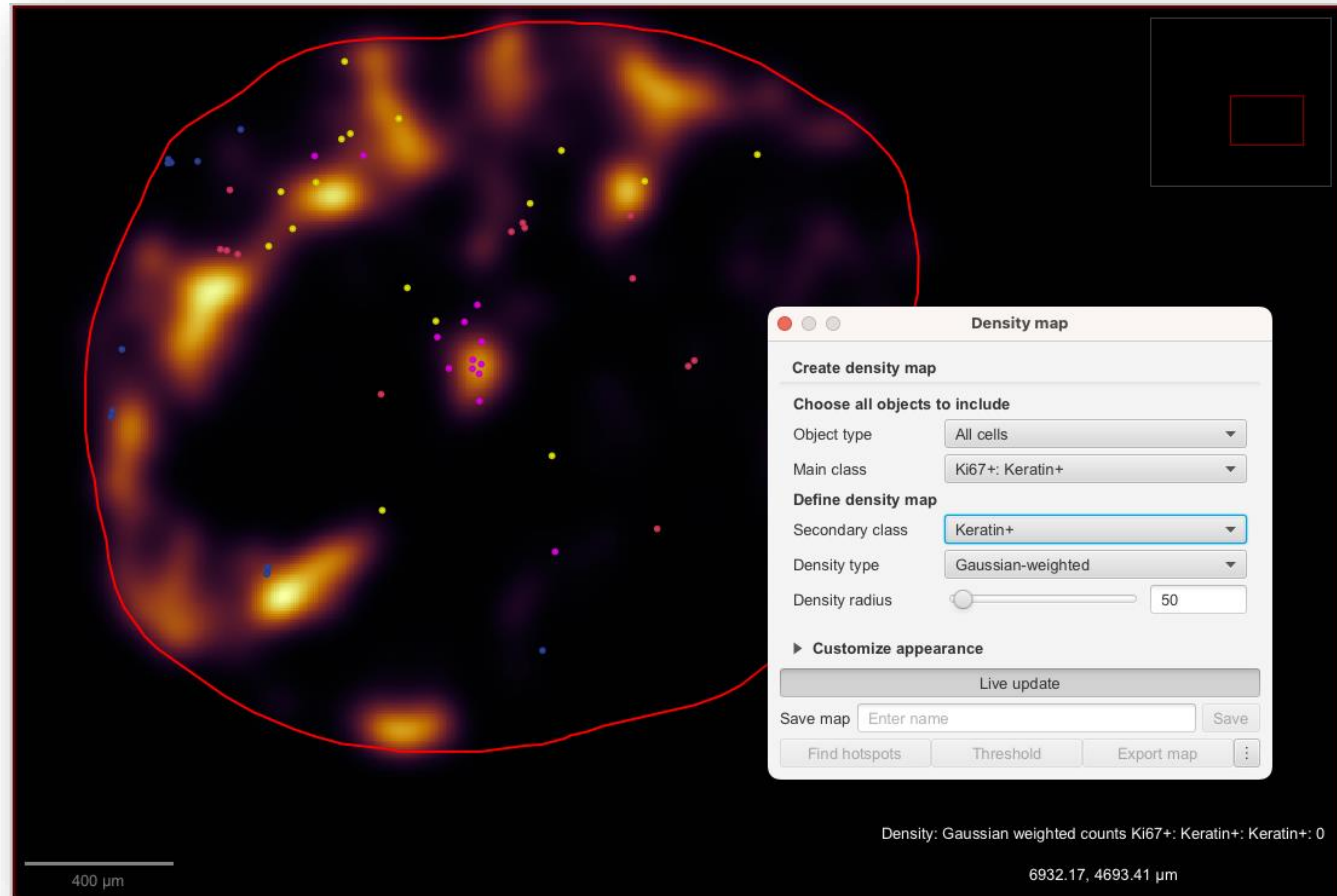
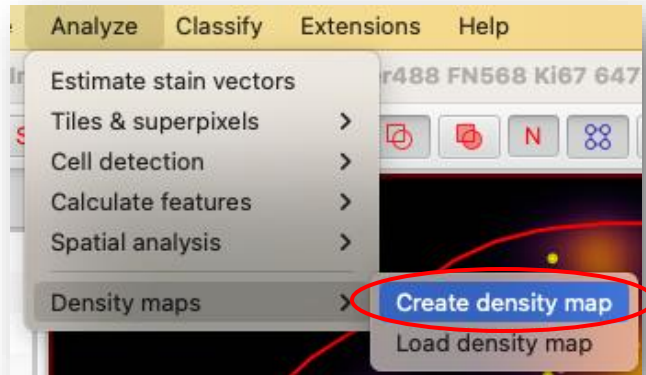
8 resulting classes!
+ ignore*

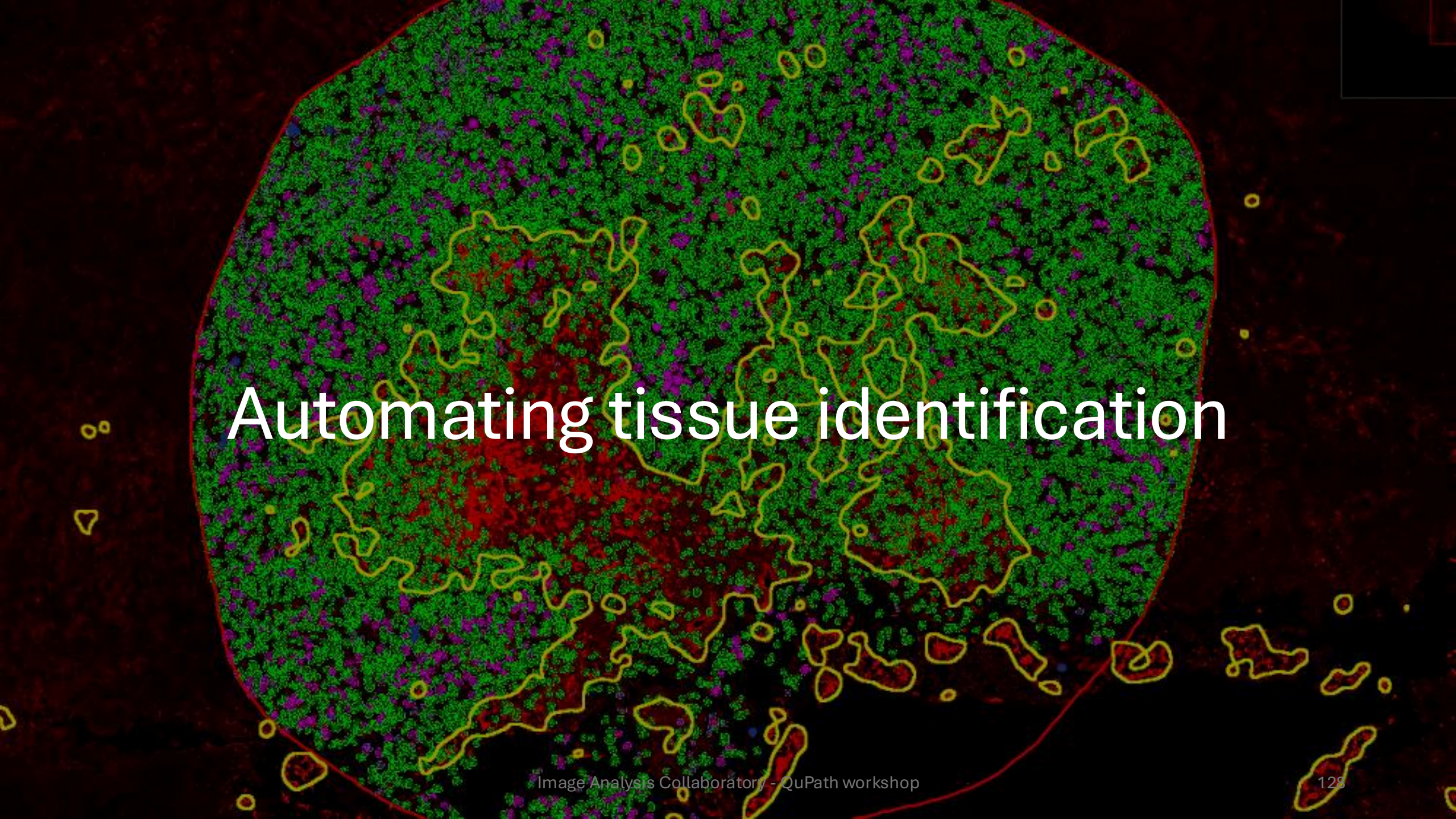
Refine your classifier

- Add more training data points
 - Classification results will change in real time if ‘Live update’ option is enabled
- Typically, *fewer*, but *well-chosen* features provides more robust results

Visualizing results using density maps

- Analyze > Density maps > Create density maps



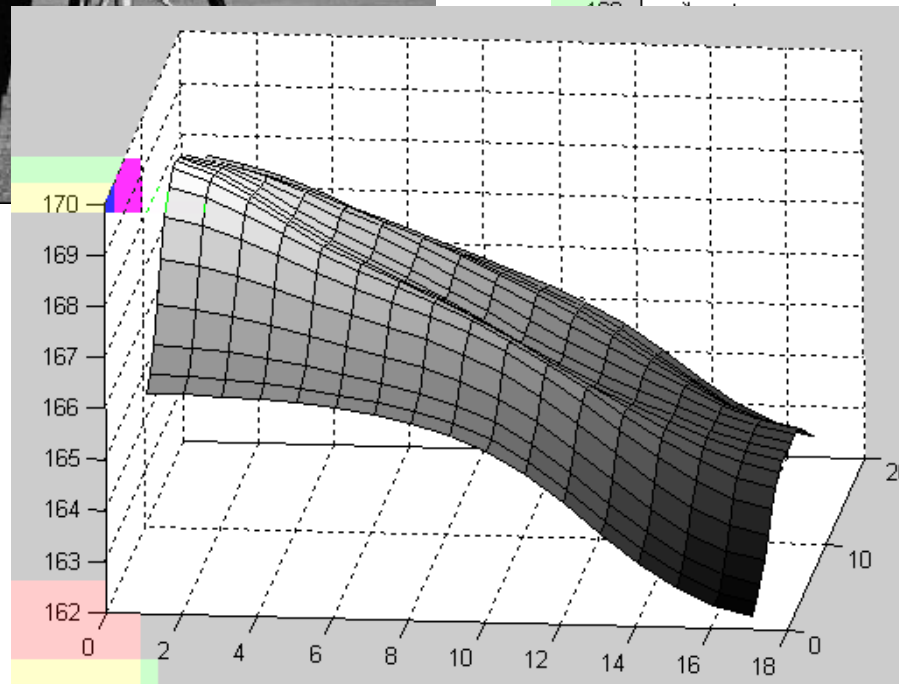
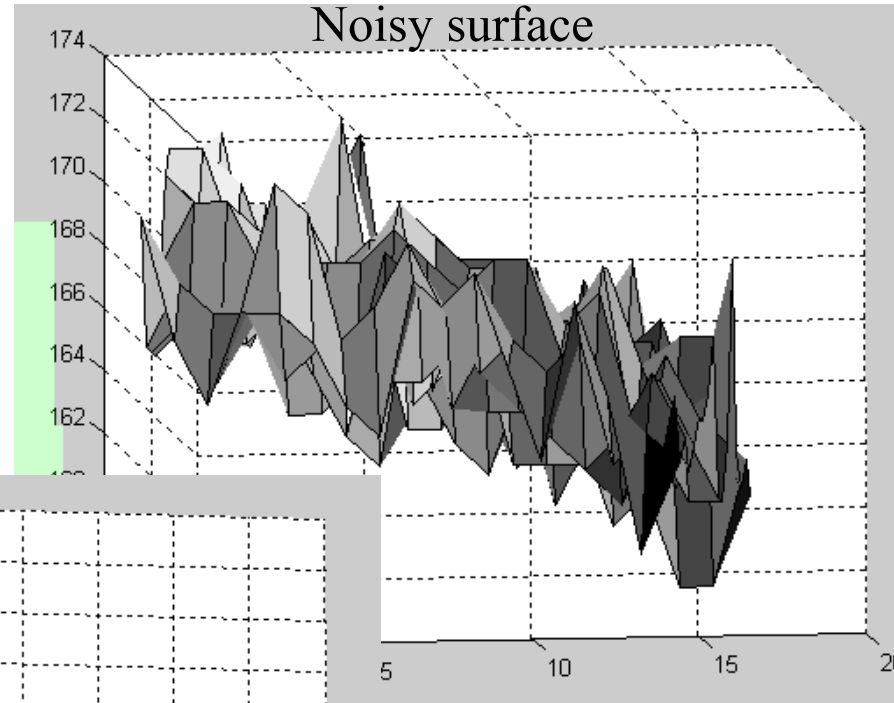
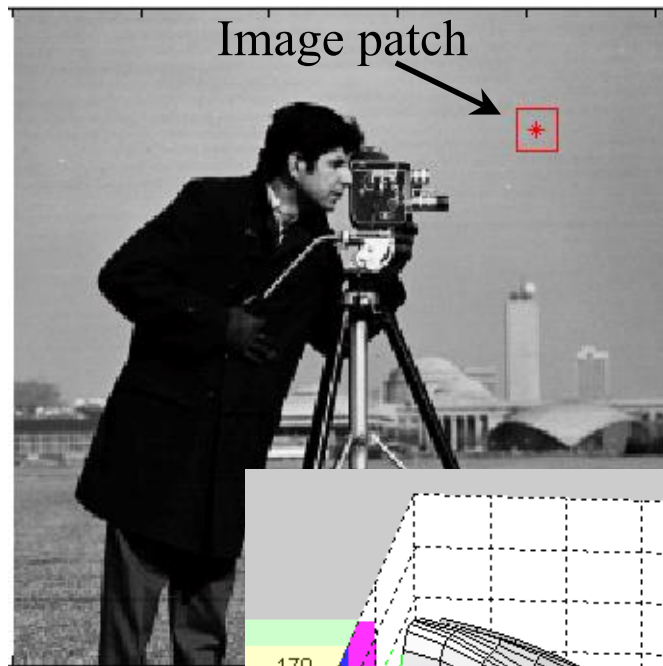


Automating tissue identification

But first, let's talk smoothing

Intermezzo aperto

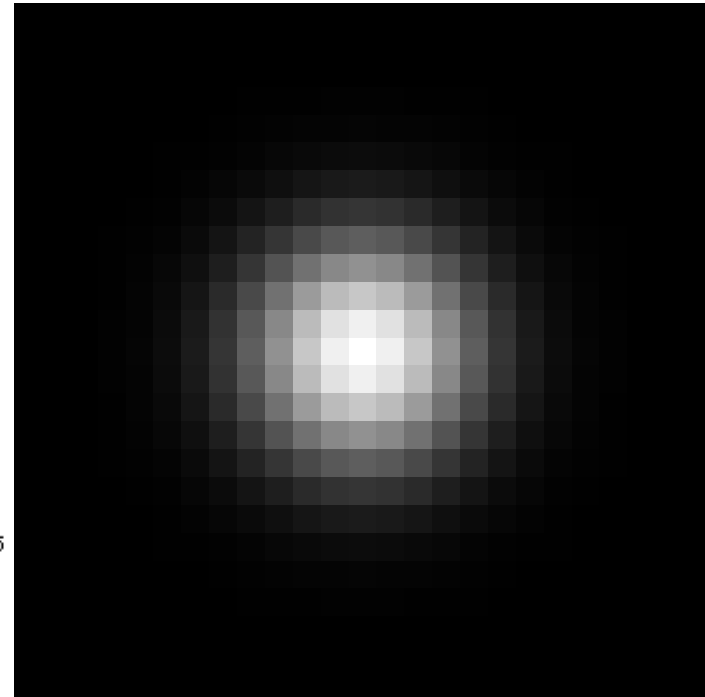
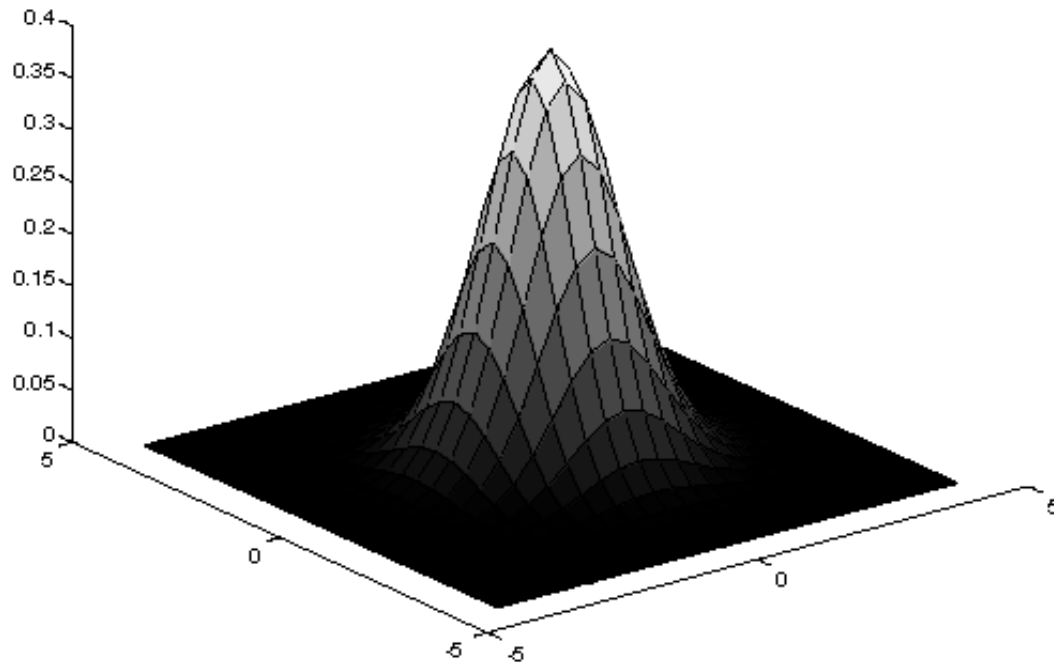
Today: Smoothing Reduces Noise



smoothing reduces noise,
giving us (perhaps) a more
accurate intensity surface.

Gaussian Smoothing Filter

An isotropic (circularly symmetric) Gaussian:



Gaussian Smoothing Example



original



sigma = 3

Robert Collins
CSE486, Penn State

Gaussian Smoothing at Different Scales



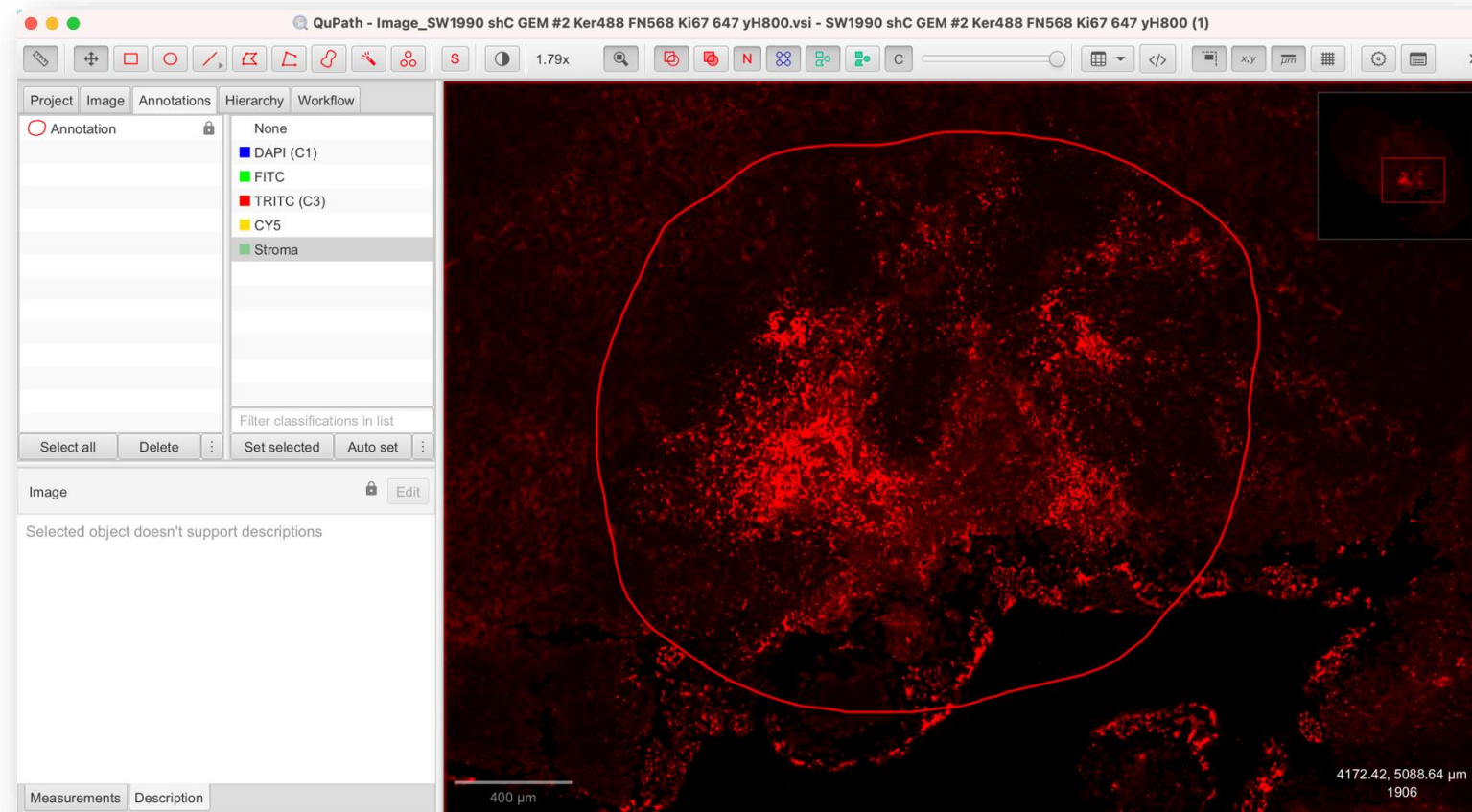
**Balancing act: smooth enough to “clean up”
the noise, but not so much as to remove
important image gradients.**

Back to QuPath

Intermezzo chiuso

Creating a region of interest

In the TRITC channel (fibronectin), create a region of interest that enclose high-fibronectin content regions aka stromal regions



Once you have finished your annotation, **lock** it:

Right-click in the viewer
> *Annotations* > *Lock*

or

Right-click on the annotation in the analysis panel > *Lock*

Pixel-based tissue annotation

- Simplest case of annotation: every pixel get assigned a class based on its intensity value – **or is a given pixel above or below a certain numeric value?**

The screenshot shows the QuPath software interface. The main window displays a histology image with a red channel (TRITC) and a 'Create threshold' dialog box open. The dialog box has the following settings: Resolution: Moderate (2.60 μm/px), Channel: TRITC (C3), Prefilter: Gaussian, Smoothing sigma: 5, Threshold: 3500, Above threshold: Stroma, Below threshold: Unclassified, Region: Any annotation ROI, Classifier name: stroma_classifier. A 'Save' button is circled in red. In the background, the 'Classify' menu is open, showing options like 'Load pixel classifier', 'Train pixel classifier', and 'Create threshold', with 'Create threshold' also circled in red. The top menu bar includes File, Edit, Tools, View, Objects, TMA, Measure, Automate, Analyze, Classify, Extensions, and Help. The status bar shows 'Wed May 1 8:39 PM'.

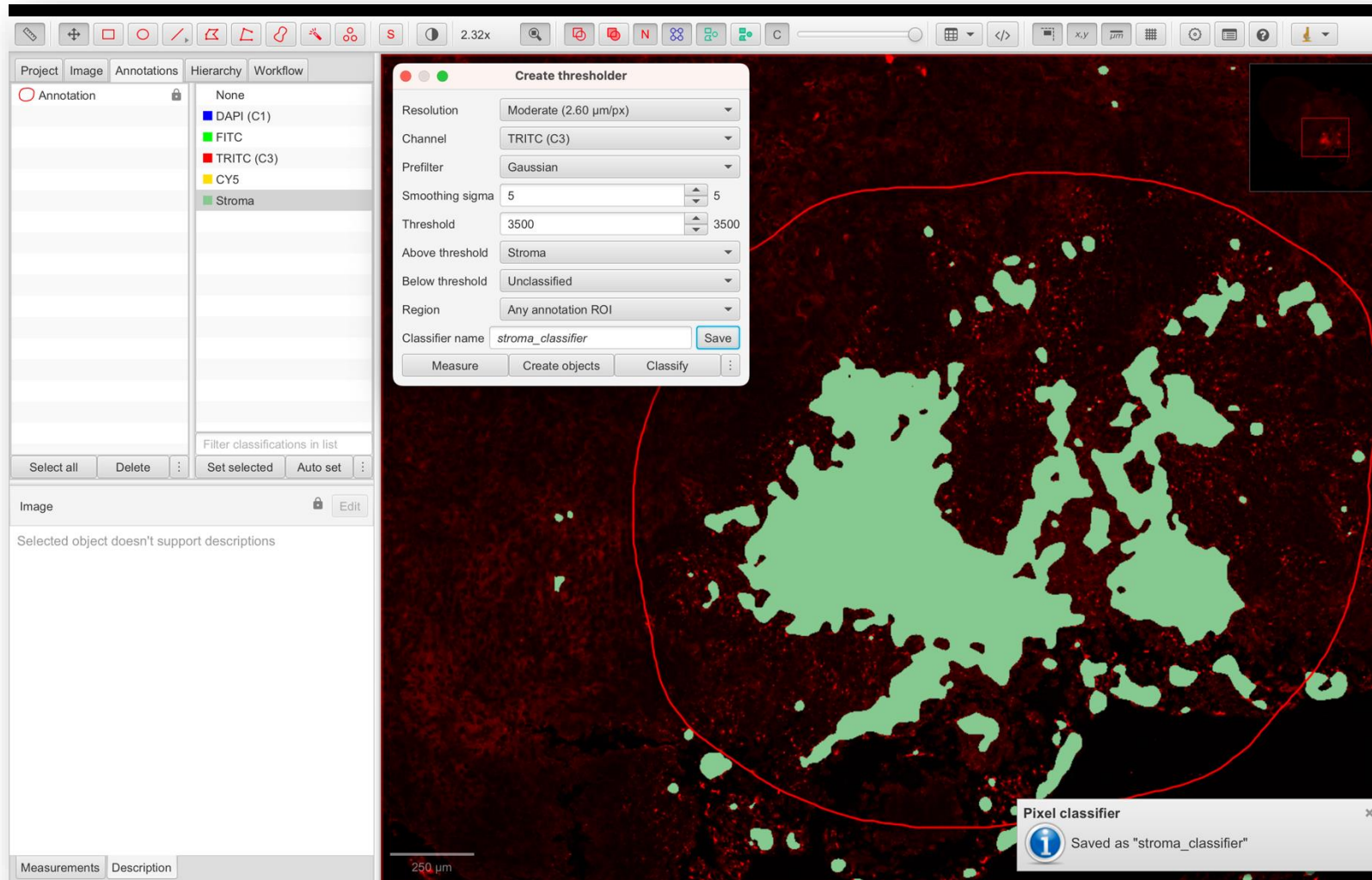
Resolution: trade-off between details and computational cost

Decide to use ROI or the full image

Pre-process images with filters e.g. smooth out noise with gaussian (sigma is the kernel size in pixels)

Save your threshold to use it!

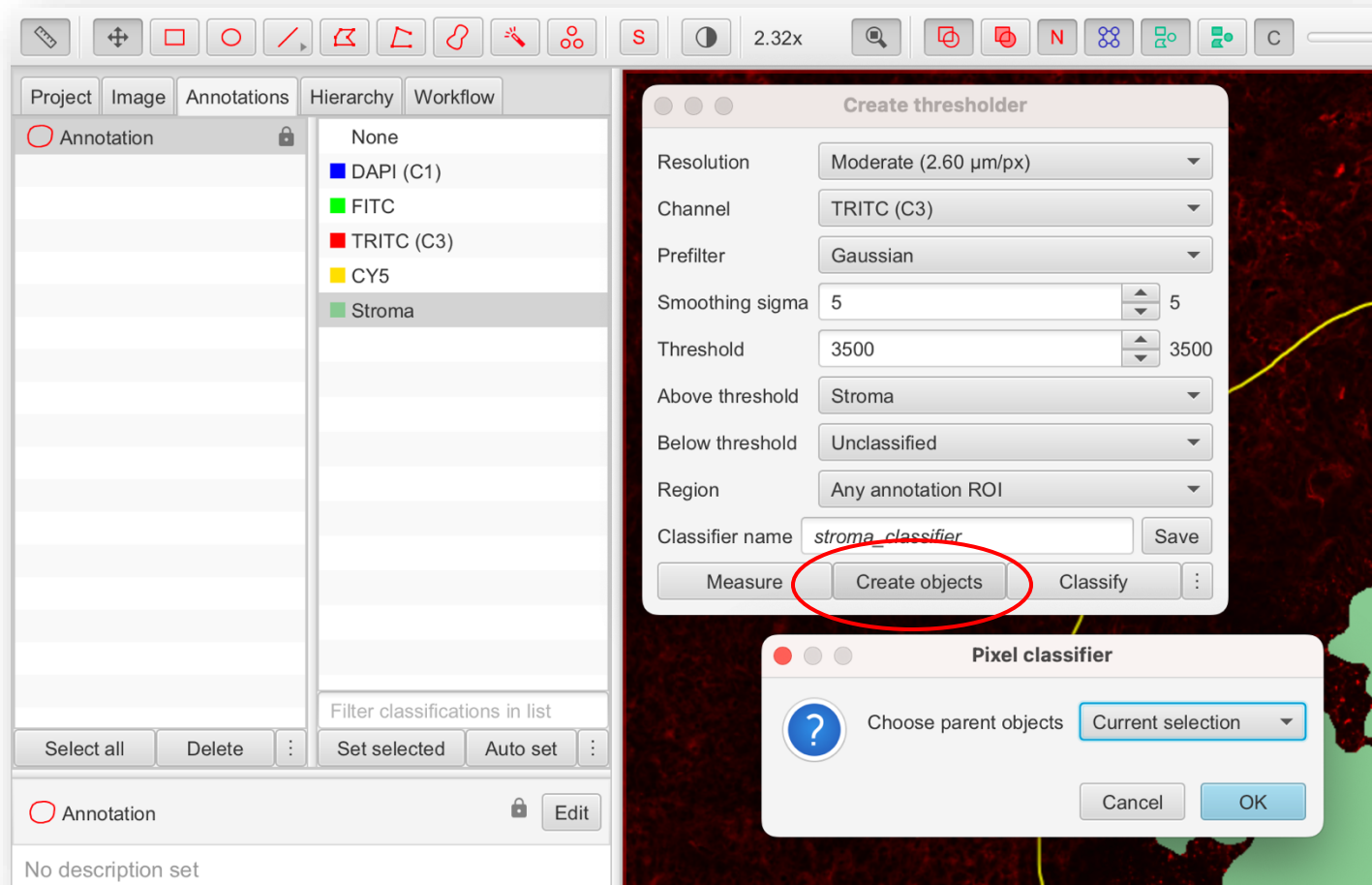
Interactive visualization of thresholding results



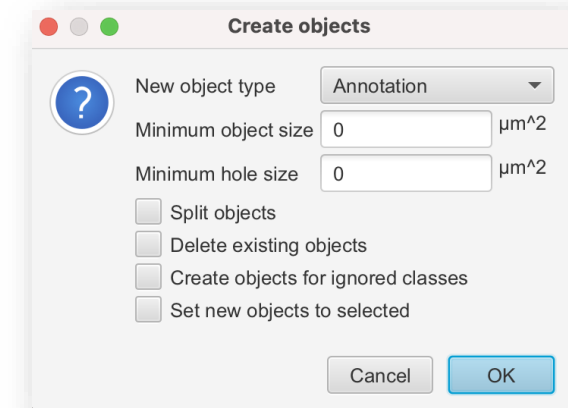
Create a class
'Stroma'

Try varying the value
of the different
parameters!

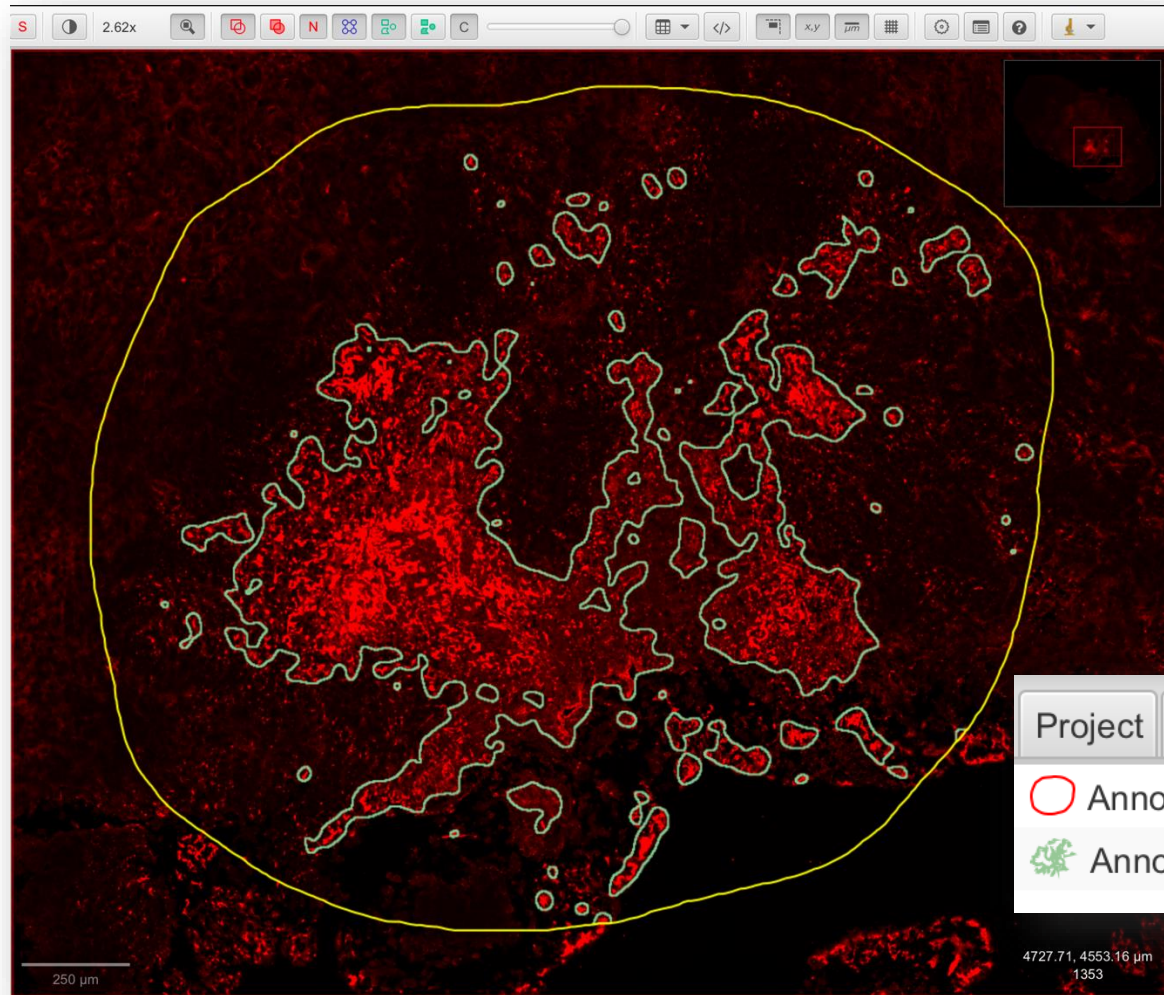
Create annotations from pixel classifier



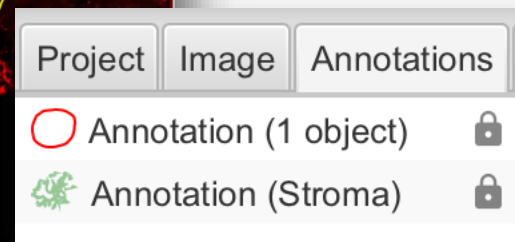
- Real-time visualization of results, once happy with it:
 1. Save your thresholder
 2. Select ROI
 3. Click *Create objects*
 4. Keep default parameters > OK



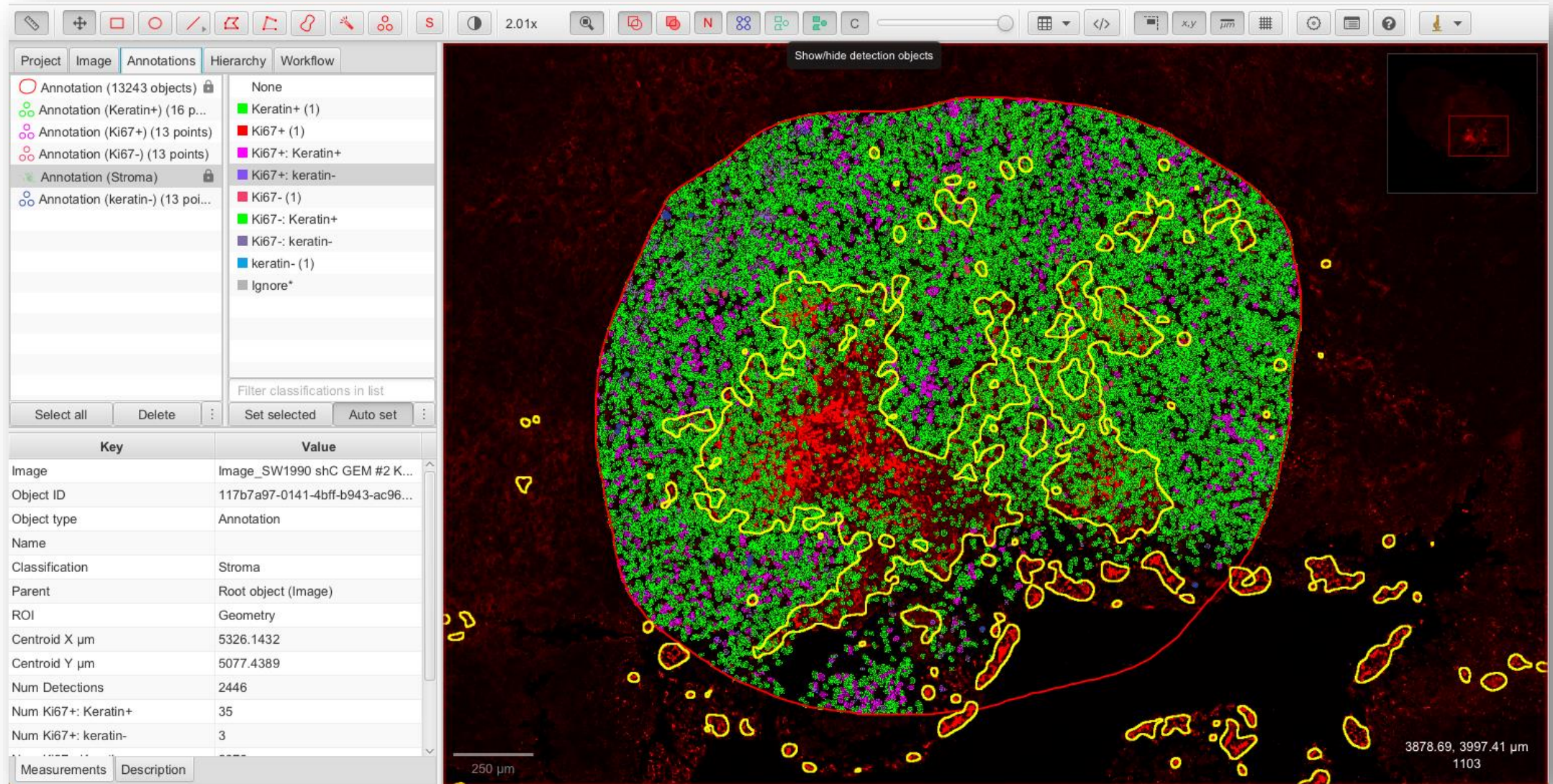
Create annotations from pixel classifier



Notice the new annotation named 'Stroma' in the *Annotations* list

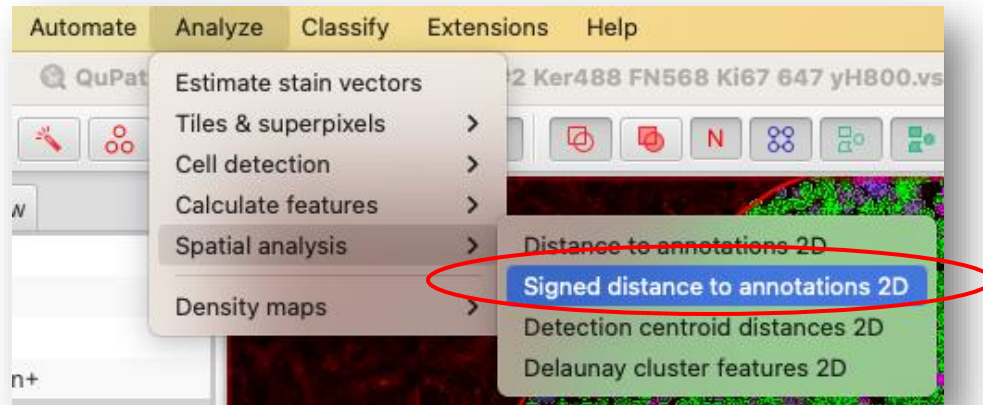


Fully annotated image



Spatial information: signed distance

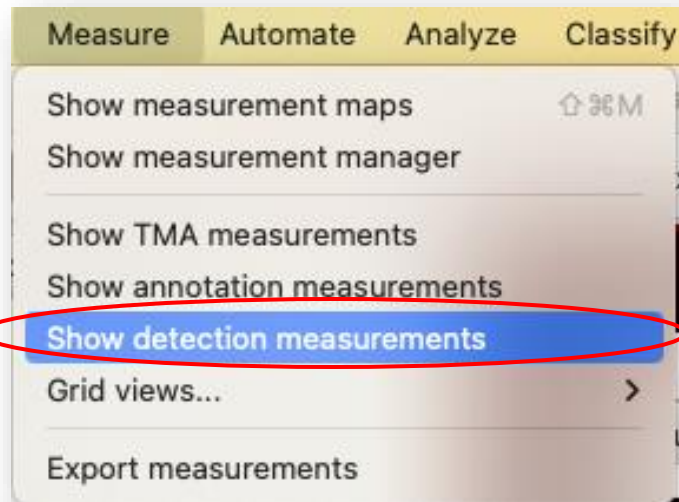
- *Analyze > Spatial analysis > Signed distance to annotations 2D*



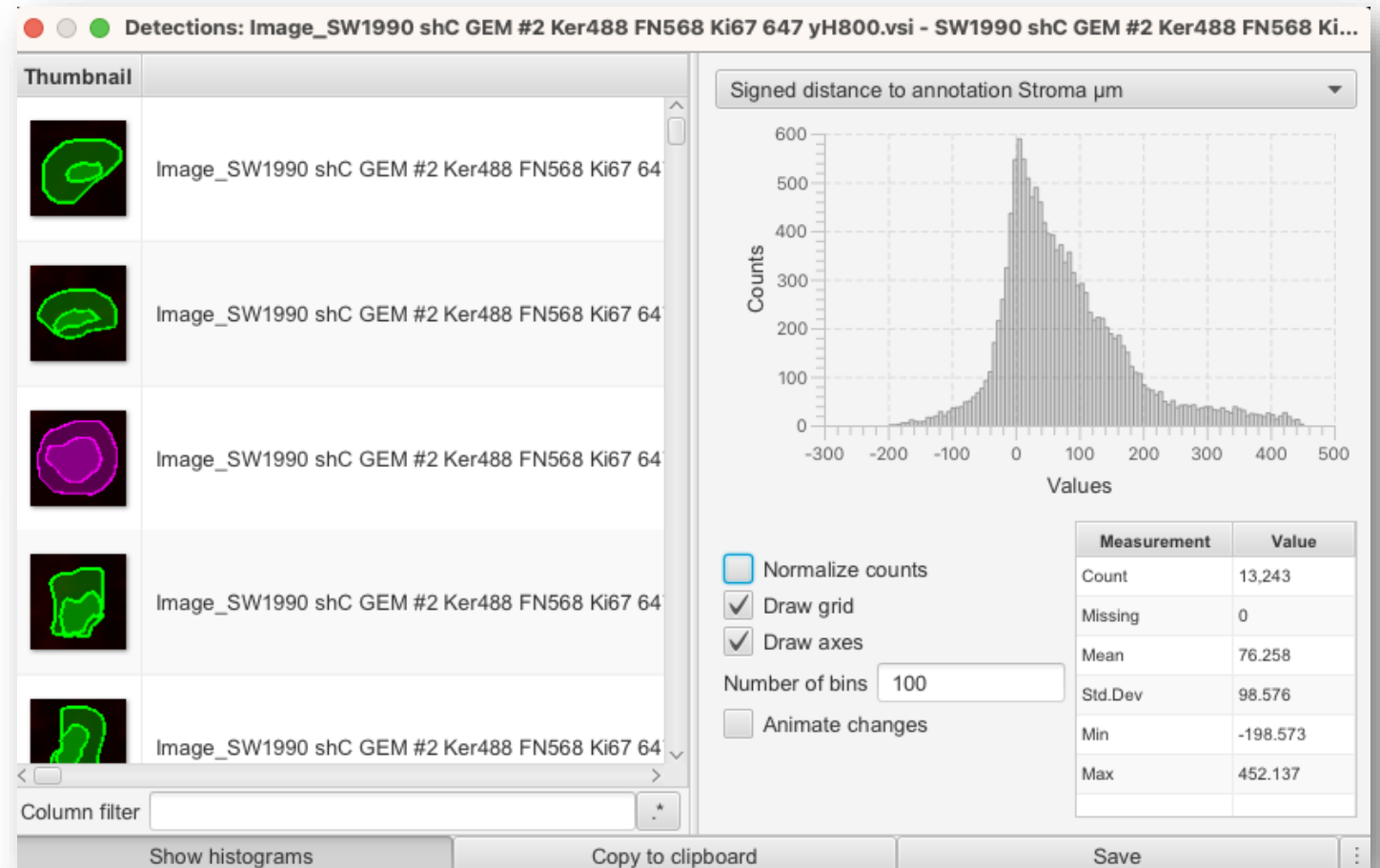
- Calculates the signed distance (2D euclidian) between cells and annotations
 - If a cell lies inside the annotation: negative distance
 - If a cell lies outside the annotation: positive distance

Spatial information: signed distance

- *Measure > Show detection measurements*



Export measurements table and use Python/R for visualization based on classes




```
String message = "Hello, Groovy!"  
int age = 25  
double pi = 3.14  
boolean isGroovy = true  
List<Integer> numbers = [1, 2, 3, 4, 5]  
Map<String, Object> person = [name: "John", age: 30, city: "New York"]
```

Scripting, workflows and batch processing

Scripting in QuPath

- QuPath uses **Groovy**, a scripting language with Java-like syntax
- Some fun facts about Groovy:
 - Created by James Strachan in 2003
 - Open-source (under the Apache License 2.0)
 - Groovy is a superset of Java and its syntax is Java-like
 - Bonus: dynamically typed (vs Java being statically typed)

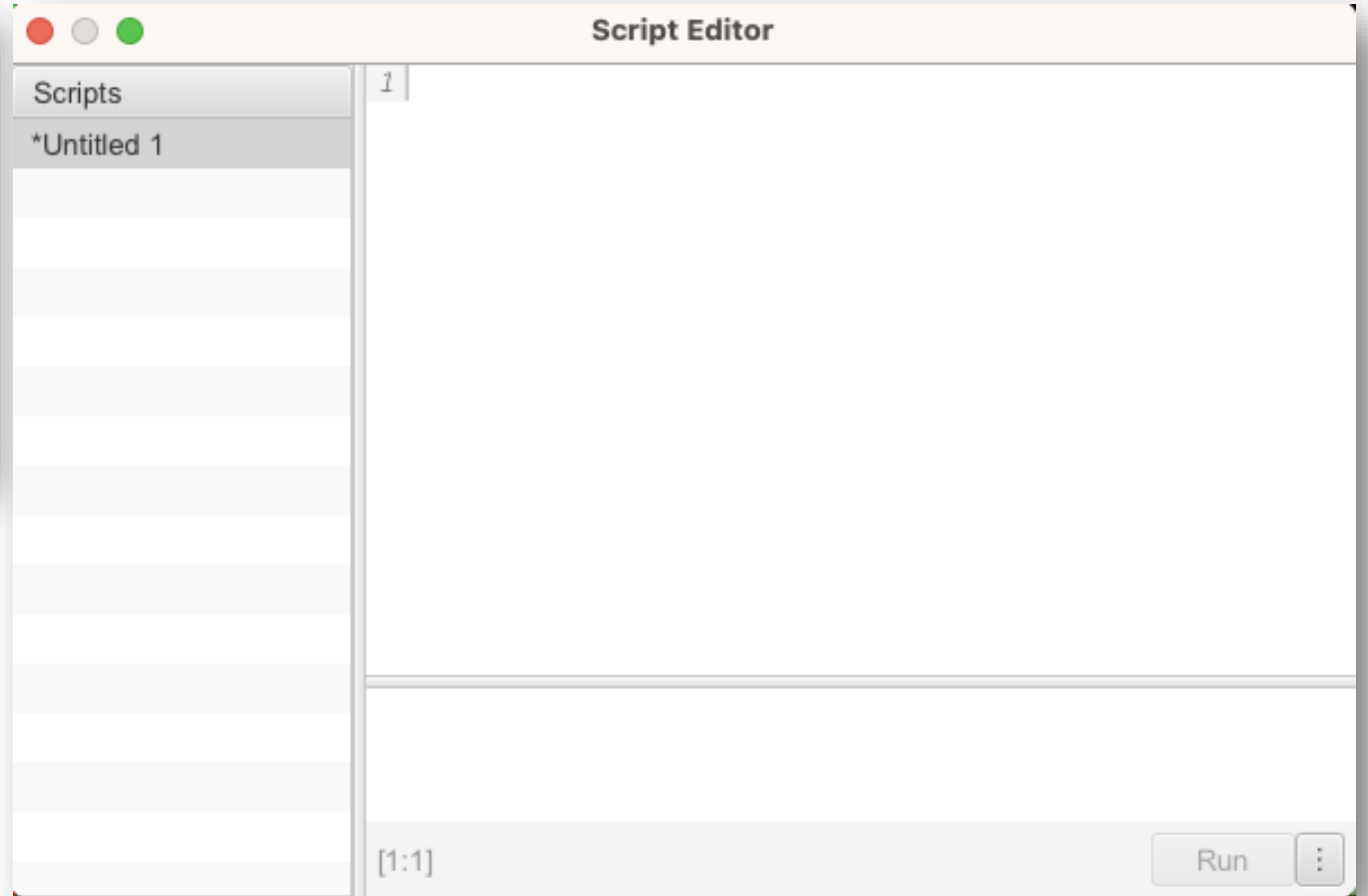
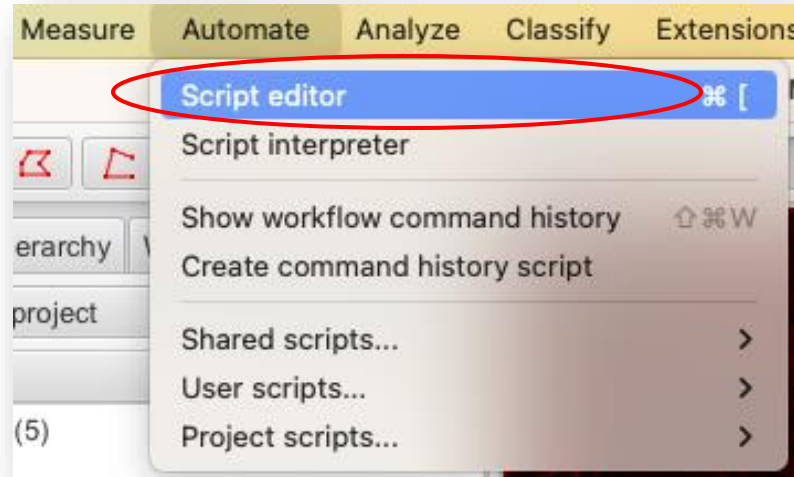
```
groovy

// Iterate over a range of numbers
for (int i = 0; i < 5; i++) {
    println("Index: $i")
}
```

for loop in Groovy

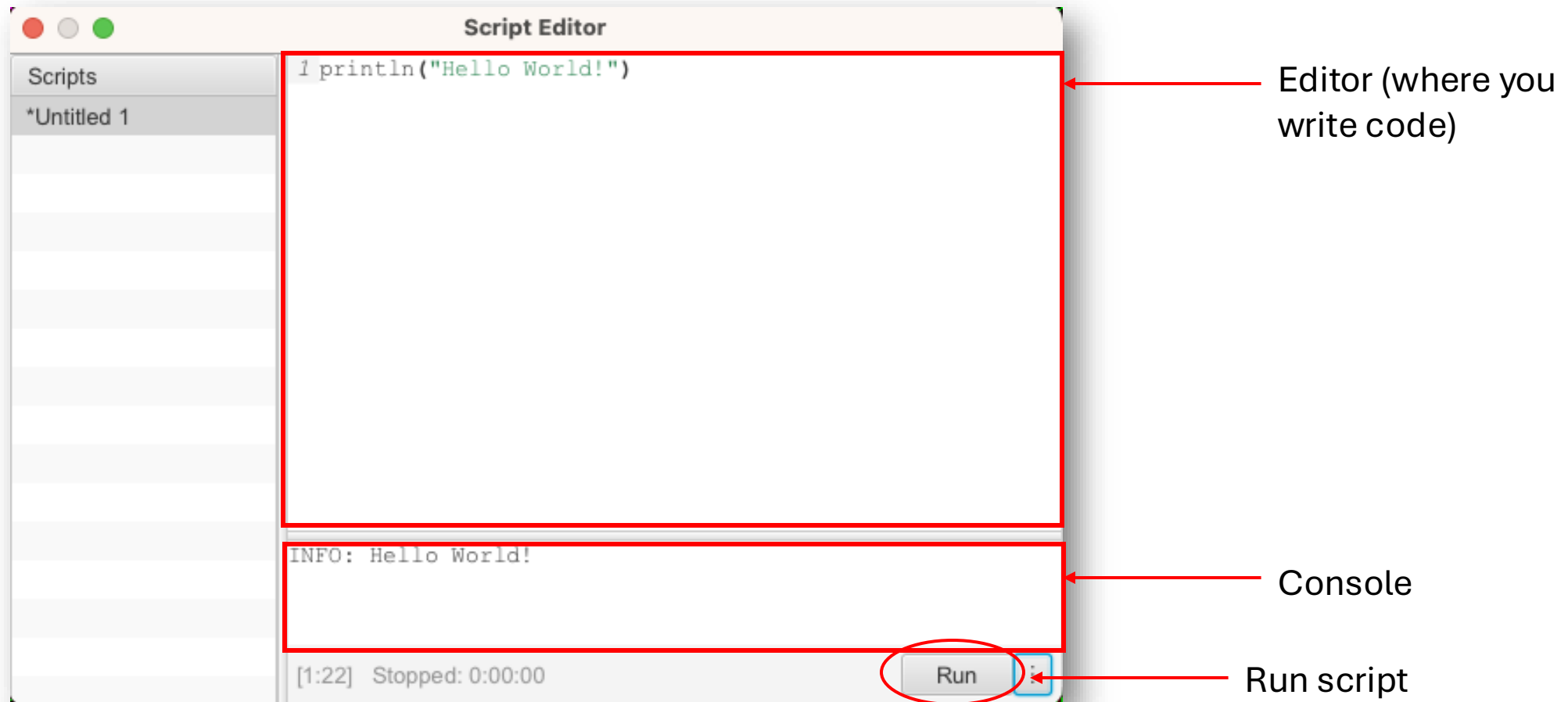
Scripting in QuPath

- *Automate > Script editor*

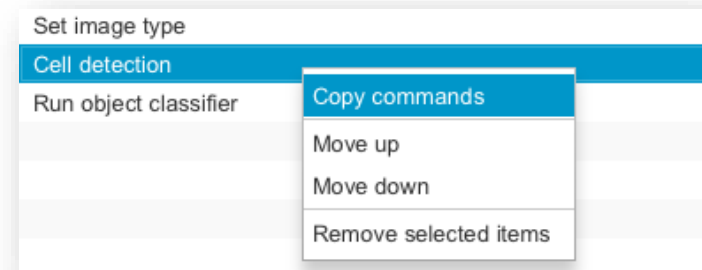
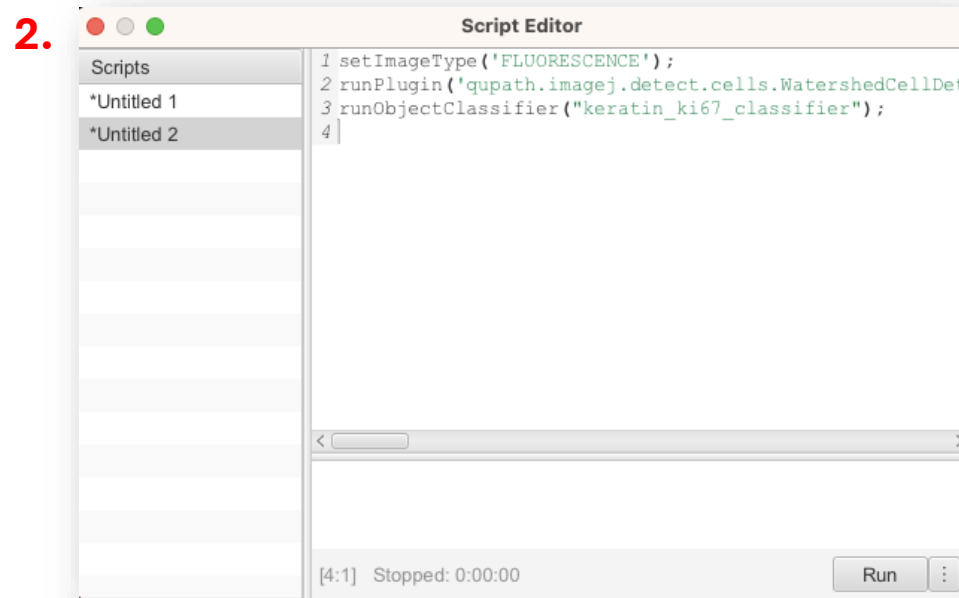
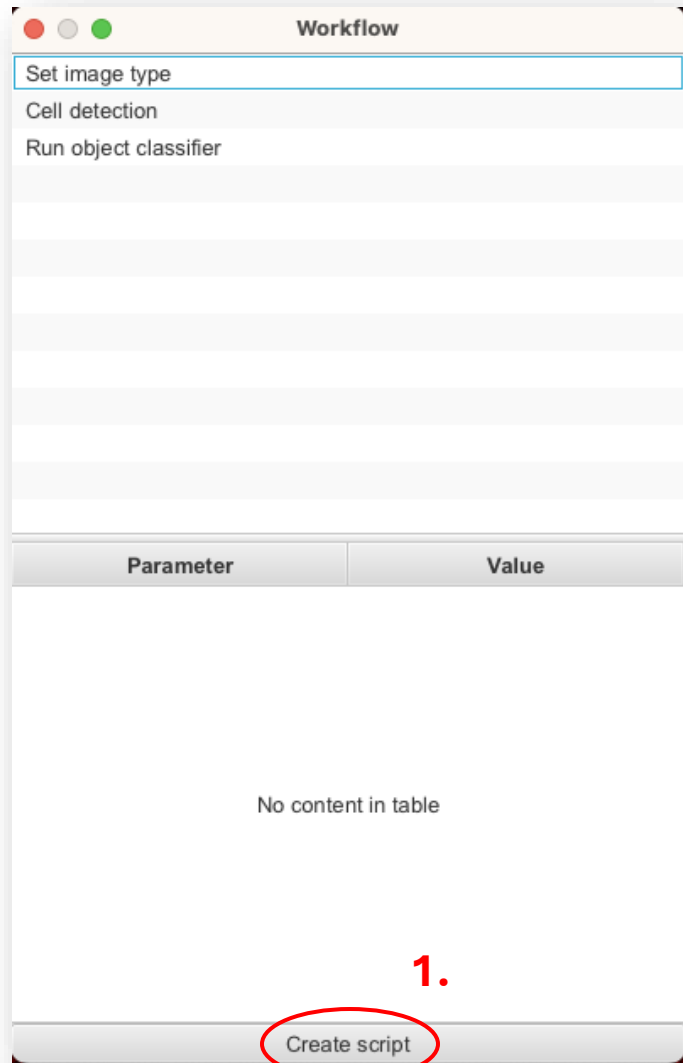


Hello World!

- Automate > Script editor

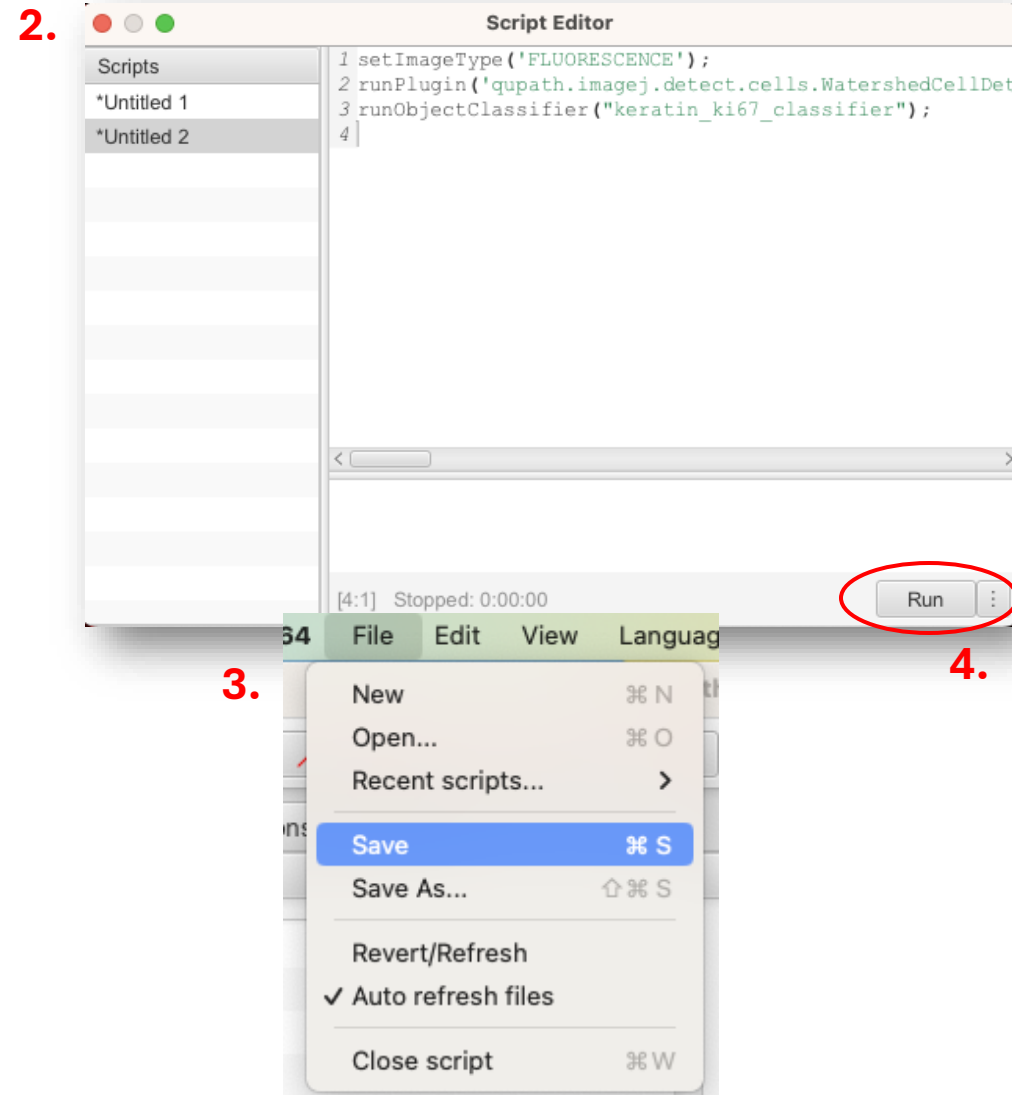
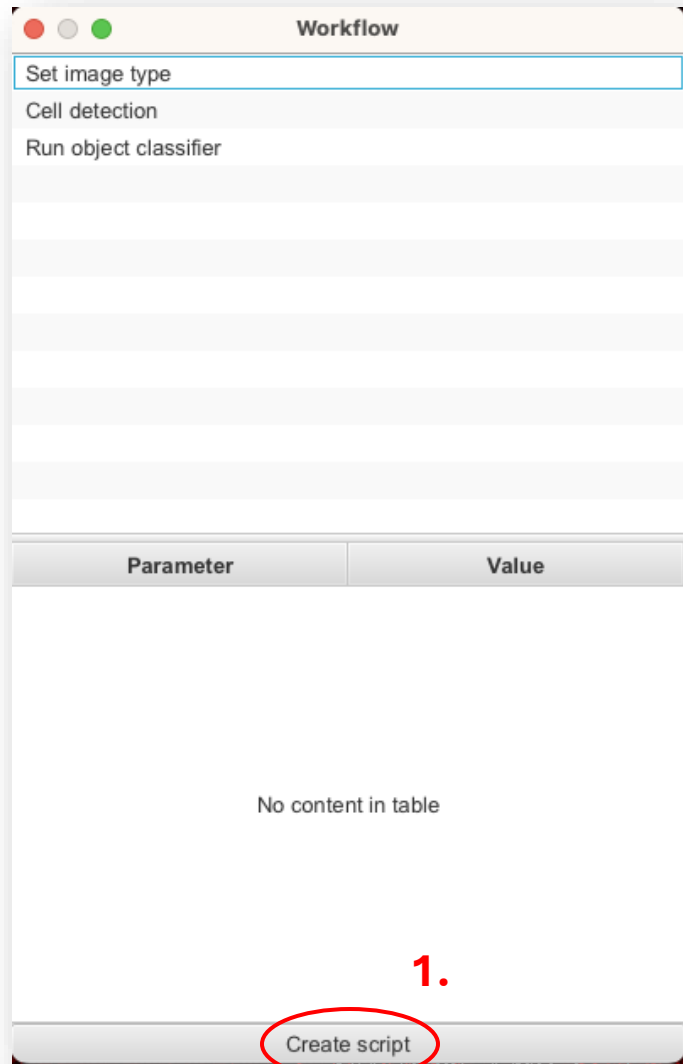


Clean your workflow for cell detection and classification



Edit the sequence of steps in the workflow using right-click

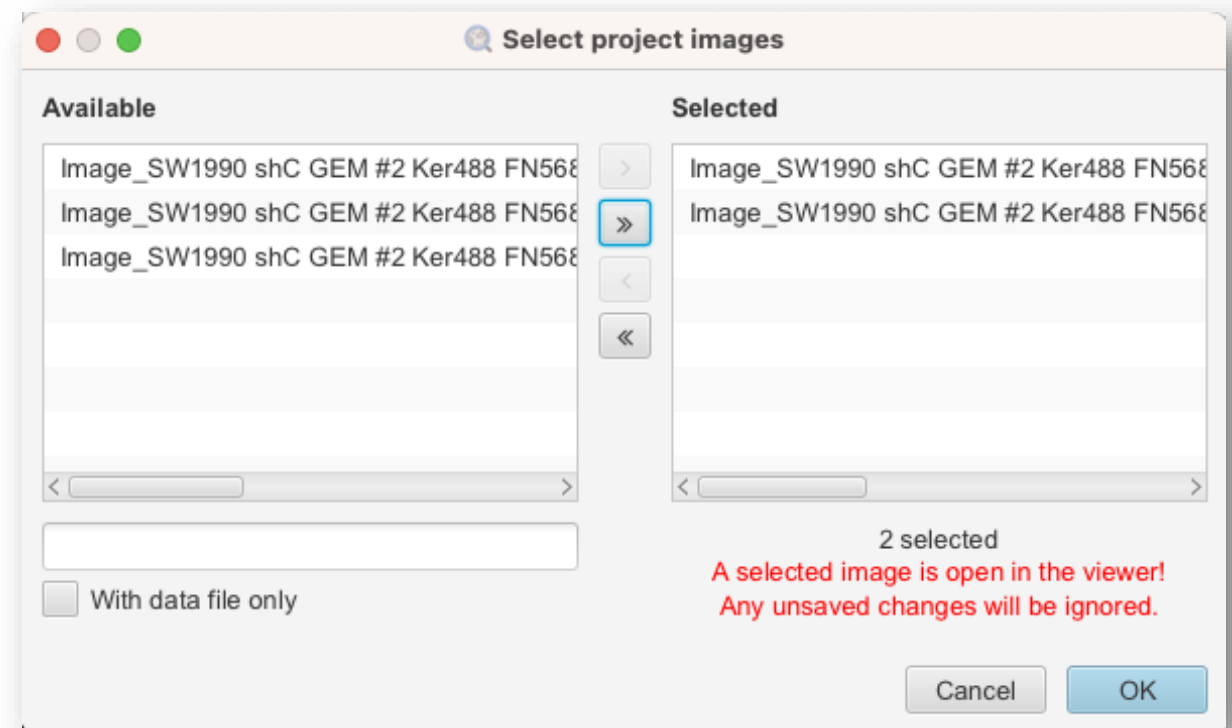
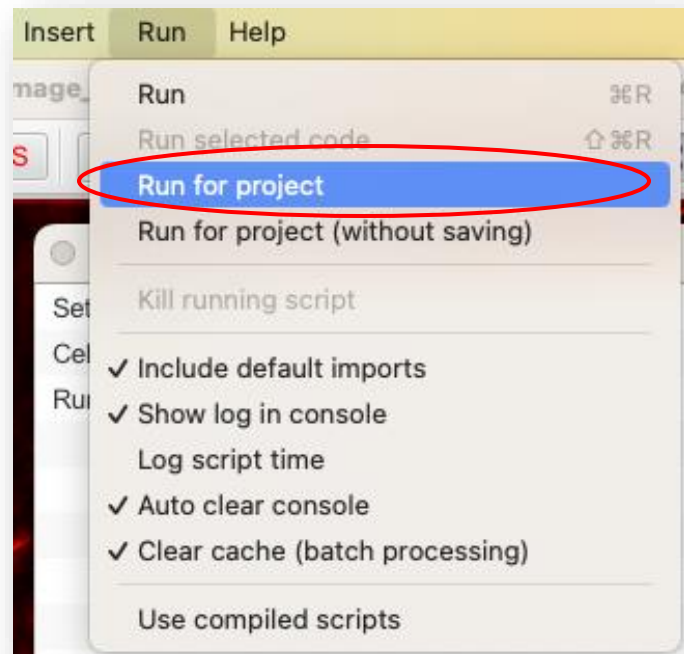
Save and run a script



Scripts can be repeated on a batch of images

QuPath allows for batch processing: scripts will run on multiple images loaded in the project.

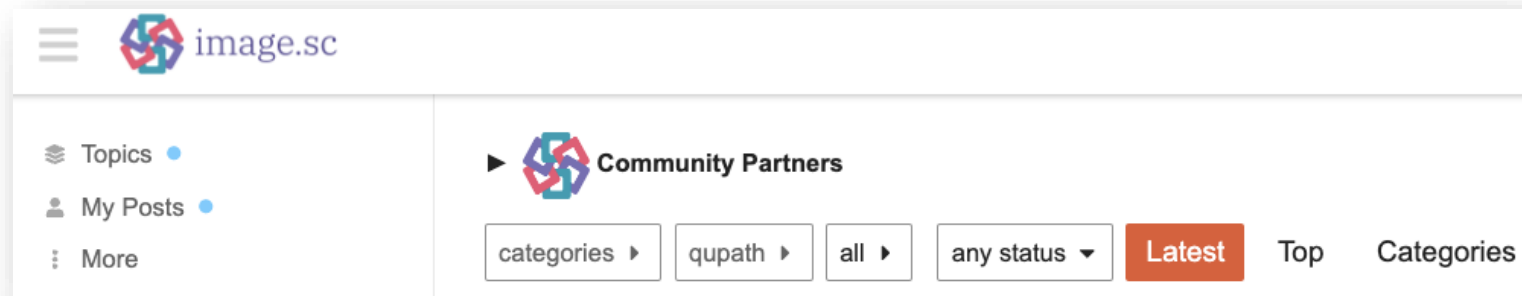
- *Run > Run for project*



Select images you wish to run the script on.

Further resources

- QuPath documentation
 - Scripting:
<https://qupath.readthedocs.io/en/latest/docs/scripting/overview.html>
 - QuPath's API docs: <https://qupath.github.io/javadoc/docs/>
- The Forum
 - Where to contact the developers of most image analysis tools
 - If you have a question, likely someone else already asked



QuPath on O2

Ranit Karmakar

Give us your feedback!



<https://tinyurl.com/52hu7bkt>

Further resources

- QuPath documentation
 - Scripting:
<https://qupath.readthedocs.io/en/latest/docs/scripting/overview.html>
 - QuPath's API docs: <https://qupath.github.io/javadoc/docs/>
- The Forum
 - Where to contact the developers of most image analysis tools
 - If you have a question, likely someone else already asked

