

Introduction Course to **Image Analysis** with **QuPath**





Introduction

- What is Qupath?
- System Requirements
- Supported Image Formats



What is QuPath?

- OS Software for **Image Analysis** focused on **Digital Pathology**.
- It facilitates the **Visualization** (WSI/standard Formats: SVS, NDPI, OME.TIFF...), **Annotation** (detailed, making easier to identify and classify regions), and **Analysis** of large whole-slide Images.



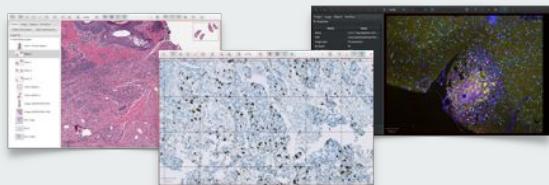
Who is QuPath designed for?

- Researchers in **biology** or **pathology**.
- Developers for extension of QuPath functionality with **Groovy** or **Java**.



For which Tasks is QuPath suitable?

- **Cell Detection**
 - Automatically **detect** and **count cells** in tissue.
 - Differentiate between **cell types** based on staining.
- **Tissue Classification**
 - Classify **different tissue regions** (e.g., tumor vs. normal tissue).
- **Fluorescence Quantification**
 - Measure **fluorescence intensity** in immunofluorescence images.
 - Analyze **multiple fluorescence channels** simultaneously.





System Requirements

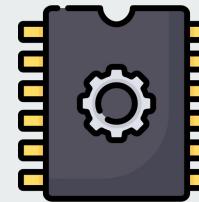
Supported OS

- **Windows:** Windows [10 or later](#)
- **macOS:** macOS [10.13](#) (High Sierra) or later
- **Linux:** Most modern distributions (e.g., [Ubuntu](#), [Fedora](#))



Hardware Requirements

- **Multi-core processor** ([Intel i7](#) or equivalent, or better)
- **Memory (RAM):** At least 8 GB ([16 GB](#) or more recommended for large images)
- **Graphics:** Dedicated graphics card ([NVIDIA](#) optional but recommended for DL Segmentation)



Additional Recommendations

- **High-Resolution Display:** For better visualization of images
- **Internet Connection:** For downloading updates and accessing online resources





Supported Image Formats

QuPath supports a **wide range of image formats**, thanks to its integration with **Bio-Formats** and **OpenSlide** libraries.

Whole Slide Image Formats

Aperio: .svs, .tif

Hamamatsu: .vms, .vmu, .ndpi

Leica: .scn

MIRAX: .mrxs

Philips: .tiff

Sakura: .svslide

Trestle: .tif

Ventana: .bif, .tif

Generic Tiled TIFF: .tif



Standard Image Formats

TIFF: .tif

JPEG: .jpg, .jpeg

PNG: .png

OME-TIFF: .ome.tif

Microscopy Formats (by using BioFormats)

CZI: .czi (Zeiss)

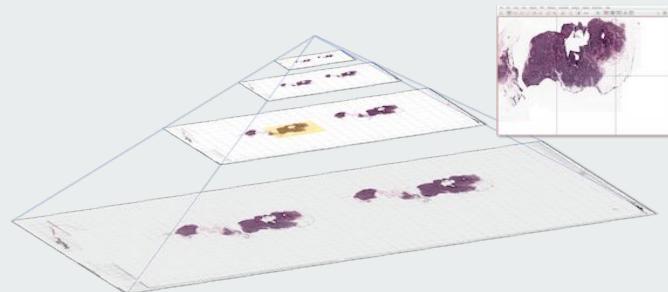
DICOM: .dcm

LIF: .lif (Leica)



Additional Notes

- **Pyramidal Images:** It handles large pyramidal images, which store **data** at **multiple resolutions**.
- **Multidimensional Data:** Supports **z-stacks**, time series, and multiplexed images.





Getting Help



- [This documentation](#)
- [Video tutorials](#)
- [forum.image.sc](#) - ***new user forum***, shared with other open source projects!
 - Fantastic post introducing QuPath from a user's perspective [here](#)
- [Issues on GitHub](#) - ***bug-reports, not general questions***
- [Pete's blog](#) - especially useful for opinions, plans & scripting info
- [Twitter](#)
- The [FAQs](#) page
- [QuPath Users Google Group](#) - ***old user forum, now replaced by forum.image.sc***

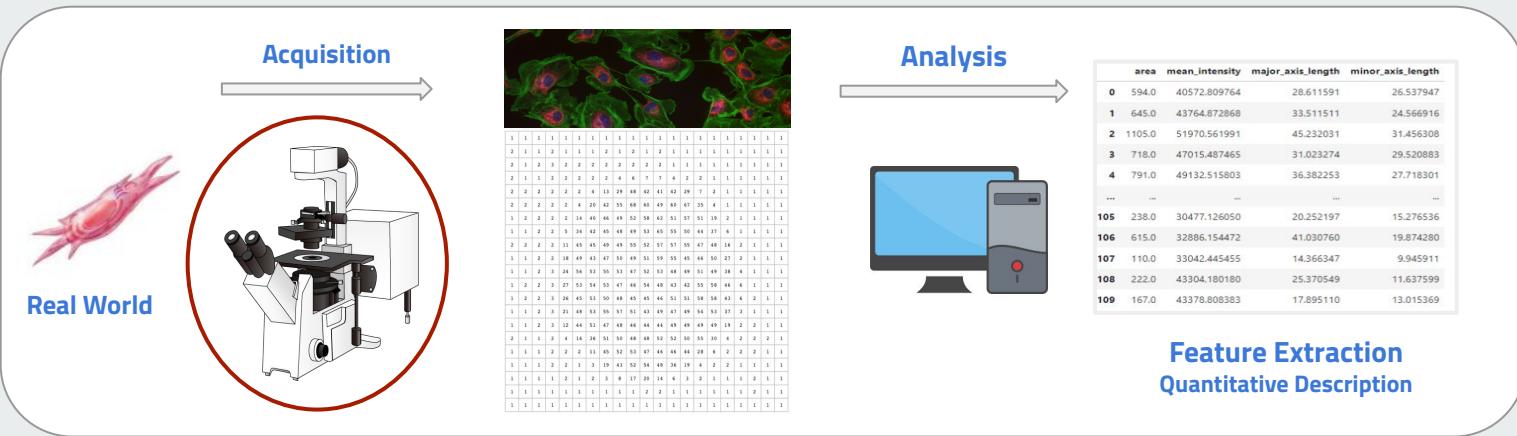


Concepts

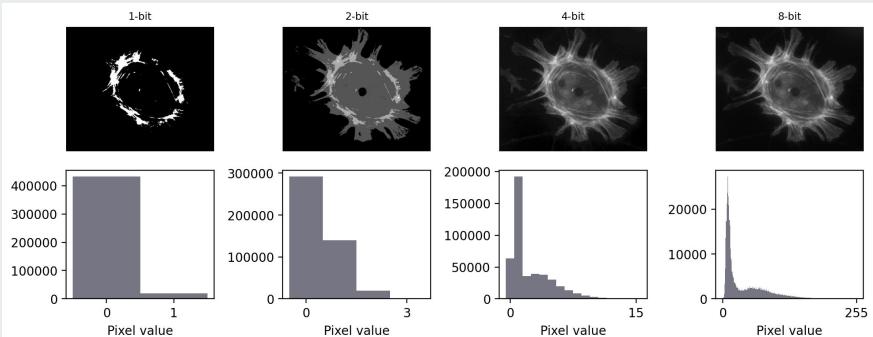
- Image Analysis Fundamentals
- Channels & Colors
- Image Processing & Analysis
- Understanding Objects
- Object Hierarchy
- Classifications & Subclassifications
- Type of Measurements



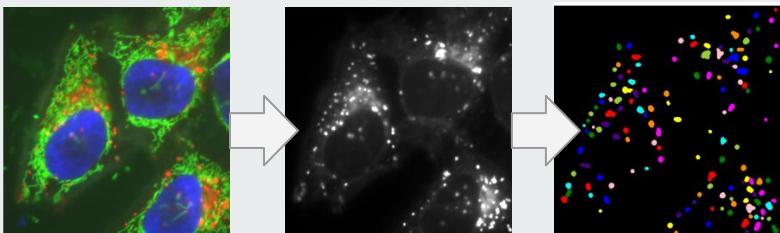
Image Analysis Fundamentals



What is Bit-Depth?



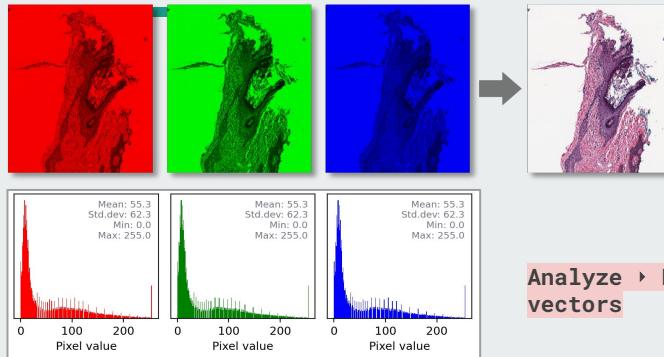
Instance Segmentation





Channels & Colors

RGB Images

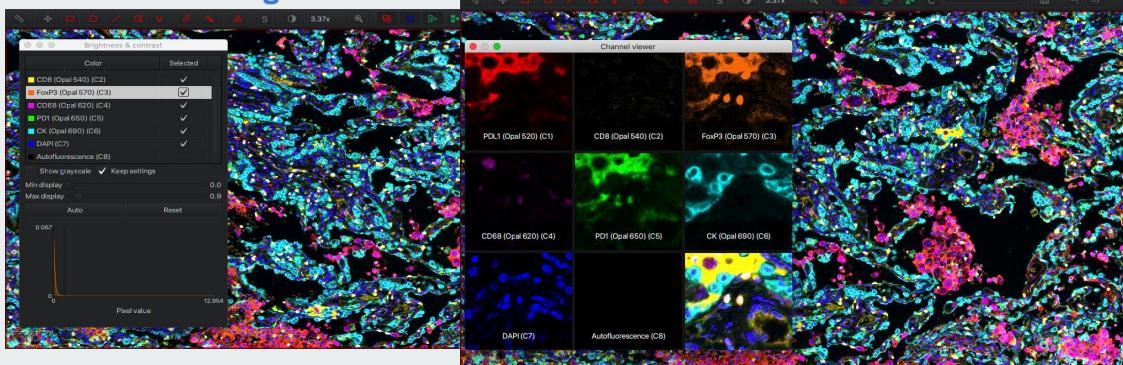


Color Deconvolution in Brightfield Images

- Each **stain** has a unique color, known as a stain vector.
- **Deconvolution Algorithm** uses stain vectors to **mathematically separate the RGB image up to 3 stains** (channels)

Analyze > Preprocess > Estimate stain vectors

Multichannel Images



"**Stain vectors** separate different stains within a **single RGB image**, while **multichannel images** capture **distinct signals in separate channels**"



Image Processing & Analysis

Image Processing

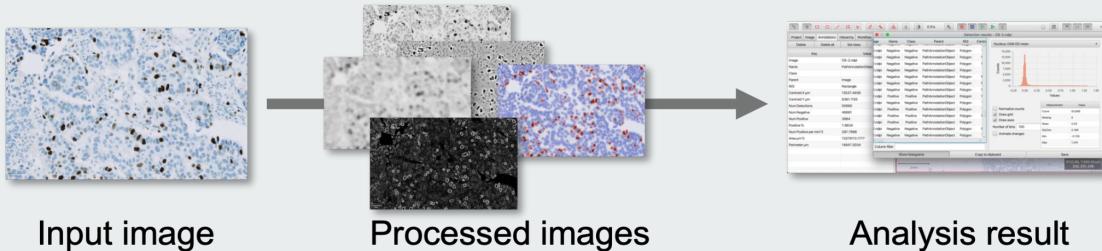


Image processing takes an **image** as input and gives **another image** as output



Image Analysis

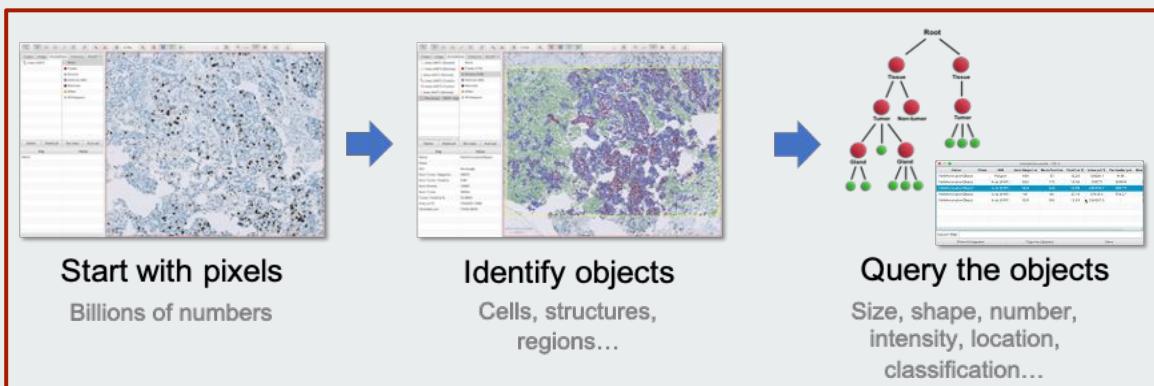


Image analysis takes an **image** as input and gives **measurements** as output



Isolate pixels with similar features
(intensity, morphology...)



Understanding Objects

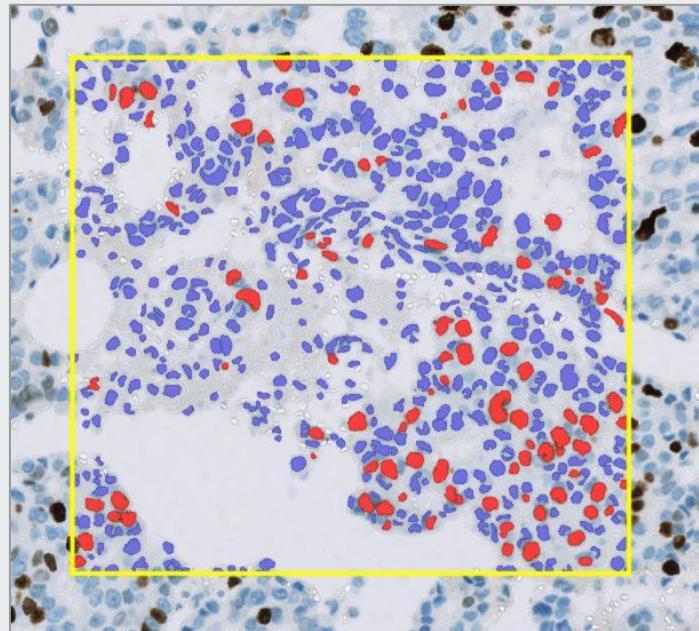
You can think of an **object** as something in an image that **QuPath** can **identify, classify and measure**.

Types of Objects

- **Annotations:** Objects that you usually create **yourself**, by drawing on the image
- **Detections:** Objects that **QuPath** usually creates **for you**, e.g. by detecting cells (where each cell is an object)

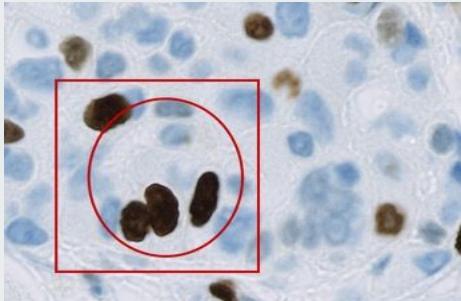
Annotations & Detections

- Annotations are **FLEXIBLE**:
 - **NO** more than **100** per image.
 - May be **edited manually**.
 - **Often** to define **regions** where **detections** are generated.
- Detections are **STATIC**:
 - Often thousands (millions) of detections per image.
 - They can be **deleted**, but **NO edited**.



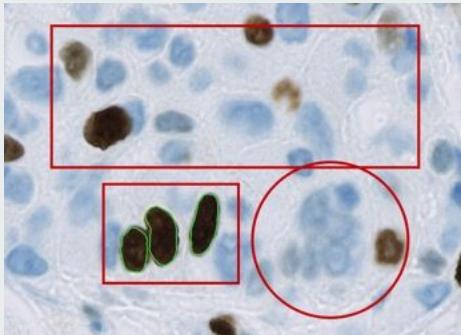
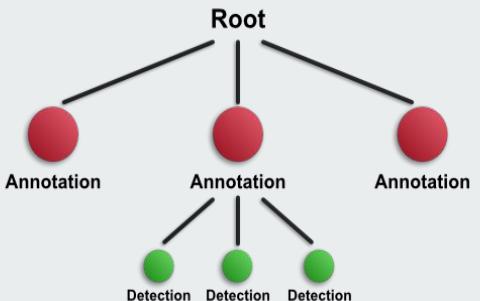


Object Hierarchies



A **rectangular annotation** containing a **circular annotation**

Objects are organized in a **tree-like structure**, where **each object** has **one parent** and **multiple child objects**.



A **rectangular annotation** containing **3 detections**

Visualizing the Hierarchy

The screenshot shows a software interface for visualizing image annotations. On the left, a tree view of the hierarchy:

- Outer rectangle (Region*) (20/98 objects)
 - Annotation (Polygon) (Region*) (5 objects)
 - Annotation (Polygon) (Region*) (5 objects)
 - Detection (Polygon)
 - Detection (Polygon)
 - Detection (Polygon)
 - Detection (Polygon)
 - Detection (Polygon)
 - Annotation (Polygon) (Region*) (11 objects)
 - Annotation (Polygon) (Region*) (14 objects)
 - Annotation (Polygon) (Region*) (7 objects)

On the right, a microscopy image of a tissue sample with various regions highlighted by colored polygons (blue, green, yellow). Below the image is a table of key statistics:

Key	Value
Image	CMU-1.svs
Name	Region*
Class	Region*
Parent	Outer rectangle
ROI	Polygon
Centroid X µm	15180.1423
Centroid Y µm	4009.2304
Num Detections	5
Area µm ²	431.1863
Perimeter µm	84.7826

Key Concepts

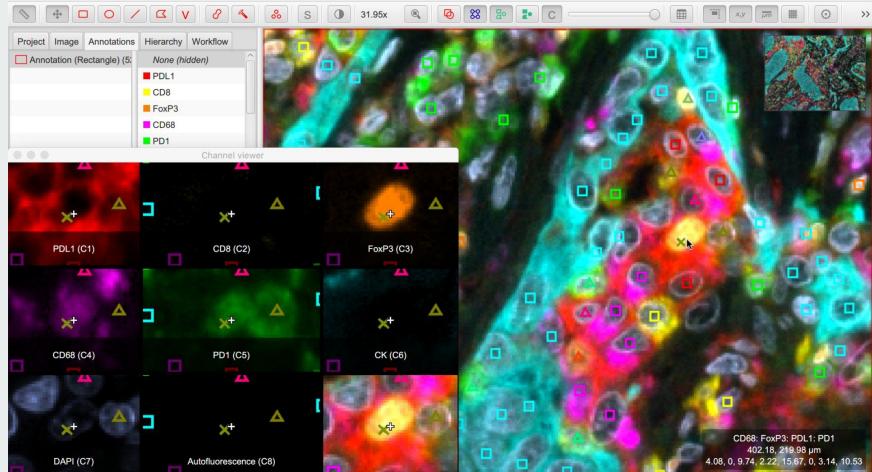
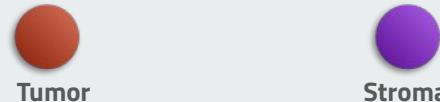
- **Root Object** is the base of the hierarchy (Image) with **NO parent**
- **Annotations** (e.g., regions of interest) and **detections** (e.g., cells)



Object Classifications & Subclassification

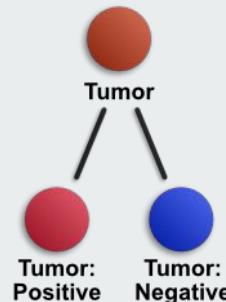
Primary Classification

- Assigns a main category to objects (e.g., Tumor, Stroma).



Subclassification

- Derived from **Primary Classifications**
- Objects can be further **subclassified** based on additional criteria (e.g., Positive, Negative).





Measurements

All **objects** have their own measurement lists.



Different Types of Measurements

Measurement Lists (Static)

- Lists of permanent measurement of **child objects (Detections)**.

Summary Statistics (Dynamic)

- Calculated based on **child objects (Detections)** related to **parent Annotation**.

Shape Statistics (Dynamic)

- Calculated for each **Annotation** and updated automatically.



Getting Started

- Installation
- Create a Project
- Two Essential Tips: Drag & Drop/Command List
- Opening an Image
- Image Properties
- Drawing Annotations
- Getting Help
- Manual Counting



Installation

1. Visit the Official Website

- Go to > QuPath GitHub Page
- Navigate to > The "Releases" section to find the latest version.



qupath/qupath
QuPath - Open-source bioimage analysis for research

A 16 Contributors 38 Issues 1k Stars 289 Forks

2. Download the Installer

- Windows (.msi/.zip), macOS(.pkg), Linux(.tar.xz)
- Download: Click the download link to get the installer file.

Releases 43

v0.5.1 Latest on Mar 4, 2024

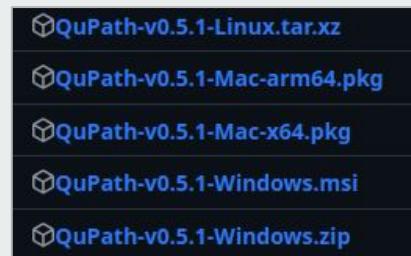
+ 42 releases

3. Install QuPath

Windows:Run the .exe file.

macOS:Open the .pkg file.

Linux: Extract the .sh file.





Two Essential Tips: Drag & Drop/Command List

Drag & Drop

Most files supported by QuPath can be opened simply by dragging them on top of the main window. This includes:

- Projects (either the `.qpproj` file or the entire folder)
- Images
- Scripts

Command List

Command	Menu Path
Add column after	TMA>Add...>
Add column before	TMA>Add...>
Add row after	TMA>Add...>
Add row before	TMA>Add...>
Align annotations within TMA core (TMA)	TMA>
Clear TMA grid	TMA>
Create cytokeratin annotations (TMA, IHC)	Analyze>
Export TMA data	File>
Find convex hull detections (TMA)	TMA>
Immune scorer (TMA only)	Analyze>
Import TMA map	File>
Kaplan Meier plot (TMA)	TMA>
Launch exported TMA data viewer	File>
Relabel TMA cores	TMA>
tma	

View > Command List or type **Ctrl + L**



Create a Project

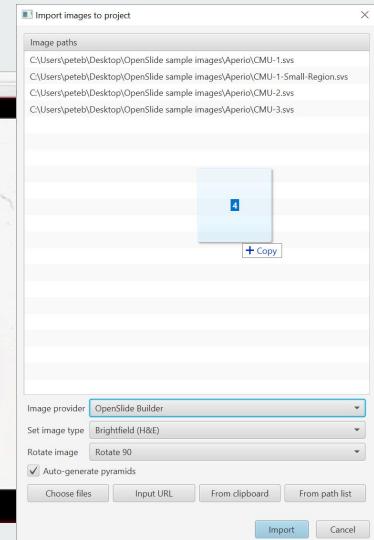
1. Choose a Folder

- **File > Project... > Create new project**
- Choose a **directory** where you want to save your project.
- or **Drag & Drop the folder** on top of QuPath



2. Add Images

- Adding single images to a project
File > Open
- Adding multiple images to a project
File > Project... > Import images to project



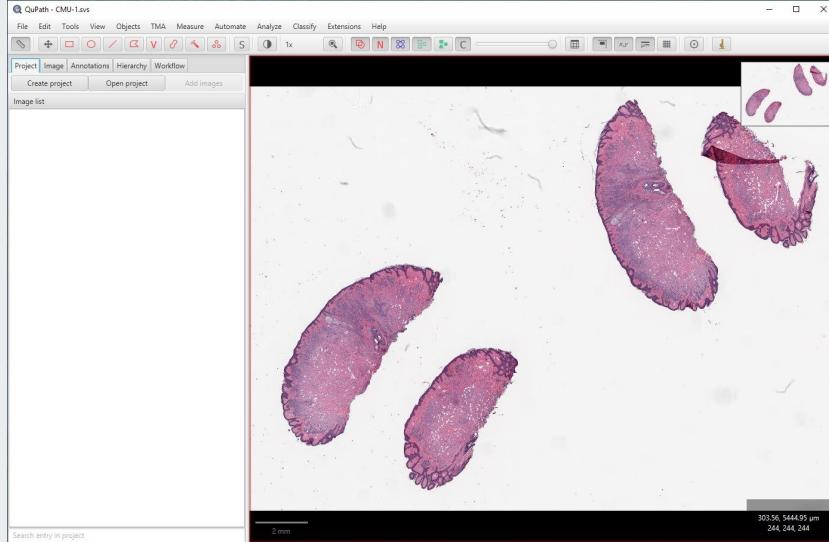
3. Reopen a project

- Drag the **.opproj** file within the project folder onto QuPath
- **File > Project... > Open project**
- **File > Recent Projects... > [your project]**

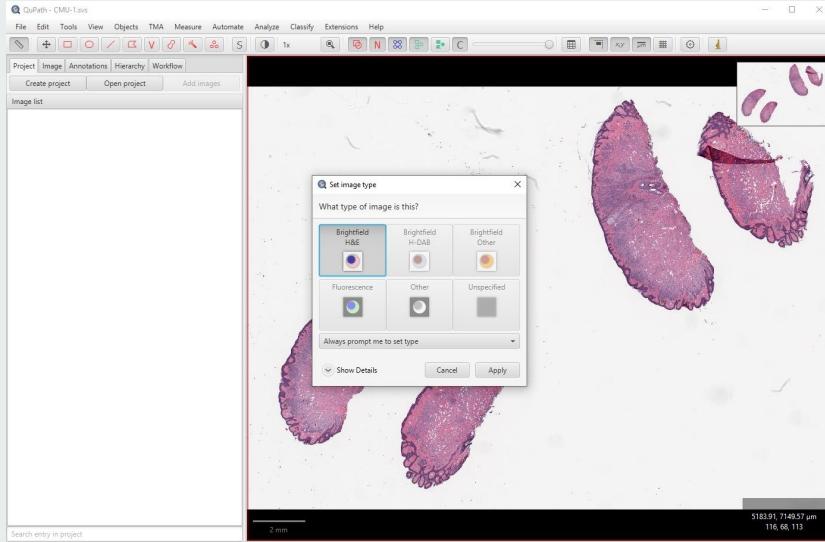


Open an Image

1. File ▶ Open... or Drag & Drop



2. Setting the Image Type

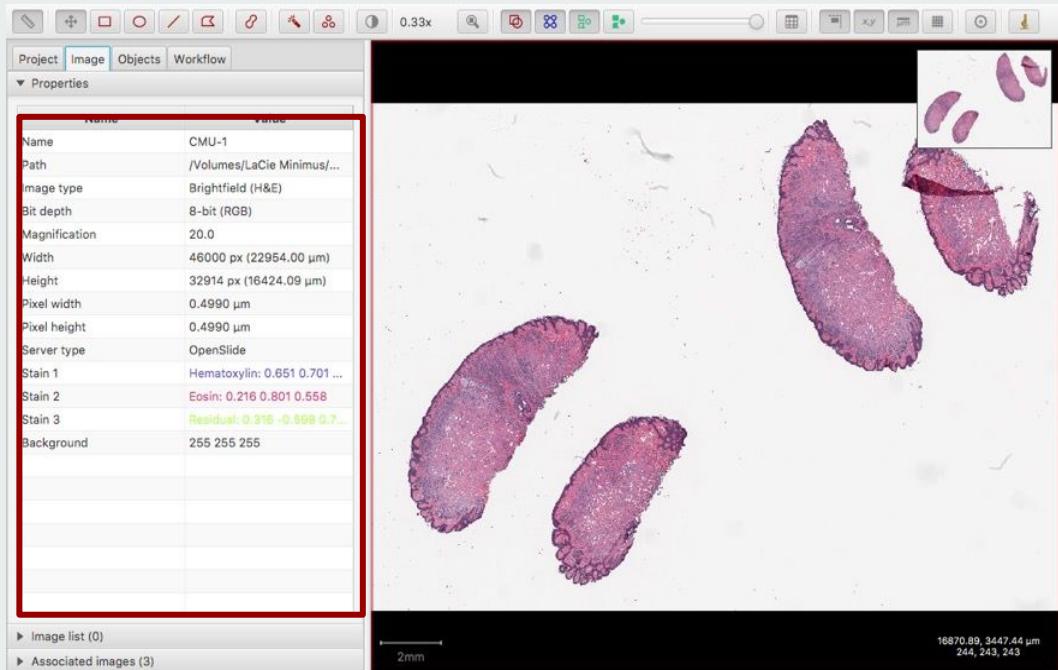


Important

- **Navigate to the Image:** In the project browser, find the image you want to open.
- **Double-Click:** Double-click on the image file to open it in the image viewer.
- **QuPath can automatically estimate the image type by [Set image type](#)**
- However, sometimes the automatic estimate is wrong!!
- **if not, under [Edit ▶ Preferences...](#)**



Image Properties



Key Features

- **Image Type:** e.g., Brightfield, Fluorescence
- **Pixel Size:** Shows the **pixel width and height in physical units**, critical for measurements.
- **Image Dimensions** in pixels (Width and Height)
- **Bit Depth** which affects the range of intensity values.

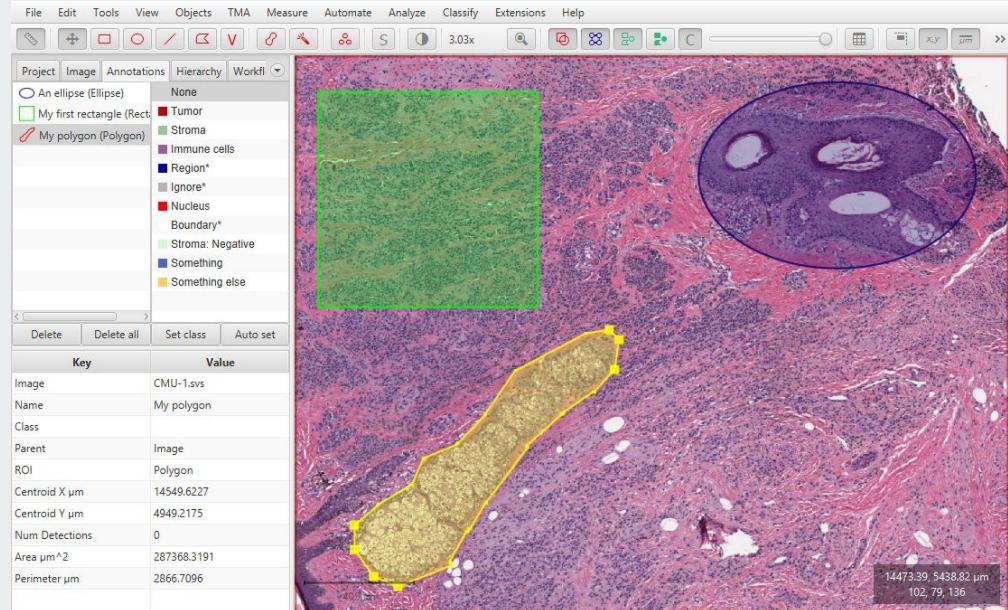




Annotation Colors & Properties

Draw Annotation Shapes

- Rectangle
- Ellipse
- Polygon
- Line
- Brush



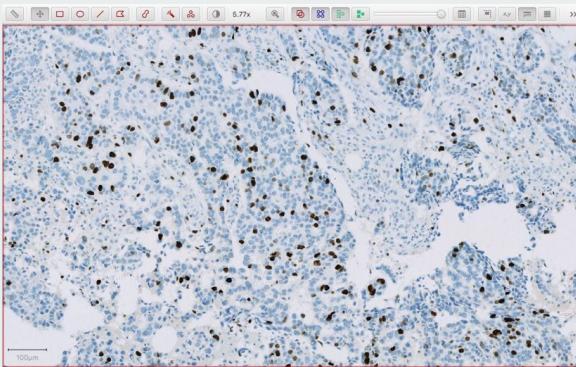
Showing & Hiding

- Show and hide annotations - shortcut **A**
- Show and hide a TMA grid (only relevant for tissue microarrays) - shortcut **G**
- Show and hide detections - shortcut **D**
- Fill and unfill detections - shortcut **F**

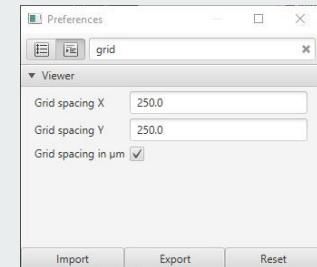
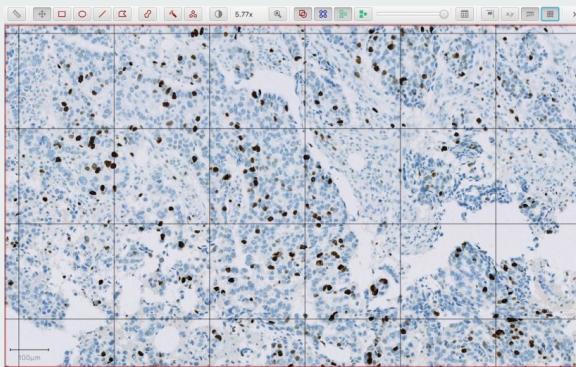


Manual Counting

1. Choosing a Region

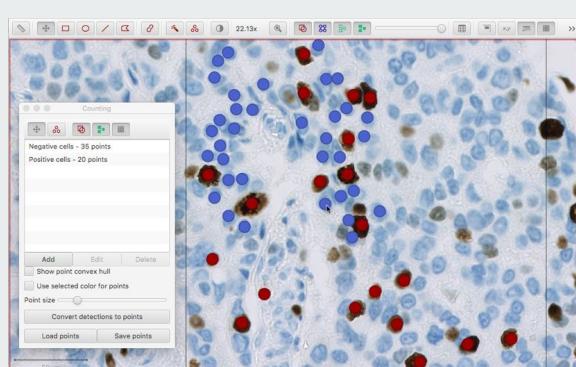
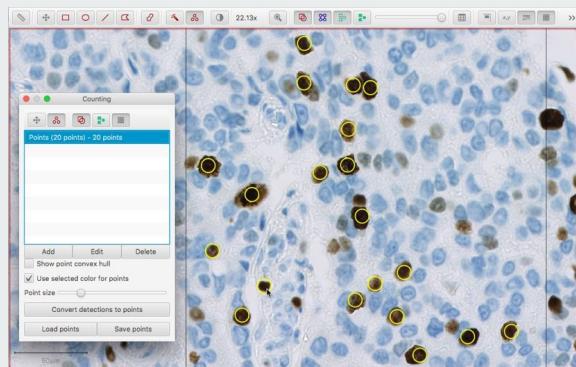


2. Show grid Toolbar Button



[Edit > Preferences...](#)

2. Clicking Cells





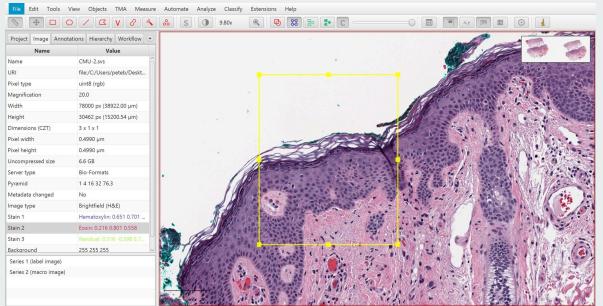
Tutorials

- Separating Stains
- Detecting Tissue
- Measuring Areas
- Cell Detection
- Positive Cell Classification
- Train a Cell Classifier based on Annotations
- Density Maps
- Exporting Measurements
- Multiplex Analysis



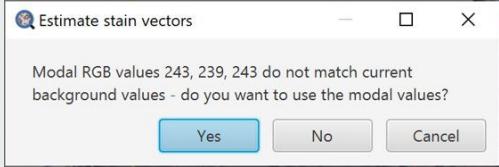
Separating Stains: Estimating Stain Vectors on H&E

1 Find a Representative Region



Manually selecting a region to for Stain estimation

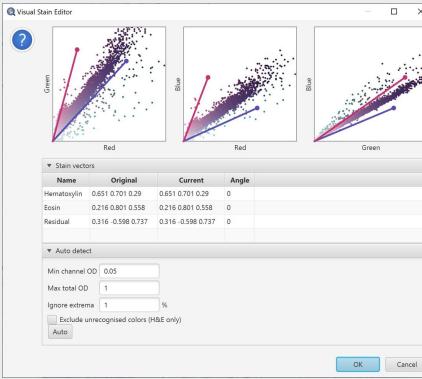
2 Run Estimate Stain Vectors



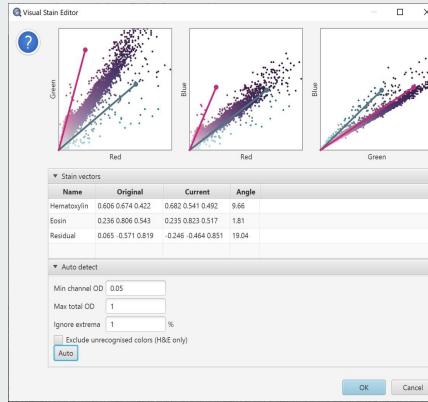
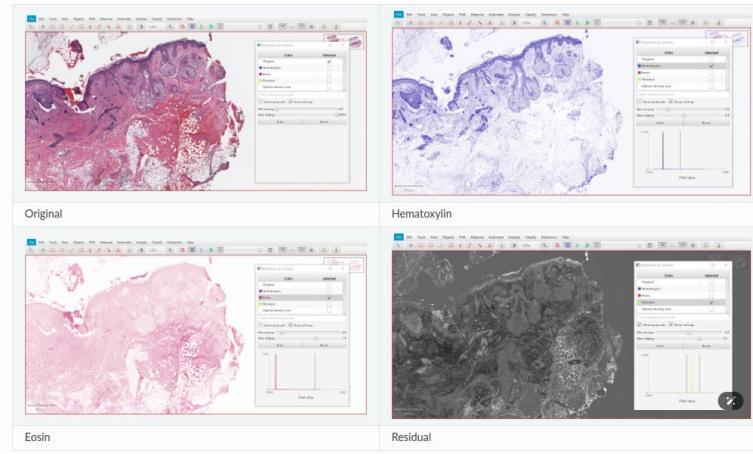
Stain estimation background prompt

Analyze > Preprocessing > Estimate
stain vectors

3 Check Scatterplots



4 View the Results

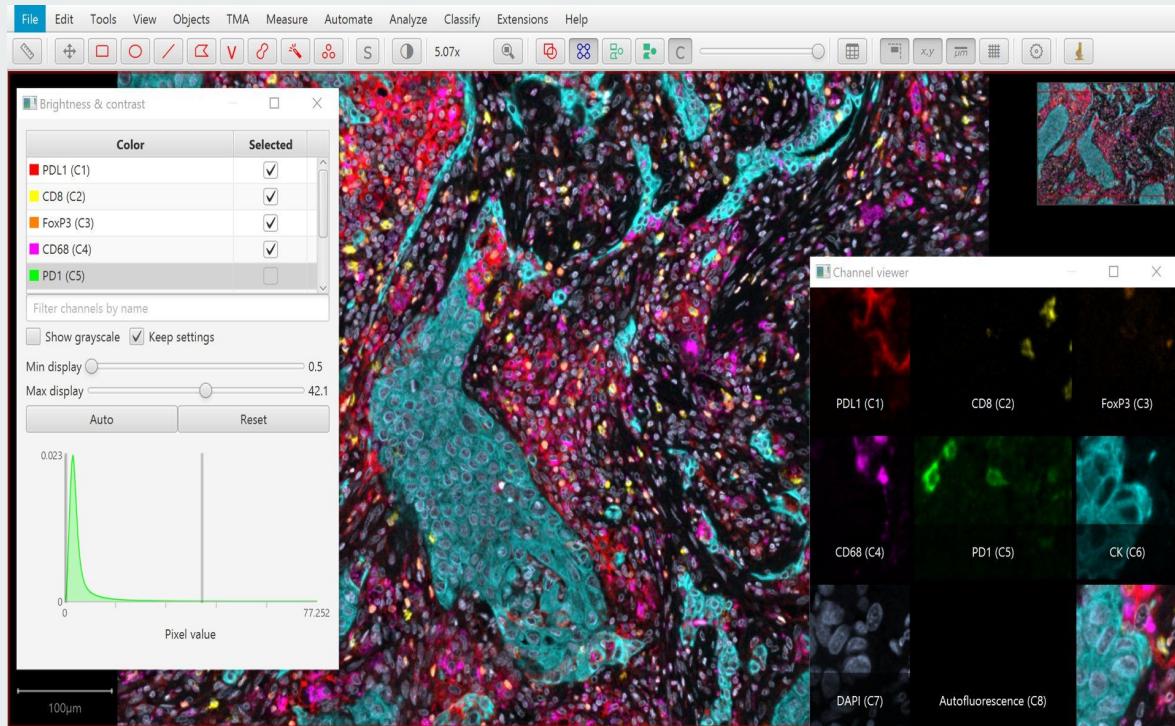


Auto-estimate
stain



Fluorescence Images: Multichannel Images

View ▶ Show channel viewer



- QuPath assumes each channel corresponds to a different marker.
- Separating these simply involves splitting the channels.

Brightness/Contrast tool and channel viewer with a multiplexed image

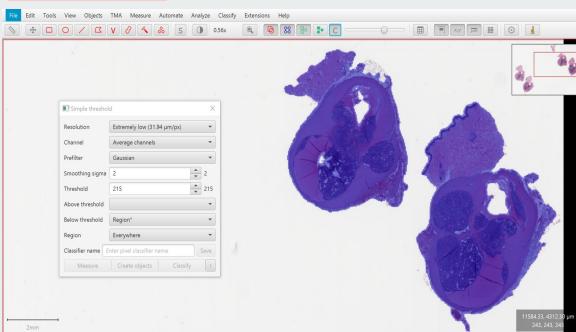
Brightness/Contrast



Detecting Tissue

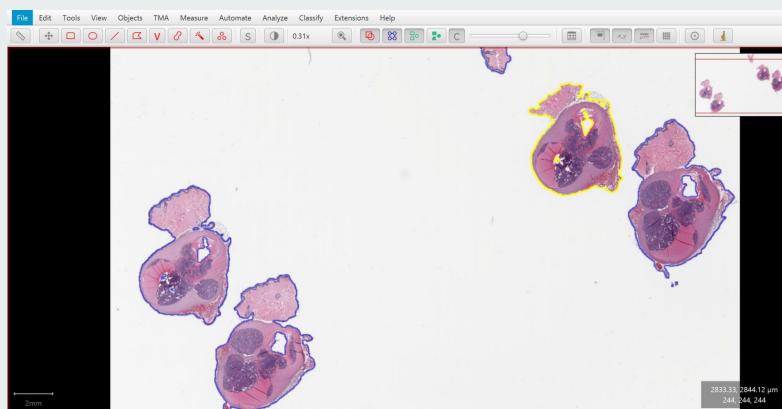
1 Thresholds in Action

Classify > Pixel classification > Create threshold

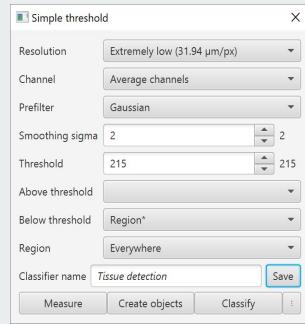


Previewing the thresholded image

4 The end result of tissue detected by thresholding



2 Adjusting Parameters & Saving your Thresholder

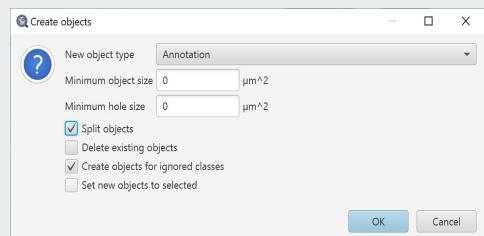


Saving a thresholder to a project

3 Creating Objects



Choose the parent objects within which to threshold, or the full image

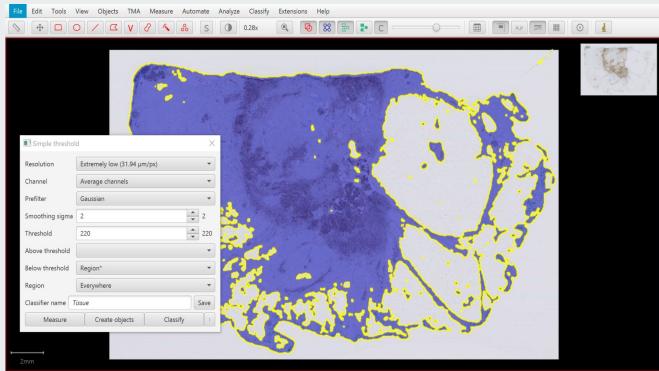


Additional options for filtering the objects created by thresholding



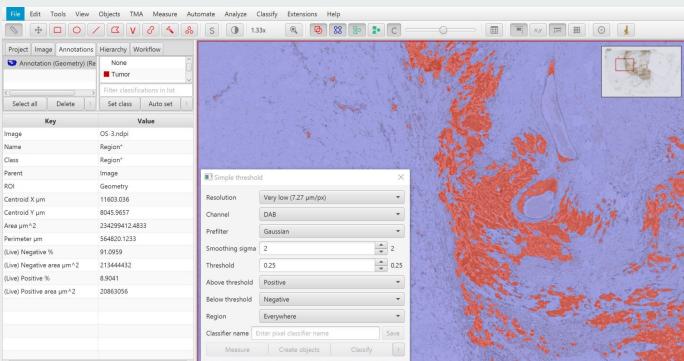
Measuring Areas

1 Define the Region of Interest



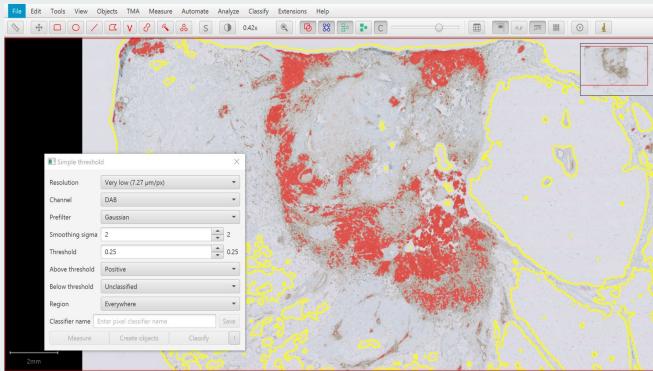
Tissue detected by thresholding

3 Viewing Measurements



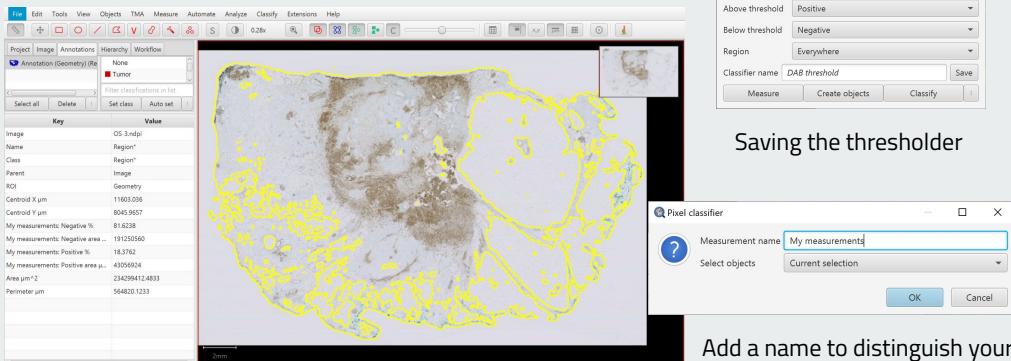
A higher threshold results in a lower stained percentage measurement

2 Threshold Stained Areas



Preview of stained area detection

4 Generating Results



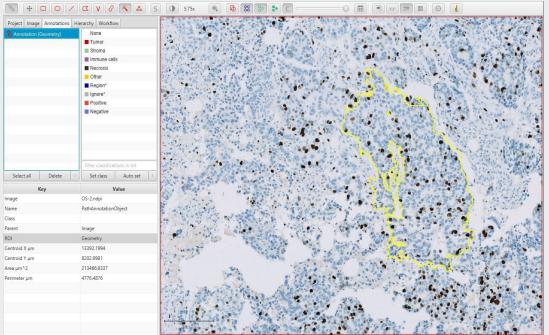
Results of calculating the stained area percentage.

Saving the thresher

Add a name to distinguish your measurements

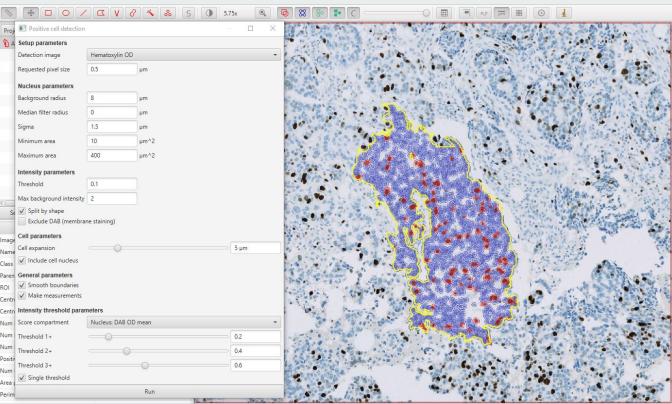
Cell Detection + Positive Cell Classification

1 Annotate ROI

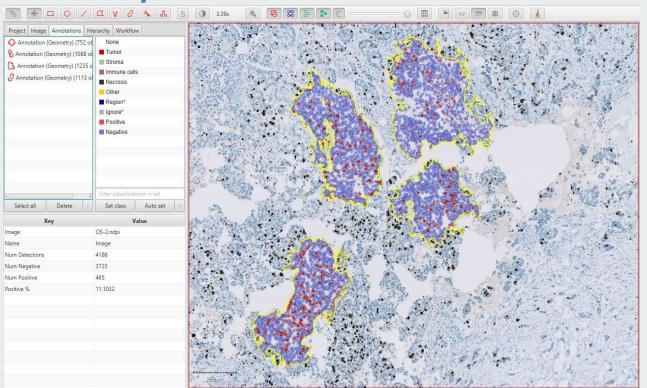


Ki67 image with annotation

2 Run Positive Cell Detection

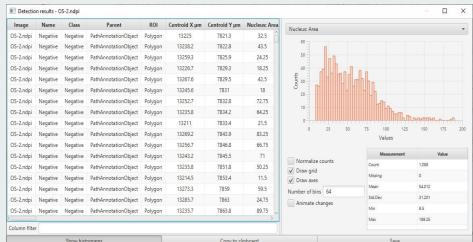


4 Analyze Additional Annotations



Positive Cell Detection in parallel for multiple annotations for Ki67 image

3 View Cell-by-Cell Results



Cell Detection Results Table

Measure > Show detection measurements

5 View Results

Analyze > Cell detection > Positive cell detection

Name	Class	Parent	ROI	Centroid X μm	Centroid Y μm	Num Detections	Num Negative	Num Positive
PathAnnotationObject	Image	Geometry	13930.1	7970.7		0	0	0
PathAnnotationObject	Image	Geometry	13392.2	8203		1088	965	123
PathAnnotationObject	Image	Geometry	13936.3	8539.6		0	0	0

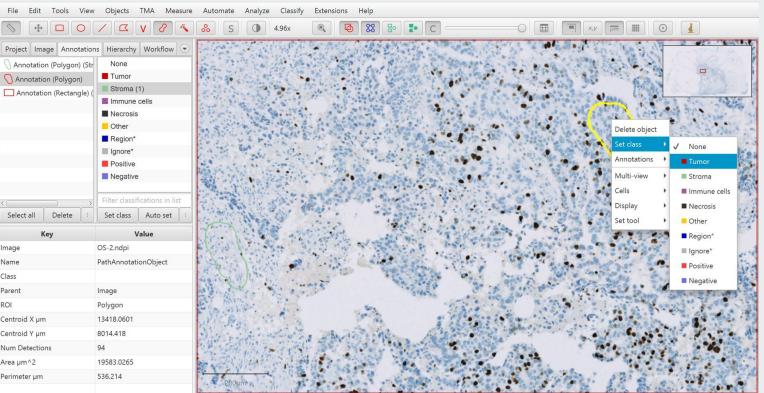
Annotation Results Table

Measure > Show detection measurements

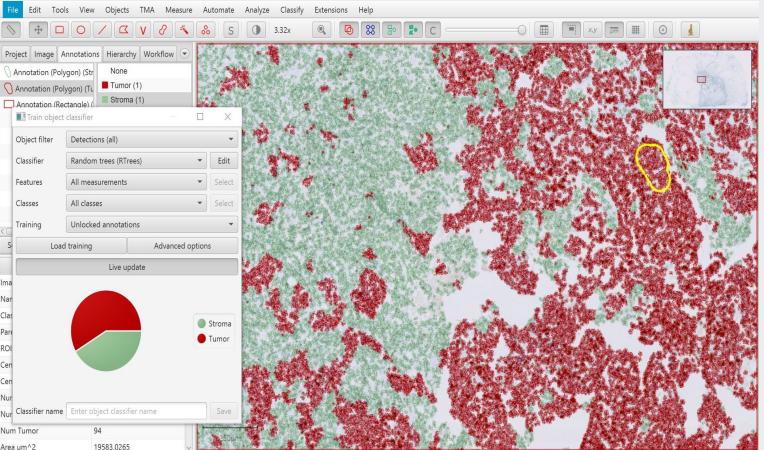


Train a Classifier based on Annotations

1 Annotate Regions Containing different Cell Types (Tumor/Stroma)

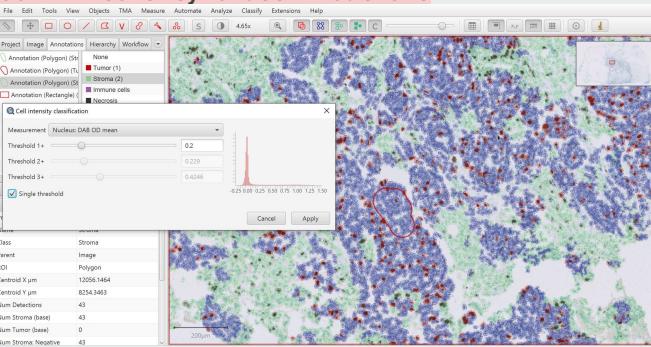


2 Training Cell Detection with live Update



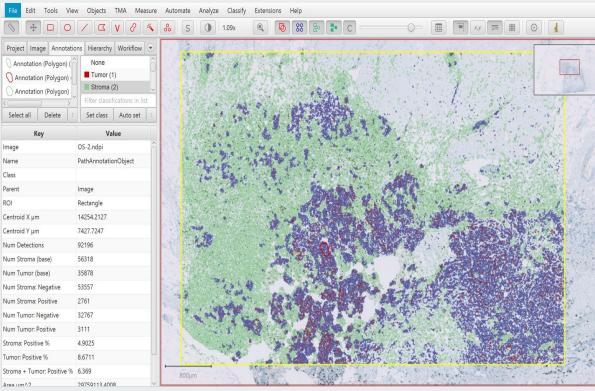
3 Save the Classifier + Apply Cell Intensity

Classify > Object classification > Set cell intensity classifications

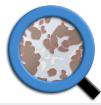


DAB staining intensity classification

4 View the Results

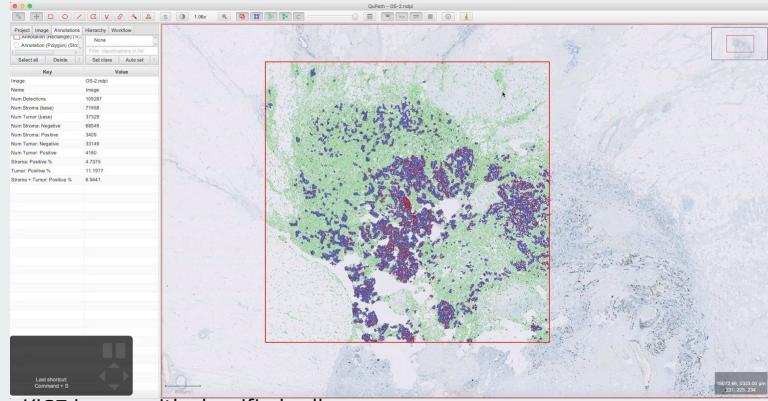


Ki67 analysis results with cell classification

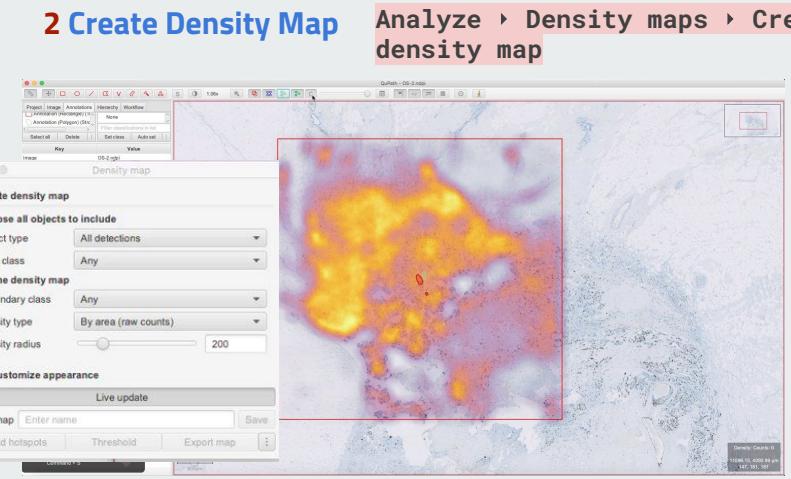


Density Maps

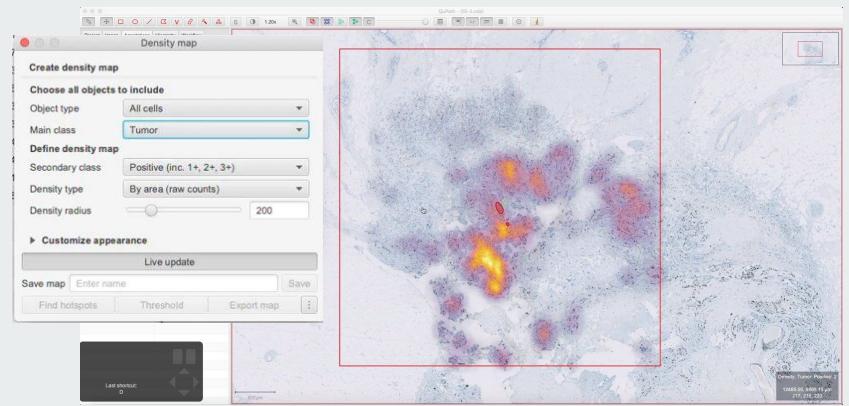
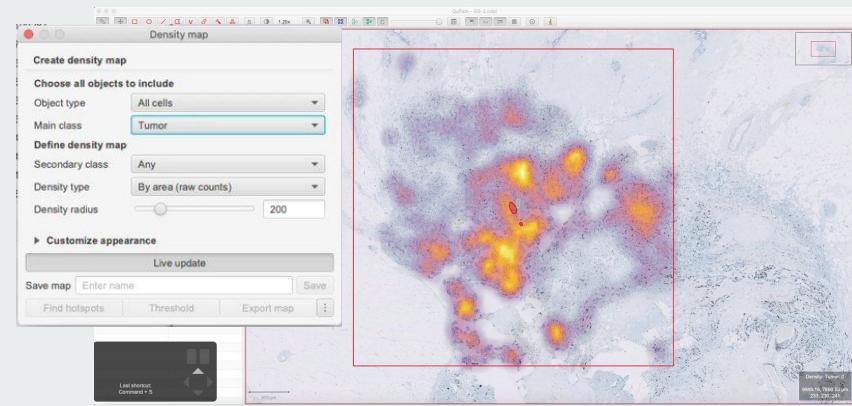
1 Starting Point: Cell Detection + Cell Classification



2 Create Density Map



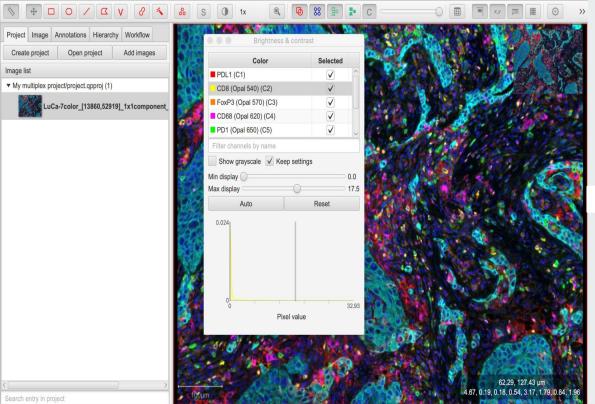
3 Customization: Choosing Objects to Include





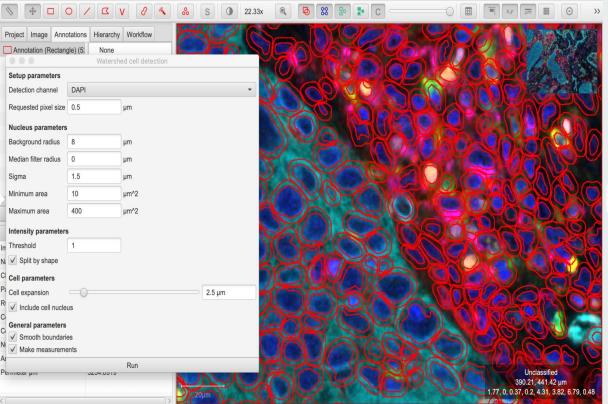
Multiplexed Analysis

1 Set Up the Channel Names



Adjusting the channel names in the Brightness & Contrast dialog

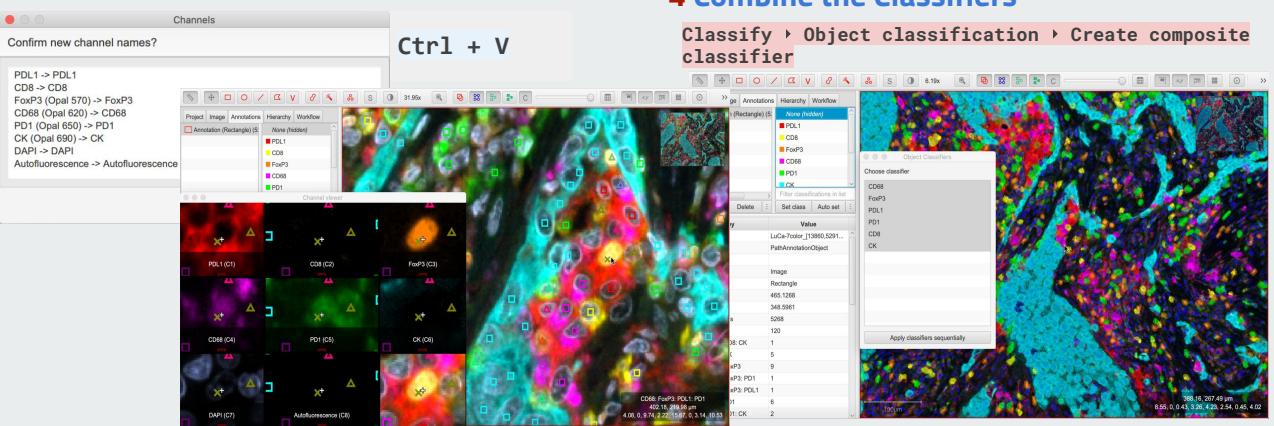
2 Detect & Measure Cells



Example of cell detection in the luca-7color image

4 Combine the Classifiers

Classify > Object classification > Create composite classifier

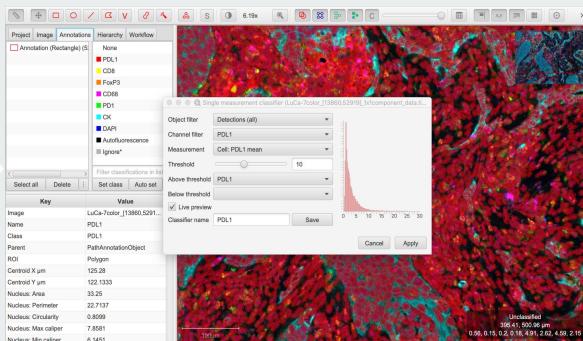


Ctrl + V

Classify > Object classification > Train object classifier

3 Create Classifier for each Marker

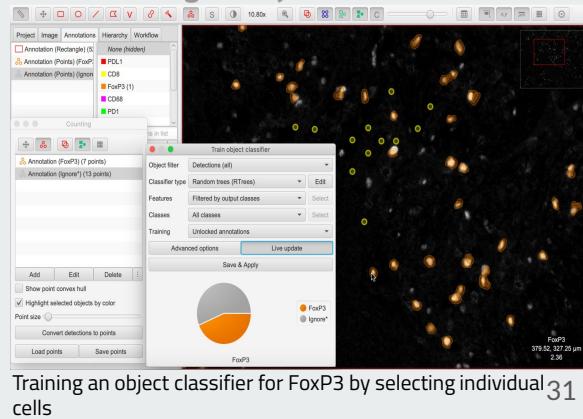
- Simple Thresholding



Creating a single measurement classifier for PDL1

Classify > Object classification > Create single measurement classifier

● Training an Object Classifier



Training an object classifier for FoxP3 by selecting individual cells



Exporting Measurements

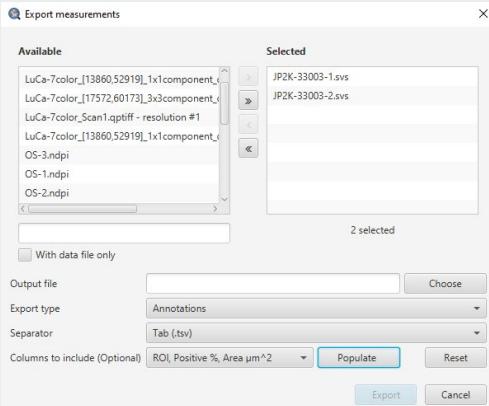
1 Via the Measurement Table

Name	Class	Parent	ROI	Centroid X μ m	Centroid Y μ m	Nucleus: Area
Stroma	Stroma	Tissue	Polygon	2433.1	9540.7	17.5
Stroma	Stroma	Tissue	Polygon	2480.3	9540.3	15.25
Stroma	Stroma	Tissue	Polygon	2503.1	9540.3	18.75
Stroma	Stroma	Tissue	Polygon	2535.3	9540.5	10.5
Stroma	Stroma	Tissue	Polygon	2569.1	9539.2	13.5
Stroma	Stroma	Tissue	Polygon	2630.7	9541.8	44.25
Stroma	Stroma	Tissue	Polygon	2653.1	9541.5	21.75
Stroma	Stroma	Tissue	Polygon	2670.7	9539.8	23
Stroma	Stroma	Tissue	Polygon	2727.7	9542	12.5
Stroma	Stroma	Tissue	Polygon	2750.3	9540.1	16.25
Stroma	Stroma	Tissue	Polygon	2766.8	9541	11
Stroma	Stroma	Tissue	Polygon	2902.1	9539.6	10.75
Stroma	Stroma	Tissue	Polygon	2914.4	9540.1	10
Stroma	Stroma	Tissue	Polygon	2397.3	9541.4	13
Stroma	Stroma	Tissue	Polygon	2417.9	9539.5	14.25
Stroma	Stroma	Tissue	Polygon	2542.1	9542.4	11.25
Stroma	Stroma	Tissue	Polygon	2549.1	9541.2	19.75

Saving cell detection measurements via the measurement table



2 Via the Measurement Exporter



The Measurement Exporter

Measure > Export measurements

3 Via Scripting

```
import qupath.lib.gui.tools.MeasurementExporter
import qupath.lib.objects.PathCellObject

// Get the list of all images in the current project
def project = getProject()
def imagesToExport = project.getImageList()

// Separate each measurement value in the output file with a tab ("t")
def separator = "\t"

// Choose the columns that will be included in the export
// Note: if 'columnsToInclude' is empty, all columns will be included
def columnsToInclude = new String[]{"Name", "Class", "Nucleus: Area"}

// Choose the type of objects that the export will process
// Other possibilities include:
// 1. CellObject
// 2. PathDetectionObject
// 3. PathRootObject
// Note: import statements should then be modified accordingly
def exporterType = PathCellObject.class

// Choose your "full" output path
def outputPath = "/measurements.tsv"
def outputFile = new File(outputPath)

// Create the measurementExporter and start the export
def exporter = new MeasurementExporter()
    .imageList(imagesToExport) // Images from which measurements will be exported
    .separator(separator) // Character that separates values
    .includeOnlyByColumn(true) // Columns are case-sensitive
    .exportType(exporterType) // Type of objects to export
    .filter(obj -> obj.getPathClass() == getPathClass("Tumor")) // Keep only objects with class 'Tumor'
    .exportMeasurements(outputFile) // Start the export process

print "Done!"
```



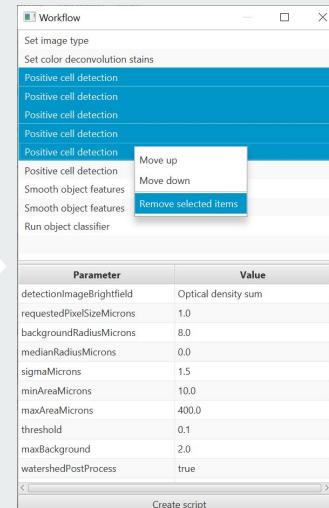
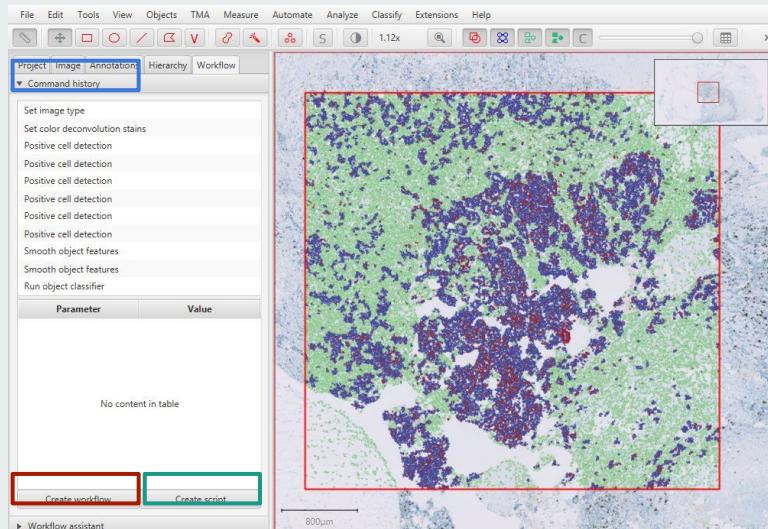
Scripting

- Workflows
- Workflows to Script



Workflows to Script & the Command History

- QuPath uses **Workflows** to represent **sequences of steps** that have been applied to an **image**.
- **Command history** is a **record** that reflects all (or most...) of what has been done to the currently open image.



```

1 setImageType("BRIGHTFIELD_H_DAB")
2 setColorDeconvolutionStains("H-DAB updated", "Stain 1": "Hematoxylin", "Values 1": "0.82")
3 runPlugin('qupath.image.detect.cells.PositiveCellDetection', 'detectionImageBrightfield': "hematoxylin")
4 runPlugin('qupath.image.detect.cells.PositiveCellDetection', 'detectionImageBrightfield': "optical density sum")
5 runPlugin('qupath.image.detect.cells.PositiveCellDetection', 'detectionImageBrightfield': "optical density sum")
6 runPlugin('qupath.image.detect.cells.PositiveCellDetection', 'detectionImageBrightfield': "optical density sum")
7 runPlugin('qupath.image.detect.cells.PositiveCellDetection', 'detectionImageBrightfield': "optical density sum")
8 runPlugin('qupath.lib.plugins.objects.SmoothFeaturesPlugin', 'fwhmMicrons': 25.0, "smoothWithinMicrons": 50.0, "smoothWithMedian": true)
9 runClassifier('C:\Users\epman\Desktop\My classifier.qpclassifier')
10 runScript("C:\Users\epman\Desktop\My classifier.qpclassifier")
11 setColorDeconvolutionStains("Name": "H-DAB updated", "Stain 1": "Hematoxylin", "Values 1": "0.82")
12

```

A script generated from a **Create Script**

Running a script for a single image

Run ➔ Run

Run ➔ Run selected

Running a script for multiple images

Run ➔ Run for project

Run ➔ Run for project
(without save)



Extensions

- Finding Extensions
- Biolmage Model Zoo
- StarDist
- Cellpose



Finding Extensions

- They make it possible to **update** new features **without needing to update** the whole software.
- They enable **other developers to add functionality to QuPath** without needing to change the core software.

Extensions > Manage extensions

Installing extensions

Extension manager

▼ Download extension from GitHub

Owner: qupath Repository: qupath-extension-djl

Installed extensions

Bio-Formats extension (Bio-Formats 7.0.1)	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
Installs support to read and write images using Bio-Formats reading and writing options using Bio-Formats (https://www.openmicroscopy.org/bio-formats/)	Core extension (part of QuPath)	
ImageJ extension	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
QuPath commands that enable integration with ImageJ - https://imagej.nih.gov/ij/	Core extension (part of QuPath)	
OpenSlide extension	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
Provides support for OpenSlide images. This includes specifying a path to an OpenSlide installation.	Core extension (part of QuPath)	
Processing & classification extension	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
Core processing & classification commands	Core extension (part of QuPath)	
Rich script editor extension	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
Adds a more attractive script editor with syntax highlighting, making use of RichTextFX - https://github.com/TomasMikula/RichTextFX	Core extension (part of QuPath)	
SVG export extension	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
Export snapshots and images in SVG format	Core extension (part of QuPath)	

Open extension directory

The extension manager window + Installation of Deep Java Library extension

Removing extensions

Extension manager

▼ Download extension from GitHub

Owner: Repository:

Installed extensions

Bio-Formats extension (Bio-Formats 7.0.1)	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
Installs support to read and write images using Bio-Formats reading and writing options using Bio-Formats (https://www.openmicroscopy.org/bio-formats/)	Core extension (part of QuPath)	
ImageJ extension	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
QuPath commands that enable integration with ImageJ - https://imagej.nih.gov/ij/	Core extension (part of QuPath)	
OpenSlide extension	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
Provides support for OpenSlide images. This includes specifying a path to an OpenSlide installation.	Core extension (part of QuPath)	
Processing & classification extension	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
Core processing & classification commands	Core extension (part of QuPath)	
Rich script editor extension	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
Adds a more attractive script editor with syntax highlighting, making use of RichTextFX - https://github.com/TomasMikula/RichTextFX	Core extension (part of QuPath)	
SVG export extension	v0.5.0-SNAPSHOT	[remove] [refresh] [update]
Export snapshots and images in SVG format	Core extension (part of QuPath)	
Deep Java Library extension	v0.2.0	[remove] [refresh] [update]
Add Deep Java Library support to QuPath	User extension (GitHub enabled)	

Open extension directory

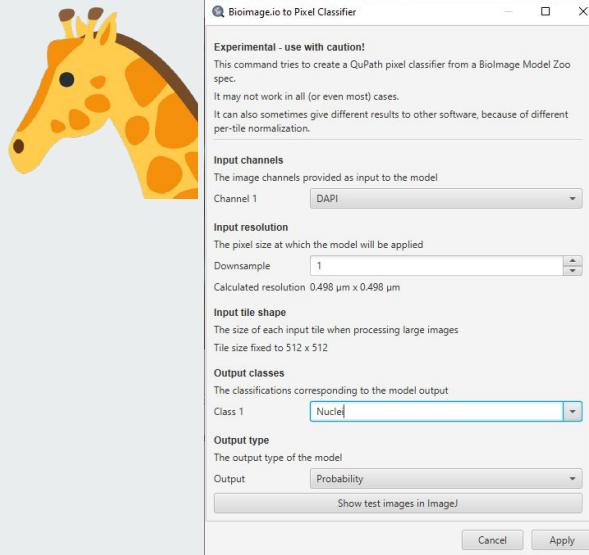
Viewing installed extensions in the extension manager, with the option to remove or update them, or to visit the extension's GitHub repository



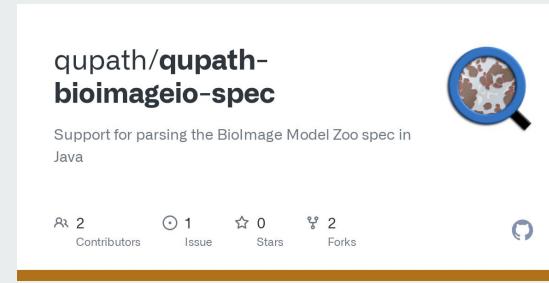
Biolmage Model Zoo

- It is a **recent** and still **experimental** initiative to make **DL models for bioimage analysis** more **accessible** and **transferable** between software.
- It provides a **place** to find and download **DL models** relevant for bioimages
- It **standardizes** the crucial **metadata** that accompanies the models

Extensions > Bioimage Model Zoo > Create pixel classifier (BIOimage Model Zoo)



Dialog showing pixel classifier options for a model zoo model



After pressing this button, QuPath will open ImageJ and show the input test image, the target prediction, the actual prediction, and the difference between them.

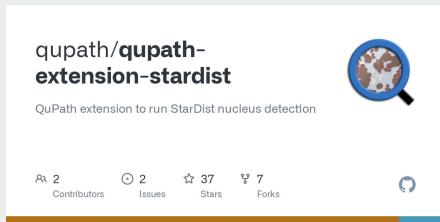


StarDist

- StarDist is a fantastic, DL-based method of **2D (in QuPath) and 3D nucleus detection** from Martin Weigert and Uwe Schmidt.
- It exists as a **Python library** and **Fiji plugin**.



Getting the StarDist Extension + Pretrained Models

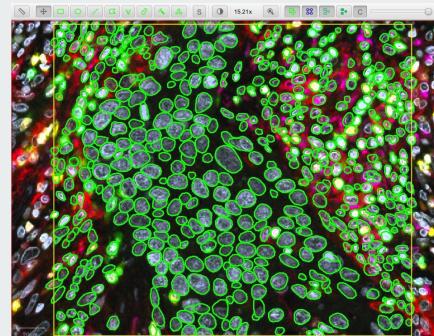
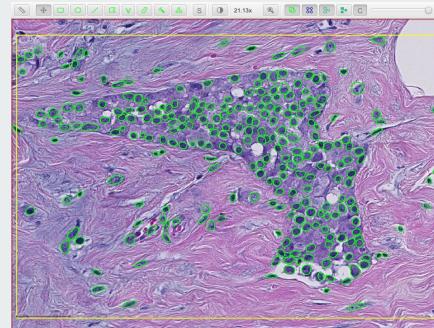


```
import qupath.ext.starDist.StarDist2D
// Specify the model file (you will need to change this!)
def pathModel = 'path/to/dab2018_heavy_augment.pb'

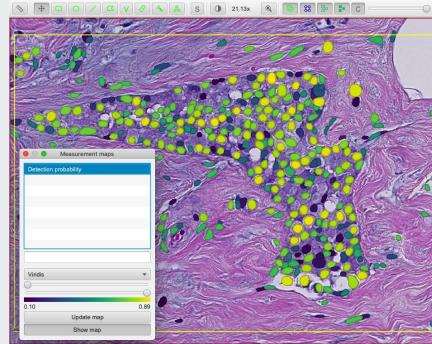
def starDist = StarDist2D.builder(pathModel)
    .threshold(0.5) // Probability (detection) threshold
    .channels('DAPI') // Select detection channel
    .normalizePercentiles(1, 99) // Percentile normalization
    .pixelsPerTissue(1000) // Approximate tissue size in pixels
    .cellExpansion(5.0) // Approximate cells based upon nucleus size
    .cellConstrainScale(1.5) // Constrain cell expansion using nucleus size
    .measureShape() // Add shape measurements
    .measureExtents() // Add cell measurements (in all compartments)
    .includeProbability(true) // Add probability as a measurement (enables later filtering)
    .build()

// Run detection for the selected objects
def imageData = getCurrentImageData()
def pathObjects = getSelectedPathObjects()
if (pathObjects.isEmpty()) {
    dialog.showErrorMessageBox("StarDist", "Please select a parent object!")
    return
}
starDist.detectObjects(imageData, pathObjects)
println 'Done!'
```

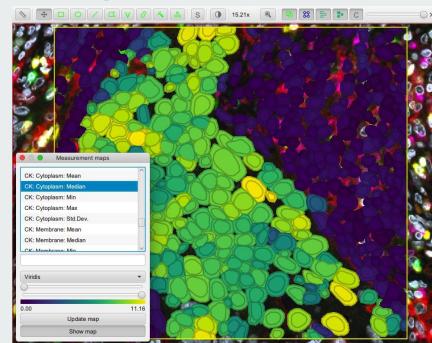
Detecting Nuclei Extensions > StarDist



Viewing probabilities



Cell expansion & Measurements





- Cellpose is a DL-based method of **2D (in QuPath)** and **3D cell detection**.
 - It exists as a **Python** library and **Fiji plugin**.



Getting the Cellpose Extension

BIOP/qupath-extension-cellpose

an extension that wraps a Cellpose environment such that WSI can be analyzed using Cellpose through QuPath

A 3 Contributors ⚡ 3 Issues ⭐ 67 Stars 📈 11 Forks

```
import qupath.ext.biop.cellpose.Cellpose2D

// Specify the model name (cyto, nuc, cyto2 or a path to your custom model)
def pathModel = "cyto2"

def Cellpose2D = Cellpose2D.builder(pathModel)
    .probabilityThreshold(0.5) // Probability threshold for detection
    .channels("Red", "Green", "Blue") // Select detection channel(s)
    .normalizePercentile(1, 99) // Percentile normalization for detection
    .pixelSize(1.8) // Average diameter of objects in pixels
    .diameter(38) // Approximate diameter of cells based upon nucleus size
    .cellConstrainScale(1.5) // Controls cell extraction using nucleus size
    .measureShape() // Add shape measurements
    .measureIntensity() // Add cell measurements (in all compartment)
    .build()

// Run detection for the selected objects
def imageData = getCurrentImageData()
def pathObjects = getSelectedObjects()
if (pathObjects.isEmpty()) {
    Dialogs.showErrorMessage("Cellpose", "Please select a parent object!")
    return
}

cellpose.detectObjects(imageData, pathObjects)
println "Done!"
```

Training/Detecting Nucleus/Cell Detection

Extensions ▶ Cellpose

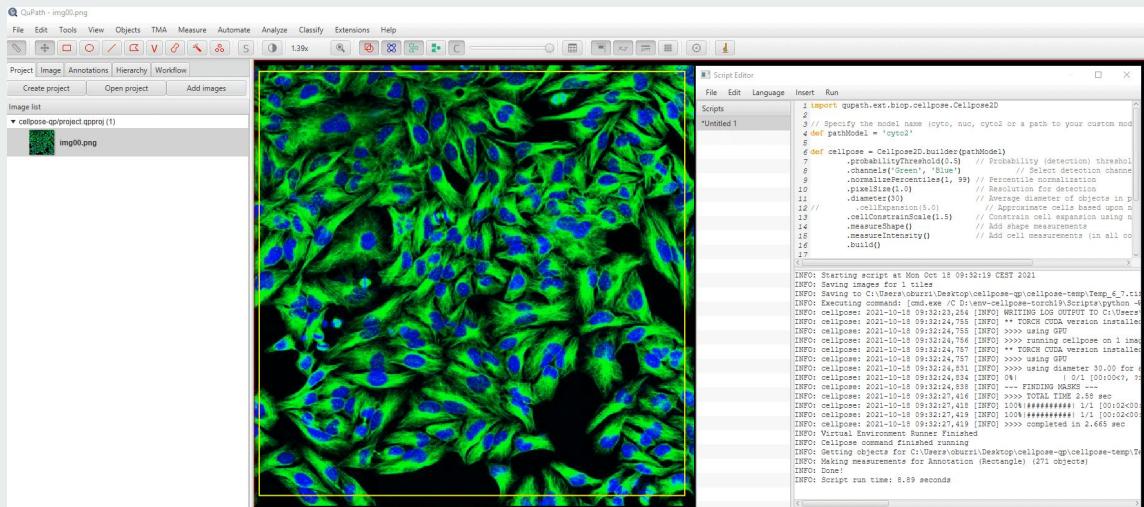




Image Analysis CMU & HCU Request Procedure

- For Fluorescence Images (CMU)
- For Brightfield Images (HCU)



BioImage Analysis Request Procedure

For Fluorescence Images (CMU)

Request CMU by Email



LD Unidad
Microscopía Confocal

microscopia.confocal@cnio.es

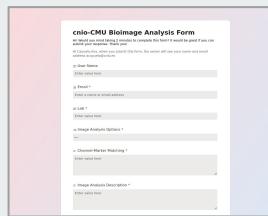
Information regarding
experimental design,
images & analysis
strategy

Define desired **Outputs**

Decide whether to use
Commercial or Open-
Source Software or
Scripts

Fill the Form on CNIO Intranet

Core Units & Facilities >
Confocal Microscopy > Links
> BioImage Analysis Form



IA Request will be added into the
Ticket System

Images in CNIO Servers



\\\sacarino.cnio.es\Temporal\
CNIO_User

\\\imgserver.cnio.es\Confocal\
IA\Projects

Deliver Pilot & Meeting for Review



Create a **Pipeline** for your
questions

Effort to develop **Generic**
Pipelines

Workstation Booking

Core Units & Facilities
► Cofocal Microscopy
Unit> Request/Booking >
Booking



Analysis Computers
PC-1
✓ PC-HARMONY
✓ PC-IMARIS
□ Jesus
✓ Manu
✓ PC-Rosalind
□ PC-POSH-Gascon

Verify you have **access** to the **CNIO**
Servers



Review **Results** together

Refine Pipeline, **if needed**

Locate your images to be analyzed
there!!

Recommended to use the
UNIT's Workstation!!



Biolmage Analysis Request Procedure

For Brightfield Images (HPU)

Request Ticket by CNIO Intranet

Verify Access in Bespin Server
with CNIO Credentials

Workstation Booking for Image
Analysis

Histopathology Unit

Getting Started Services / Technologies Request / Booking Price List Resources + Info

Booking Mouse / Xenograft Human Mouse / Xenograft help Human help Image Analysis Request Ticket



`\bespin.cnio.es\Stage\Image_Analysis`

Core Units & Facilities ›
Histopathology
Request/Booking ›
Image Analysis Request
Ticket

In the HPU we work with images
located in the **bespin server**.

Please, verify that you have
access to:

`\bespin\Stage\Image_Analysis.`

If you cannot access the
Image_Analysis folder request:

histopatologia@cnio.es

Core Units & Facilities ›
Histopathology Request/Booking ›
Booking

Histopathology Request

Requester name: Calle de Mira Beristain, María Requester Grp / Unit: Microscopos Confocal Requester email: MCGYB@NOVAES.CNIO.ES

SAP Service: Select service Date: _____ Amount: _____ SAP Project: Select project

Comments: _____

Submit



Workstation 1

User: HP50061\analysis_user
Password: cnio-HPU

Workstation 2

User: HPW50062\analysis_user
Password: cnio-HPU