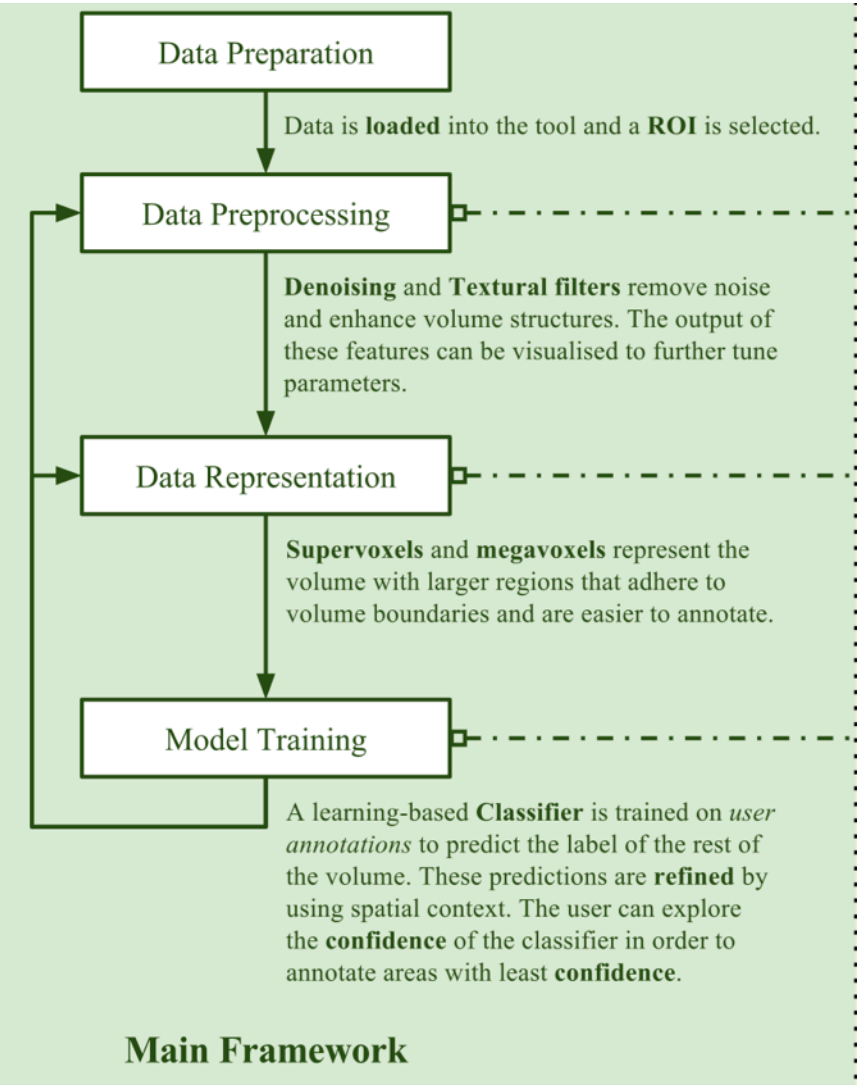
Practical Uses of SuRVoS Workbench

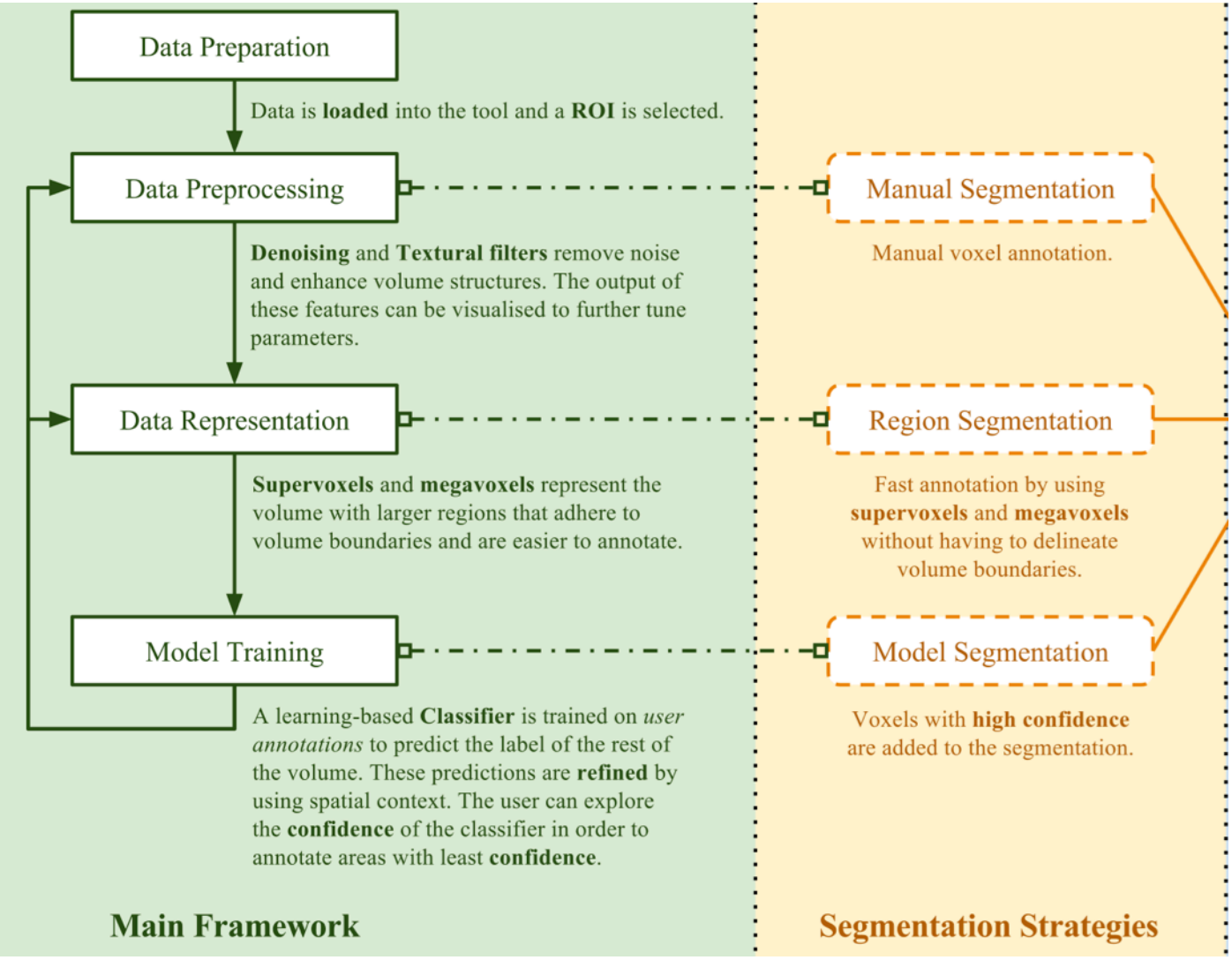
Michele C. Darrow & Matt Spink

Beamline B24
Diamond Light Source

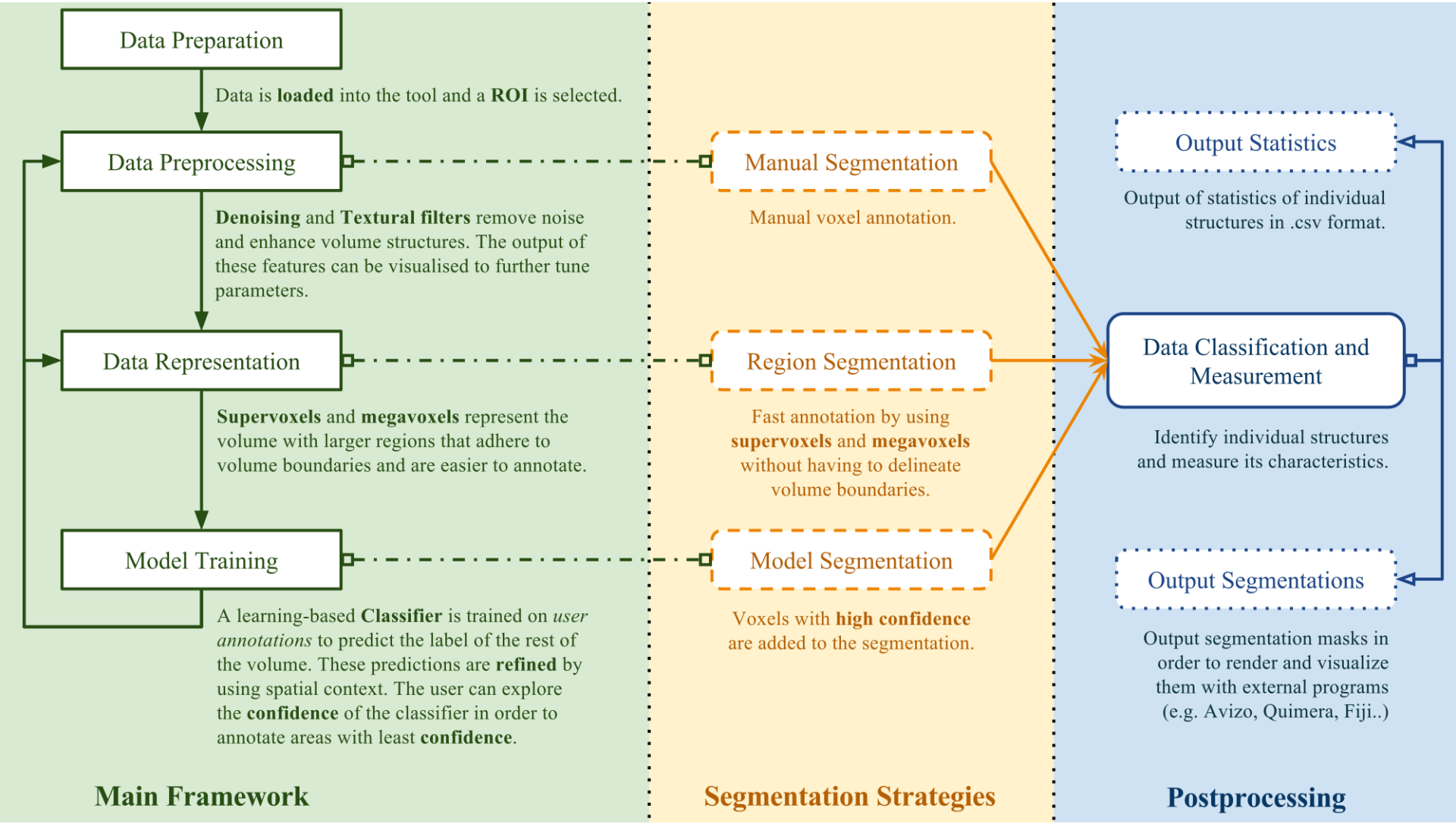
Main Framework



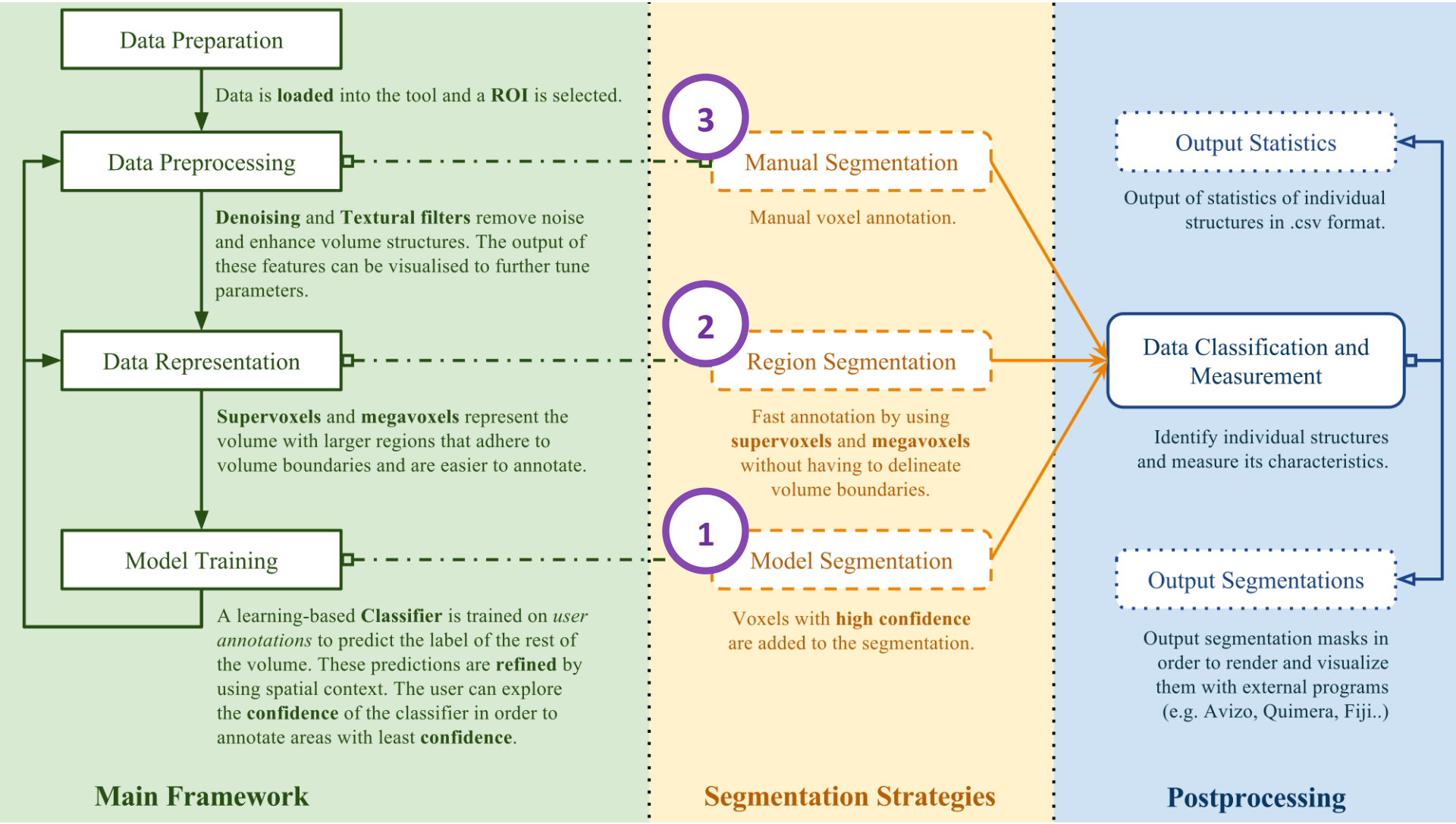
Main Framework & Segmentation Strategies



Main Framework, Segmentation Strategies & Postprocessing



The Plan for Today:



Starting SuRVoS

- In NoMachine, open a terminal (right click on the desktop and choose “open in terminal”)

- To log on to the cluster:
  - Type: module load global/cluster
  - Hit enter
  - Type: qlogin -q medium.q@@com10 -l exclusive -l gpu=1
  - Hit enter
  - When prompted, put in your password, then hit enter

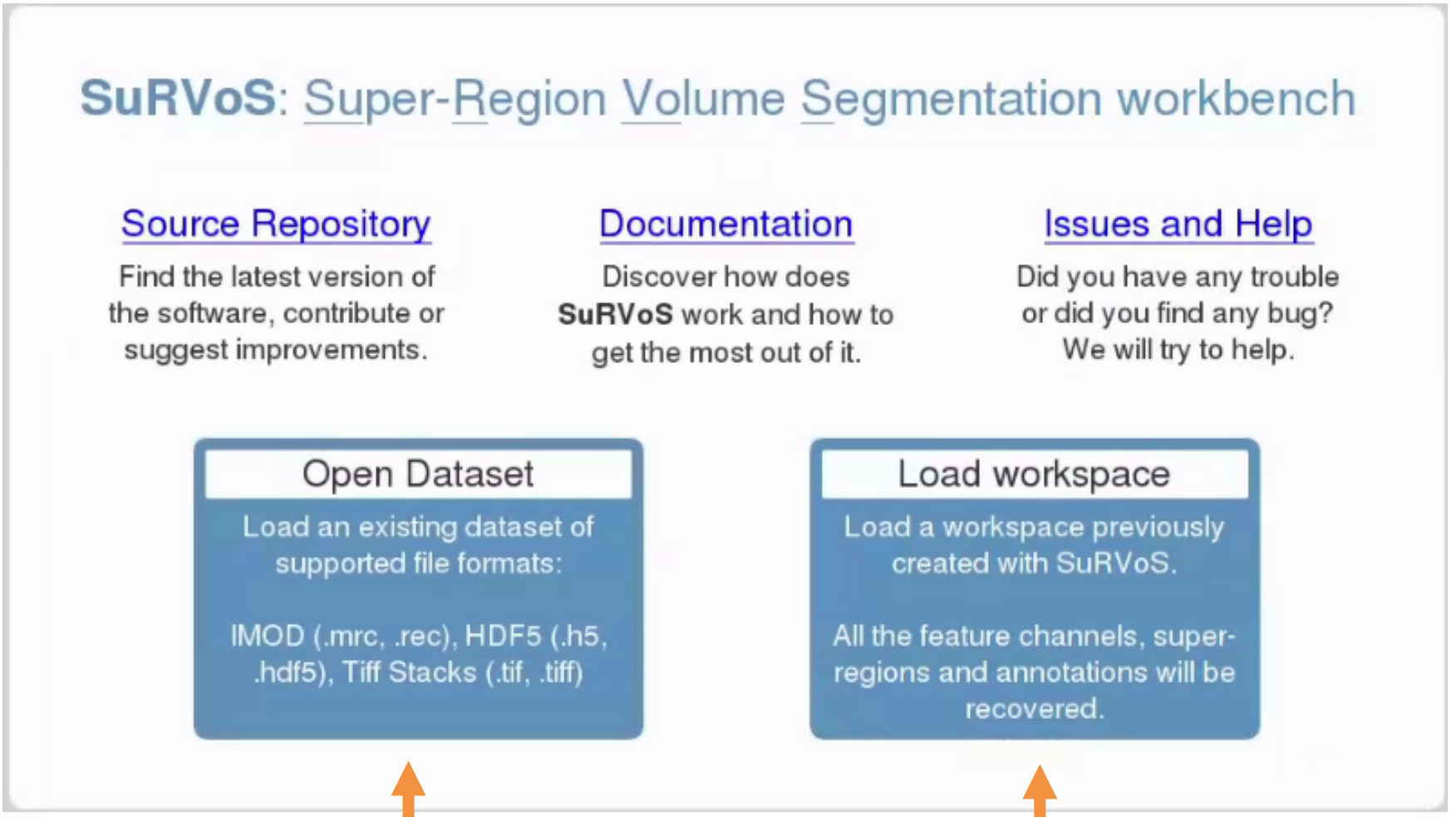
- To start SuRVoS
  - Type: module load survos
  - Hit enter
  - Type: survos
  - Hit enter

Quick Notes

We would like to use the segmentations you complete today. They will be used to test new features of SuRVoS and to assess subjectivity in segmentation. Please let us know if you do not wish your segmentations to be used in these ways. We appreciate your help to further develop SuRVoS.

We will show slides with examples for each task and detailed instructions, followed by some time to complete the task. Matt and I will circulate throughout the room to help and answer individual questions during the hands-on time. Feel free to work along with the presentation if you're confident, but rest assured that there will be time built in to complete each task we talk about.

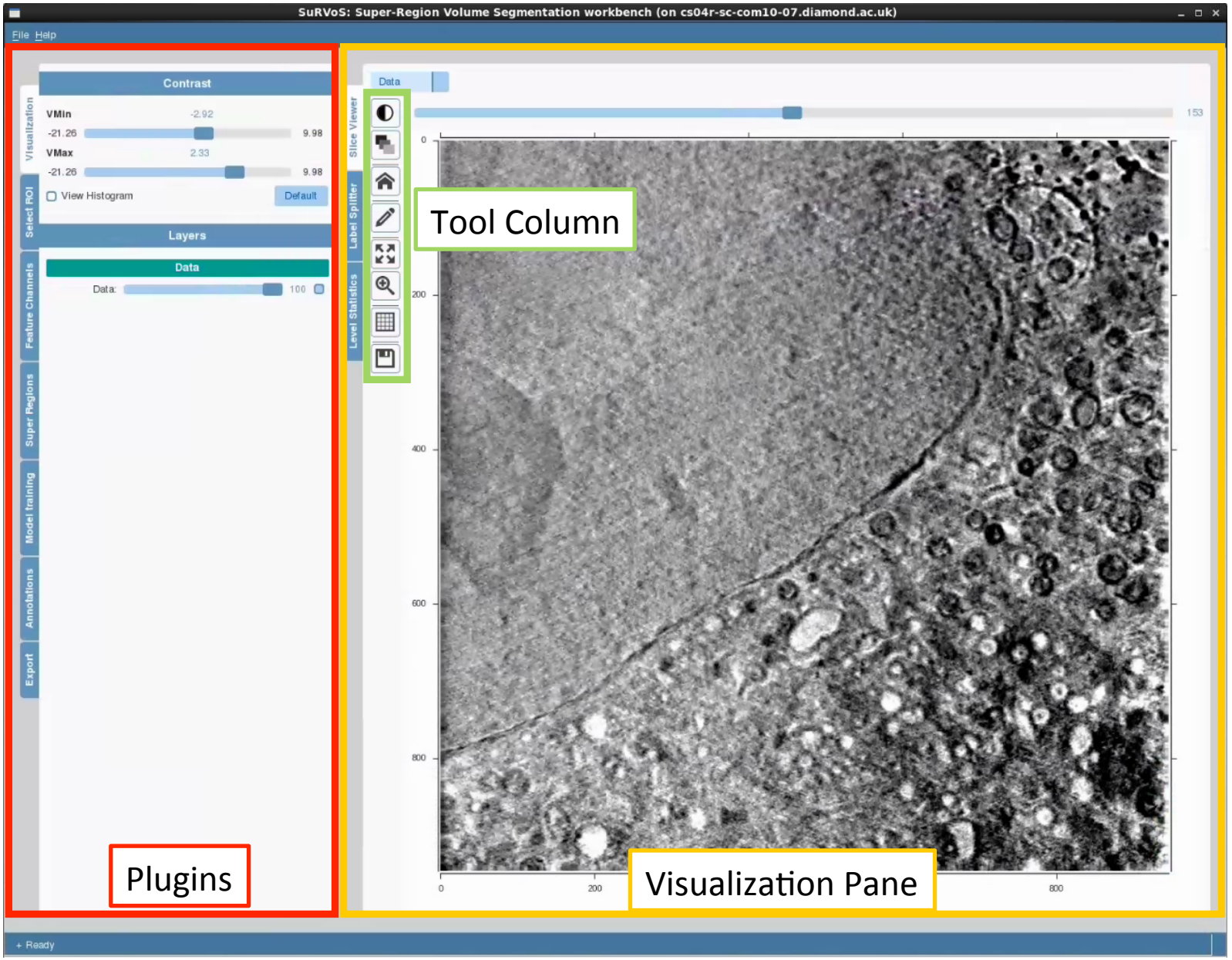
Starting or Continuing Project



Starting a New Project

Continuing a Project

Orientation to SuRVoS



Tool Column, Shortcuts

Adjust contrast

Viewer window

Home (go back to original view)

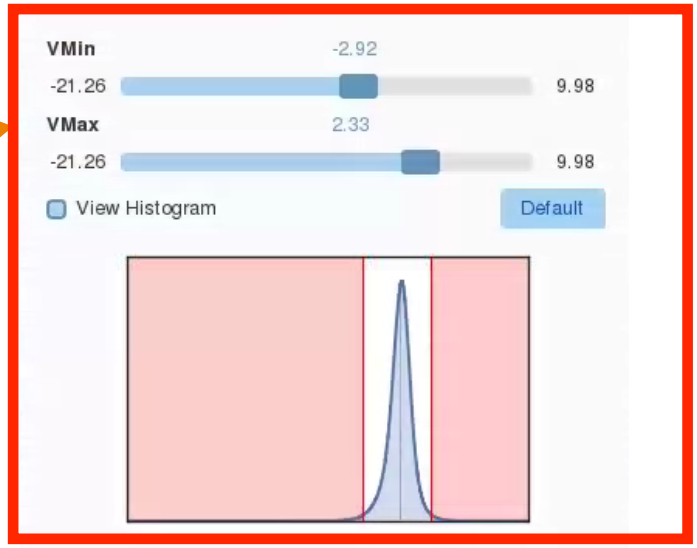
Annotate

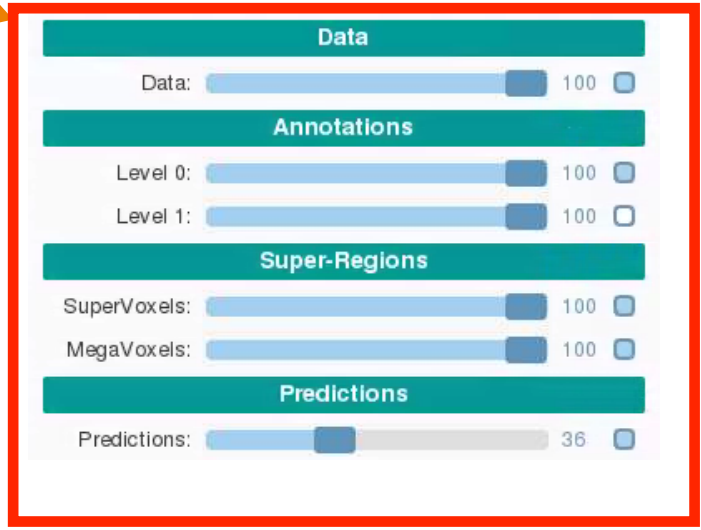
Pan Axis with the left mouse button, zoom with right

Draw a rectangle and zoom to that area

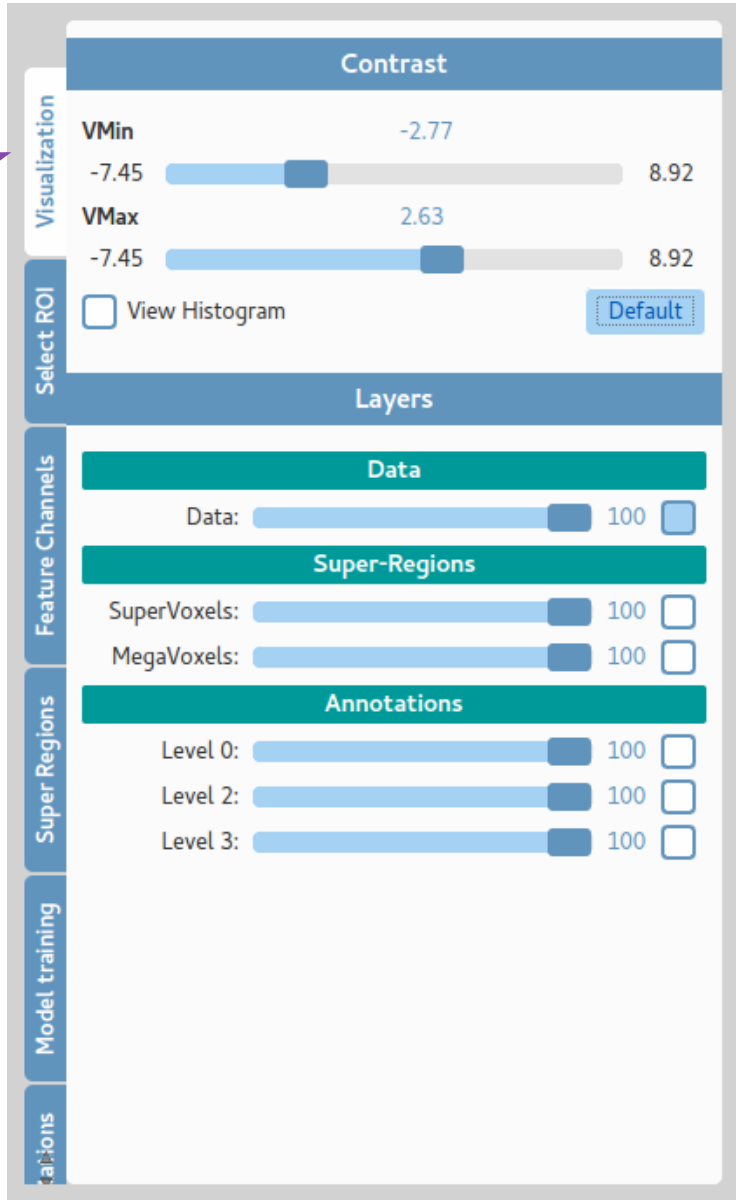
Turn on/ off a grid

Save current image





Visualization Tab in GUI



Region of Interest Tab in GUI

Note: order is Z, Y, X

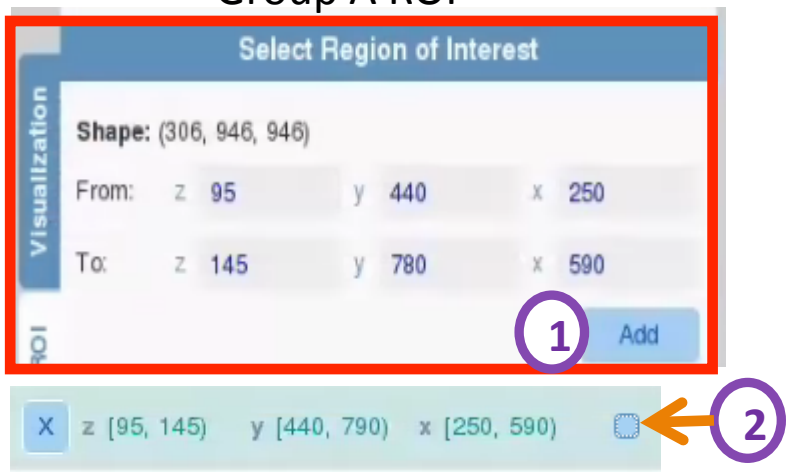
Filled in means selected

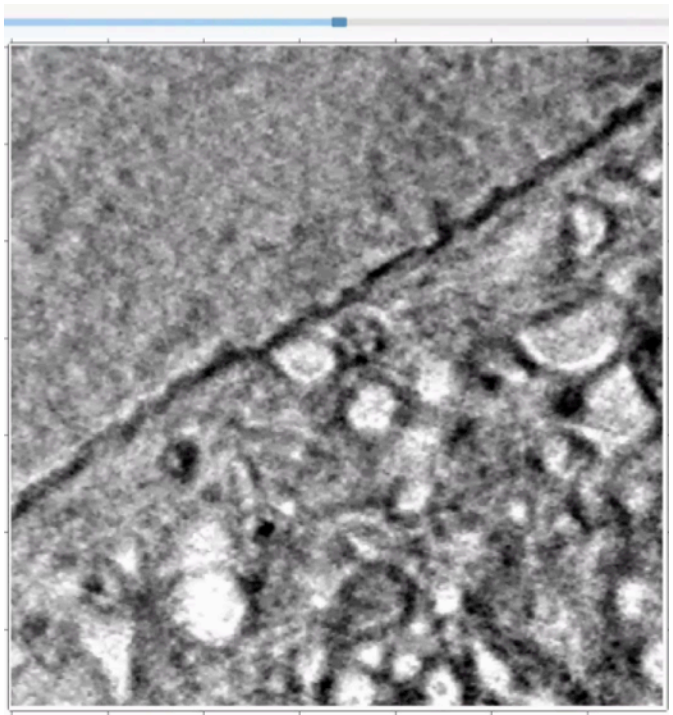
All downstream calculations will be performed on the selected region.

We suggest you start with a small, representative region of interest to optimize parameters. Then, select the whole area to be segmented and reapply appropriate parameters.

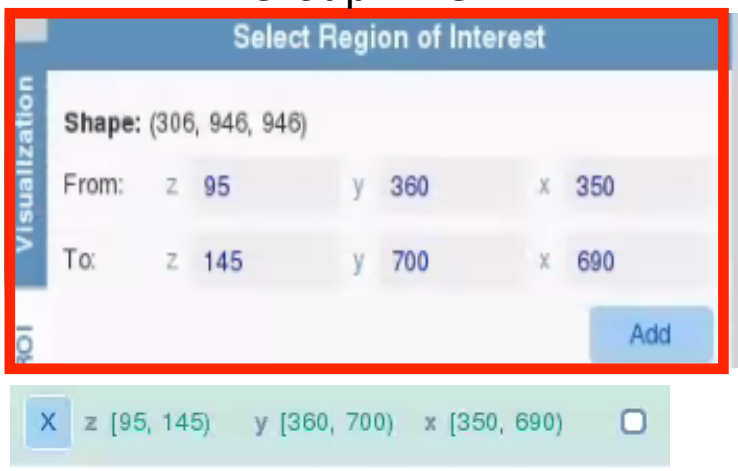
1) Insert the desired Z, Y and X values and click Add
2) Select the created Region of interest (ROI) that you have just created

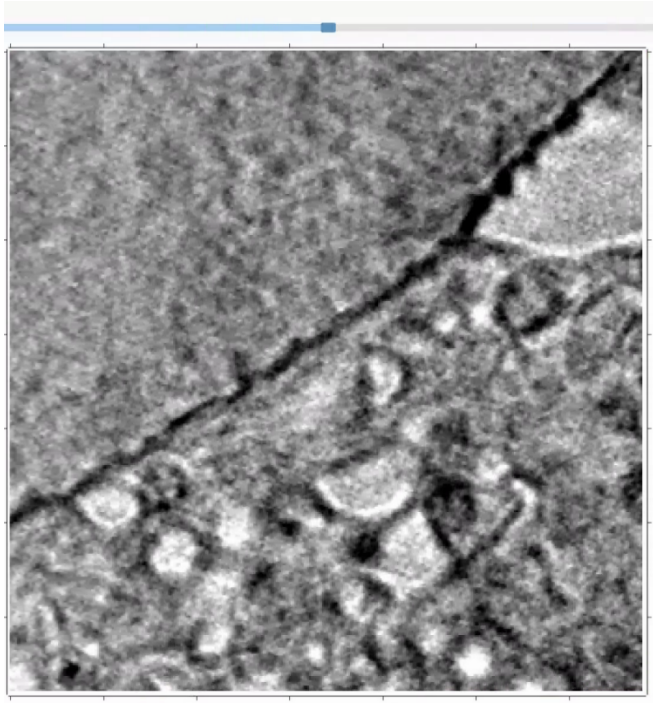
Group A ROI





Group B ROI





Time – 2 minutes

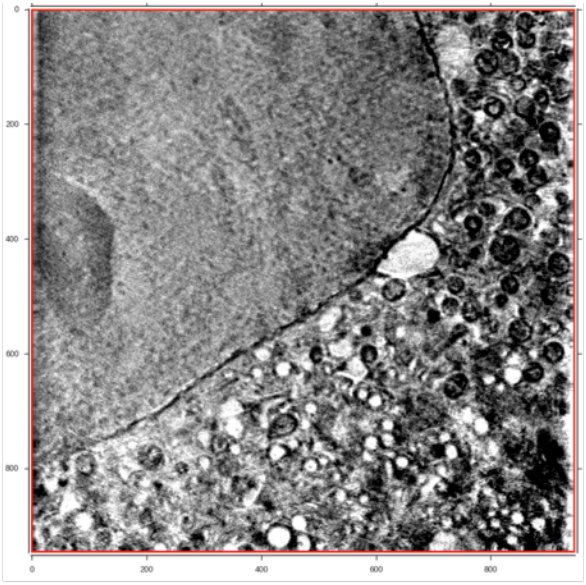
Available Filter and Feature Algorithms

- Raw
  - Threshold
  - Invert Threshold
- Denoising
  - Gaussian Filter
  - Total Variation Filter
- Local Features
  - Local Mean
  - Local Standard Deviation
  - Local Normalization
  - Local Gradient Magnitude
  - Local Gradient Orientation
- Gaussian Features
  - Gaussian Normalization
  - Gaussian Gradient Magnitude
  - Gaussian Gradient Orientation
- Blob Detection
  - Difference of Gaussian
  - Laplacian of Gaussian
  - Determinant of Hessian
  - Determinant of Structure Tensor
- Texture and Structure
  - Hessian Eigenvalues
  - Structure Tensor Eigenvalues
  - Gabor Filter
- Robust Features
  - (SI) Gaussian
  - Derivative Rotation Invariant
  - (SI) Difference of Gaussians
  - (SI) Laplacian of Gaussian
  - (SI) Determinant of hessian
  - (SI) Frangi Filter
- Activation Layers
  - Maximum Response
  - Rectified Linear Unit

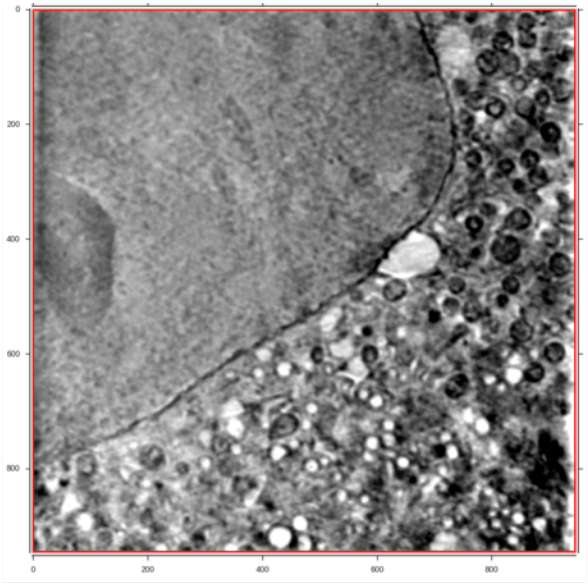
Available Filter and Feature Algorithms

- **Raw**
  Binary selection of data
- **Denoising**
  Removes noise
- **Local Features**
  Uses info from neighboring voxels based on a cube
- **Gaussian Features**
  Uses info from neighboring voxels based on a Gaussian neighborhood
- **Blob Detection**
  Uses info from neighboring voxels based on a Lapcian neighborhood
- **Texture and Structure**
  Highlights textural differences
- **Robust Features**
  Uses Gaussian, Blob and Texture and Structure Features with varying parameters and identifies the largest response
- **Activation Layers**
  In development

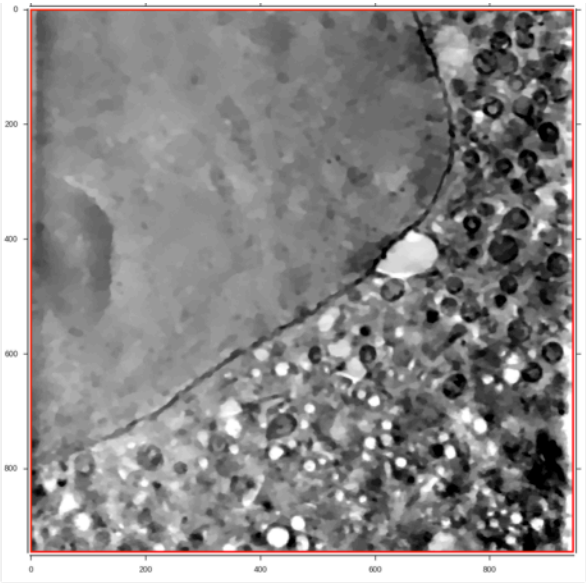
Available Filter Algorithms



(a) Raw SIRT reconstruction



(b) Gaussian Smooth



(c) Total Variation

Recommend

Sigma: 2.0 (range 1-5)

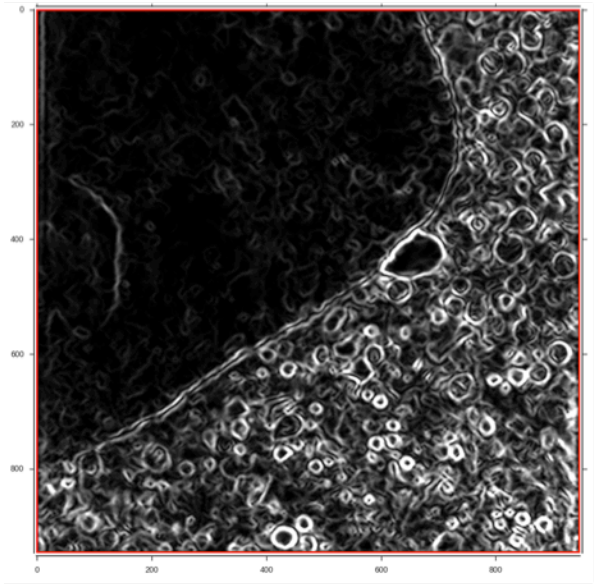
Recommend

Lambda: 10 (range 1-15)
(lower more denoising)

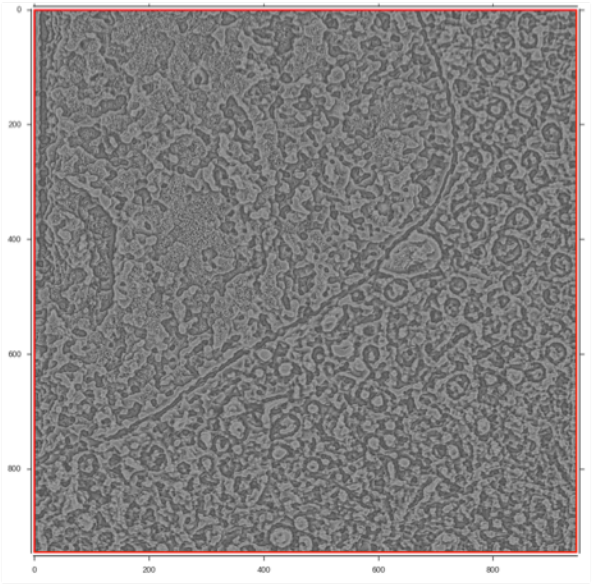
Maxiter: 100 (range 50-500)
(more iterations more denoising)

Note: Recommend to use Gaussian filter for supervoxel and megavoxel calculations, total variation for filter and feature calculations.

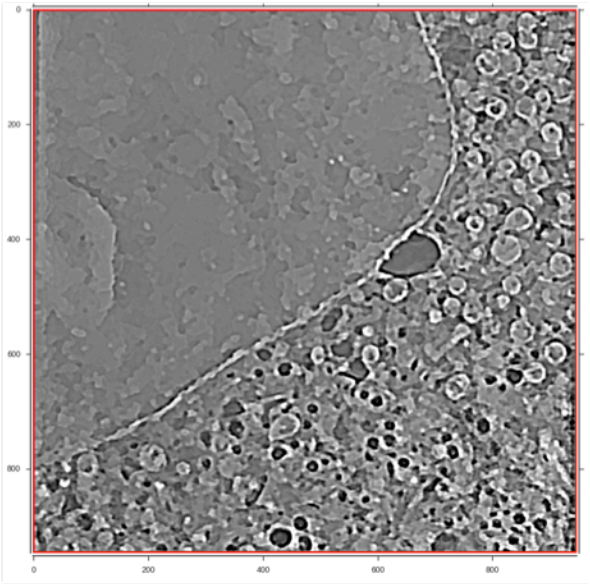
Selected Feature Filters



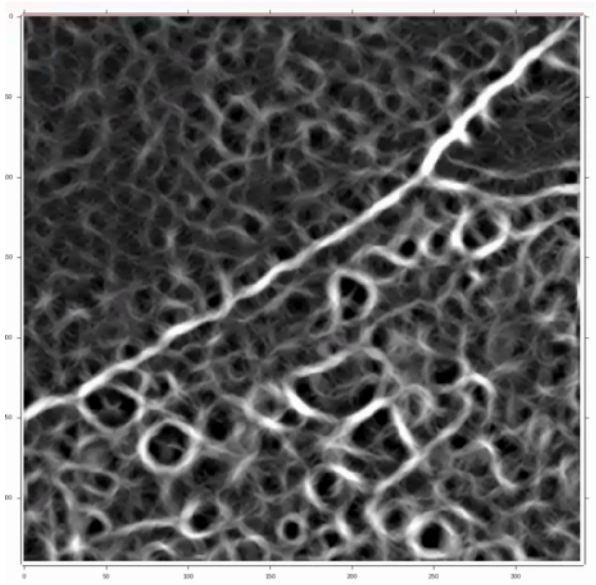
(d) Gradient Magnitude



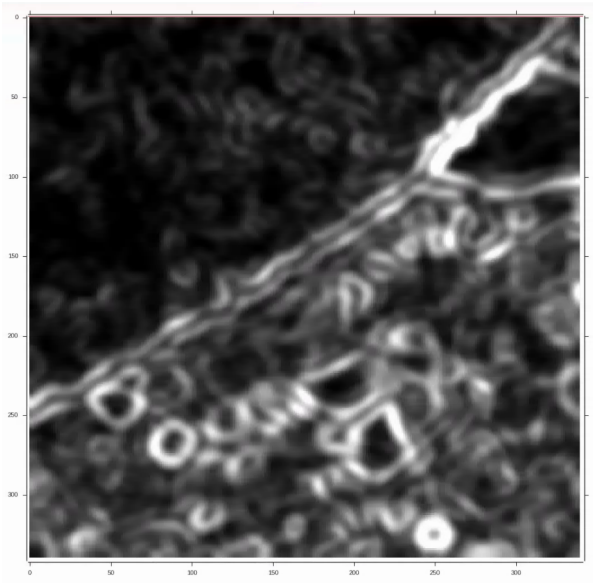
(e) Gaussian Local Normalization



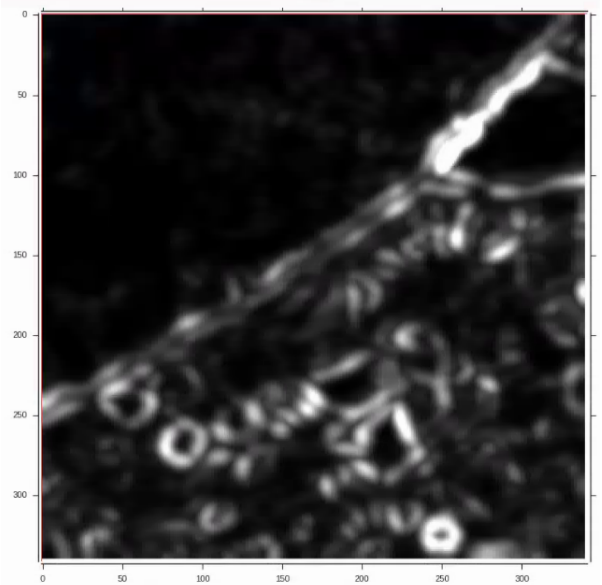
(f) Laplacian of Gaussian



(g) Hessian Eigenvalues

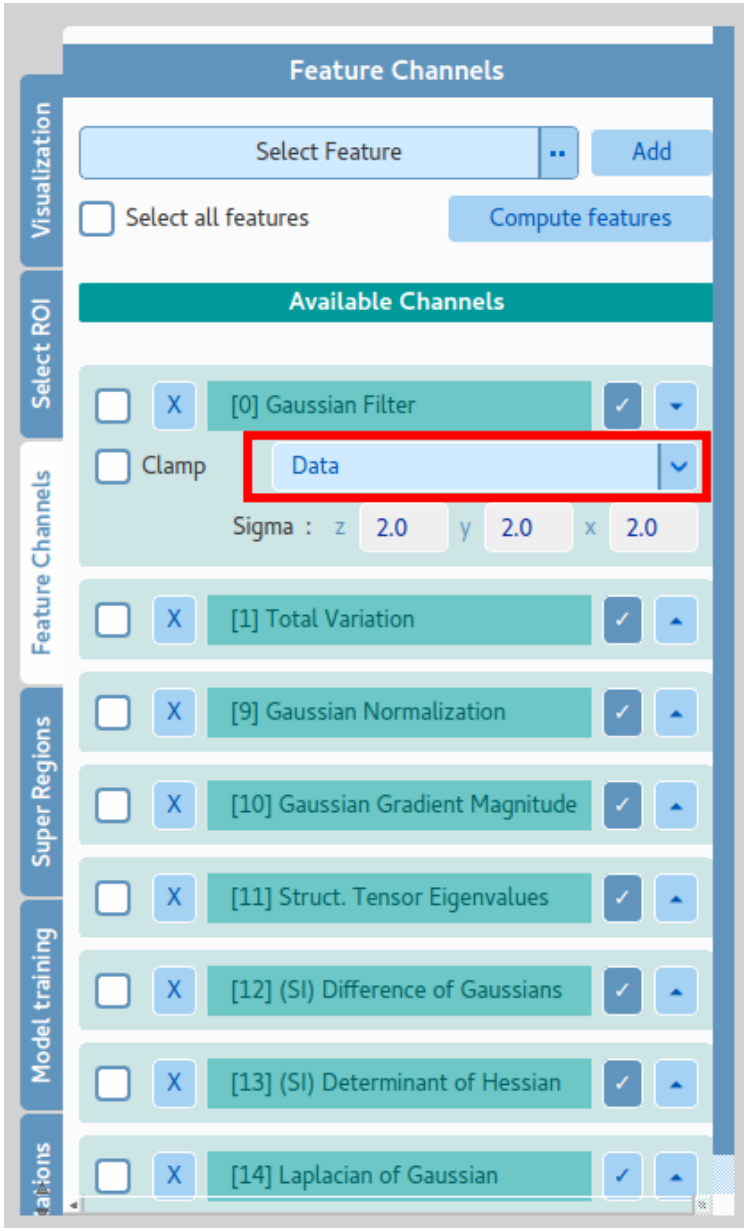


(h) Local Gradient Magnitude



(i) Structure Tensor Eigenvalues

Available Filter Algorithms



Select feature, then add
Click on name, or arrow to see details
Change input and algorithm values if needed
Click tick mark to run

To compute multiple features at once, select all (using box at top) or by selecting individual channel boxes to left of name, then click compute features (at top)

Note: For filters that use them, coordinate order is Z, Y, X

Time – 10 minutes

Introduction to Super Regions

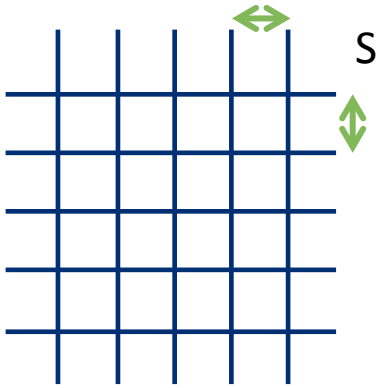
Image Supervoxels Megavoxels

- Supervoxels and megavoxels are **edge preserving**, and **three-dimensional**.
- Segmentations are **less time and labor intensive** and should be **less subjective**.

Super Region Parameters: Supervoxels

SuperVoxels:

- SP shape
- Spacing
- Compactness

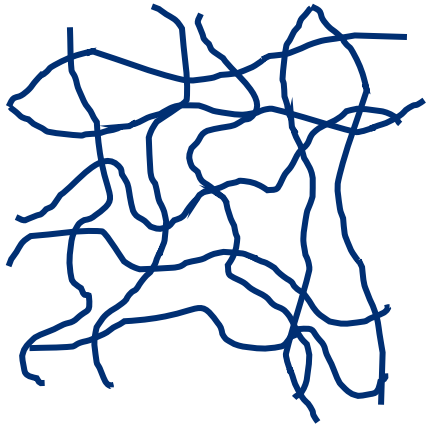


SP Shape: i.e. # of voxels to include in supervoxel grid

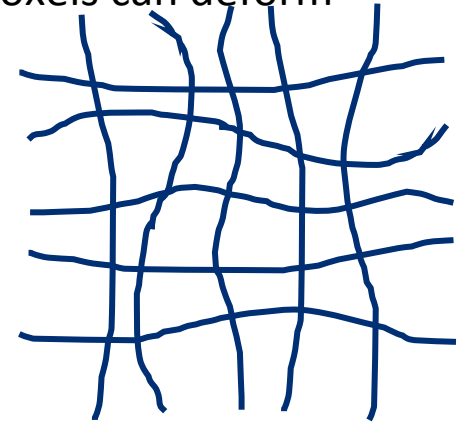
Note: order is Z, Y, X

Spacing: Importance of boundaries in each direction.

Compactness: how much supervoxels can deform



Low compactness #



High compactness #

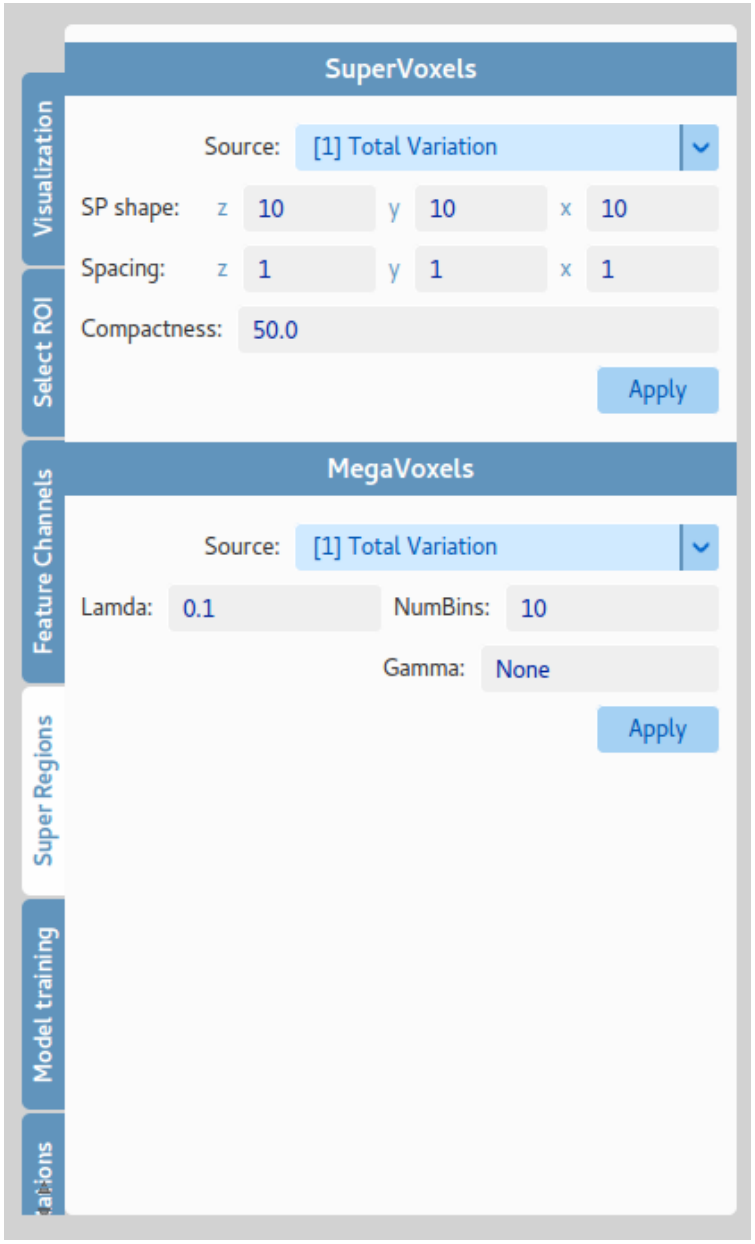
Super Region Parameters: Megavoxels

Megavoxels:

- Lambda
- NumBins
- Gamma

Gamma: controls the cost of merging two supervoxels together

Super Region Parameters: Recommendations



SuperVoxels:

- Source: Data channel
- SP shape: z=10 y=10 x=10
- Spacing: 1x1x1 (range 1-5)
- Compactness: 20 (range 10-100)

MegaVoxels:

- Source: Data channel
- Lamda: 0.1 (range 0.001-0.5)
- NumBins: 20 (range 10-50)
- Gamma: Either “None” or “auto”

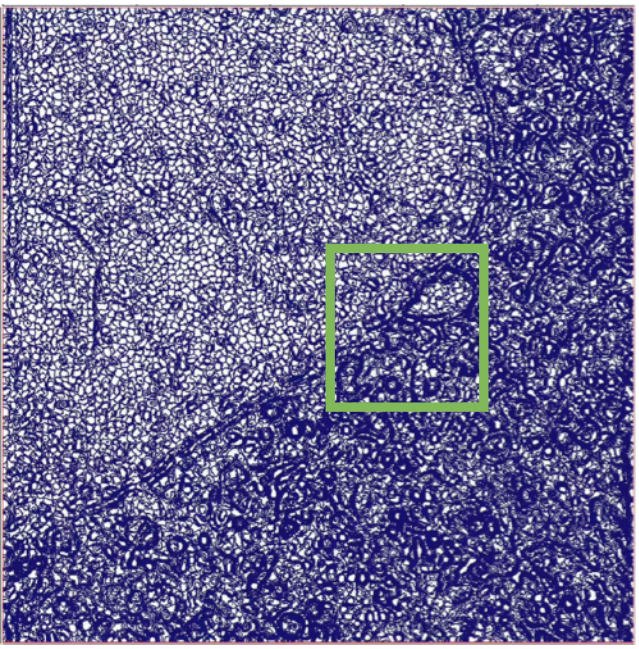
Notes:

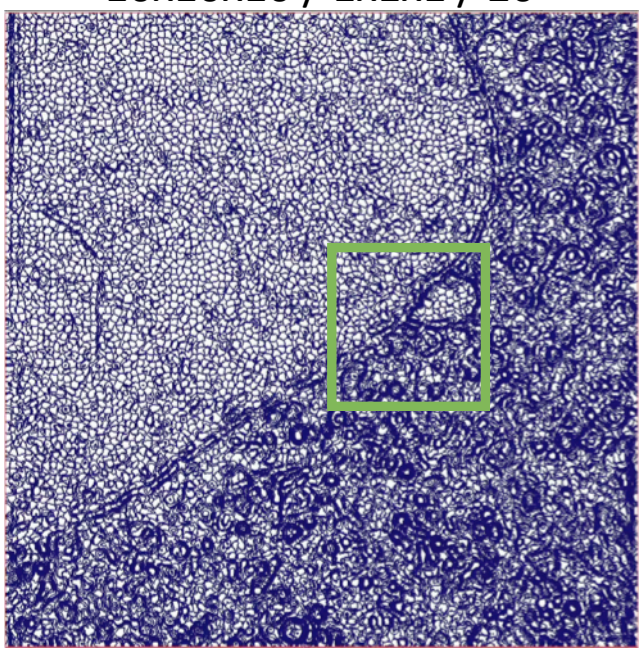
- SP shape parameters multiplied together have to equal less than 1024
- Order is Z, Y, X

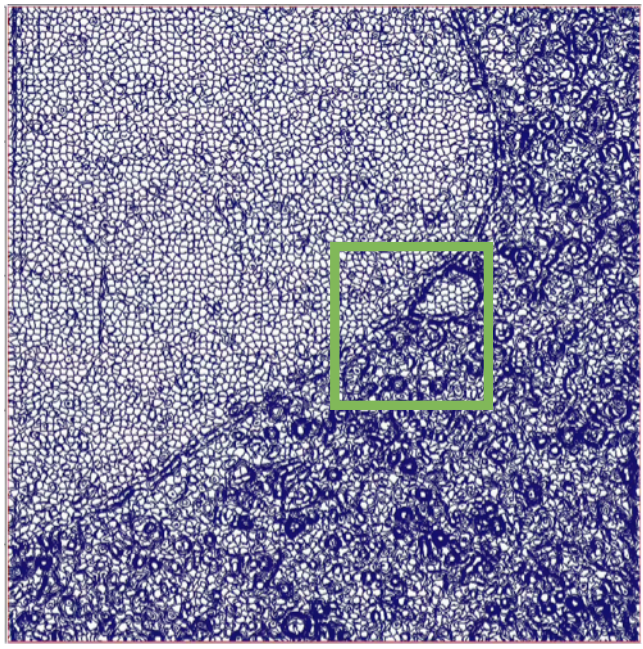
10x10x10 / 1x1x1 / **10**

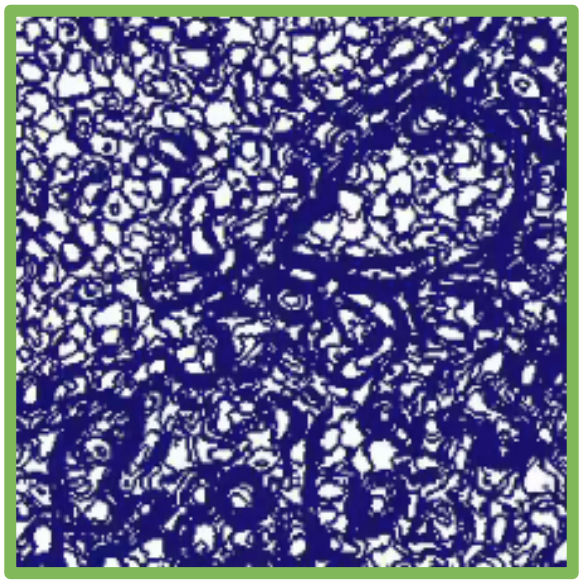
Default:
10x10x10 / 1x1x1 / 20

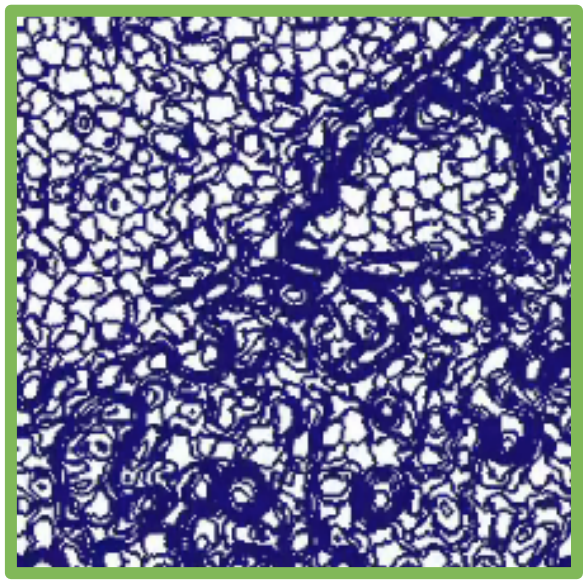
10x10x10 / 1x1x1 / **30**

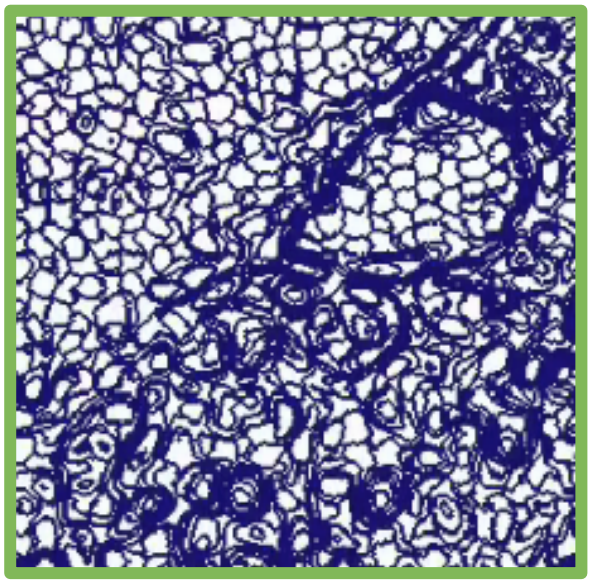








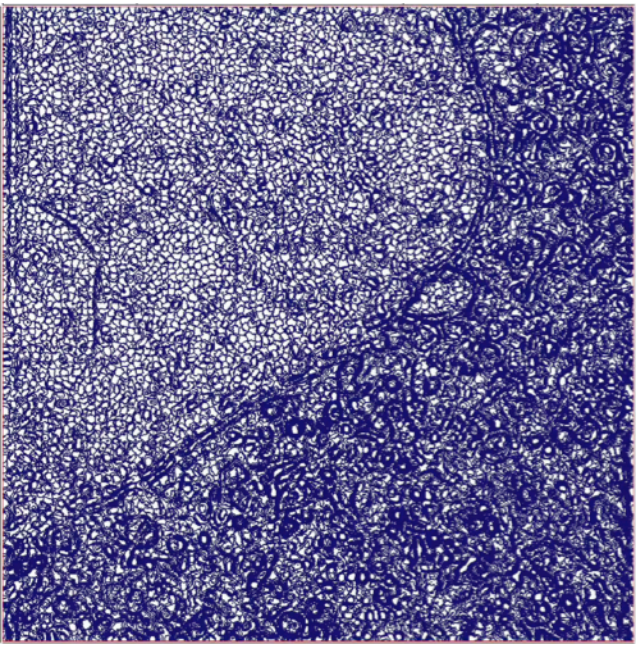
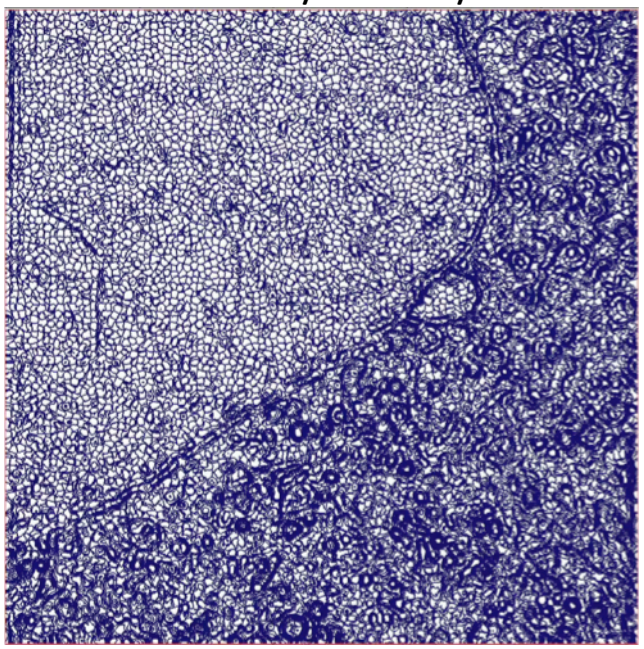
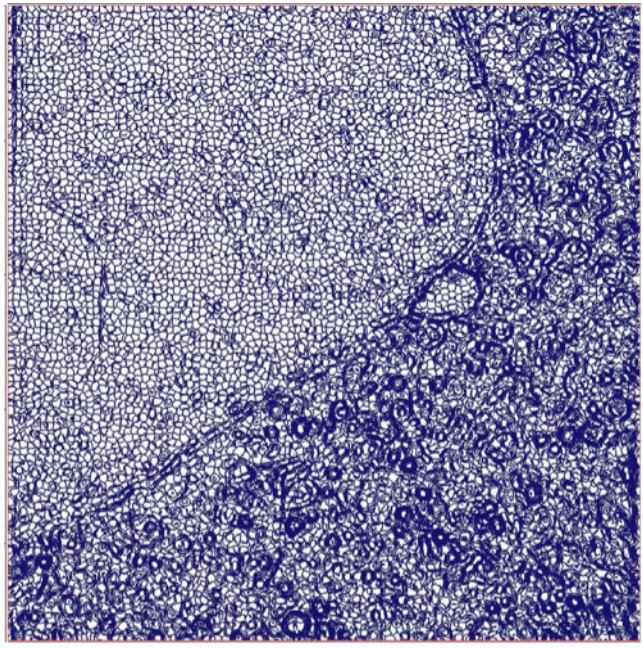
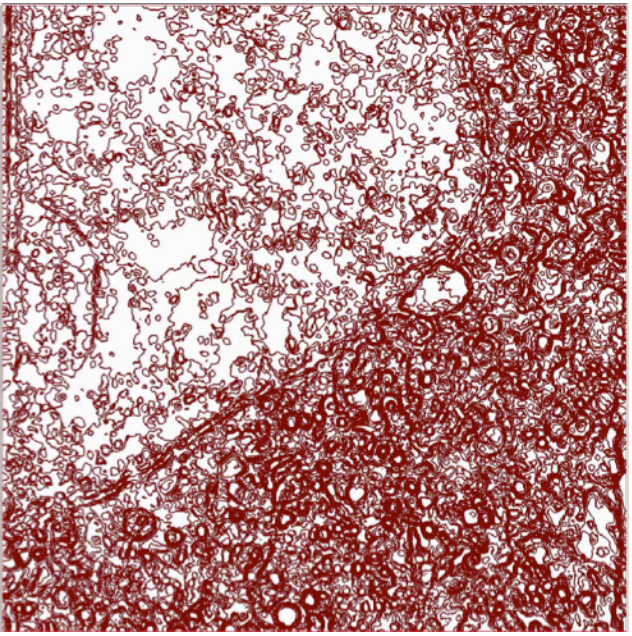
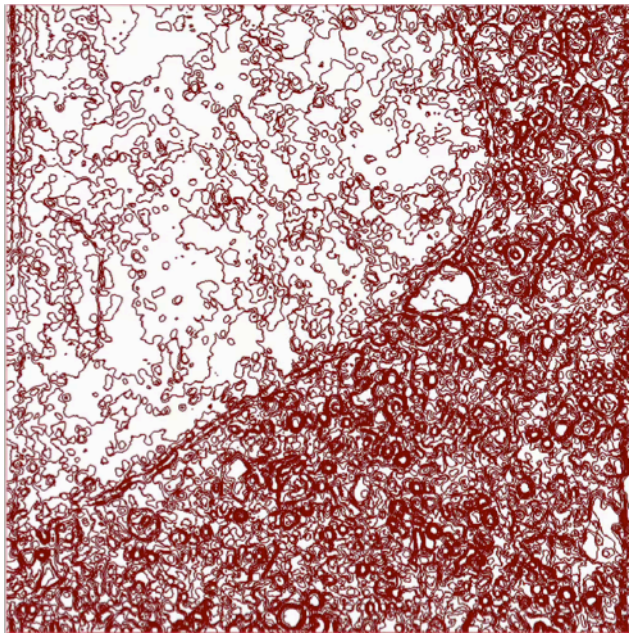
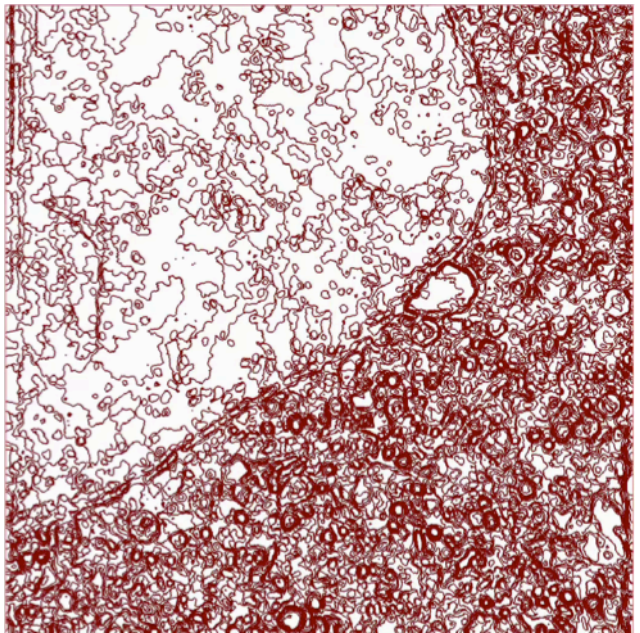




10x10x10 / 1x1x1 / **10**

Default:
10x10x10 / 1x1x1 / 20

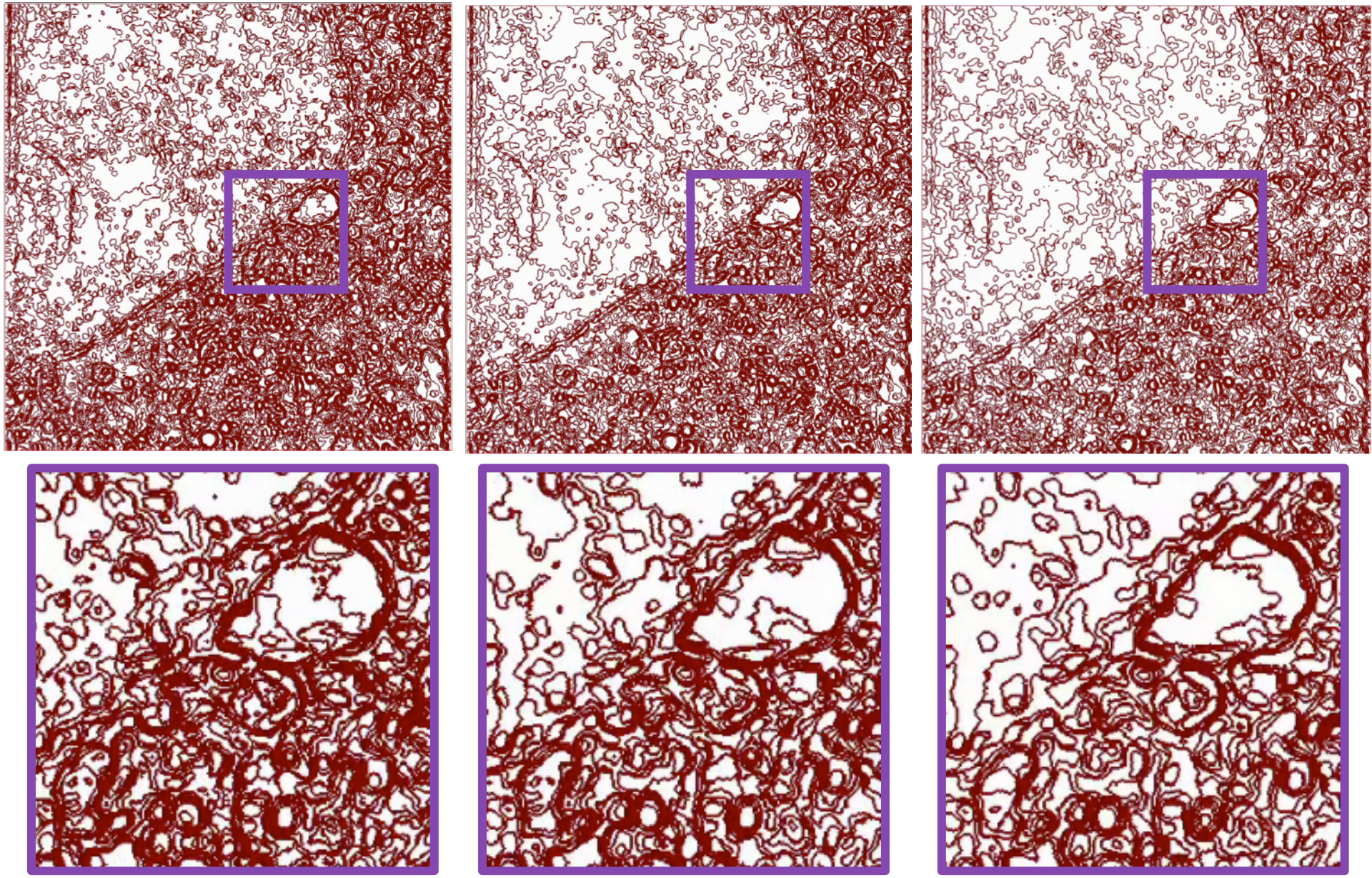
10x10x10 / 1x1x1 / **30**

10x10x10 / 1x1x1 / **10**

Default:
10x10x10 / 1x1x1 / 20

10x10x10 / 1x1x1 / **30**

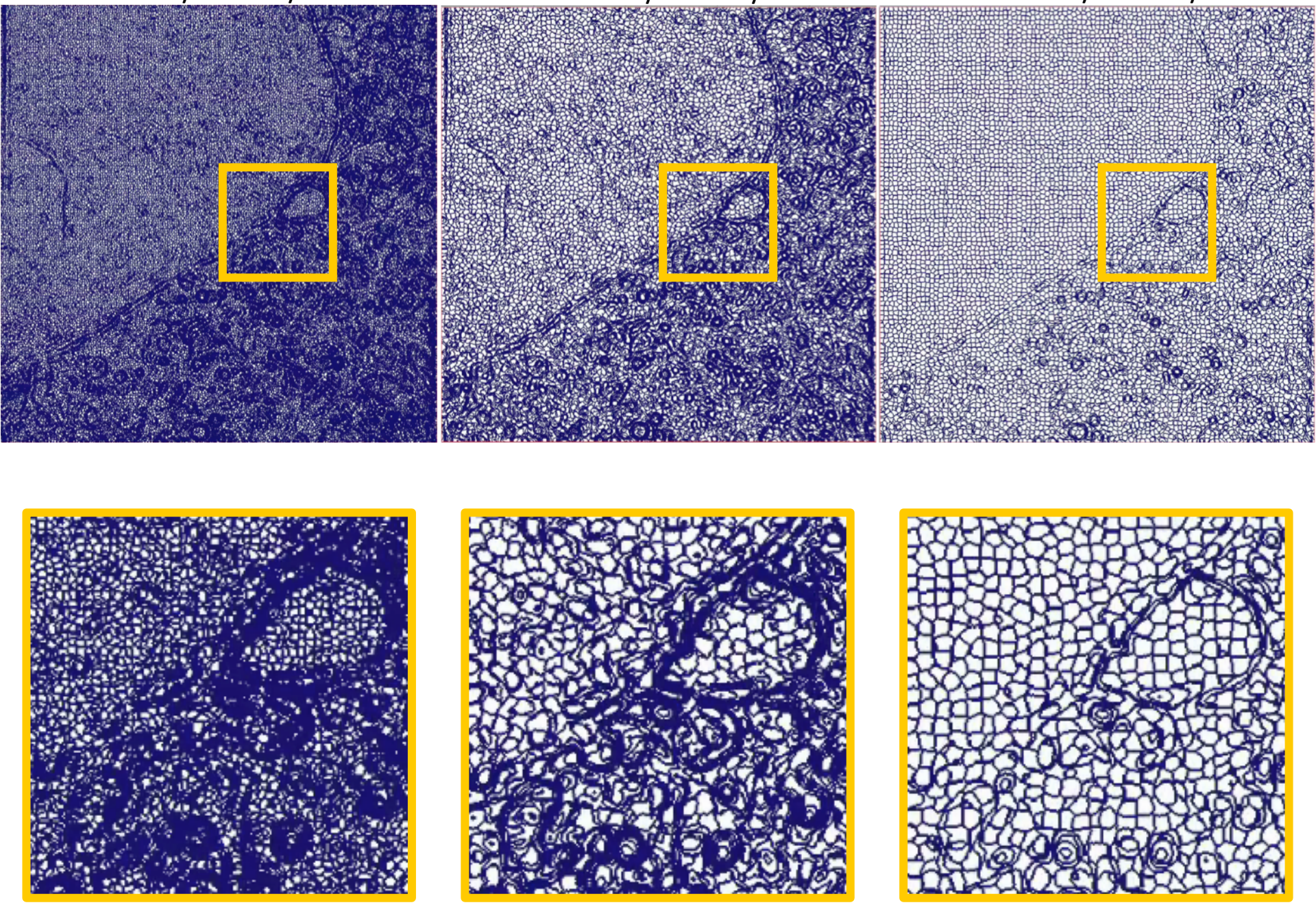


Default:

5x5x5 / 1x1x1 / 20

10x10x10 / 1x1x1 / 20

10x10x10 / **3x3x3** / 20

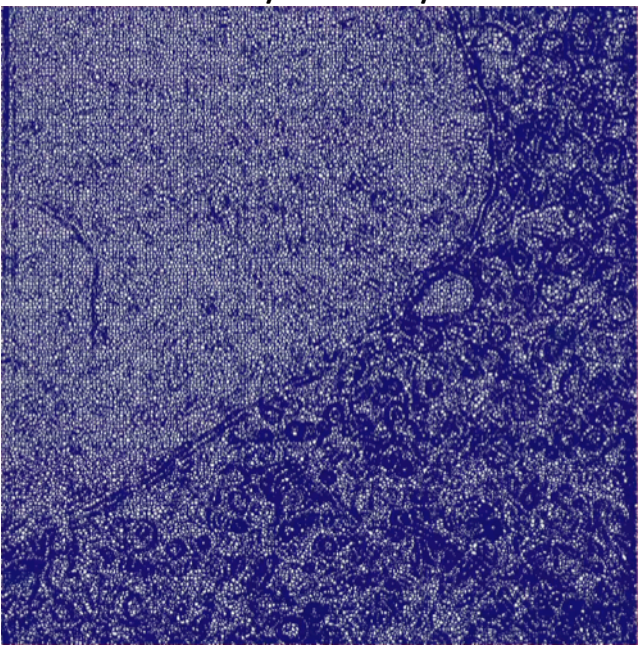


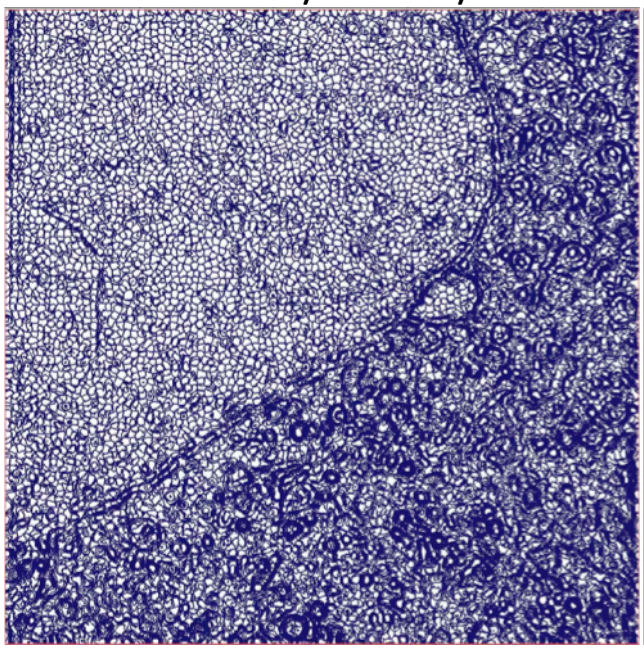
Default:

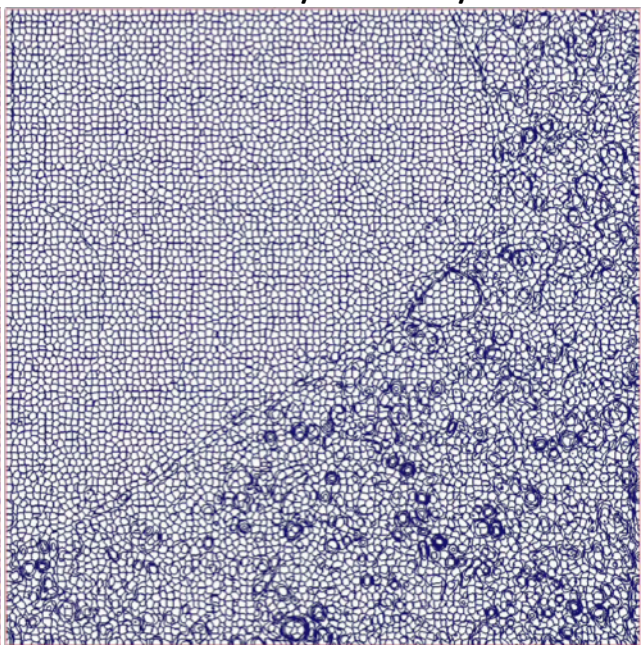
5x5x5 / 1x1x1 / 20

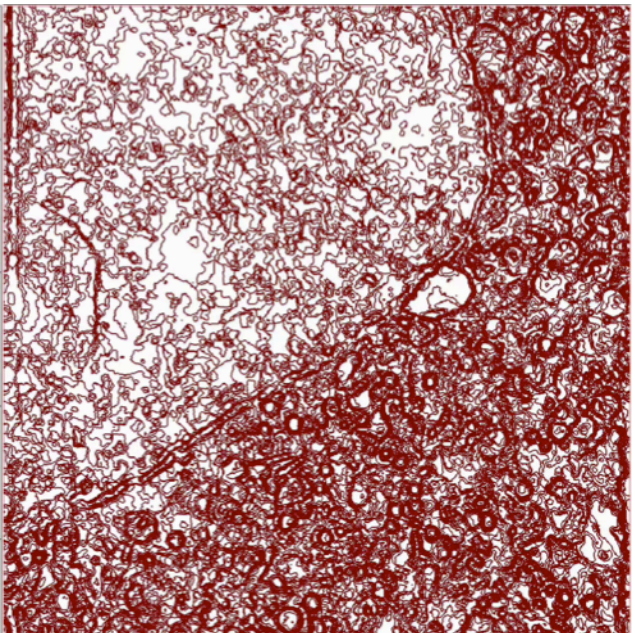
10x10x10 / 1x1x1 / 20

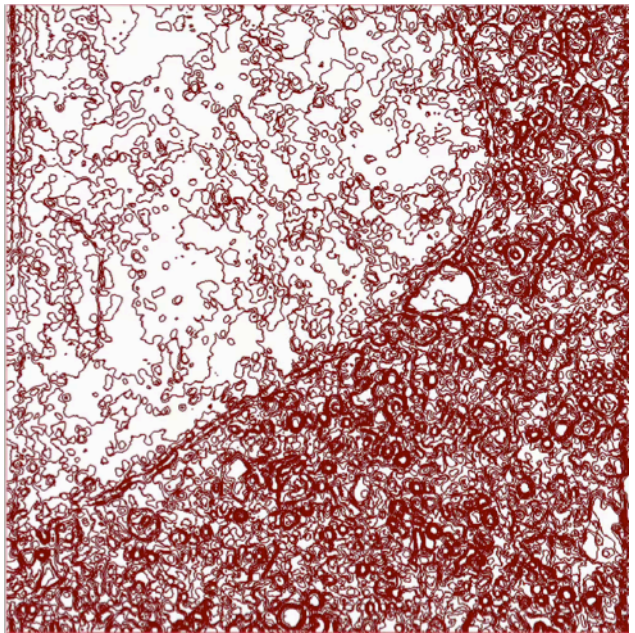
10x10x10 / **3x3x3** / 20

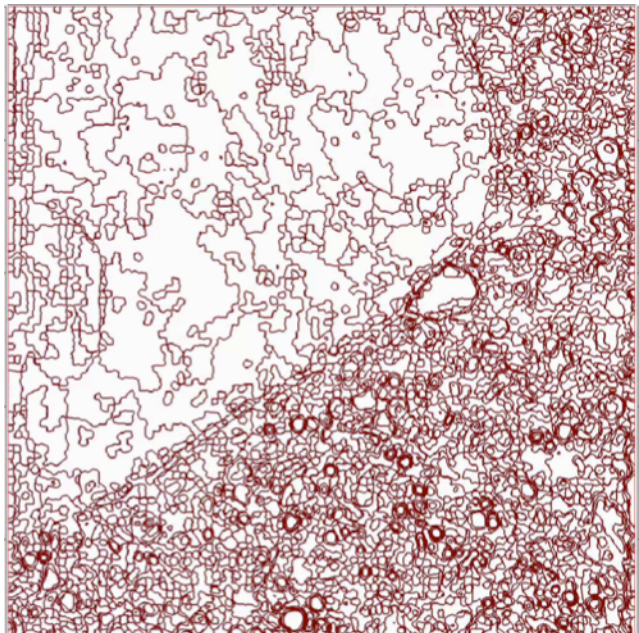










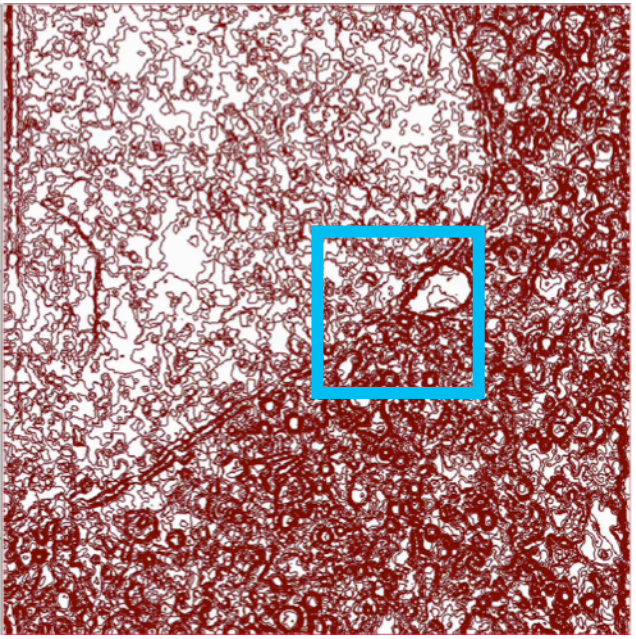


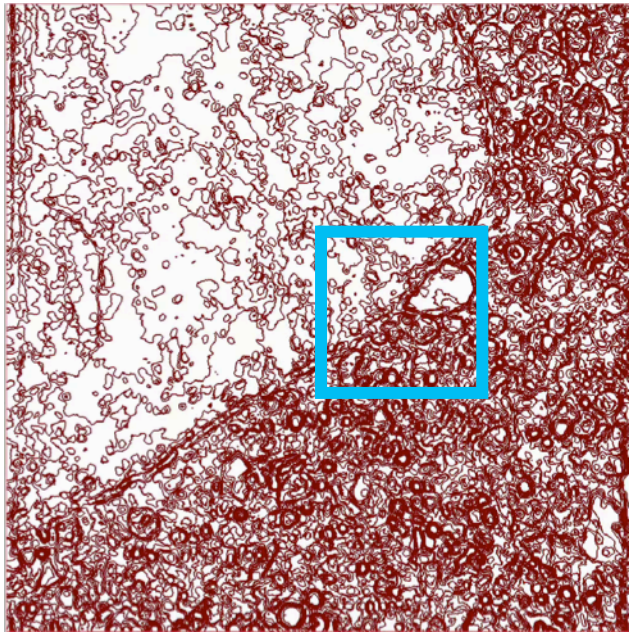
Default:

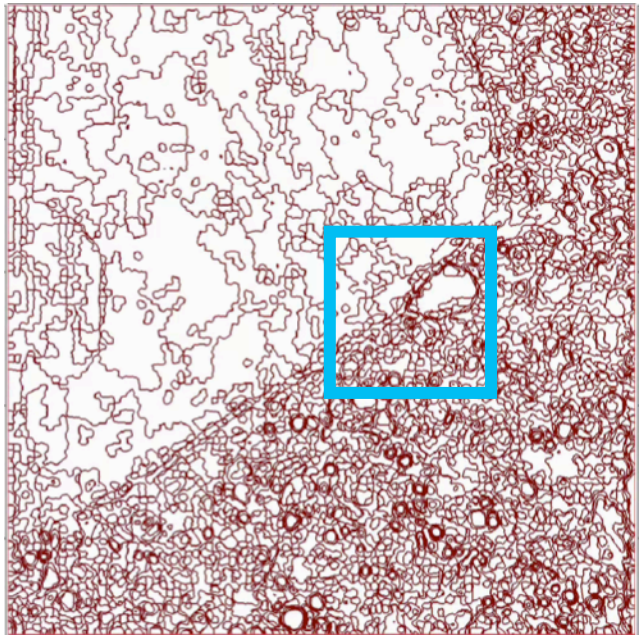
5x5x5 / 1x1x1 / 20

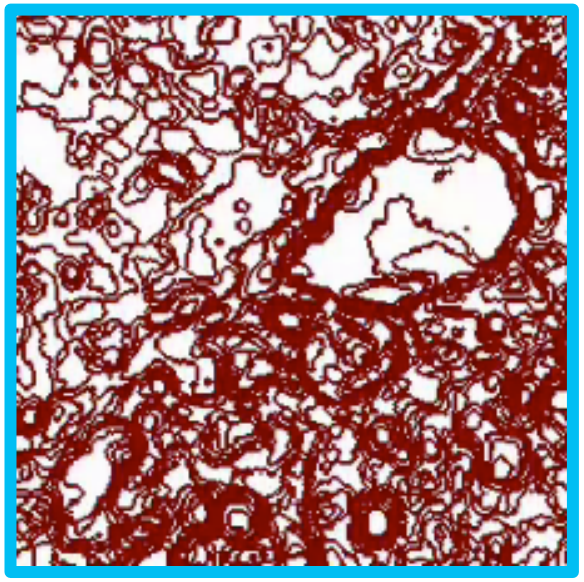
10x10x10 / 1x1x1 / 20

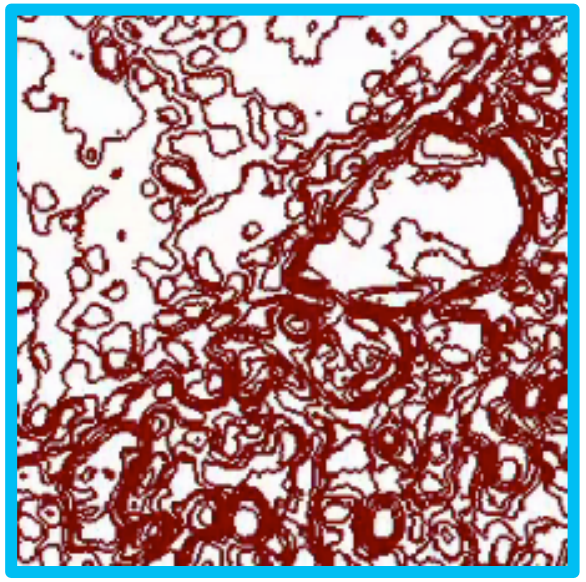
10x10x10 / **3x3x3** / 20

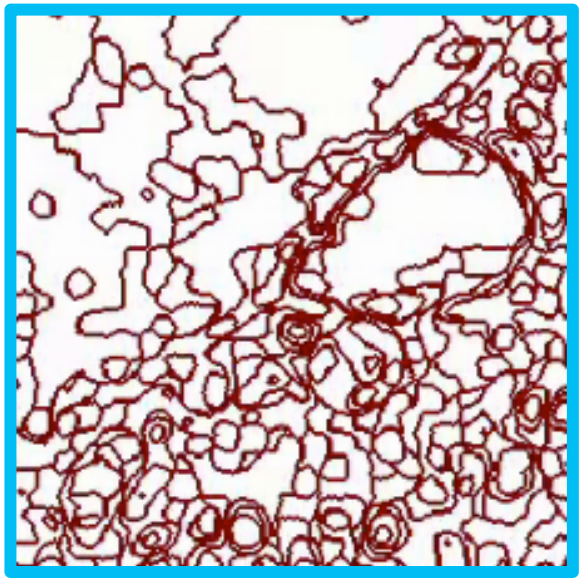




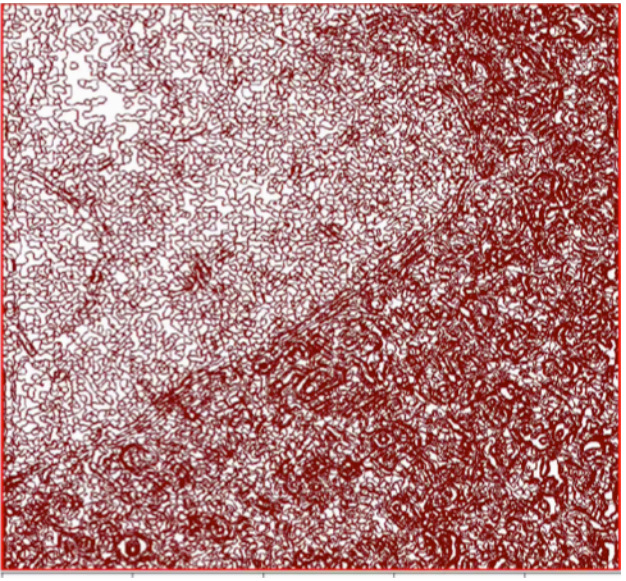




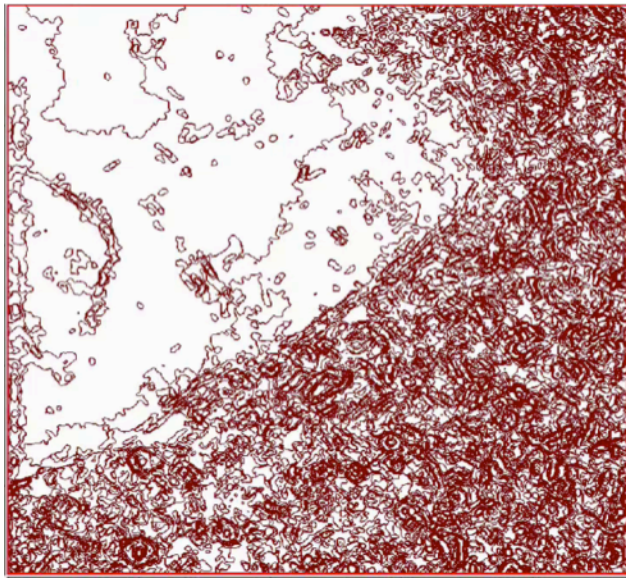




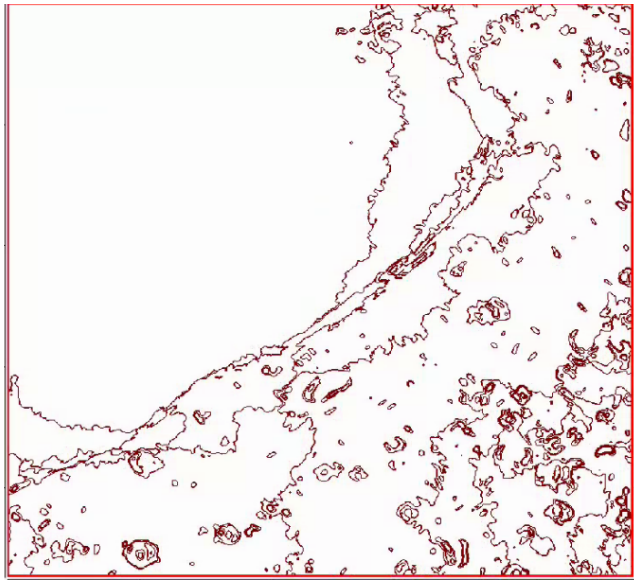
λ=0.01/NumBins=20



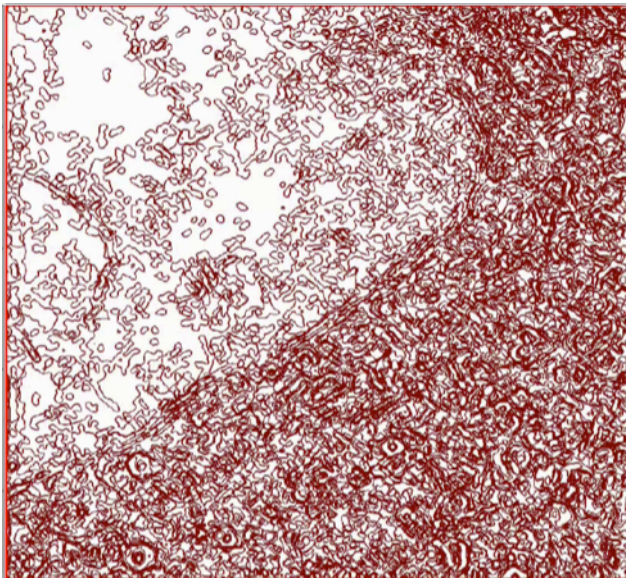
Default:
λ=0.1/NumBins=20



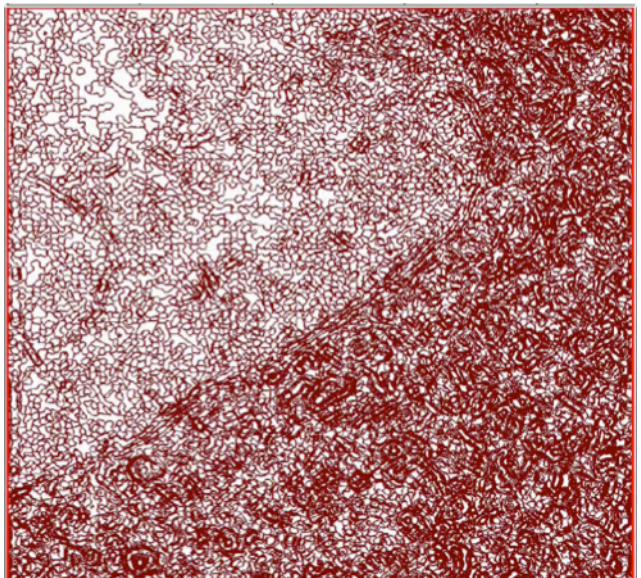
λ=1/NumBins=20



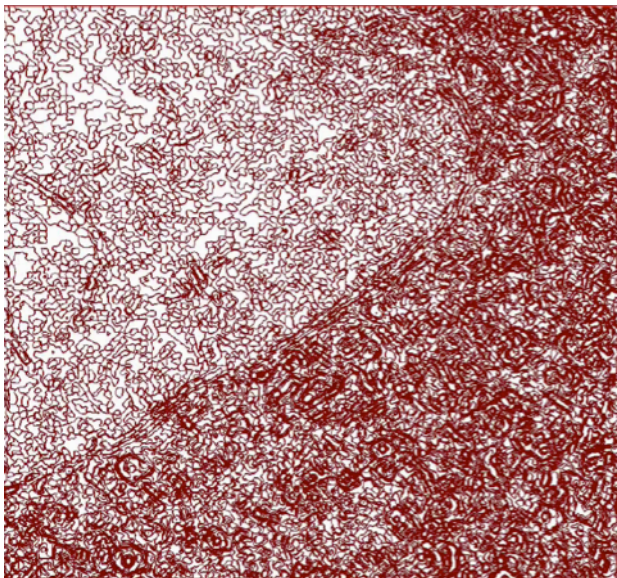
λ=0.1/**NumBins=10**



λ=0.1/**NumBins=30**



Defaults, **gamma=auto**



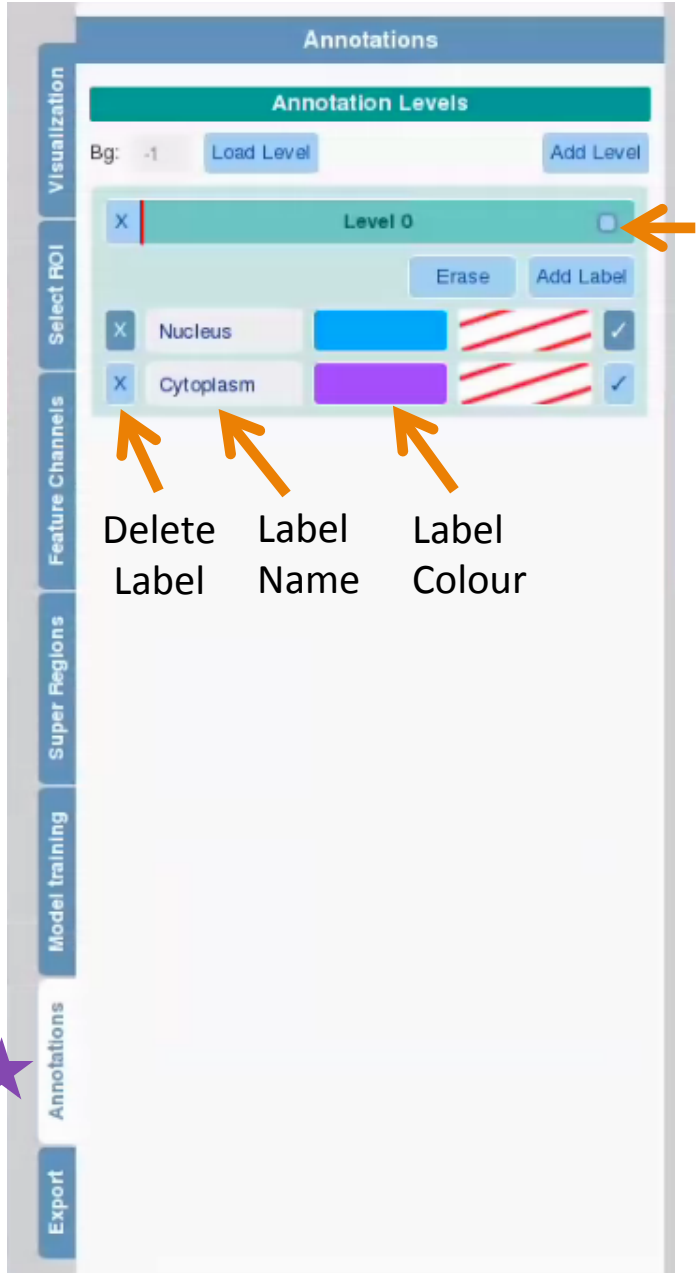
Time – 15 minutes

Switch to Matt

The Annotation Tab

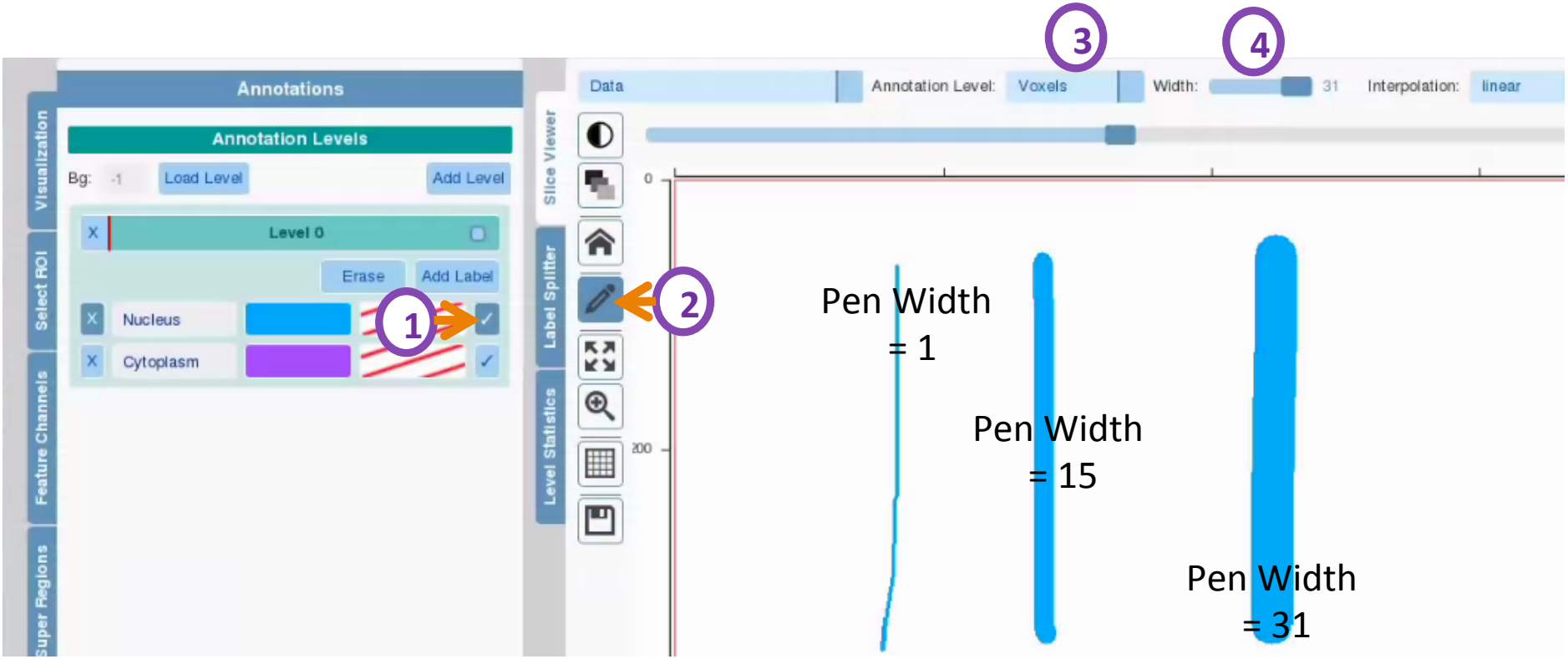
- To add a Level click on Add Level.
- To add Labels to a Level, click Add Label. Multiple labels can be added to any Level.

The Annotation Tab



- To add a Level click on Add Level.
- To add Labels to a Level, click Add Label. Multiple labels can be added to any Level.
- Labels can be given names, the colour used to represent them in annotations can be changed. Or if they are no longer needed, labels can be deleted.

Annotation Using Voxels



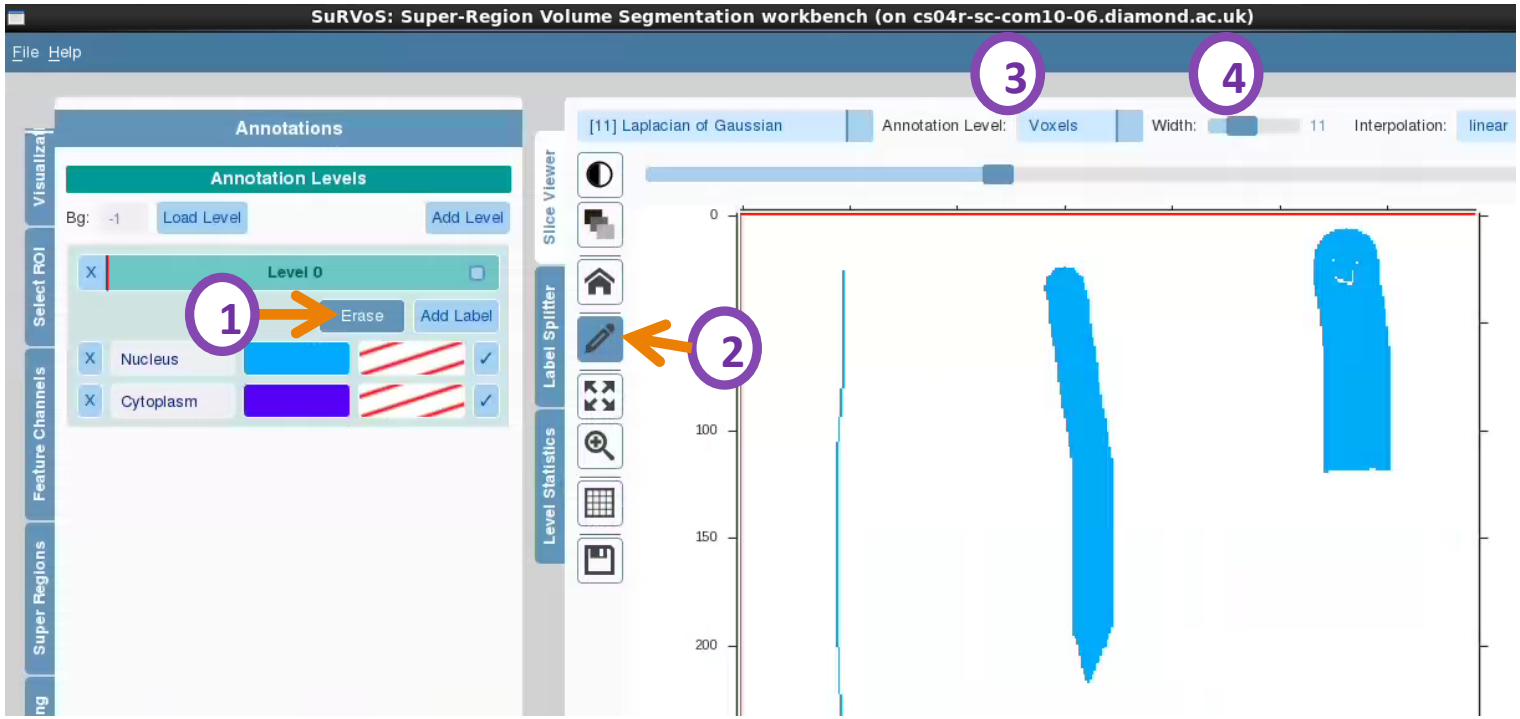
1) In the annotation tab select the label you wish to annotate with

2) Select the pencil tool

3) Select annotation level (**voxel**, supervoxel, or megavoxel)

4) Choose an appropriate width for the feature that you are annotating and draw using the left mouse button

Erasing Annotations



1) In the annotation tab select erase on the level you wish to use the use the eraser on

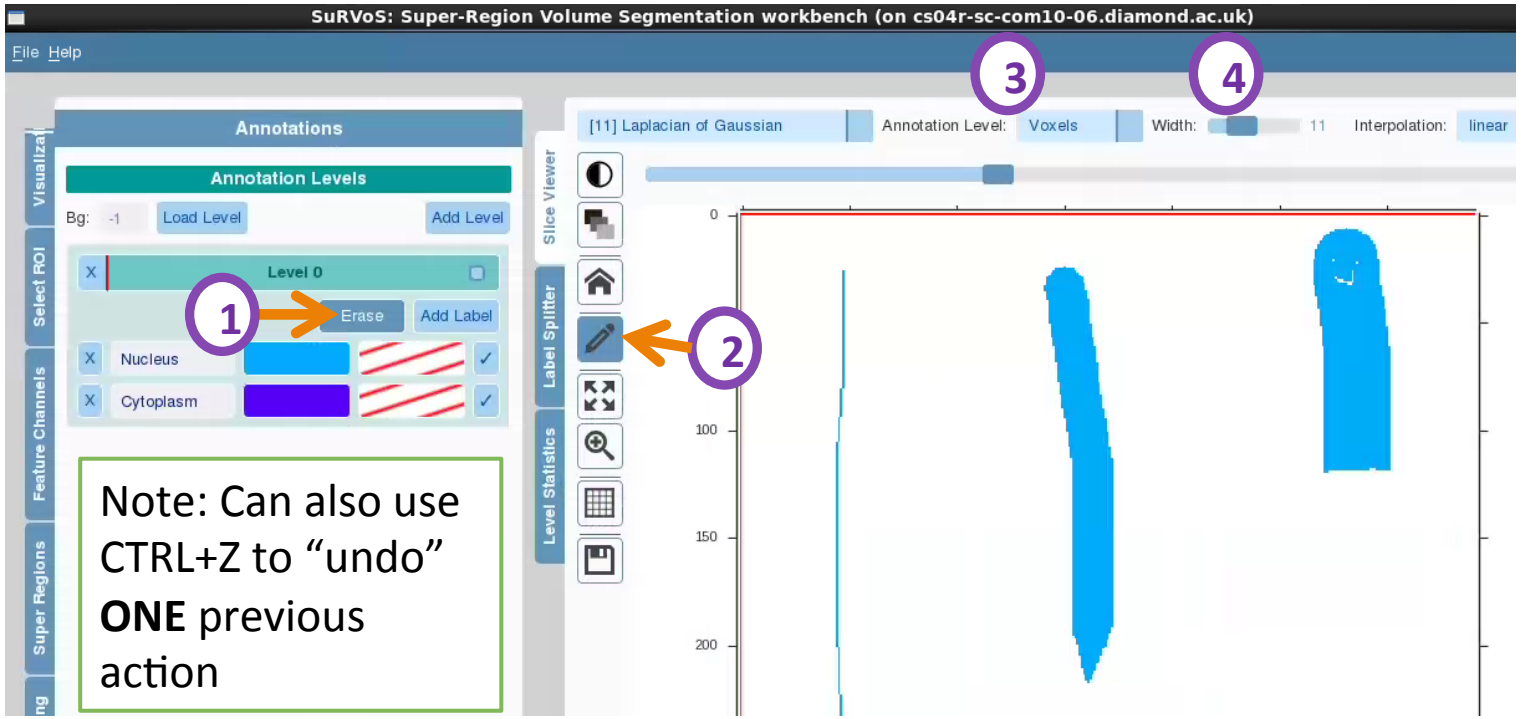
2) Select the pencil tool (if not already selected)

3) Select annotation level (voxel, supervoxel, or megavoxel)

4) Choose an appropriate width for the feature that you are erasing and erase using the left mouse button

5) To begin annotating again, select the appropriate label and parameters.

Erasing Annotations



1) In the annotation tab select erase on the level you wish to use the use the eraser on

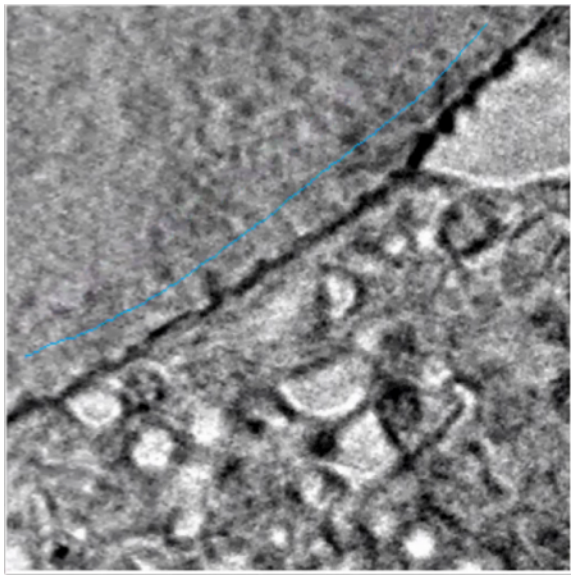
2) Select the pencil tool (if not already selected)

3) Select annotation level (voxel, supervoxel, or megavoxel)

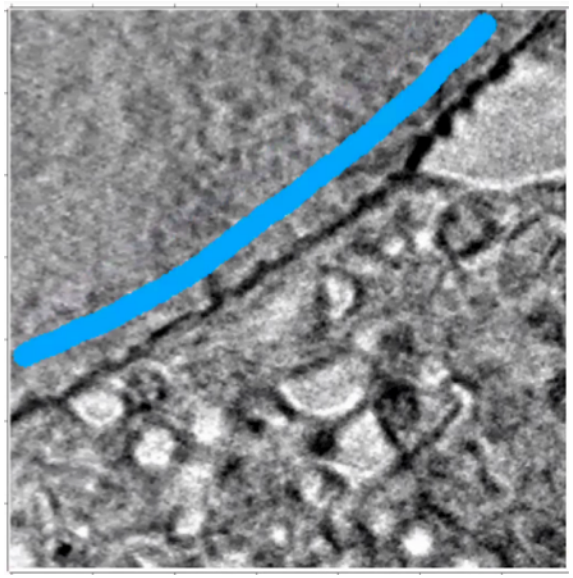
4) Choose an appropriate width for the feature that you are erasing and erase using the left mouse button

5) To begin annotating again, select the appropriate label and parameters.

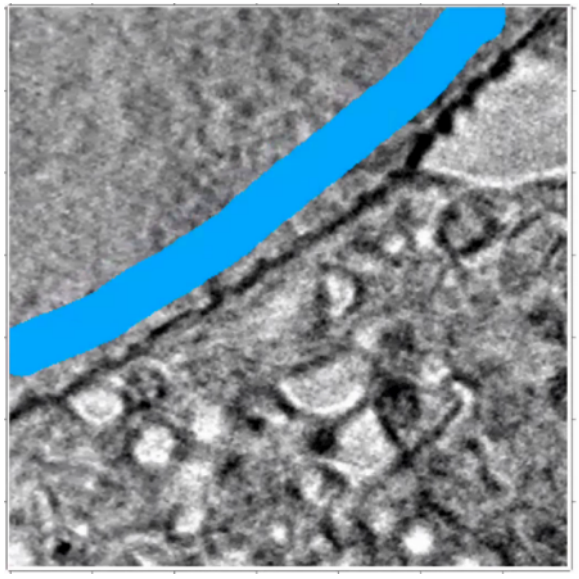
Annotation Using Voxels



Pen Width = 1



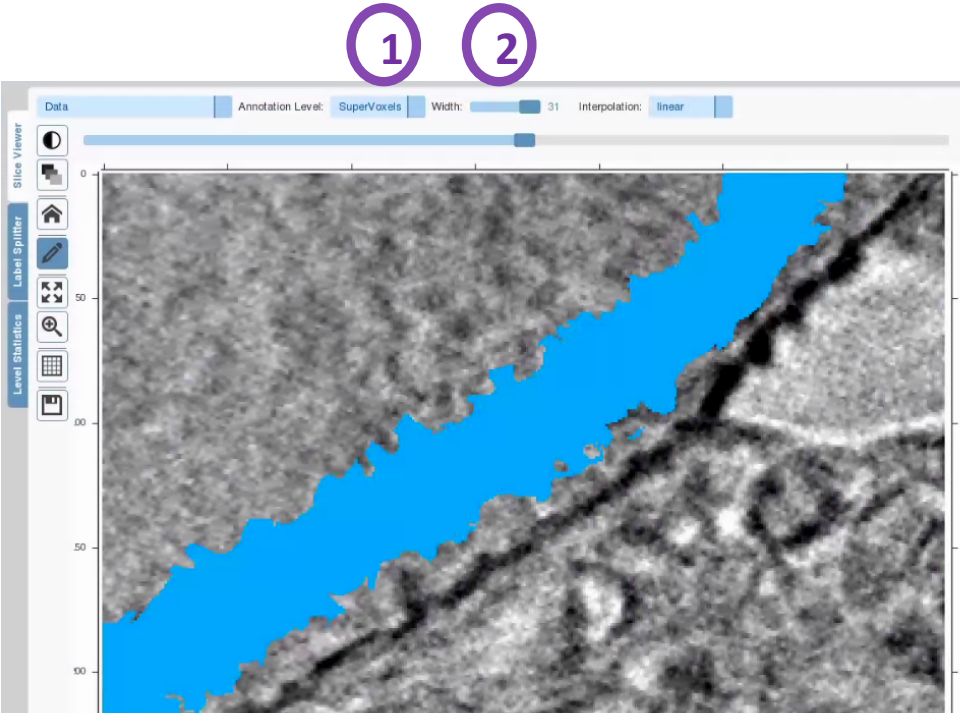
Pen Width = 15

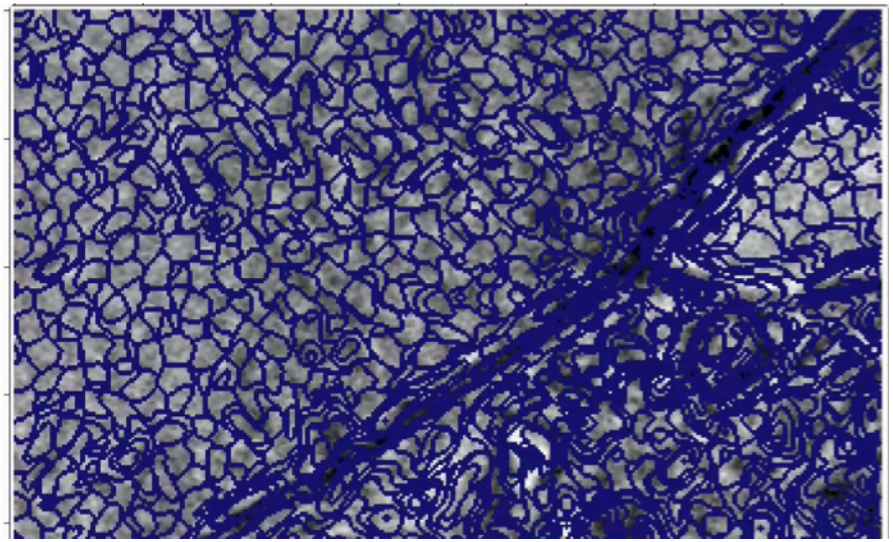


Pen Width = 31

- Annotation using voxels can be done with different pen widths.
- Whilst annotating in voxels annotations do not penetrate the volume in Z

Annotation Using Supervoxels

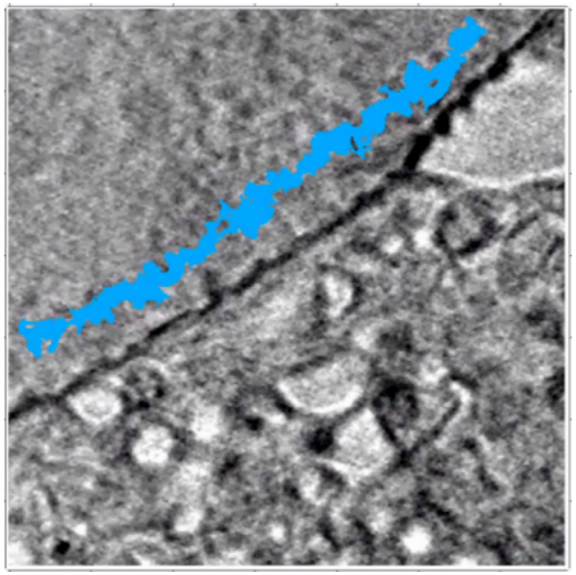




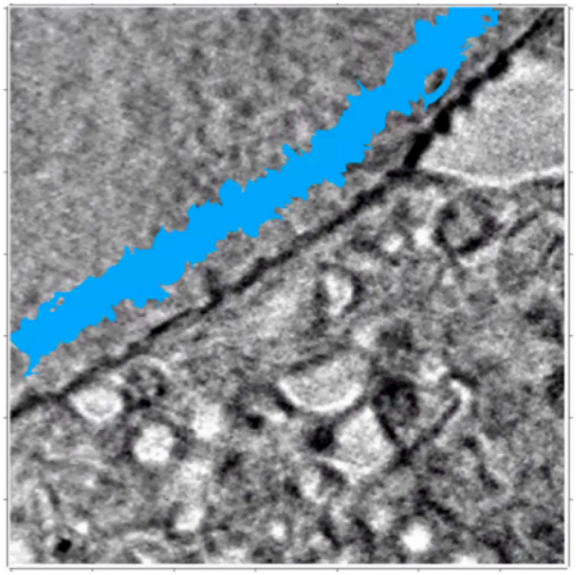
1) Select annotation level (voxel, **supervoxel**, or megavoxel)

2) Choose a width appropriate for the feature that you are annotating and draw using the left mouse button.

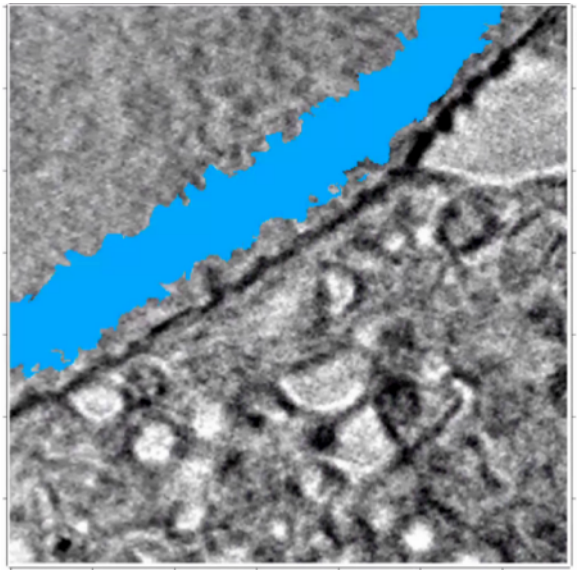
Annotation Using Supervoxels



Pen Width = 1

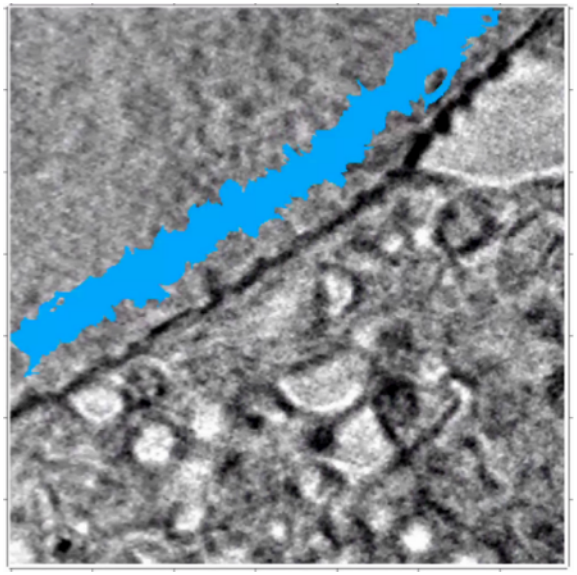


Pen Width = 15



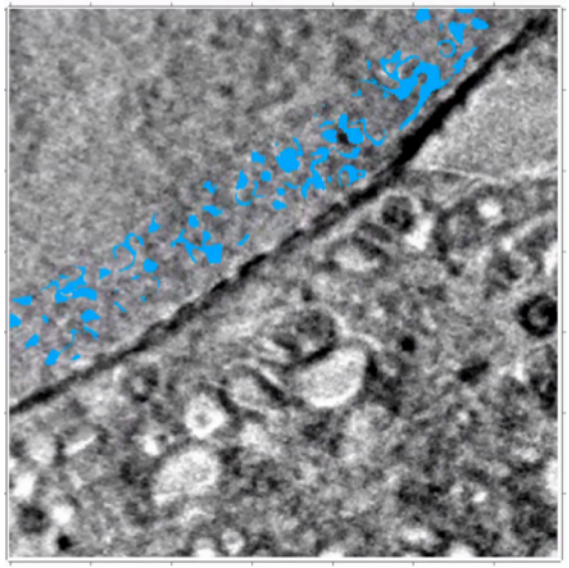
Pen Width = 31

Annotation Using Supervoxels



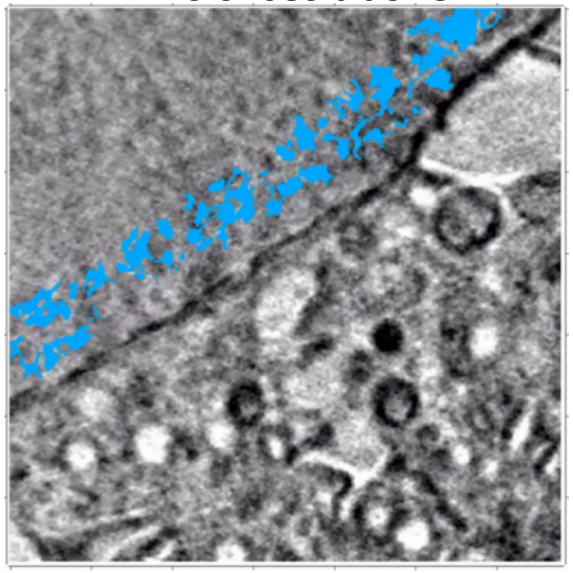
Pen Width = 15
Center Slice

10 slices below

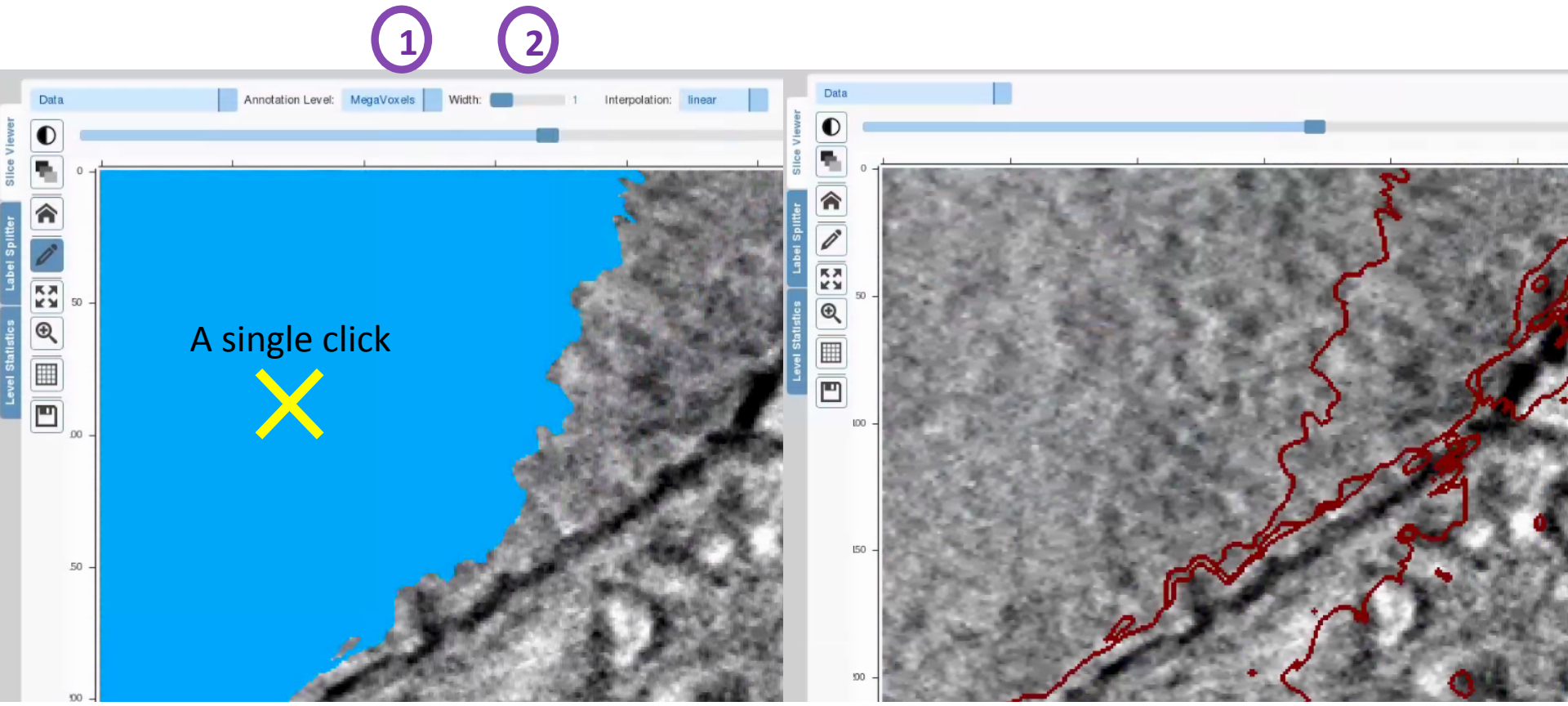


Annotating using
supervoxels penetrates
multiple Z-slices

10 slices above



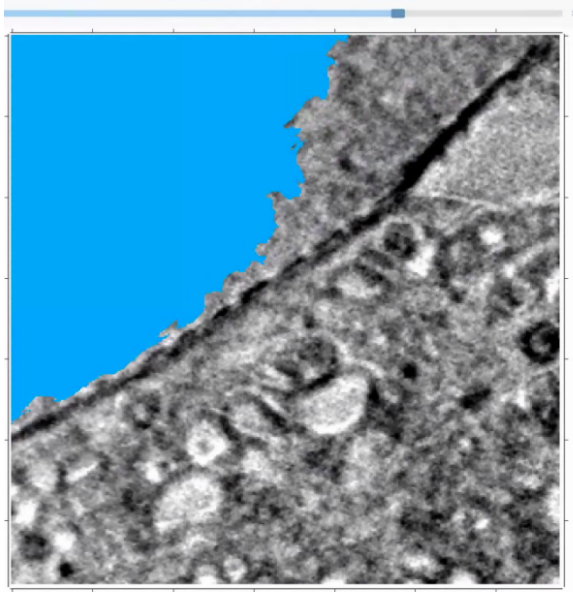
Annotation using Megavoxels



1) Select annotation level (voxel, supervoxel, or **megavoxel**)

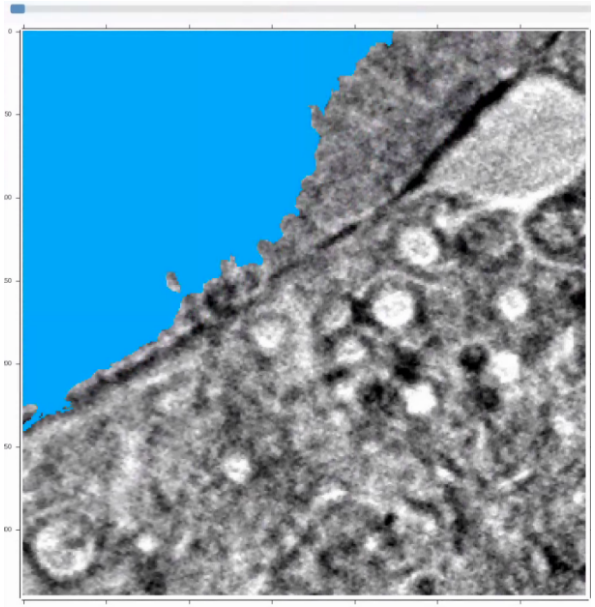
2) Choose a width appropriate for the feature that you are annotating and draw using the left mouse button.

Annotation using Megavoxels



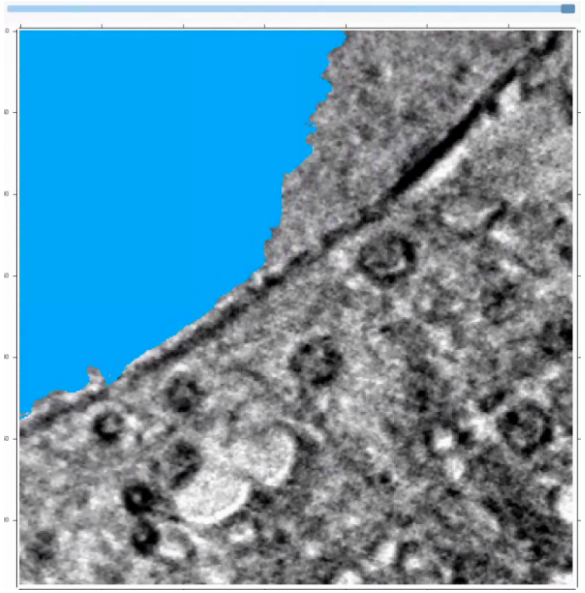
Center Slice

First Slice in Stack

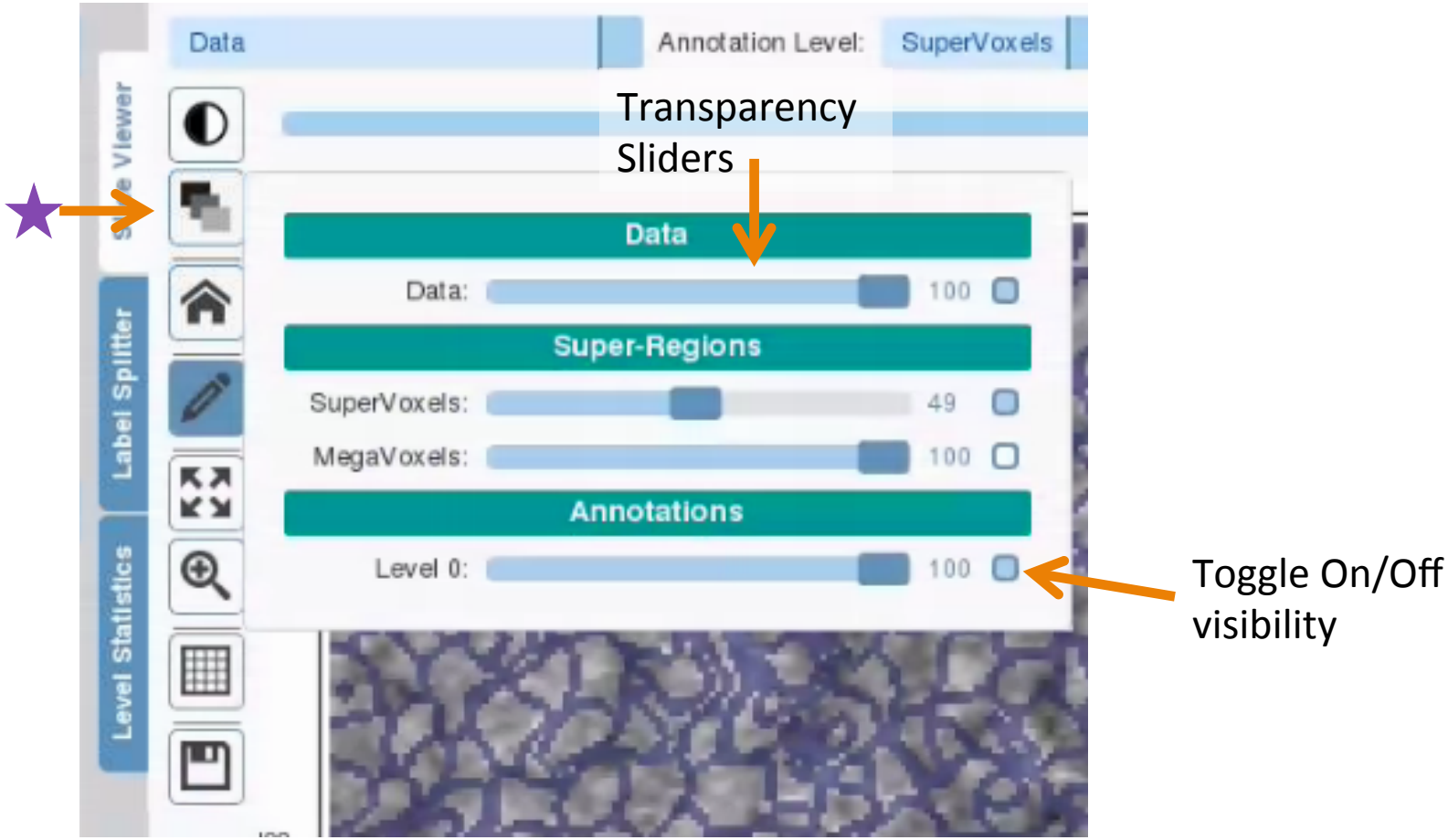


Annotating using megavoxels penetrates multiple Z-slices

Last Slice in Stack



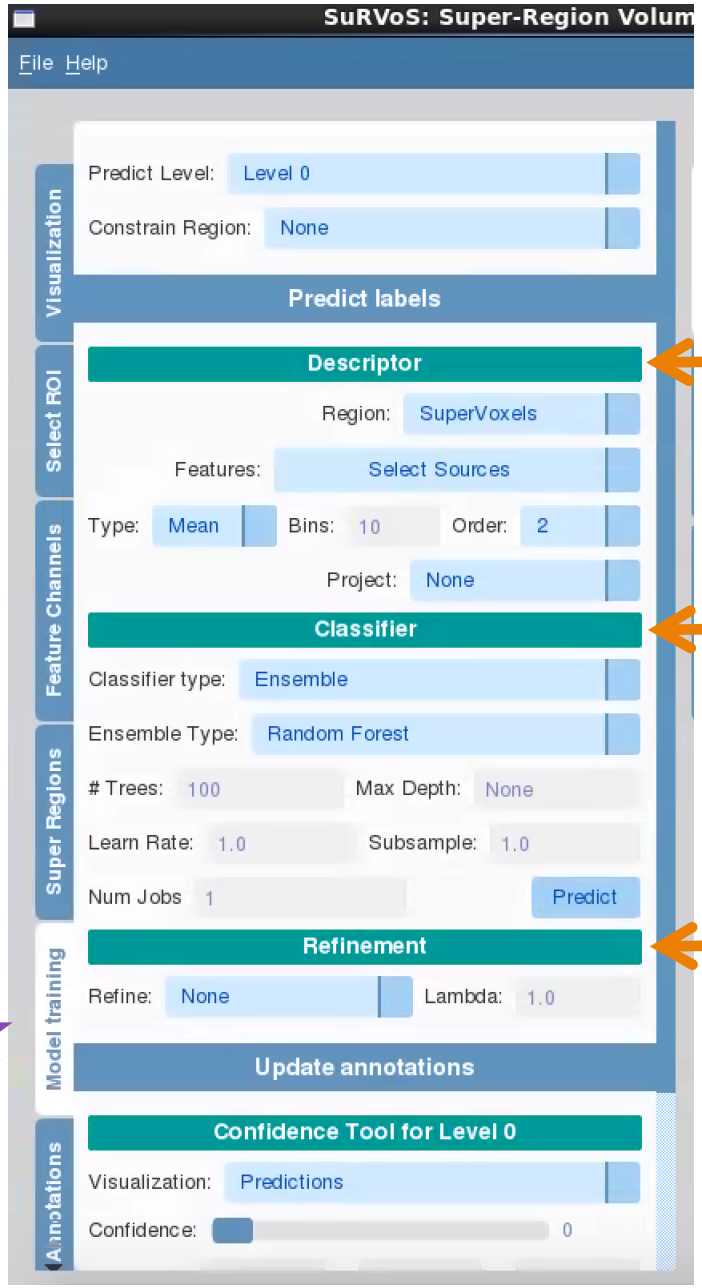
Reminder: Visualization Shortcut



Time – 10 minutes

Switch to Michele

Model Training Introduction



Describes the data at the region level you choose.

Learns a model from the descriptor and classifies the unlabeled data based on this model.

Improves the prediction of the resultant model.

Model Training: How to Use Features

Small amount of annotations

Large amount of annotations

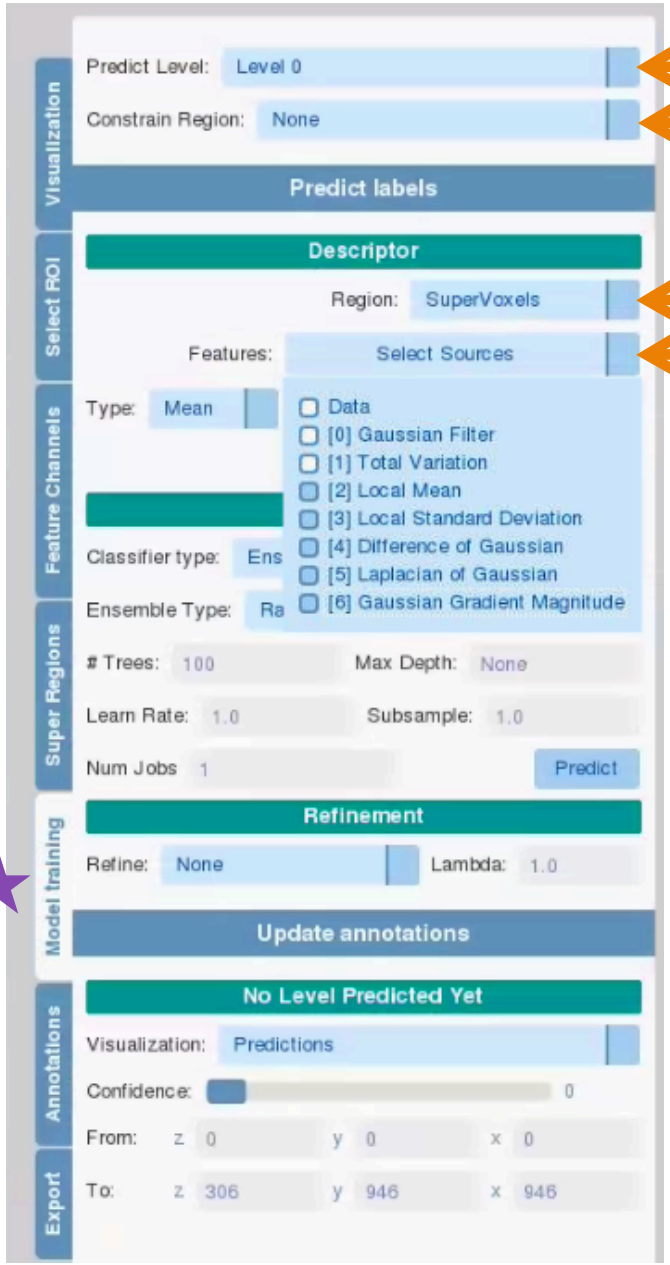
A few robust features

- Blob Detection
- Texture and Structure
- Robust Features

More, varied features

- Any available, as long as you use more of them

Model Training Introduction



Choose which level to train with

Optional (can constrain model training region based on annotation levels)

Choose to train with either voxels or supervoxels

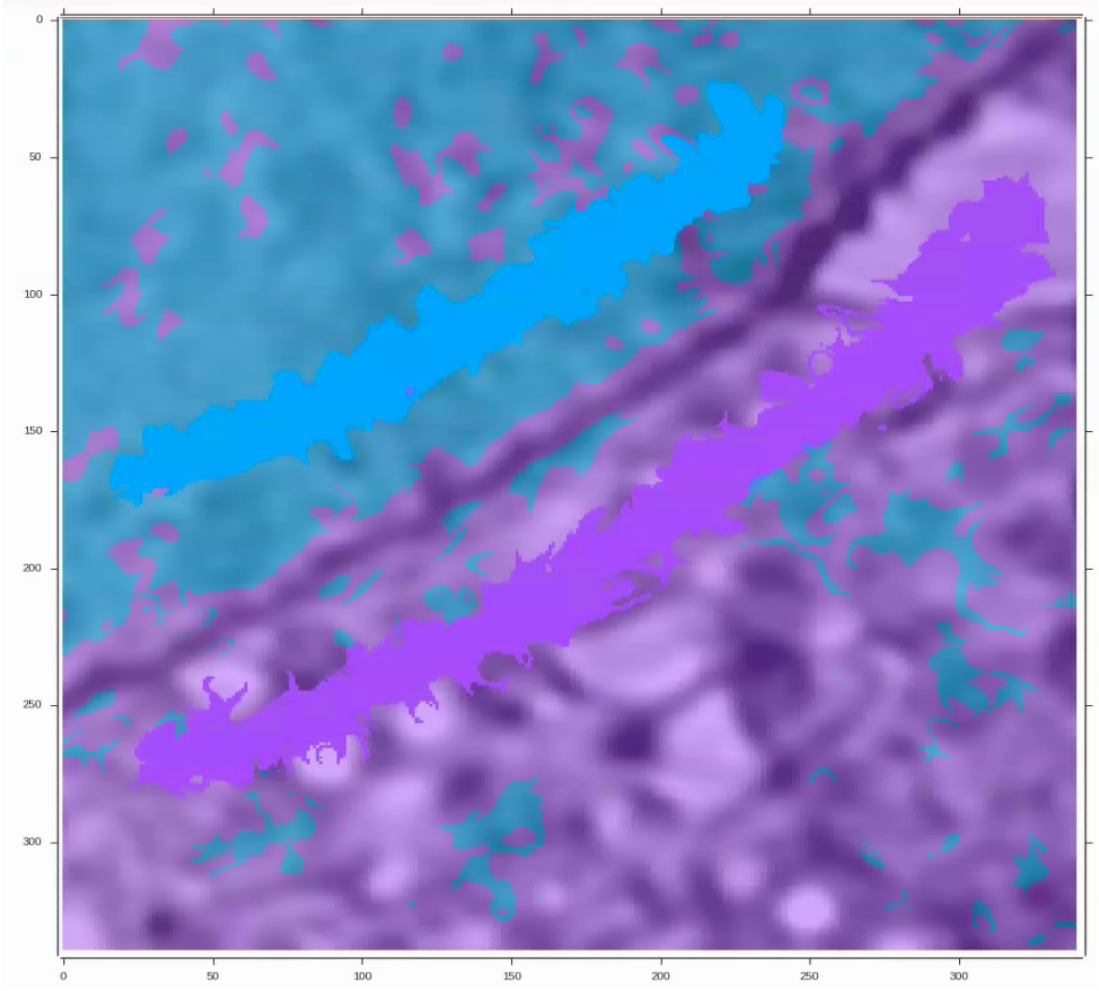
Choose sources to train the descriptor

- Generally want to choose as many sources as possible
- Generally do not choose raw data
- For large areas try: Total Variation, Blob Detection, Texture and Structure, and Robust Feature Filters
- For small areas try: the above, plus Local Feature and Gaussian Feature Filters

Note:

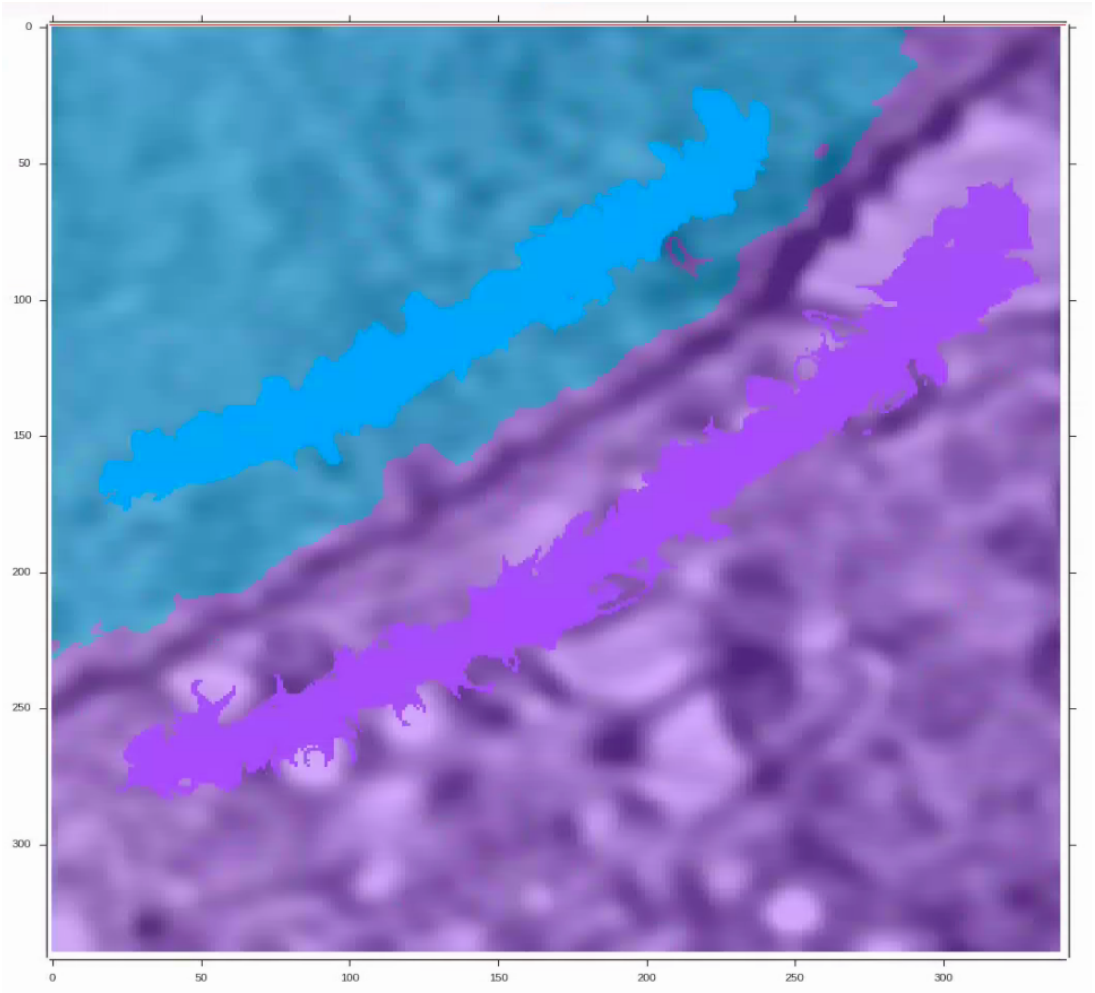
- Smaller objects will require more input annotations to accurately train the model.
- Model training with voxels will be quite slow for large region of interest.

Model Training Options



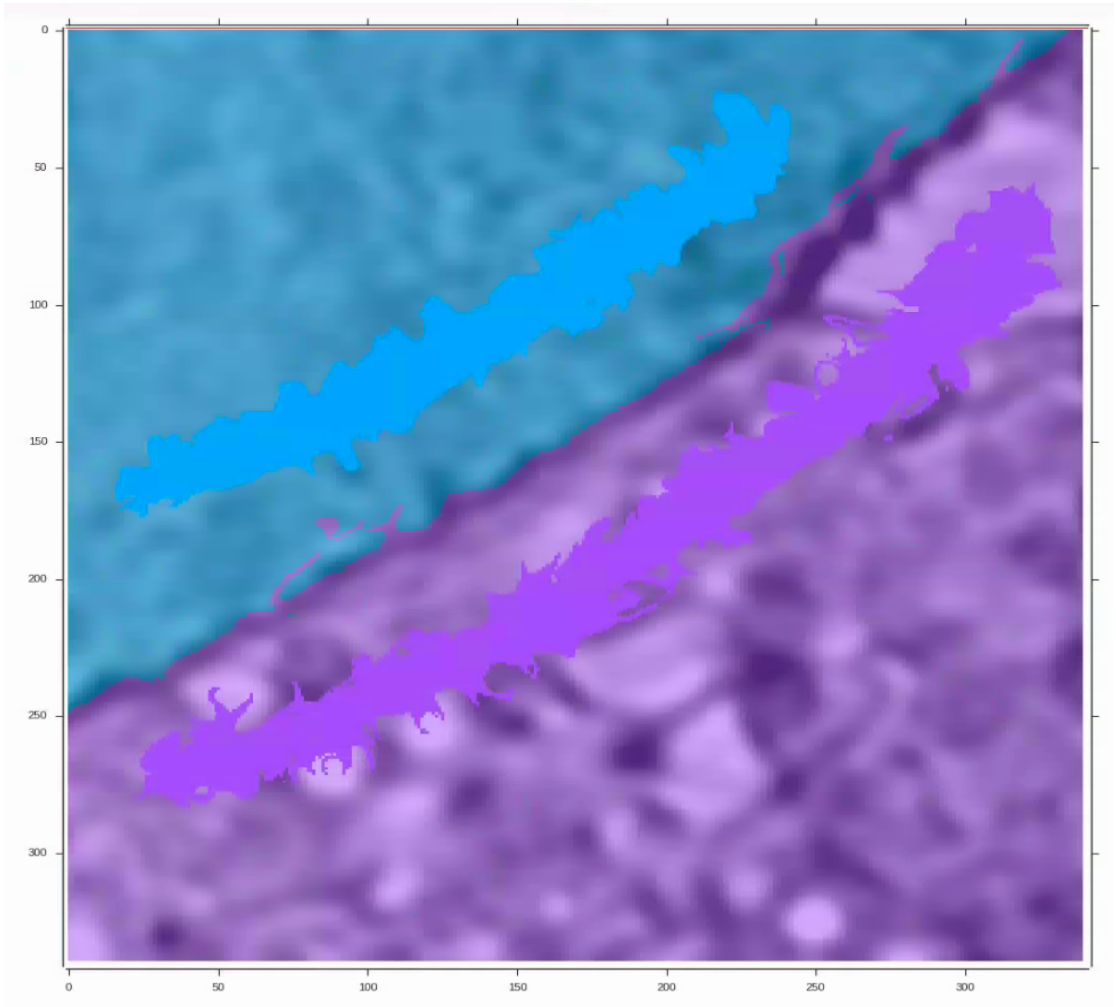
Classifier: Ensemble
Ensemble Type: Random Forest (with defaults)
Refine: None

Model Training Options



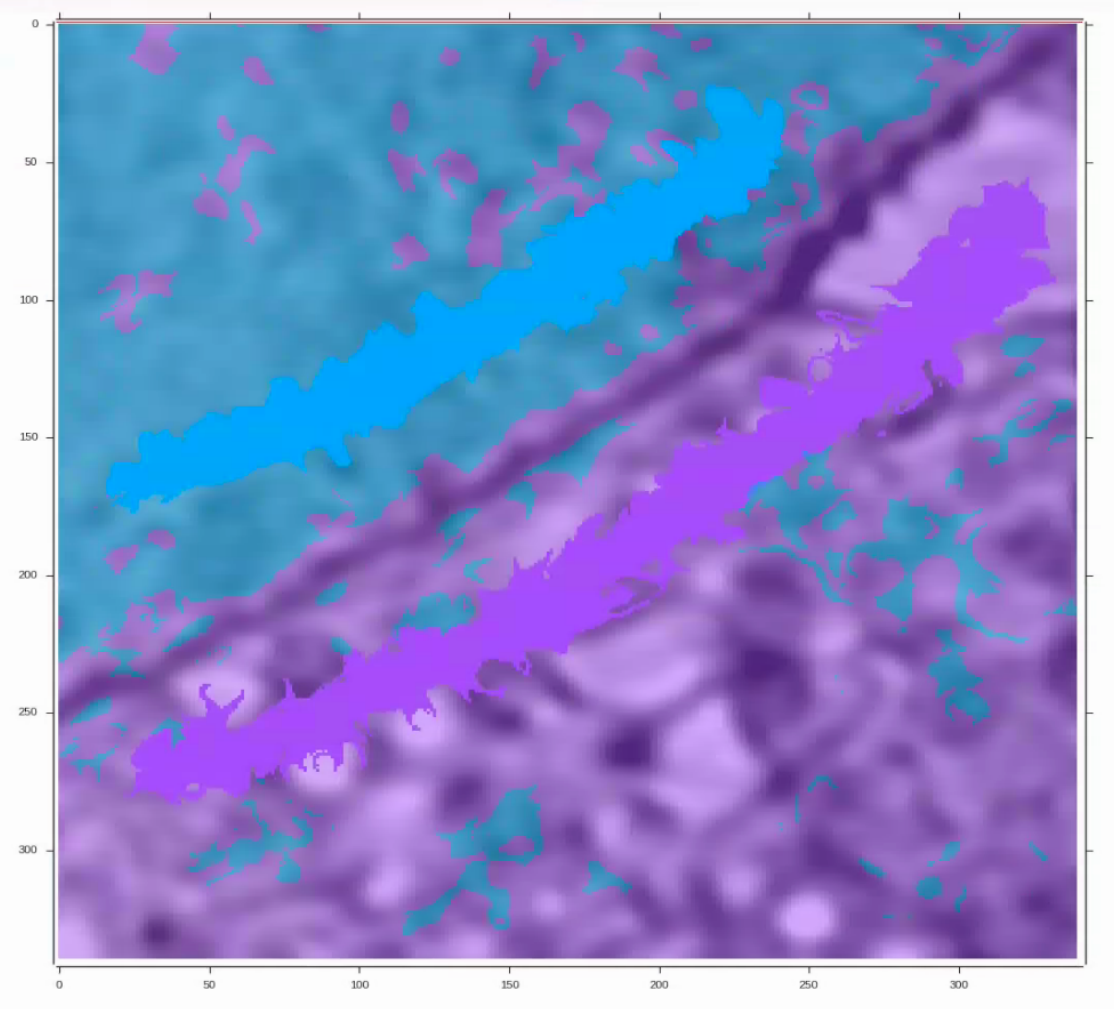
Classifier: Ensemble
Ensemble Type: Random Forest (with defaults)
Refine: Potts (50)

Model Training Options



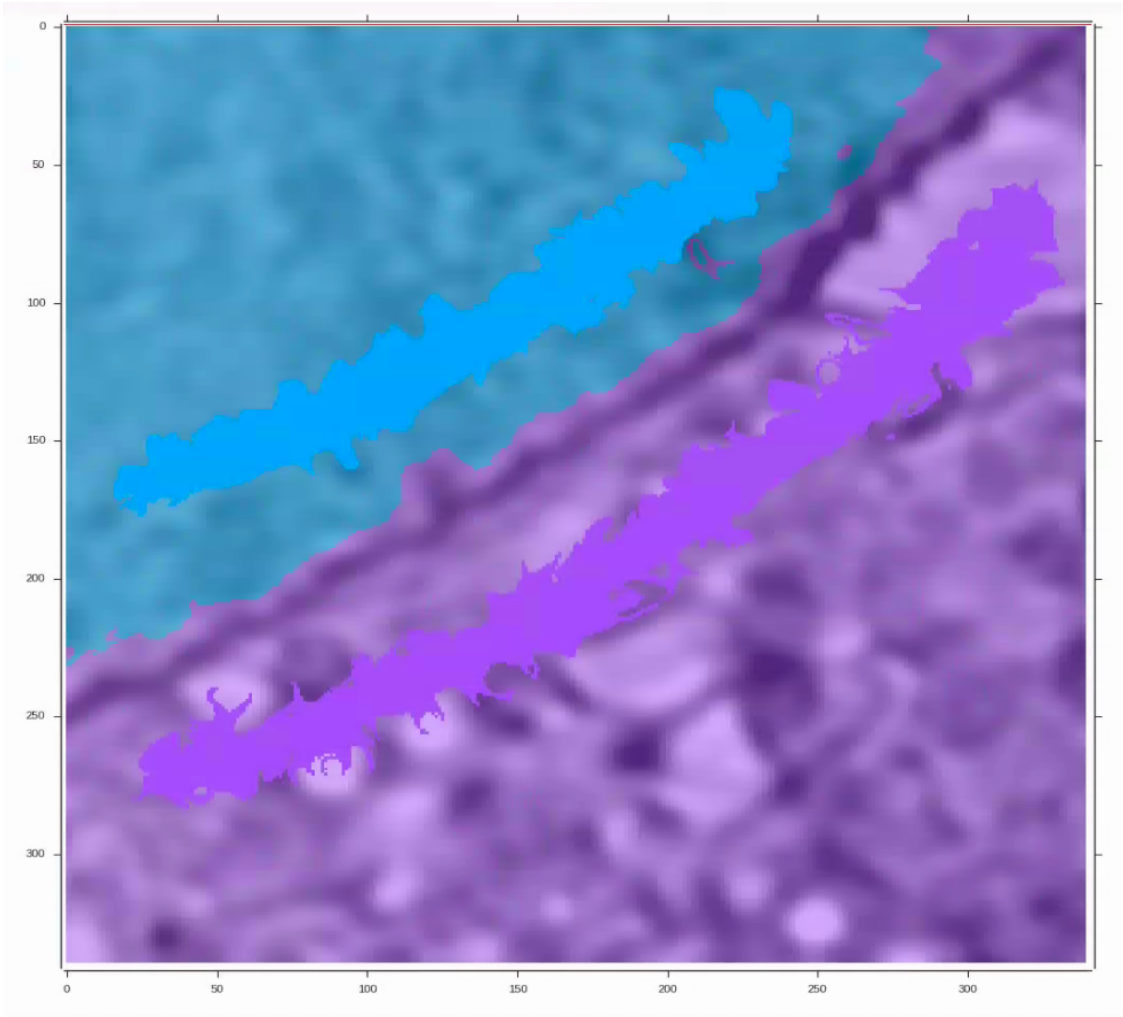
Classifier: Ensemble
Ensemble Type: Random Forest (with defaults)
Refine: Appearance (50)

Model Training Options



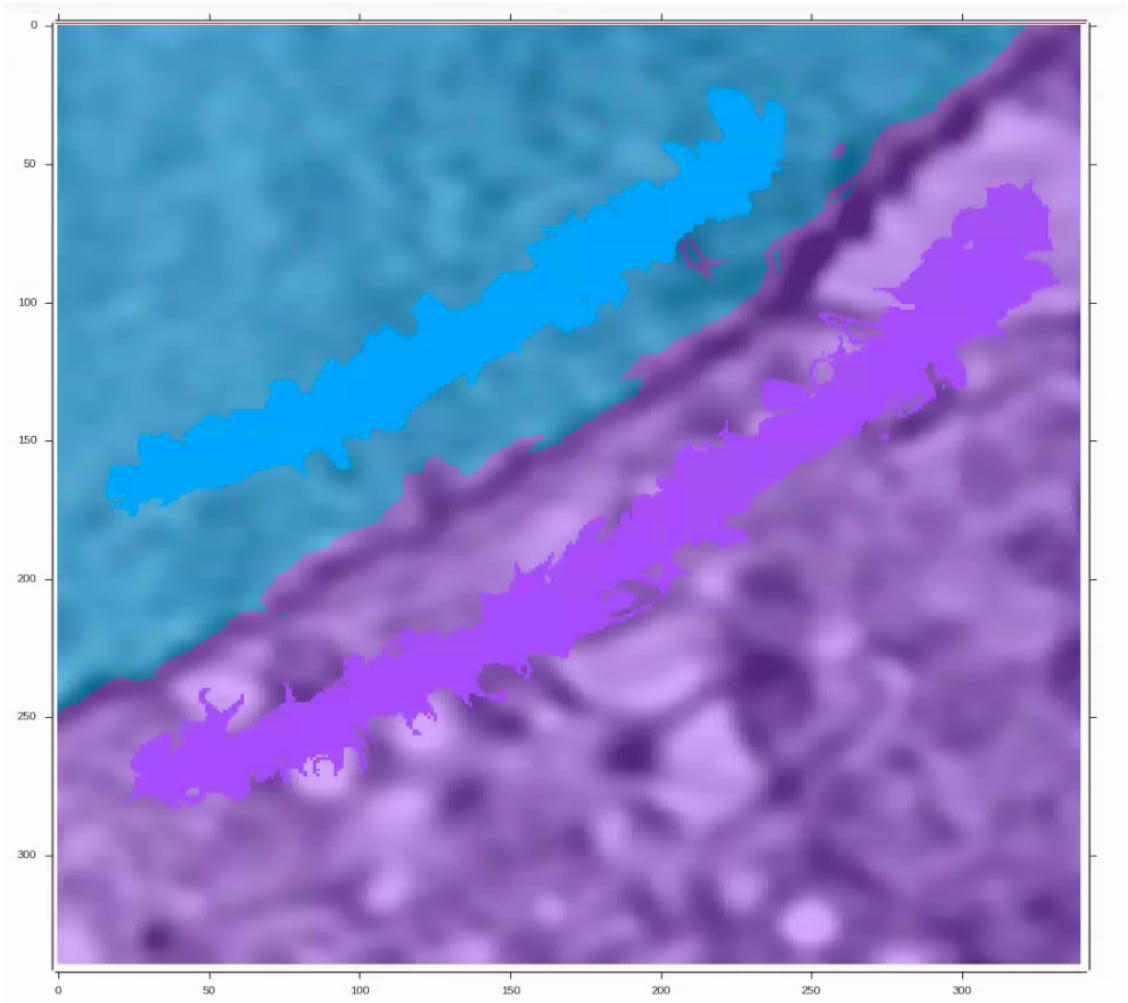
Classifier: Ensemble
Ensemble Type: Extra Random Forest (with defaults)
Refine: None

Model Training Options



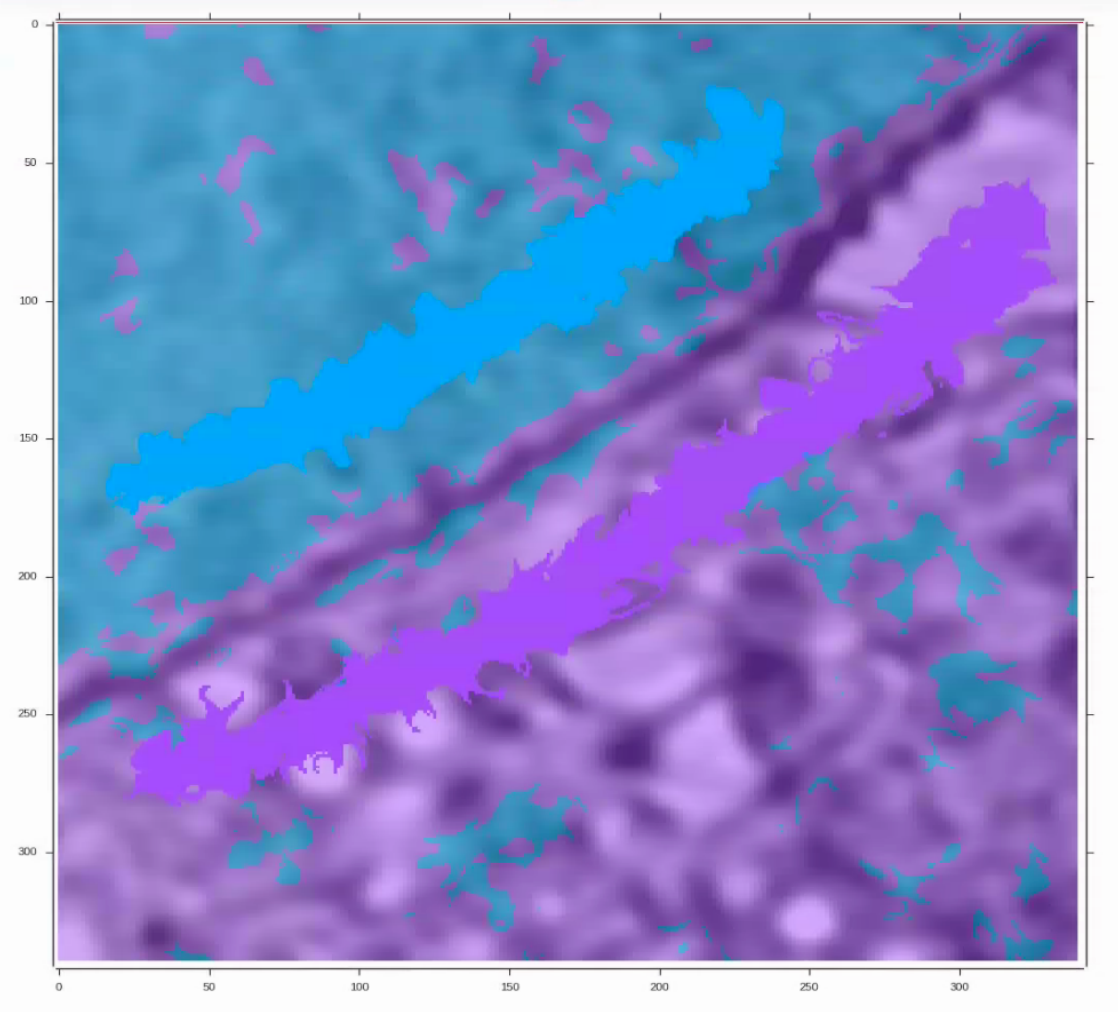
Classifier: Ensemble
Ensemble Type: Extra Random Forest (with defaults)
Refine: Potts (50)

Model Training Options



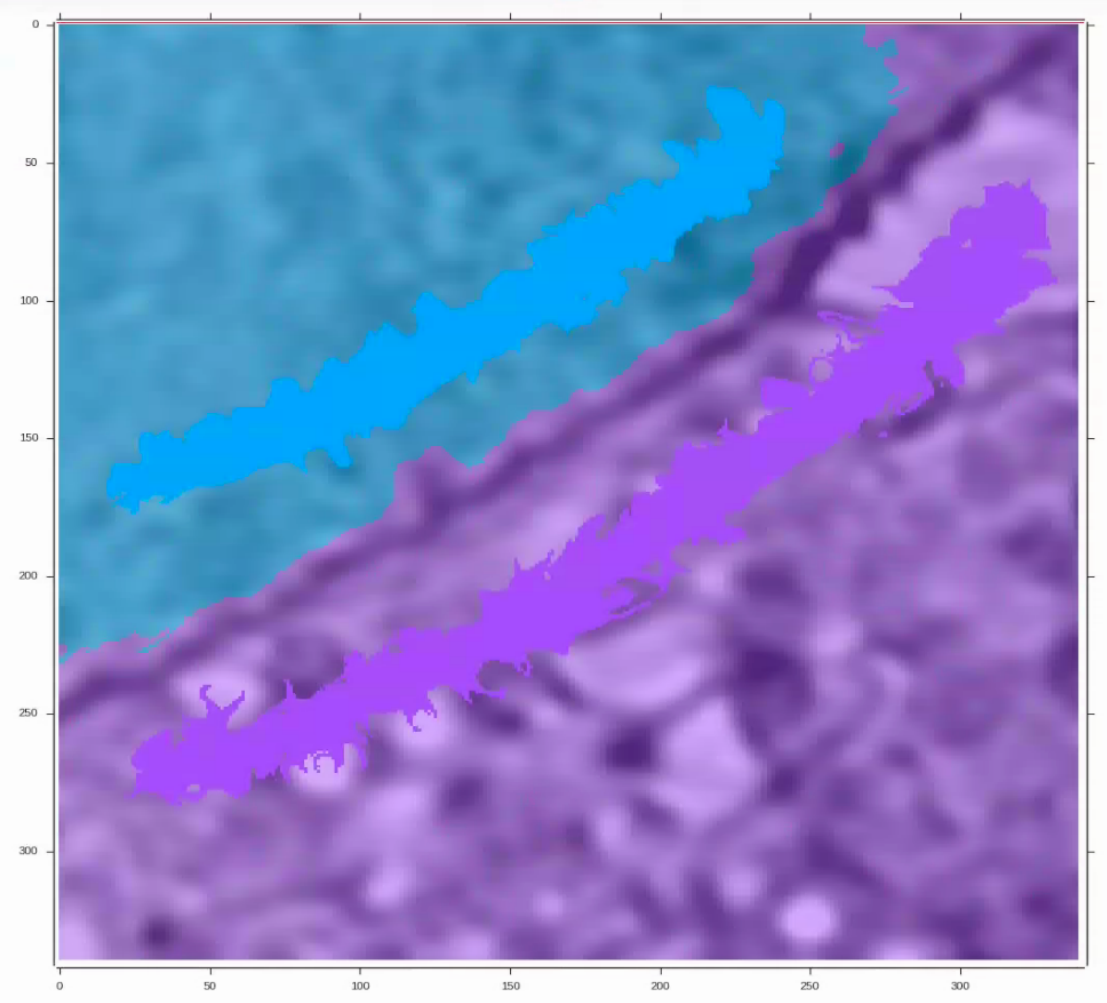
Classifier: Ensemble
Ensemble Type: Extra Random Forest (with defaults)
Refine: Appearance (50)

Model Training Options



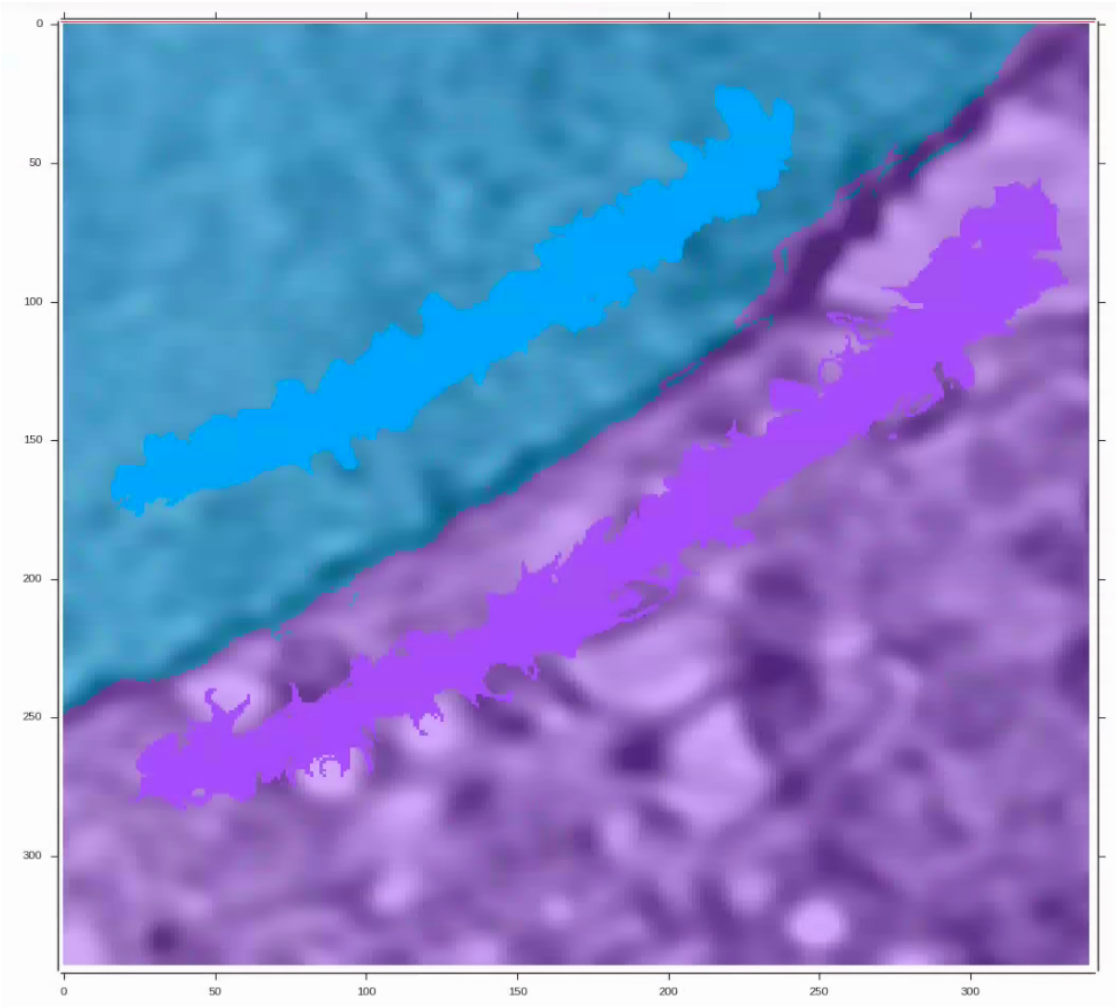
Classifier: SVM
Kernel Type: rbf (with defaults)
Refine: None

Model Training Options



Classifier: SVM
Kernel Type: rbf (with defaults)
Refine: Potts (50)

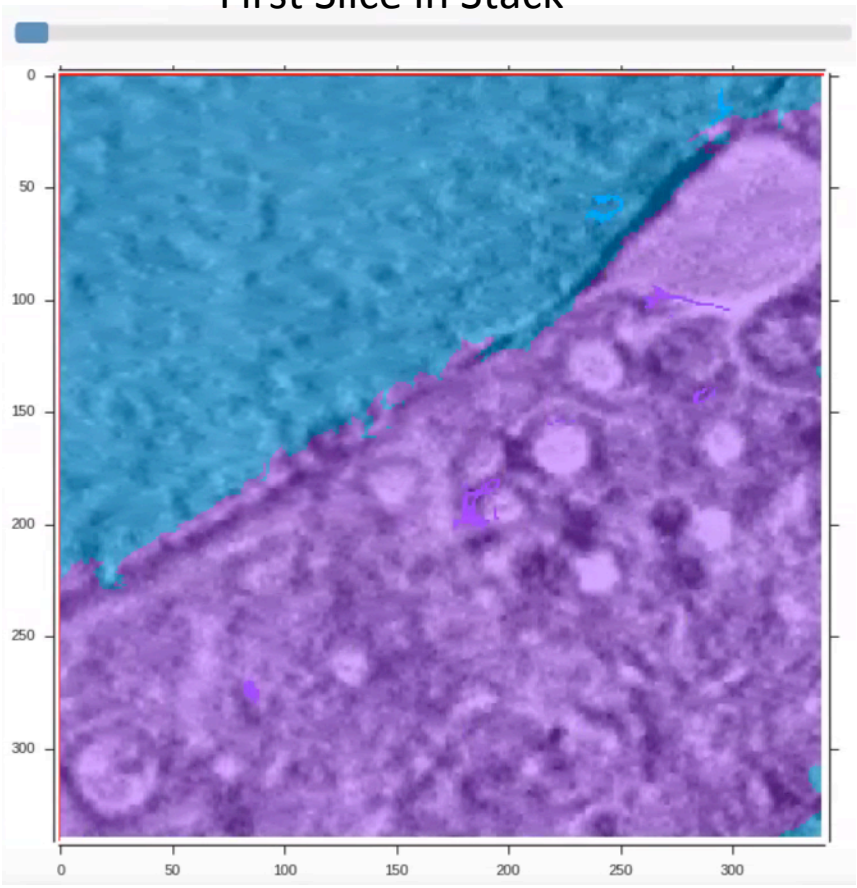
Model Training Options



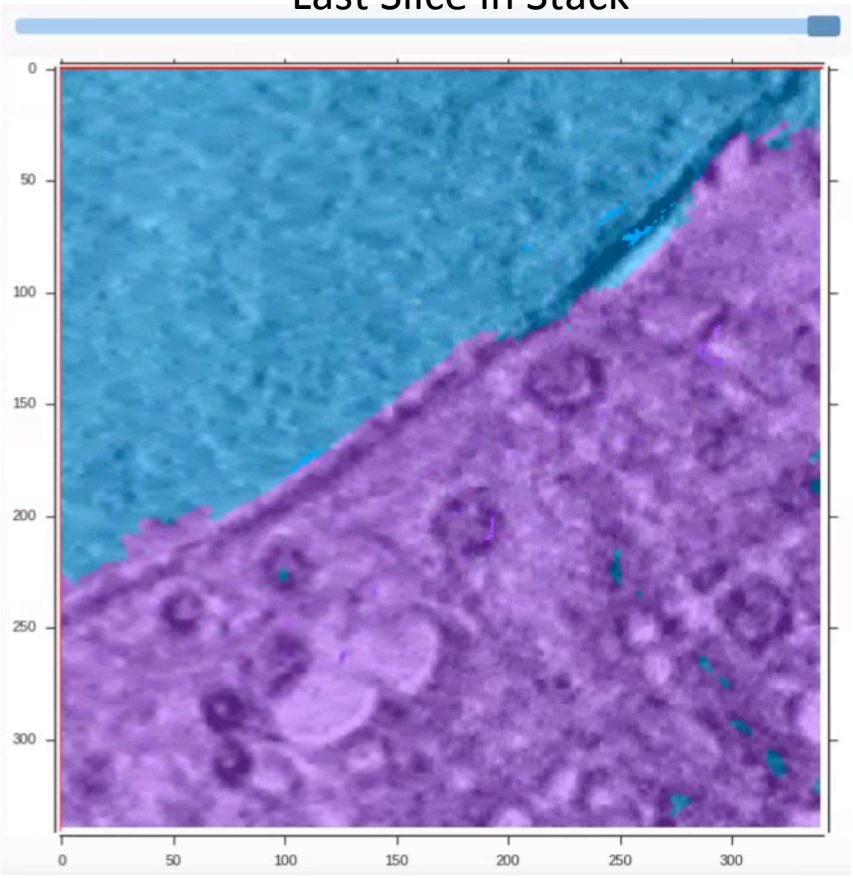
Classifier: SVM
Kernel Type: rbf (with defaults)
Refine: Appearance (50)

Model Training is in 3D

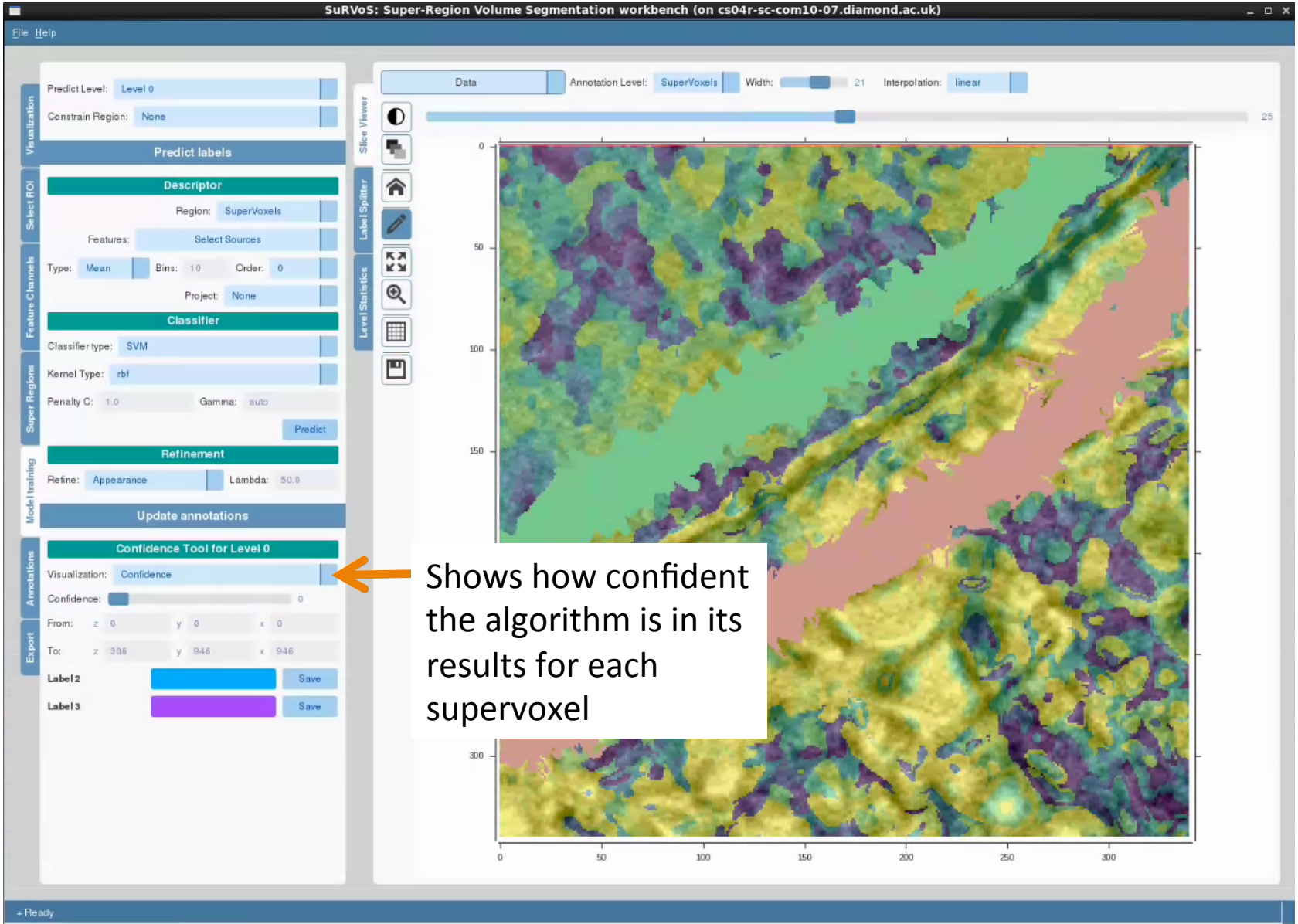
First Slice in Stack



Last Slice in Stack

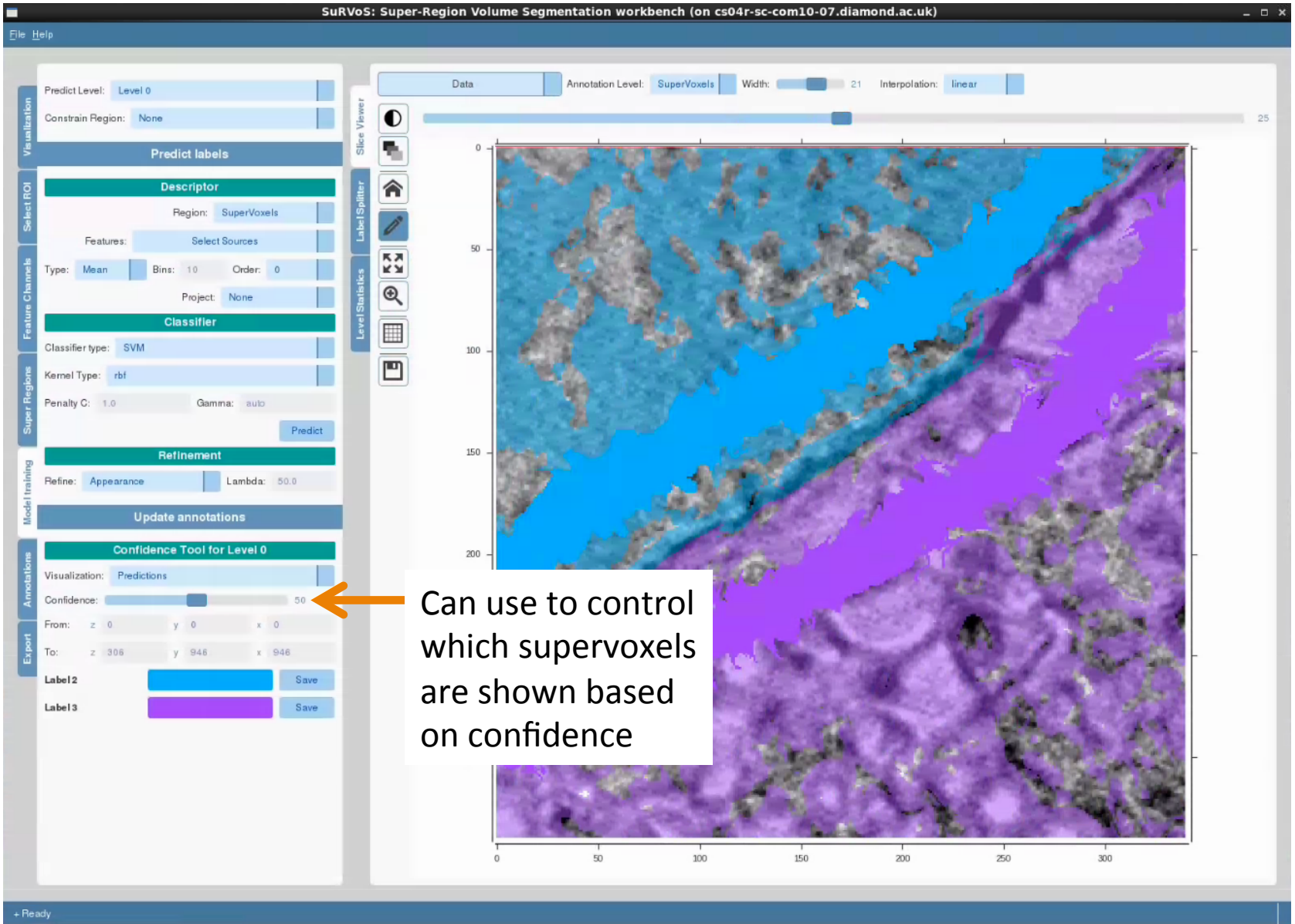


Model Training: Confidence and Update Annotations

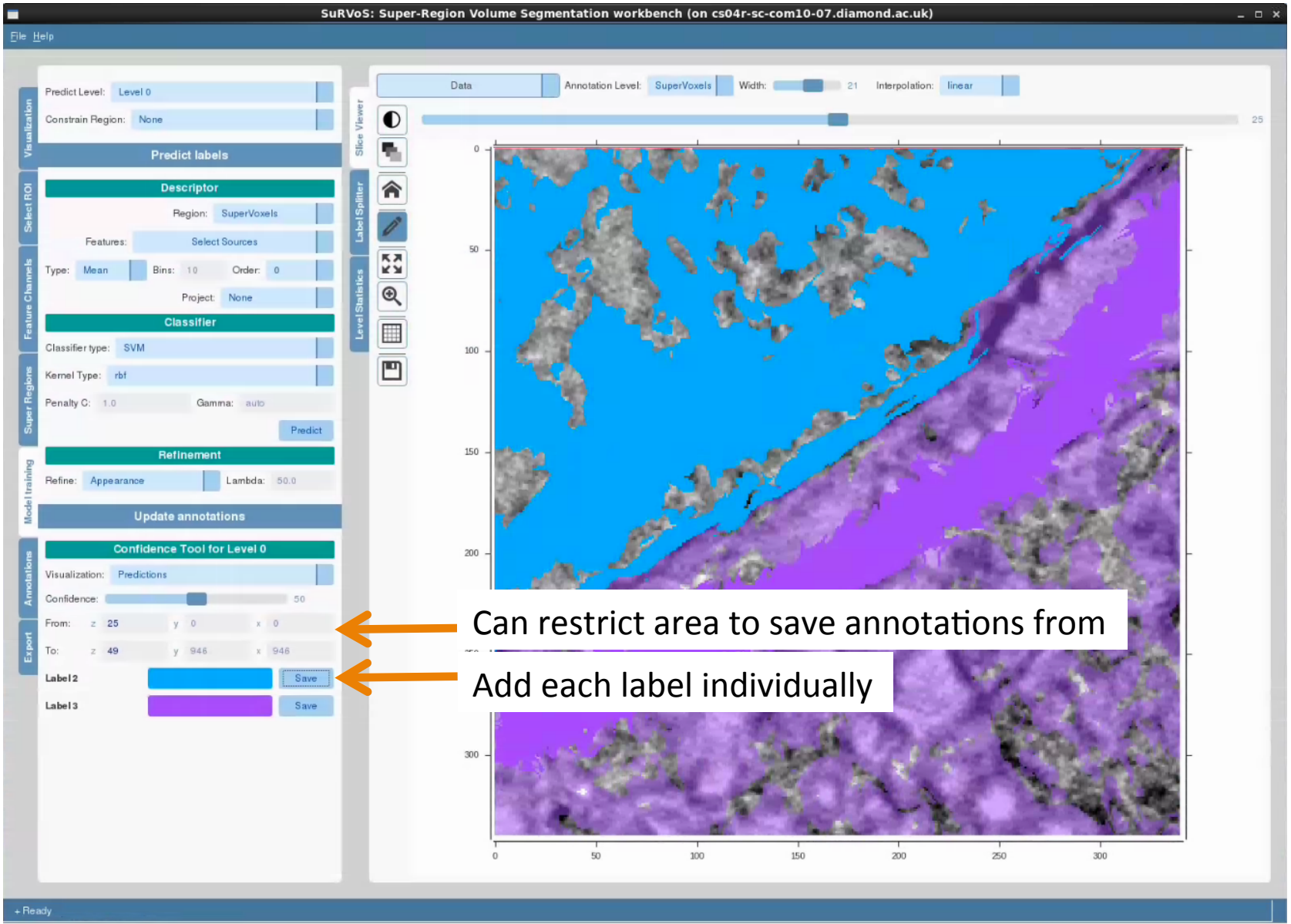


Shows how confident the algorithm is in its results for each supervoxel

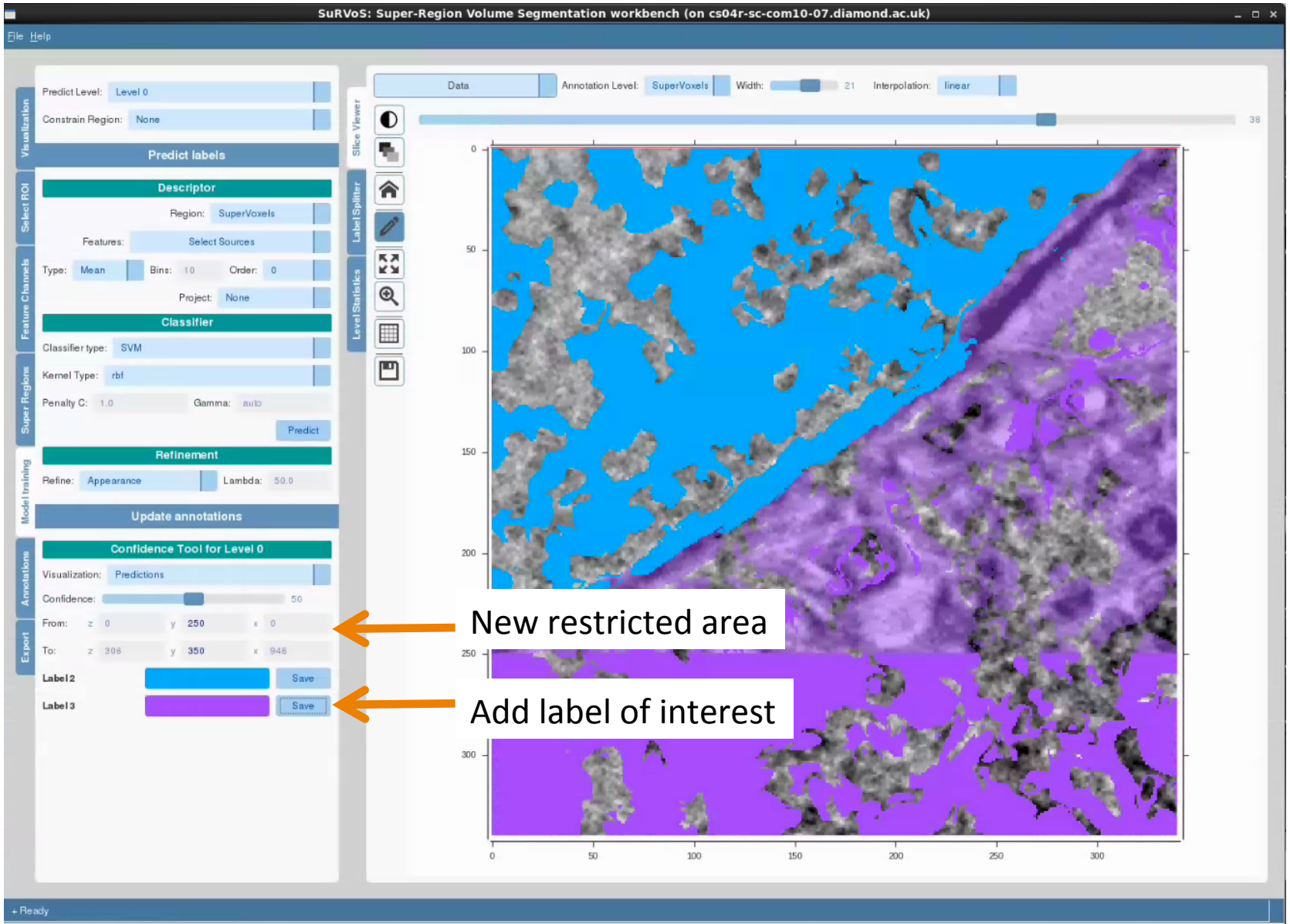
Model Training: Confidence and Update Annotations



Model Training: Confidence and Update Annotations

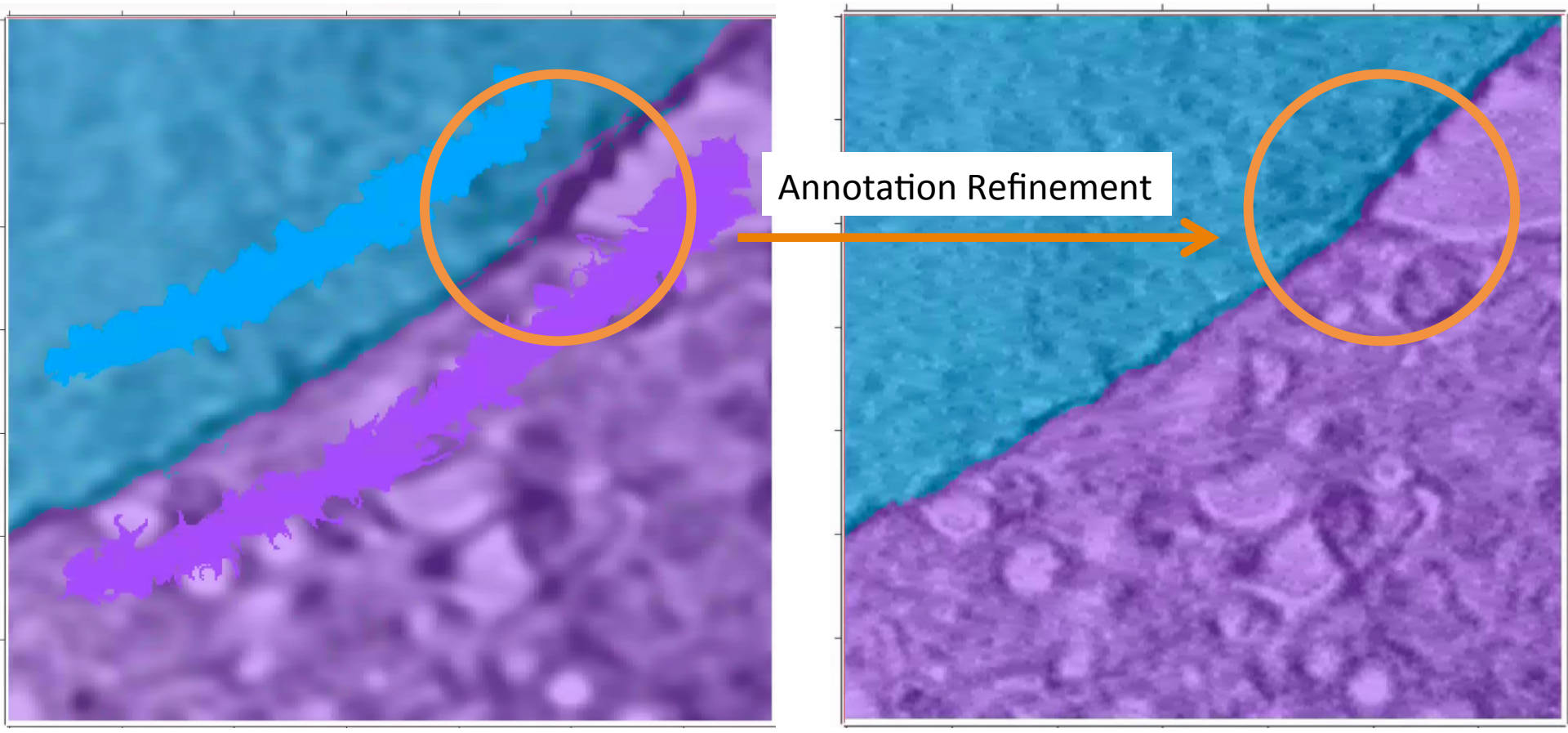


Model Training: Confidence and Update Annotations

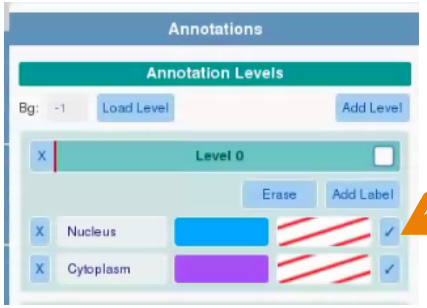


Model Training: Refining Annotations

Don’t worry about getting the model training perfect.
The annotations can be refined.



Model Training: Refining Annotations



In the annotation tab
select a label to refine.

We can refine a label on
- This slice
- All slices 2D (recommended)
- Whole volume 3D

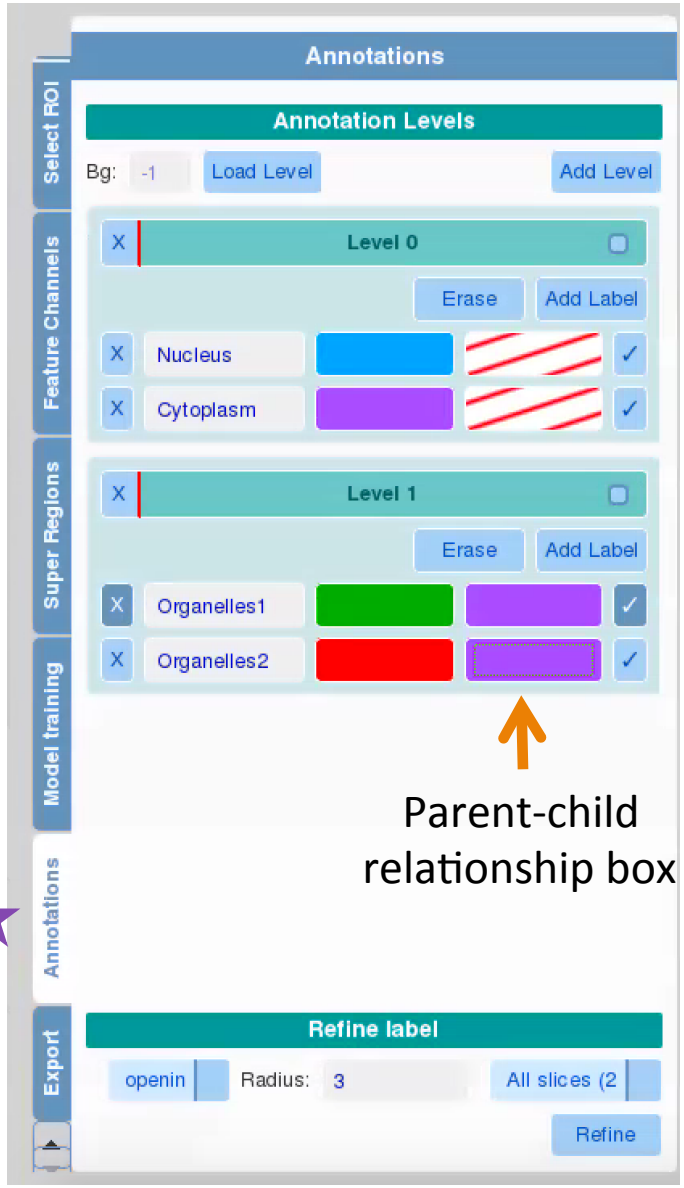
Refinement methods:

- **Dilation -** This adds pixels to the inner and outer boundary of the annotation.
- **Erosion -** This strips away layers of pixels from the inner and outer boundary of the annotation. Small objects can be deleted easily with erosion.
- **Opening** - Erosion followed by dilation.
- **Closing** - Dilation followed by erosion.
- **Fill holes** - Fills holes when a label surrounds them entirely.

Time – 15 minutes

Switch to Matt

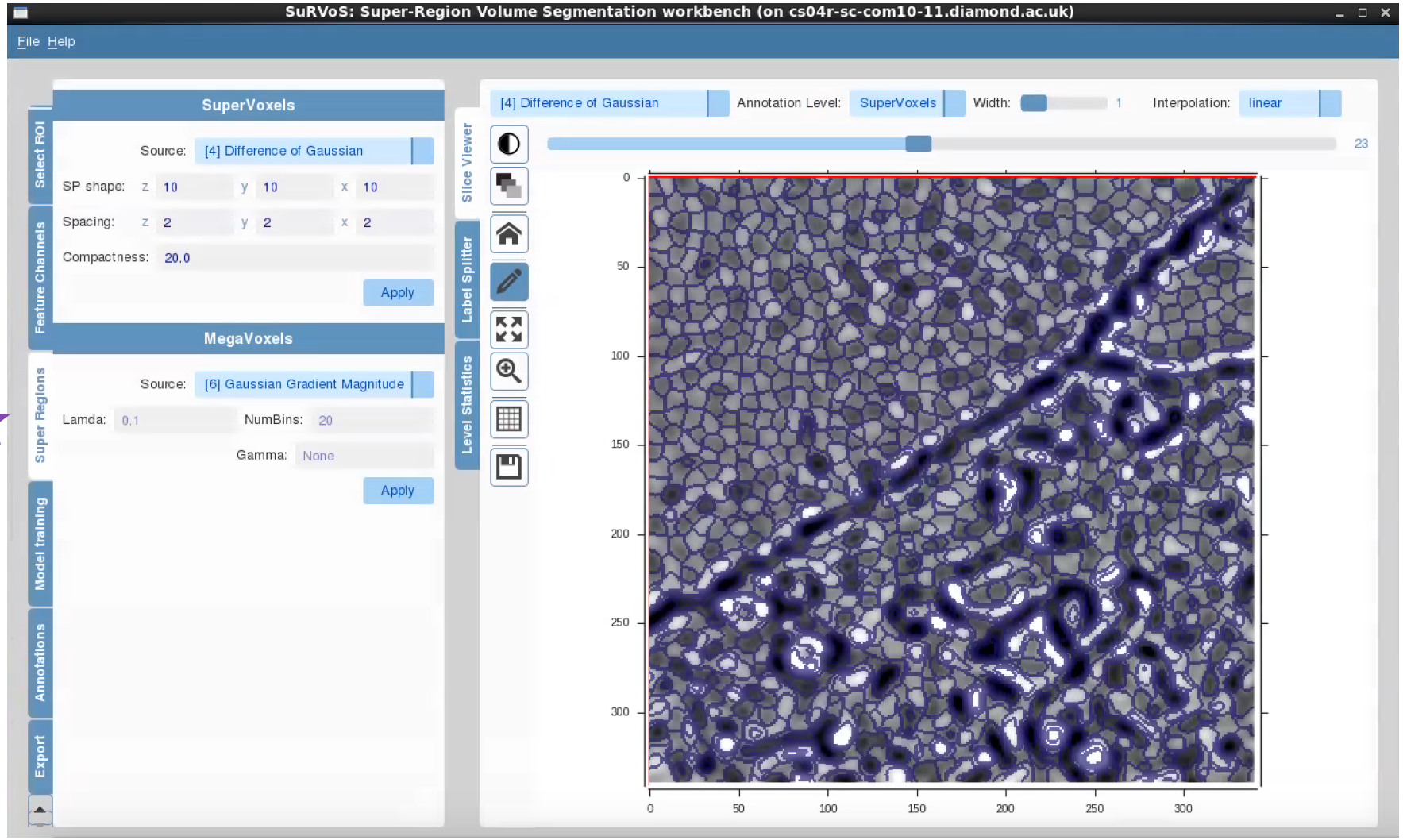
Parent-child Relationship Between Levels



- Add a new level
- Add labels to this level
- A parent of the label can be specified by clicking on the parent-child relationship box for each label
- In the example shown (left) we are wanting to segment organelles which are located in the cytoplasm, so we choose the parent label: Level 0/Cytoplasm

Note: Objects next to each other must be different labels to be considered different objects!

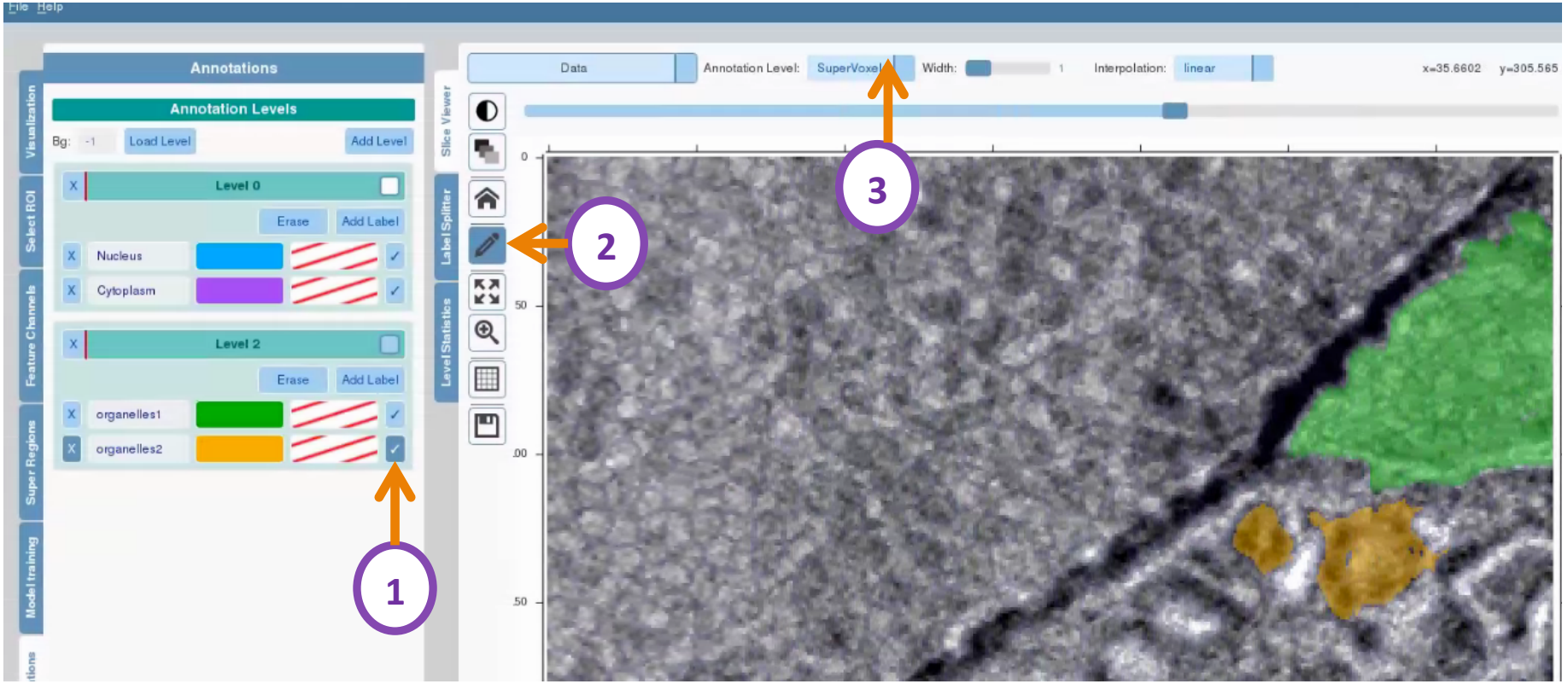
Choosing New Supervoxels for Organelle Segmentation



- **Supervoxels can be recalculated at anytime, using any source**.
- This is useful for generating supervoxels that represent specific organelles.

Time – 5 minutes

Segmenting Organelles Using Supervoxels

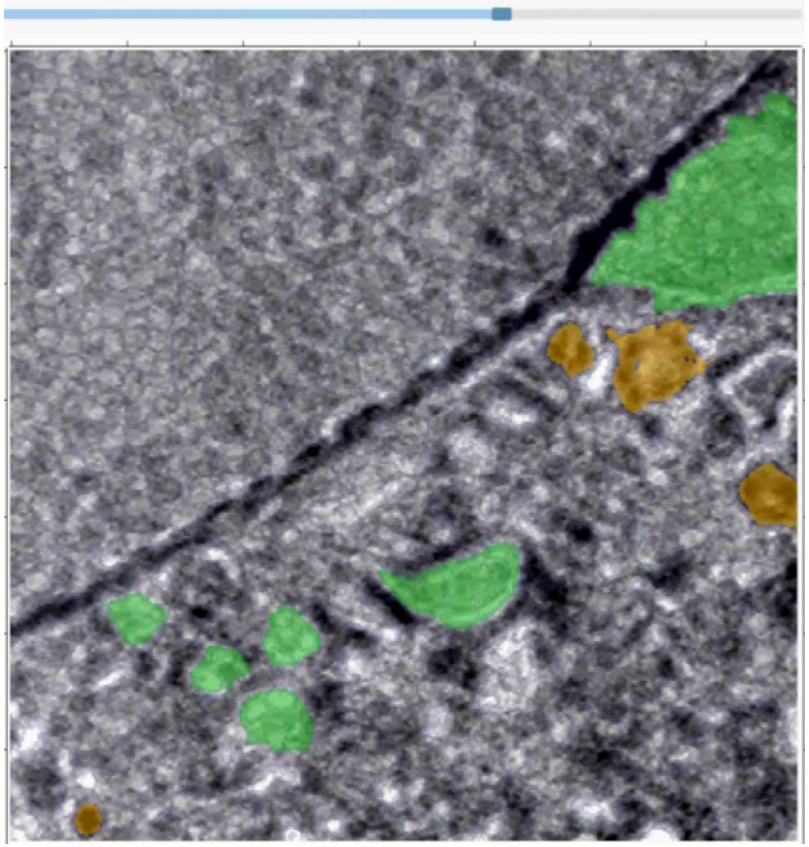


Reminder:

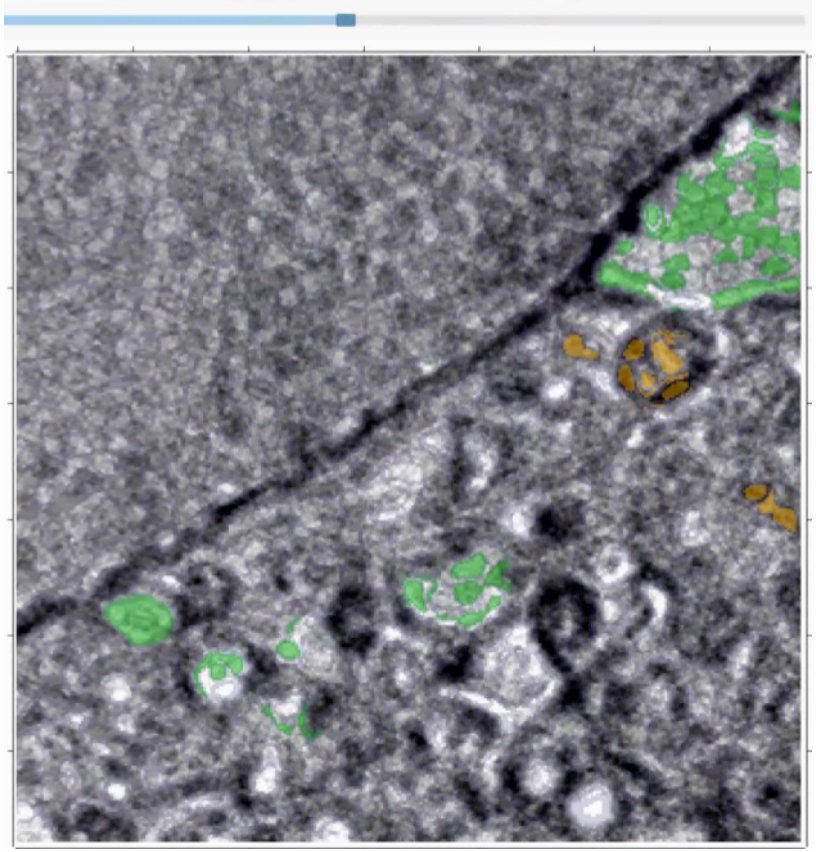
1) Select the label to annotate with
2) Select the pen tool
3) Select annotation level to supervoxels and left click to annotate

Segmenting Organelles Using Supervoxels

Segmented Z-slice

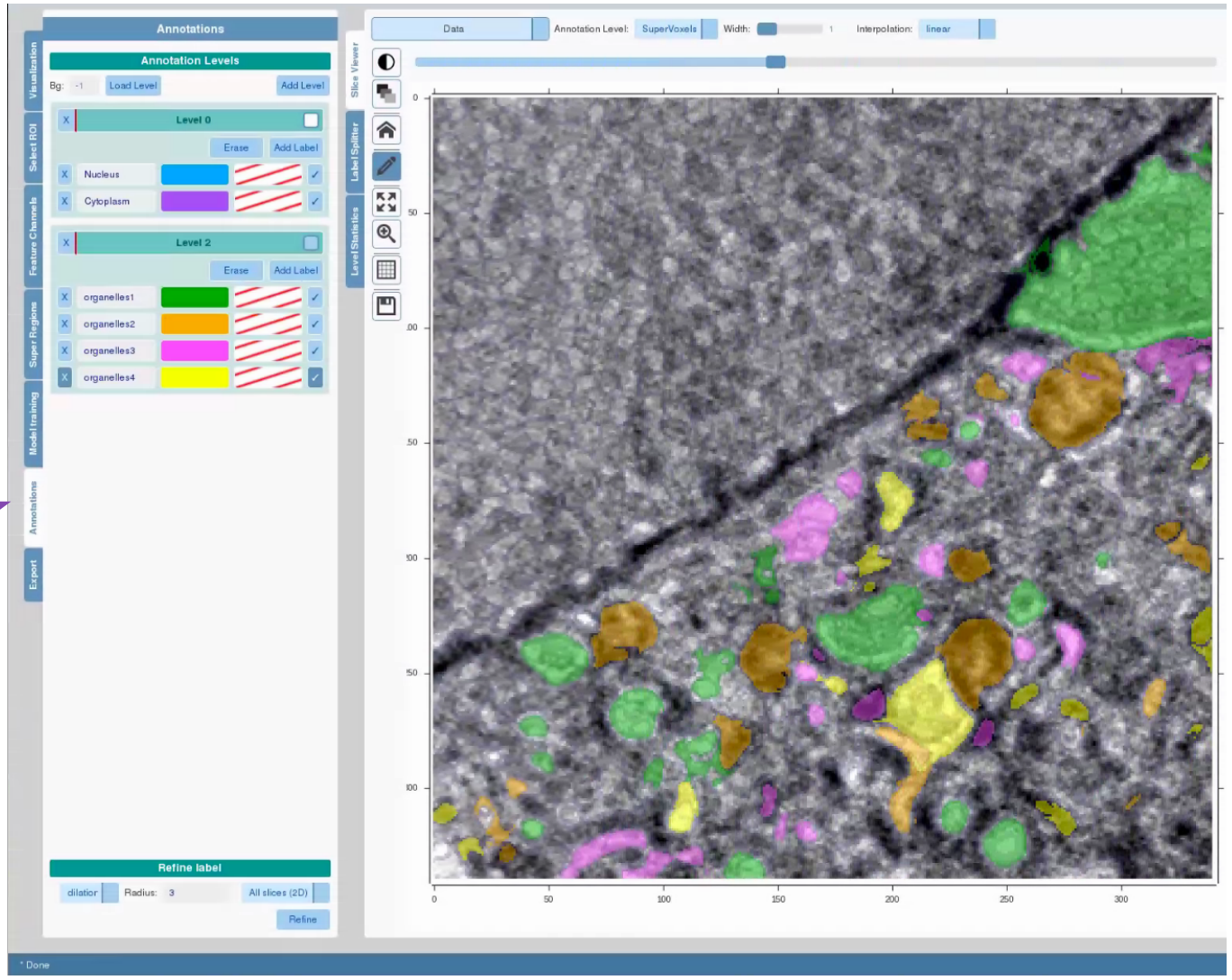


10 slices below



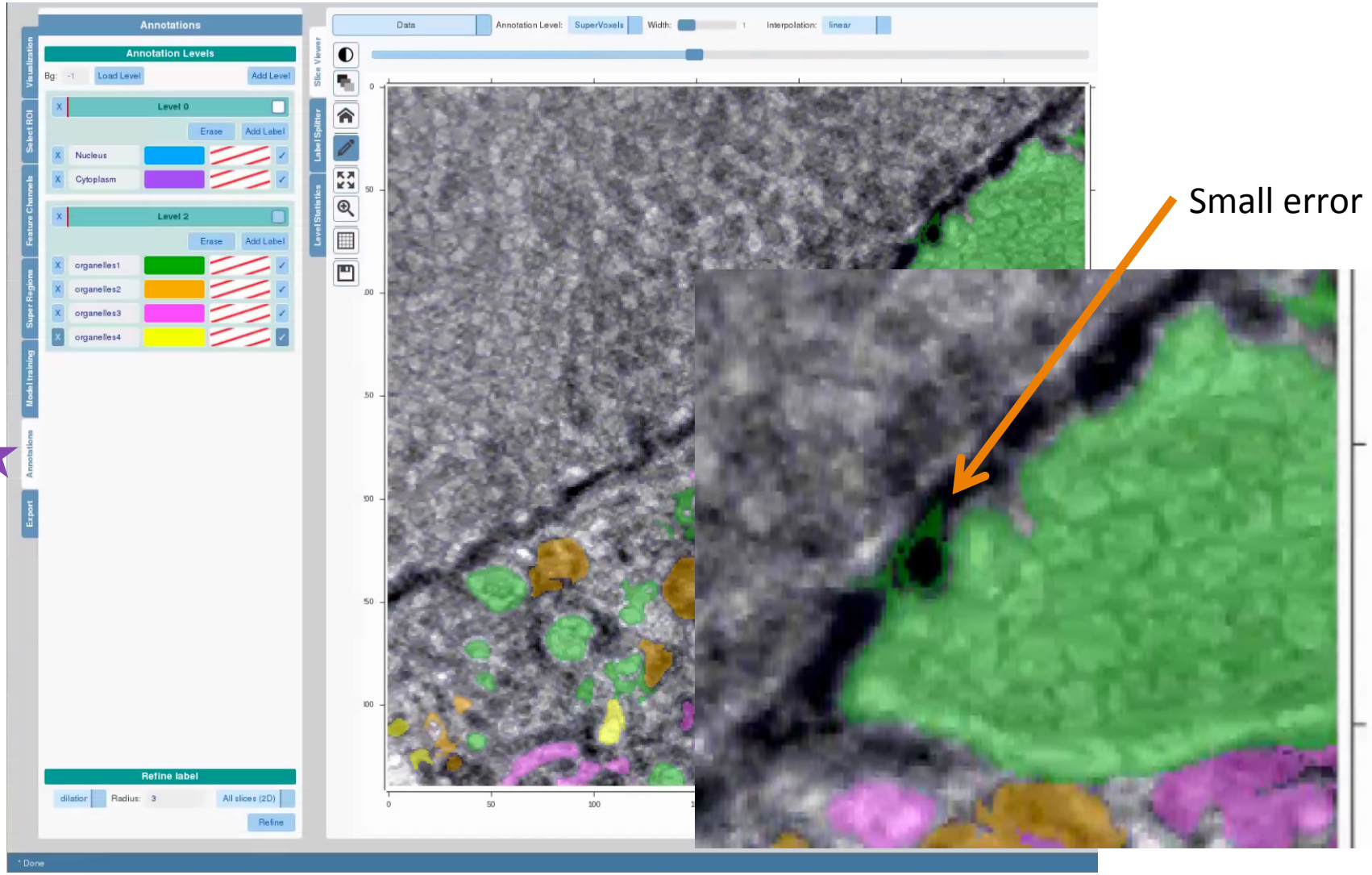
- Segment organelles in one slice
- Move through a few slices in Z
- Re-annotate keeping organelles the same colour

Segmenting Organelles Using Supervoxels



Once happy that the organelles have been segmented apart from some small errors then the annotation can be refined.

Segmenting Organelles Using Supervoxels



Once happy that the organelles have been segmented apart from some small errors then the annotation can be refined.

Segmenting Organelles: Refinement

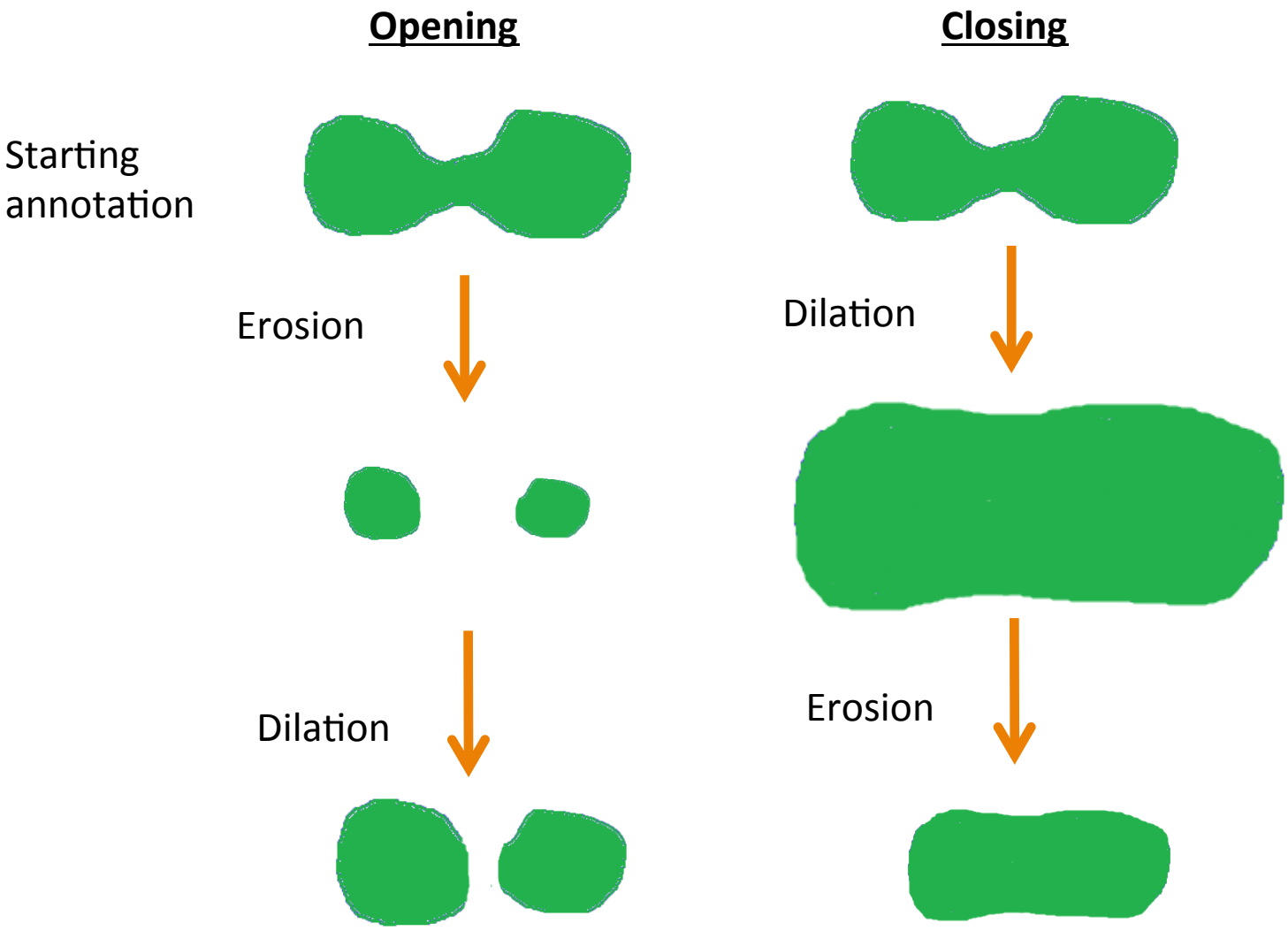
In the annotation tab
select a label to refine.

We can refine a label on
- This slice
- All slices 2D
- Whole volume 3D

Refinement methods:

- **Dilation -** This adds pixels to the inner and outer boundary of the annotation.
- **Erosion -** This strips away layers of pixels from the inner and outer boundary of the annotation. Small objects can be deleted easily with erosion.
- **Opening** - Erosion followed by dilation.
- **Closing** - Dilation followed by erosion.
- **Fill holes** - Fills holes when a label surrounds them entirely.

Segmenting Organelles: Refinement



Segmenting Organelles: Refinement

General Suggested Procedure:

1) Fill holes - typically use a large radius

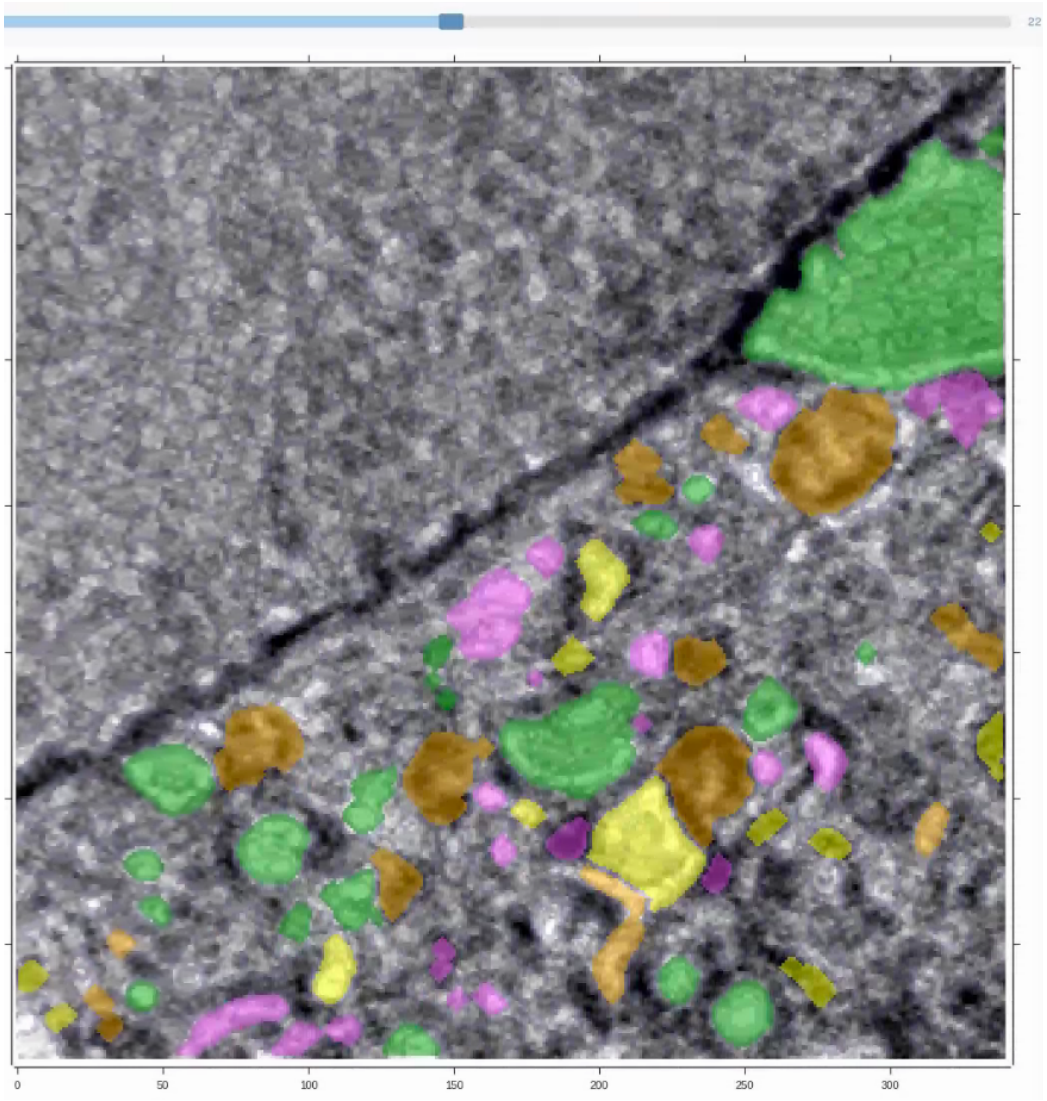
2) Opening - Radius 5

3) Dilation - Radius 2-3

4) Erosion - Radius 1-2

Do each step on each similar label before moving on to next step.

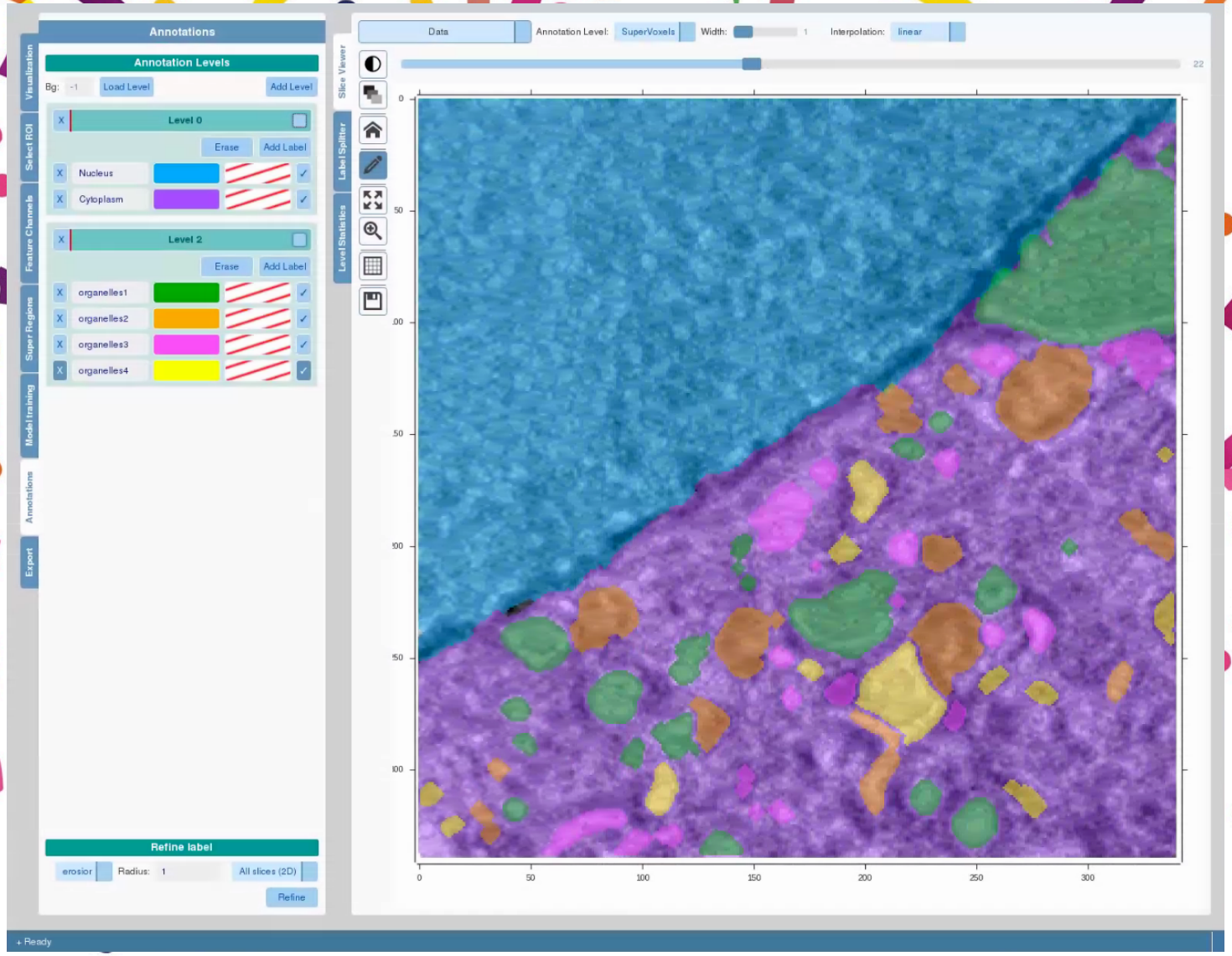
Can check the effects of each step on an individual slice to optimize parameters



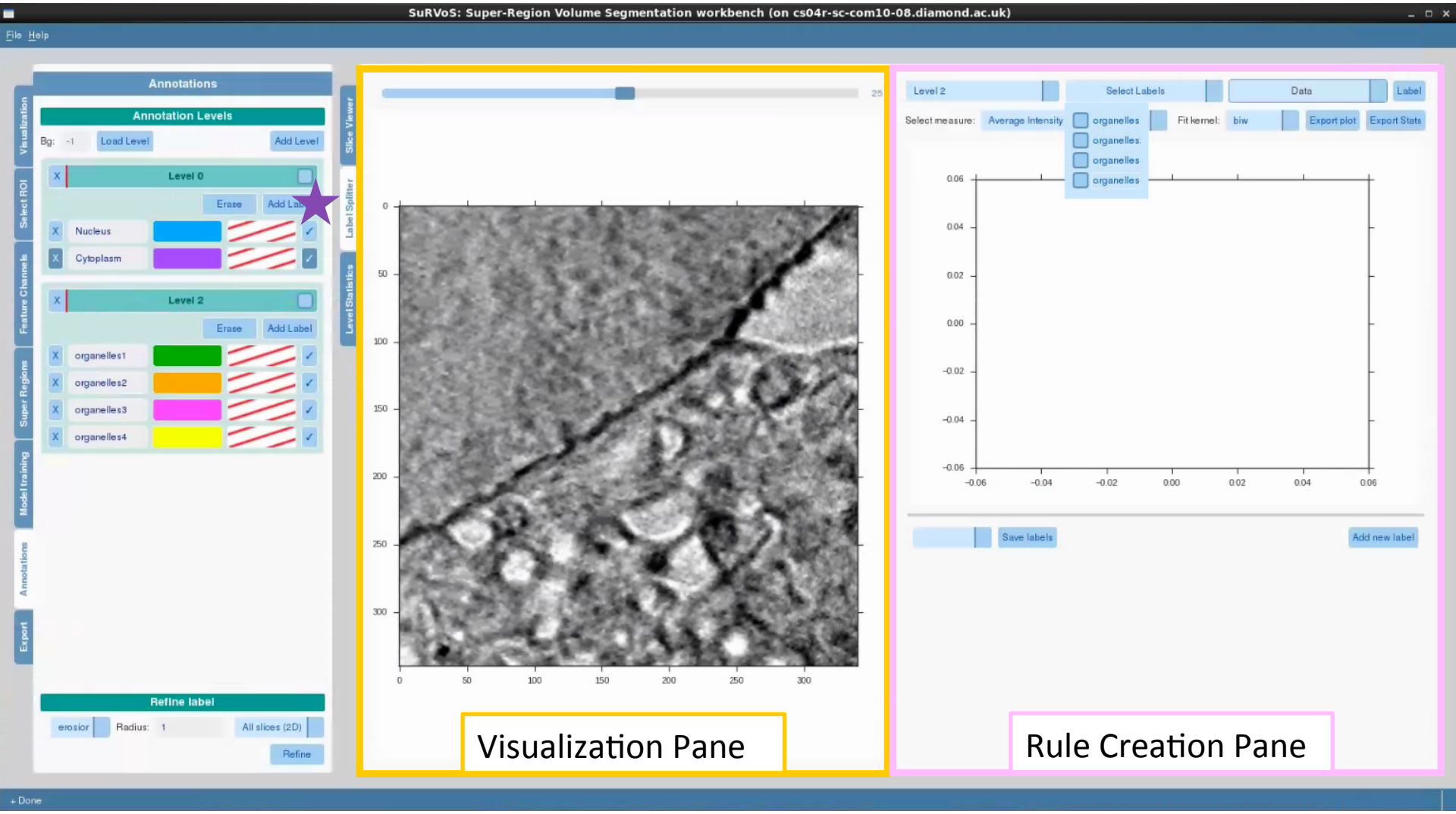
Time – 25 minutes

Switch to Michele

Finished Segmentation

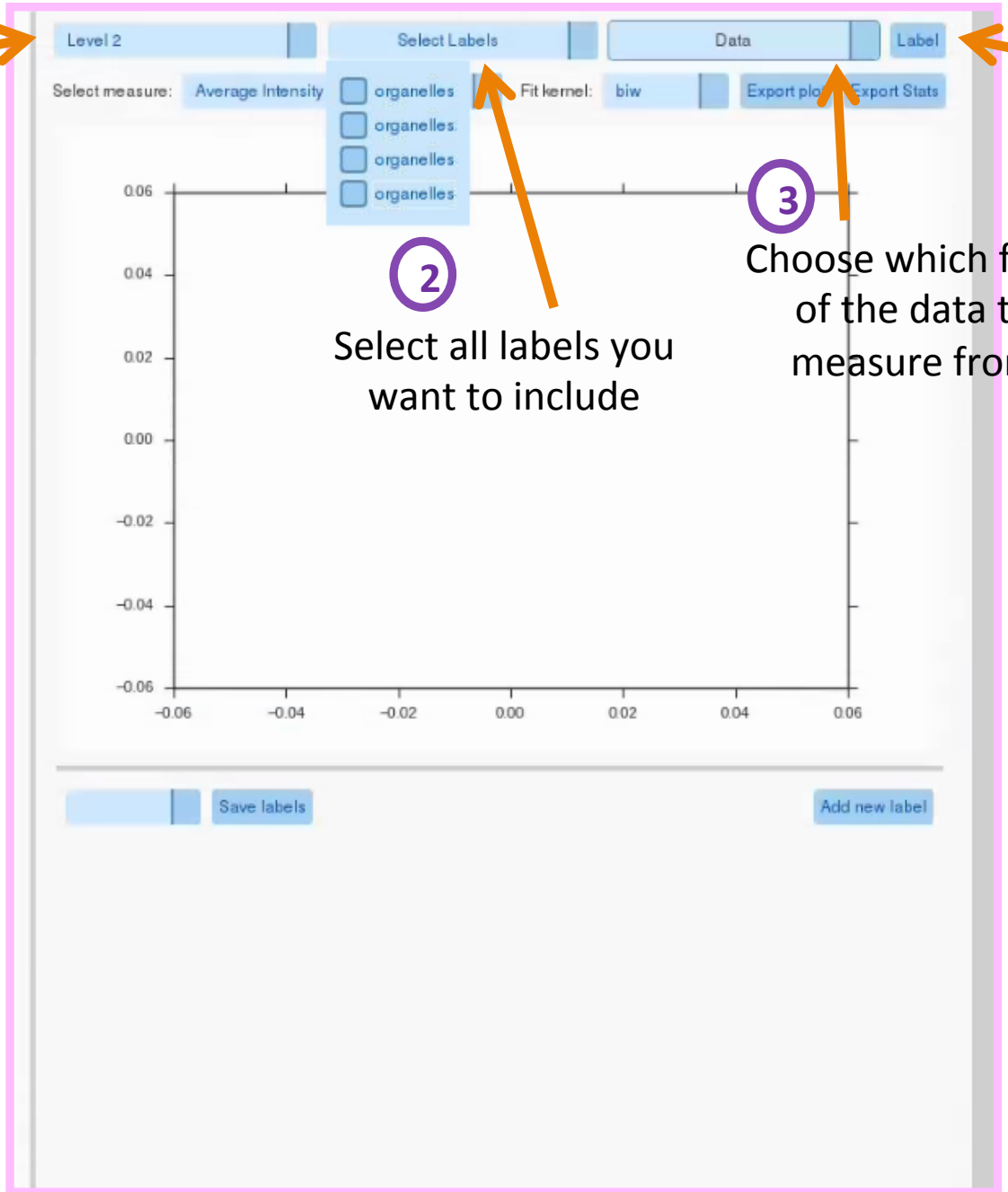


Label Splitter



Label Splitter: Rule Creation

1. Choose level you will work with

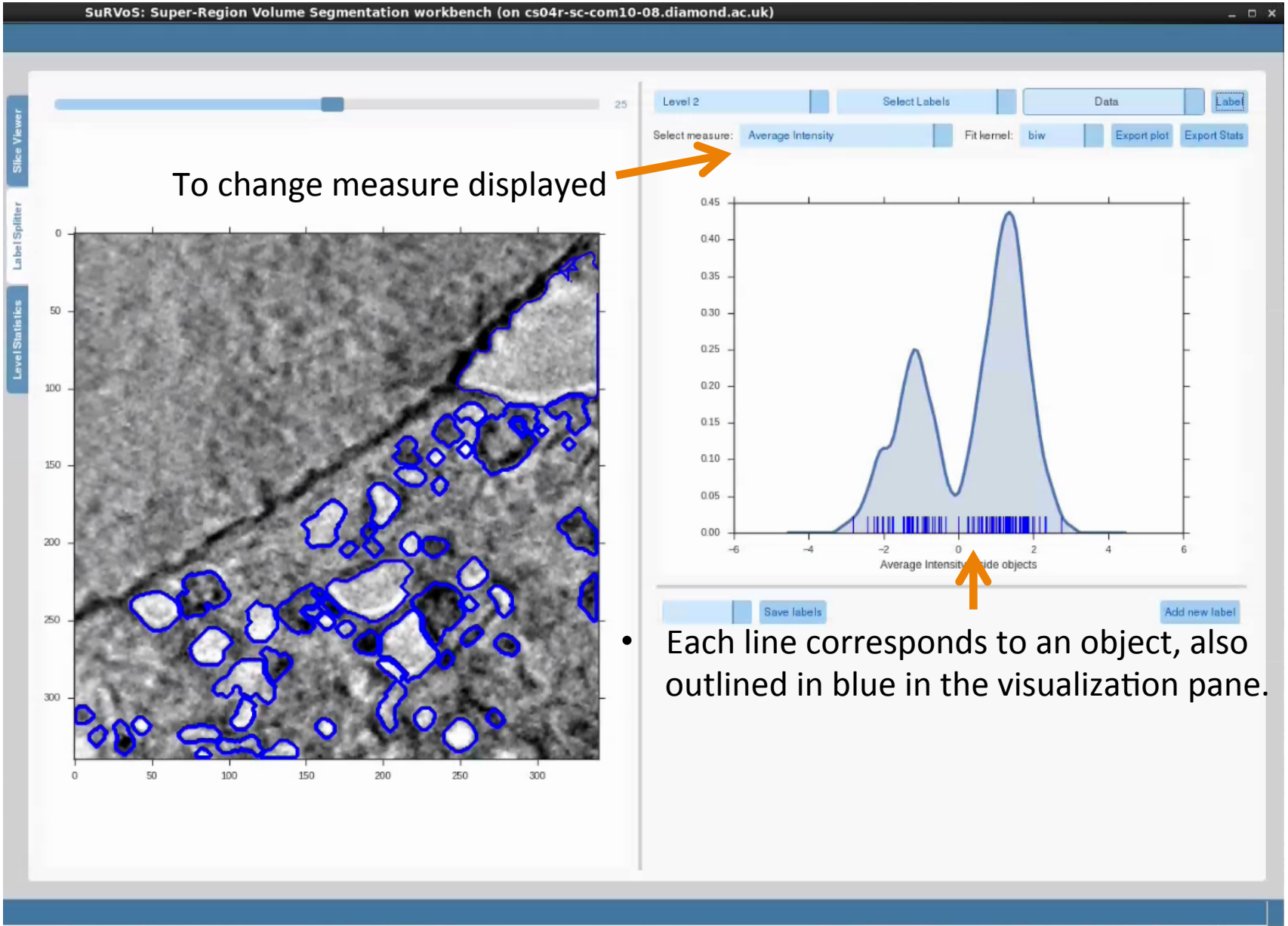


2. Select all labels you want to include

3. Choose which form of the data to measure from

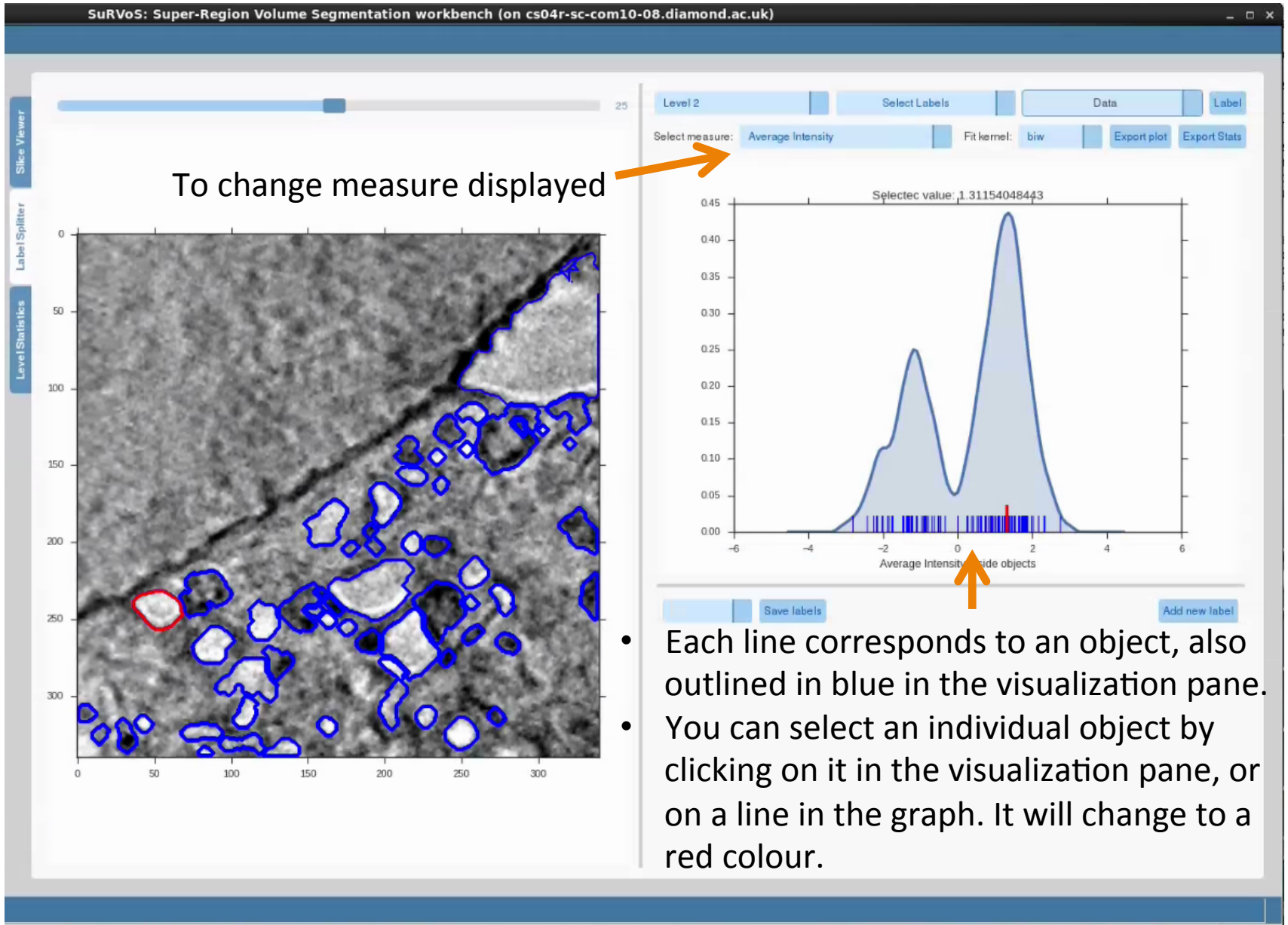
4. Click Label

Label Splitter: Rule Creation



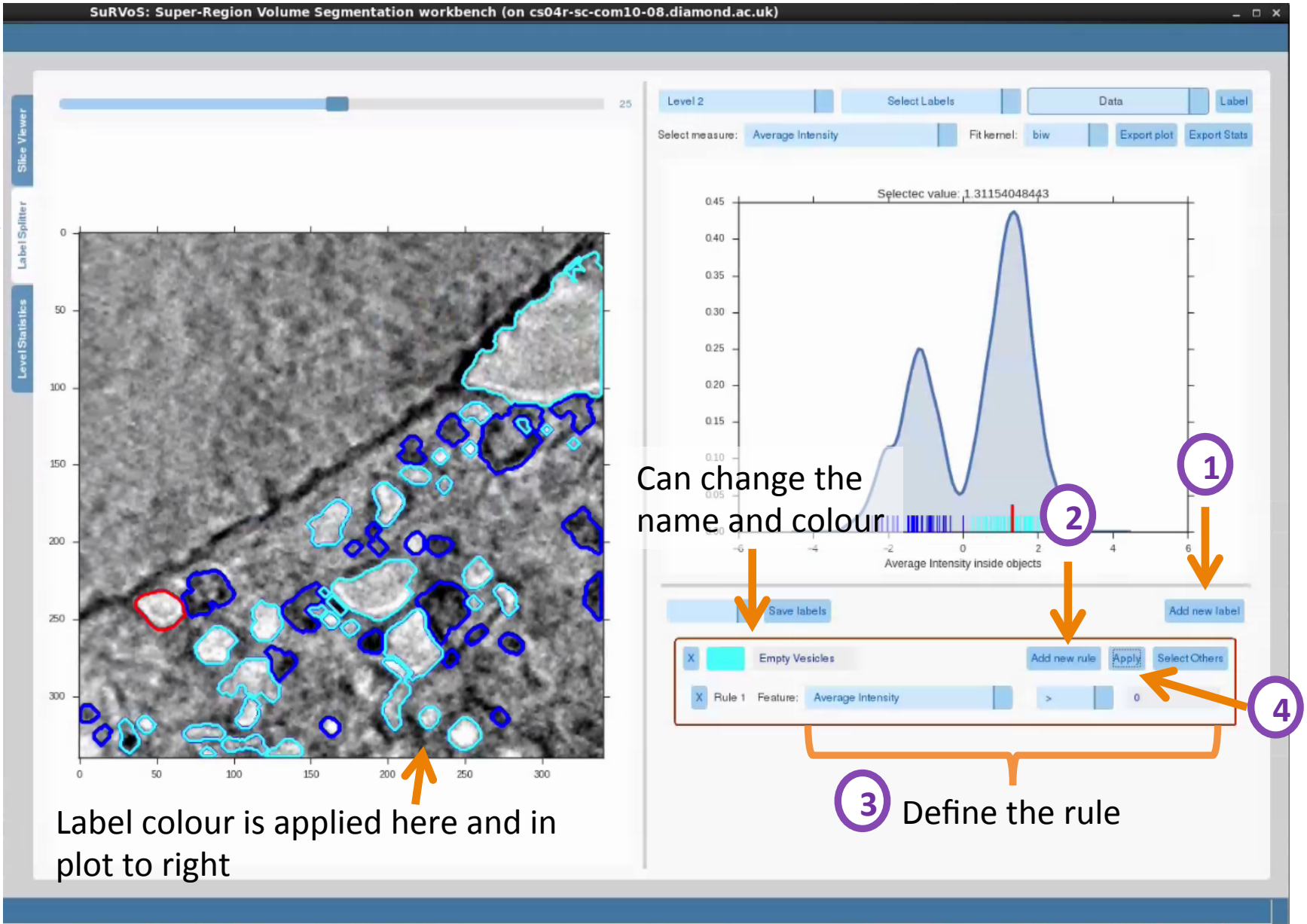
- Each line corresponds to an object, also outlined in blue in the visualization pane.

Label Splitter: Rule Creation

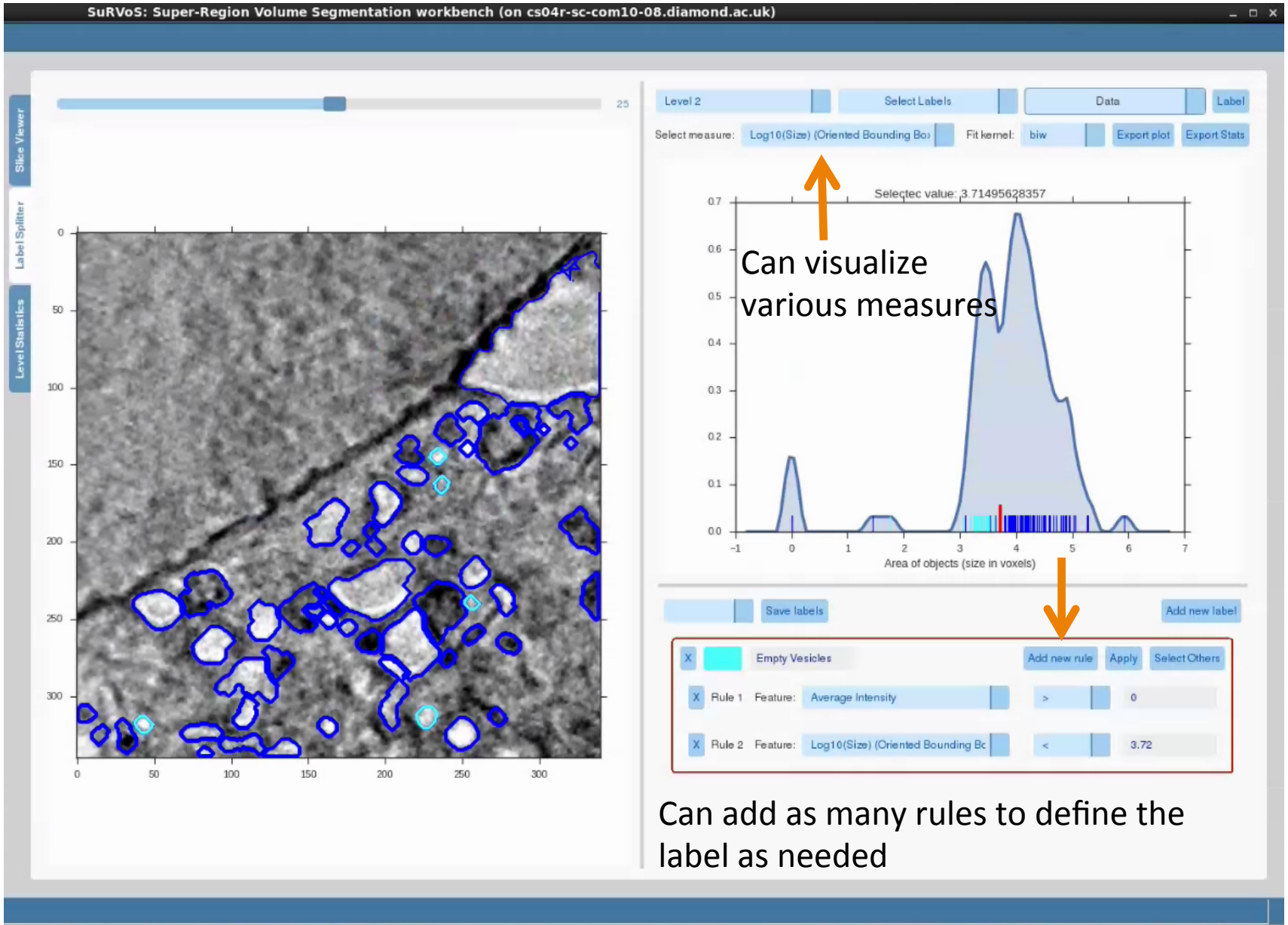


- Each line corresponds to an object, also outlined in blue in the visualization pane.
- You can select an individual object by clicking on it in the visualization pane, or on a line in the graph. It will change to a red colour.

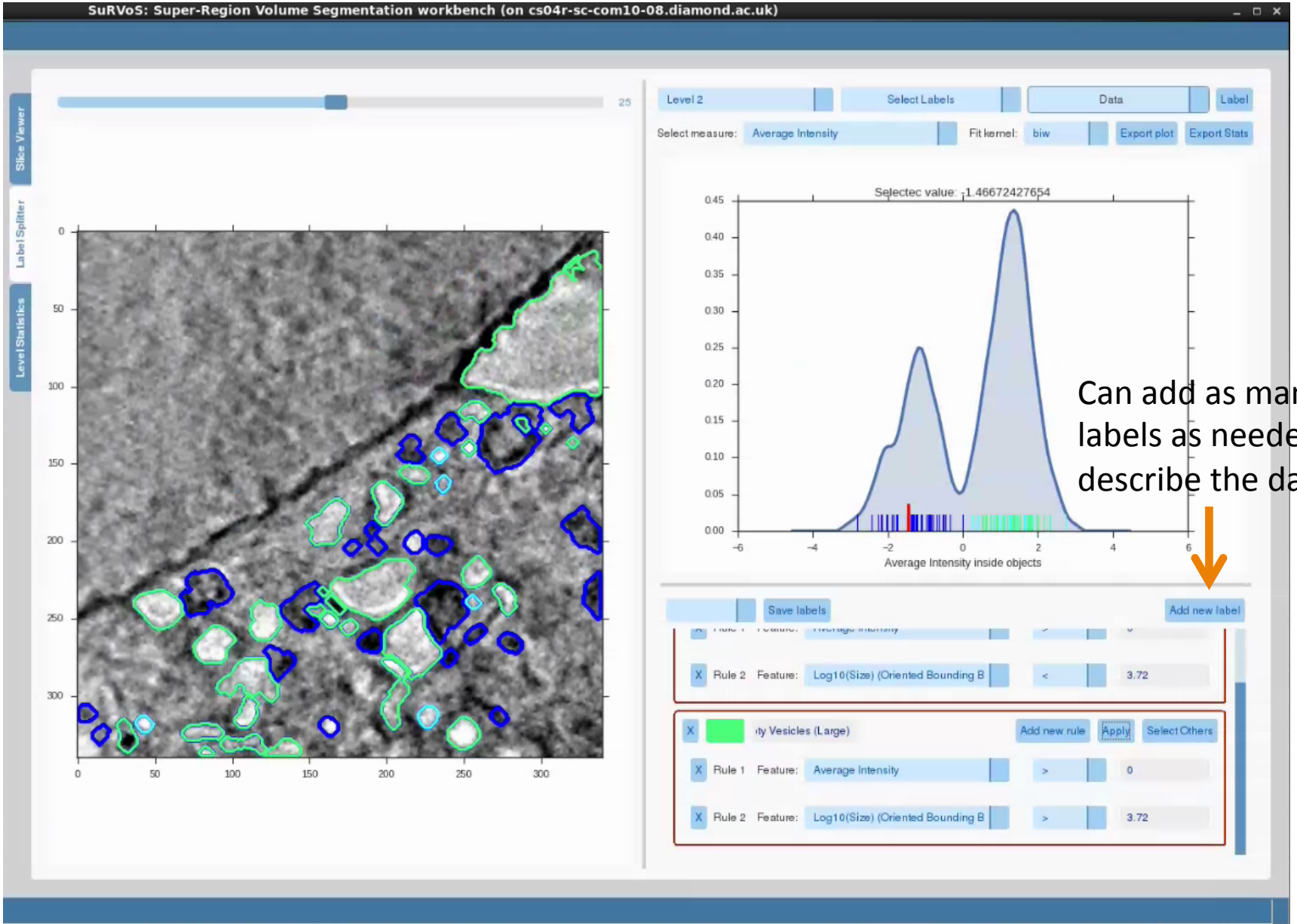
Label Splitter: Rule Creation



Label Splitter: Rule Creation

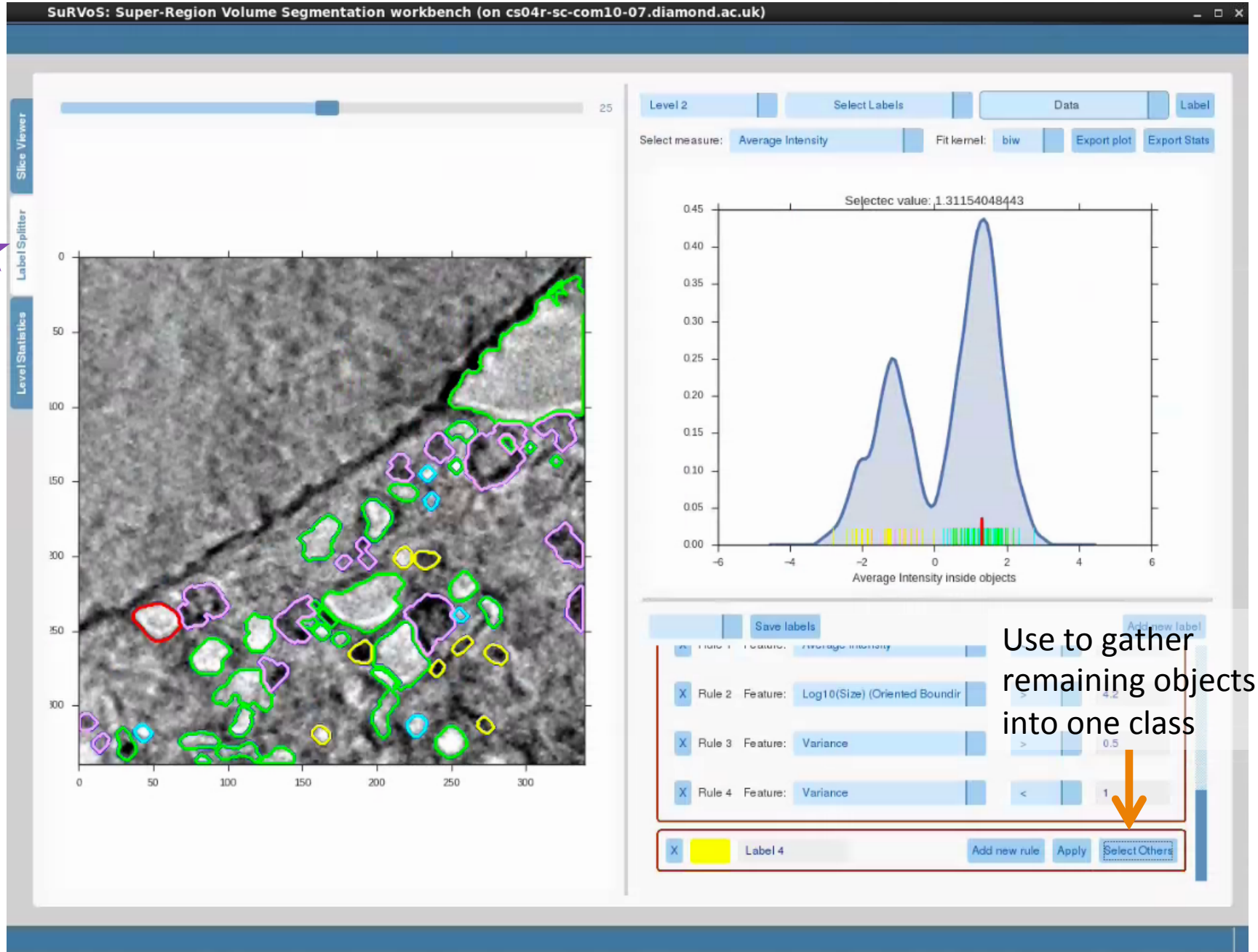


Label Splitter: Rule Creation

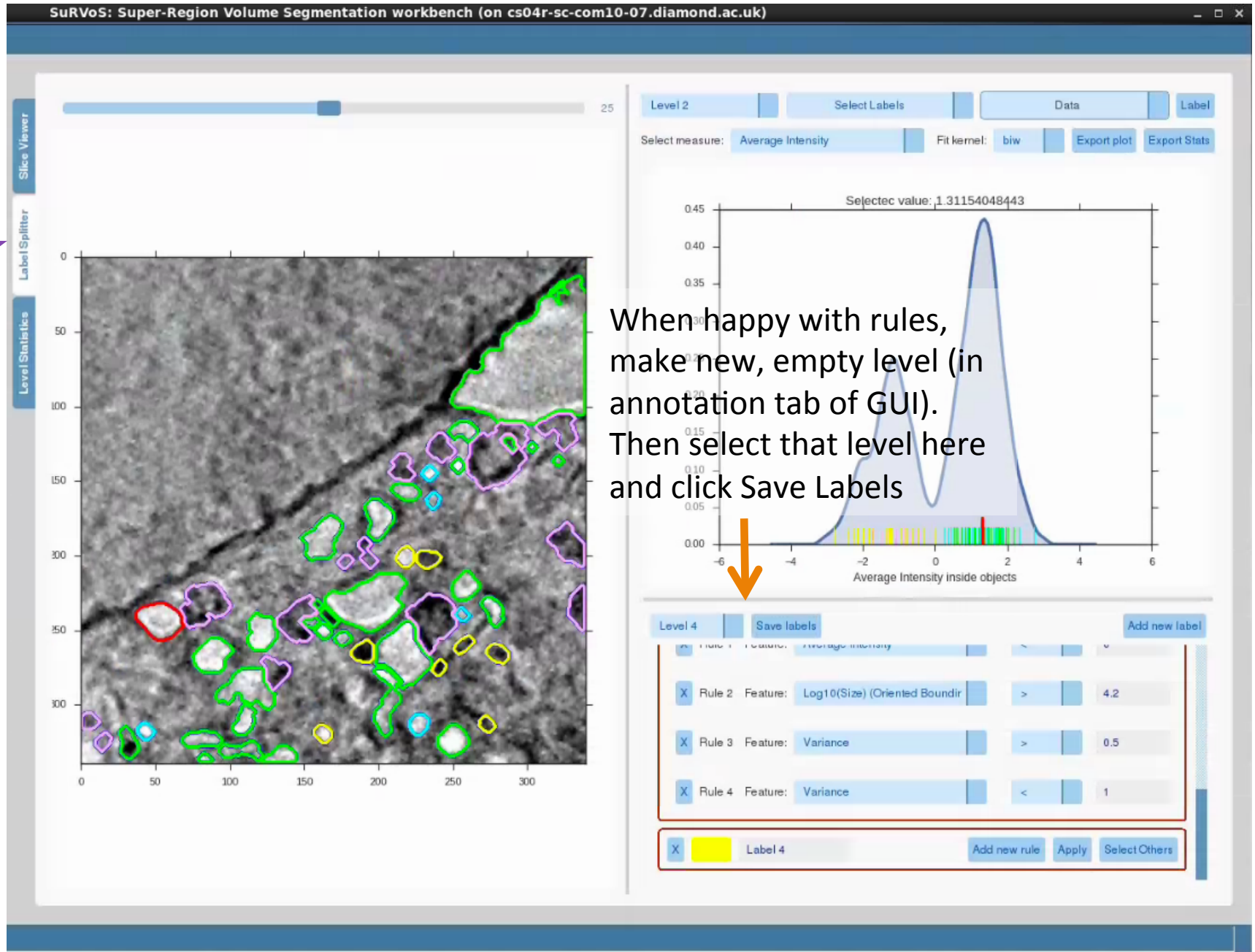


Can add as many labels as needed to describe the data

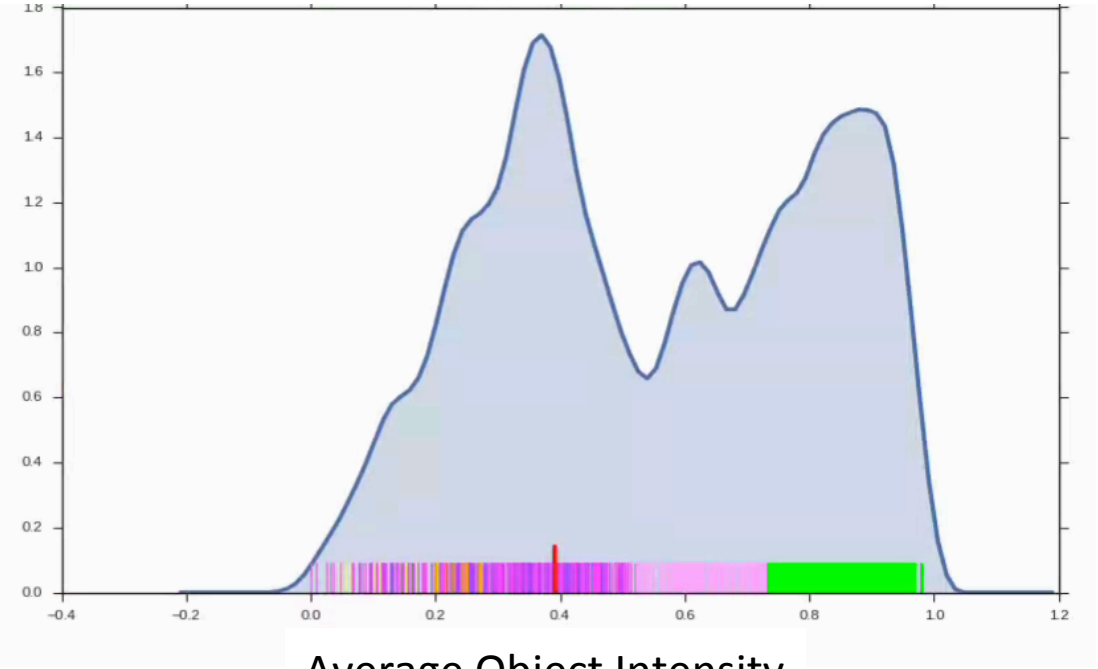
Label Splitter: Rule Creation



Label Splitter: Rule Creation



Label Splitter: Rule Creation



Average Object Intensity

Measures:

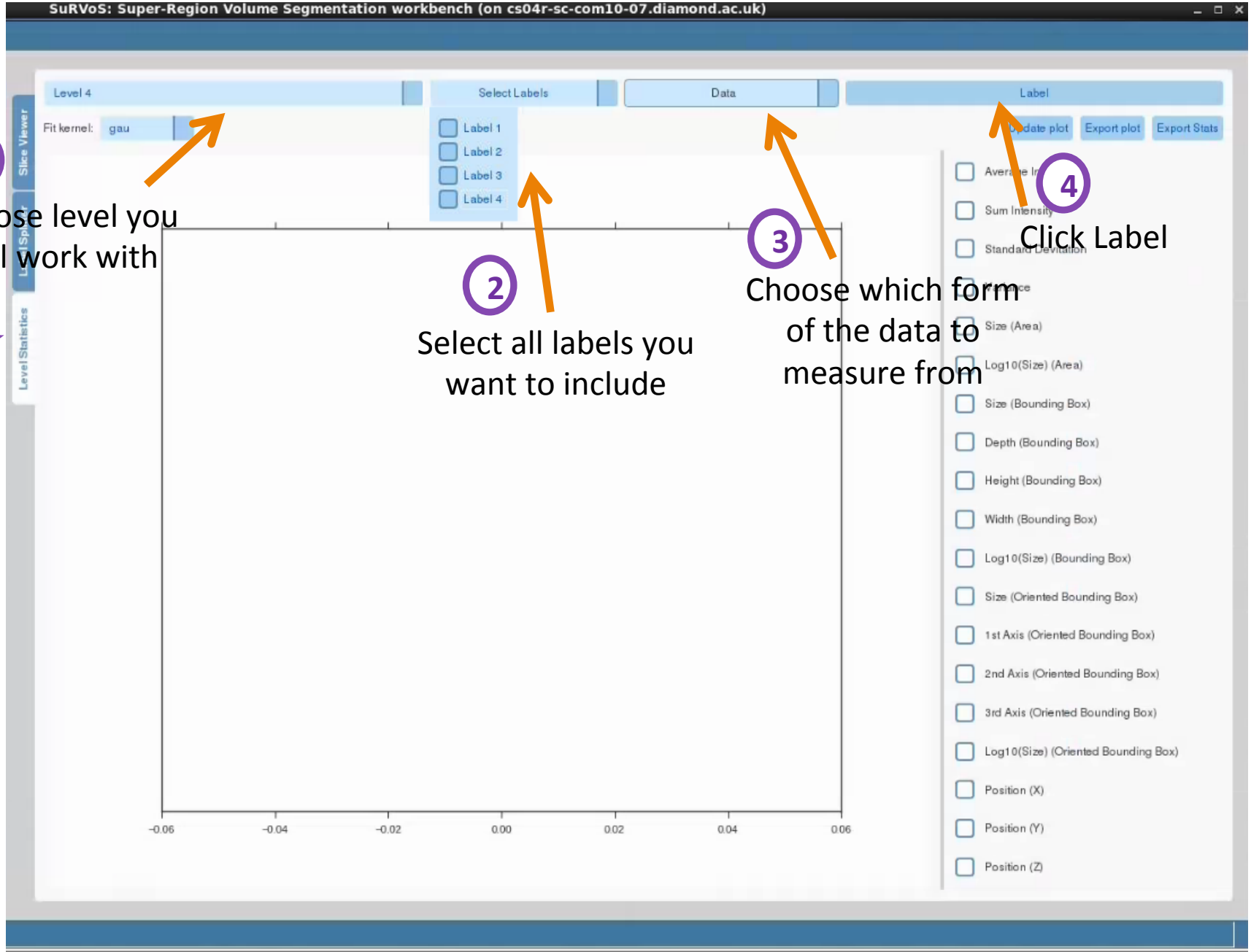
- Average Object Intensity
- Sum of Object Intensity
- Standard Deviation of Object Intensity
- Variance of Object Intensity
- Area of Object
- Bounding Box of Object
  - Depth, width, height
- Oriented Bounding Box of Object
  - Depth, width, height
- X, Y, Z positions of Object

Two Ways to Use:

1. Use these characteristics to create rules to separate features into classes.
2. Use other methods to separate features into classes and then use characteristics to describe classes.

Time – 15 minutes

Label Statistics



1 Choose level you will work with

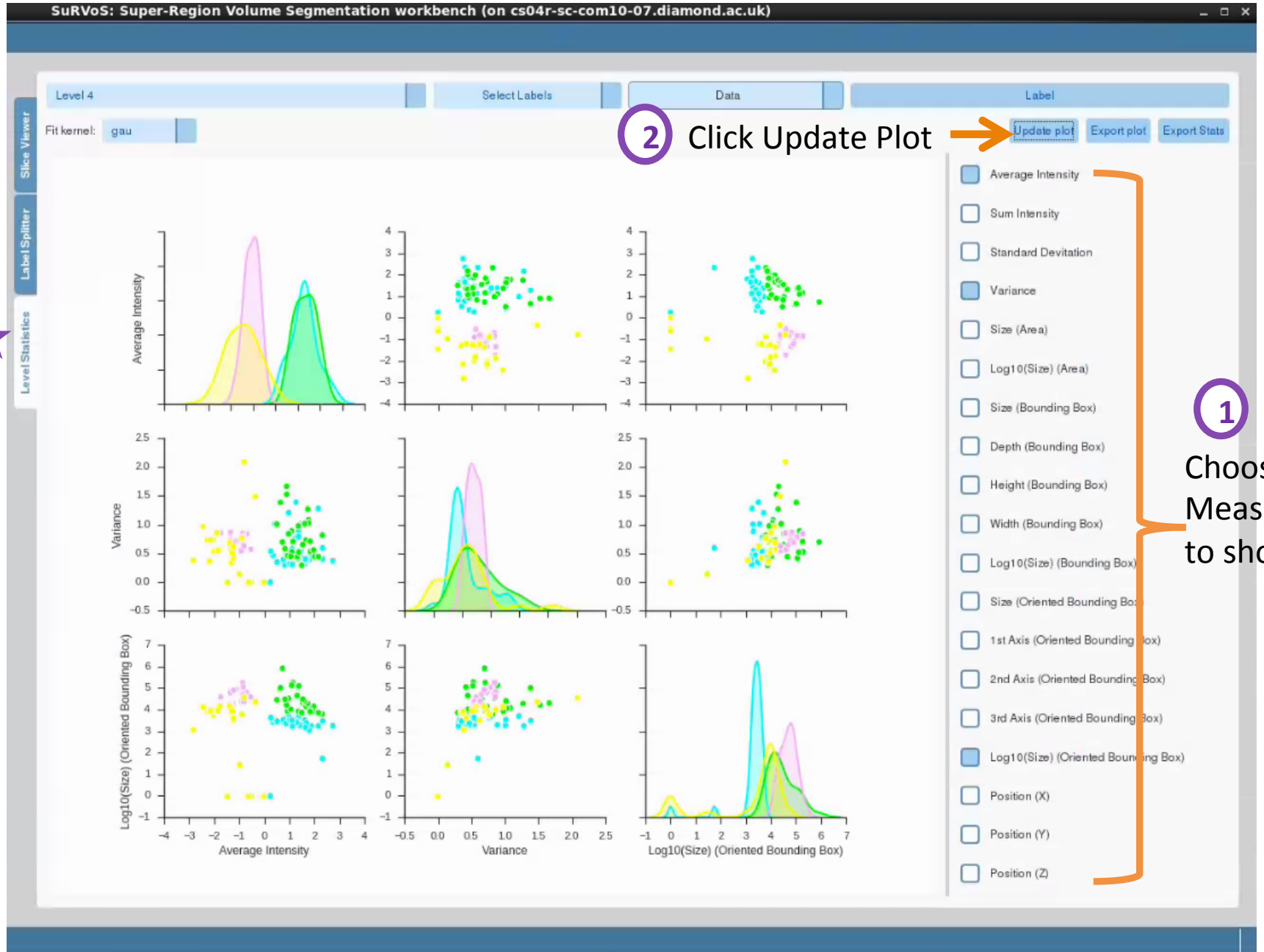
2 Select all labels you want to include

3 Choose which form of the data to measure from

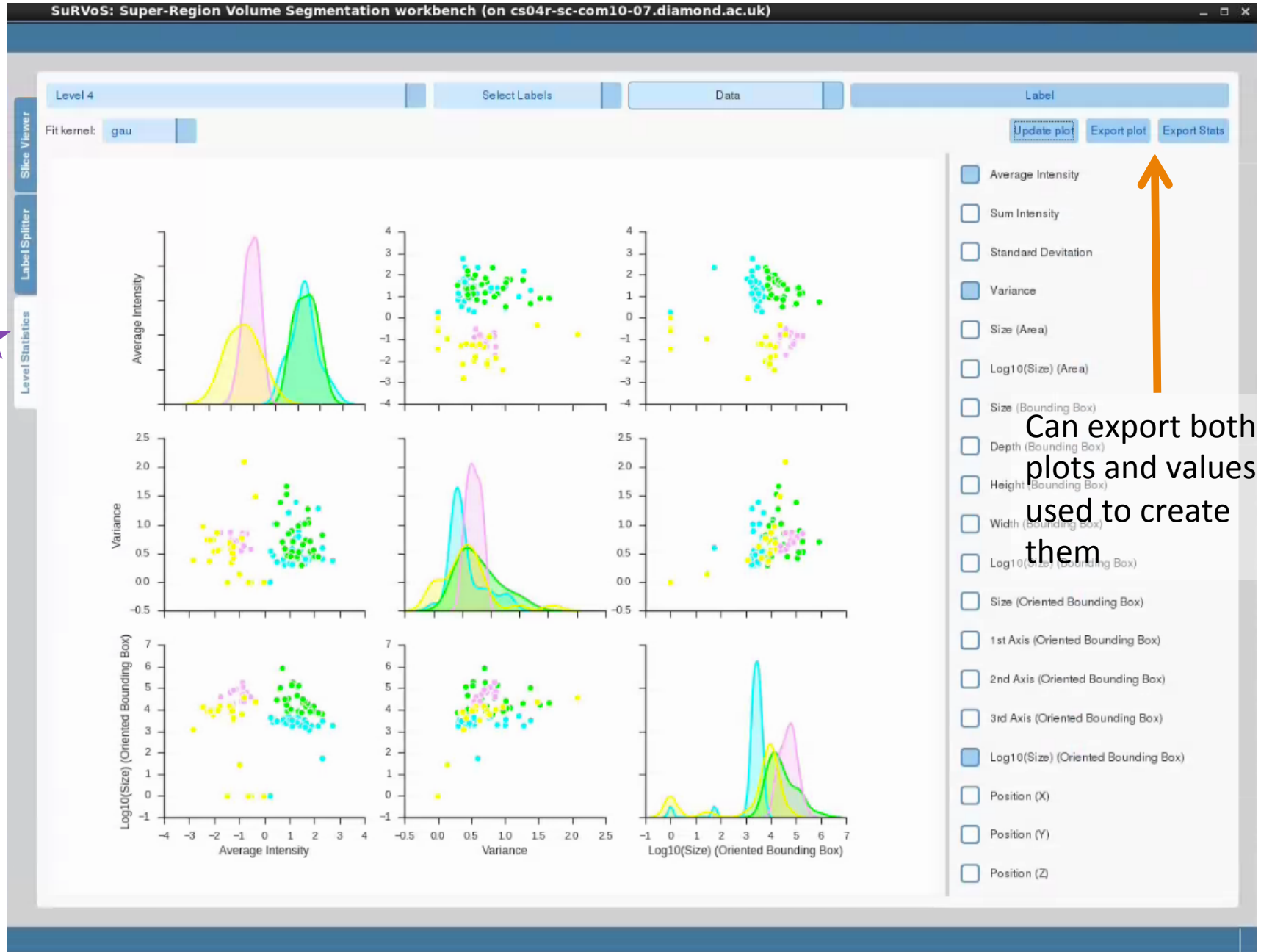
4 Click Label

★

Label Statistics



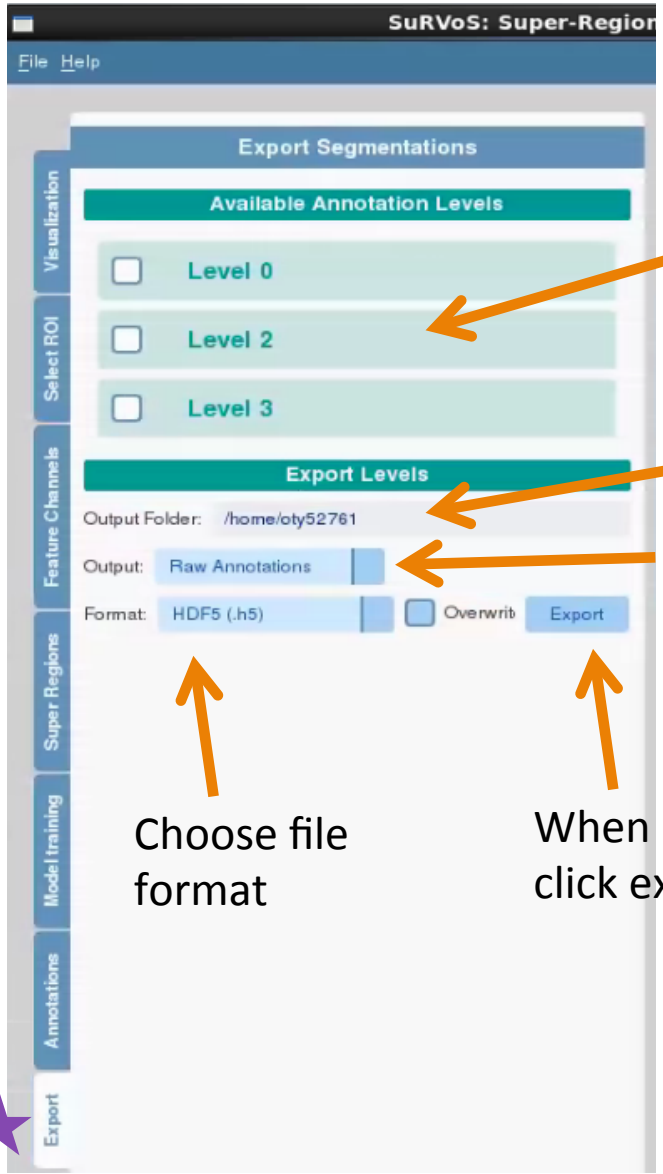
Label Statistics



Can export both plots and values used to create them

Time – 5 minutes

Exporting Data



Choose levels to export

Choose location to export data to

Choose way of exporting data

Choose file format

When ready, click export

Output options:

- Raw Data
- Raw Annotations
- Segmentation Masks
- Masked Data

Format:

- HDF5
- MRC
- Tiff stack

Note: If you choose Masked Data, you can choose which dataset to pull data from (i.e. Gaussian, Total Variation, etc). We recommend using the raw data. You can also choose whether to scale the data (0 to 1) or to invert. This will depend on your downstream application

Acknowledgements

Nottingham University
Imanol Luengo (joint between Nottingham and Diamond)
Andrew P. French
Tony Pridmore

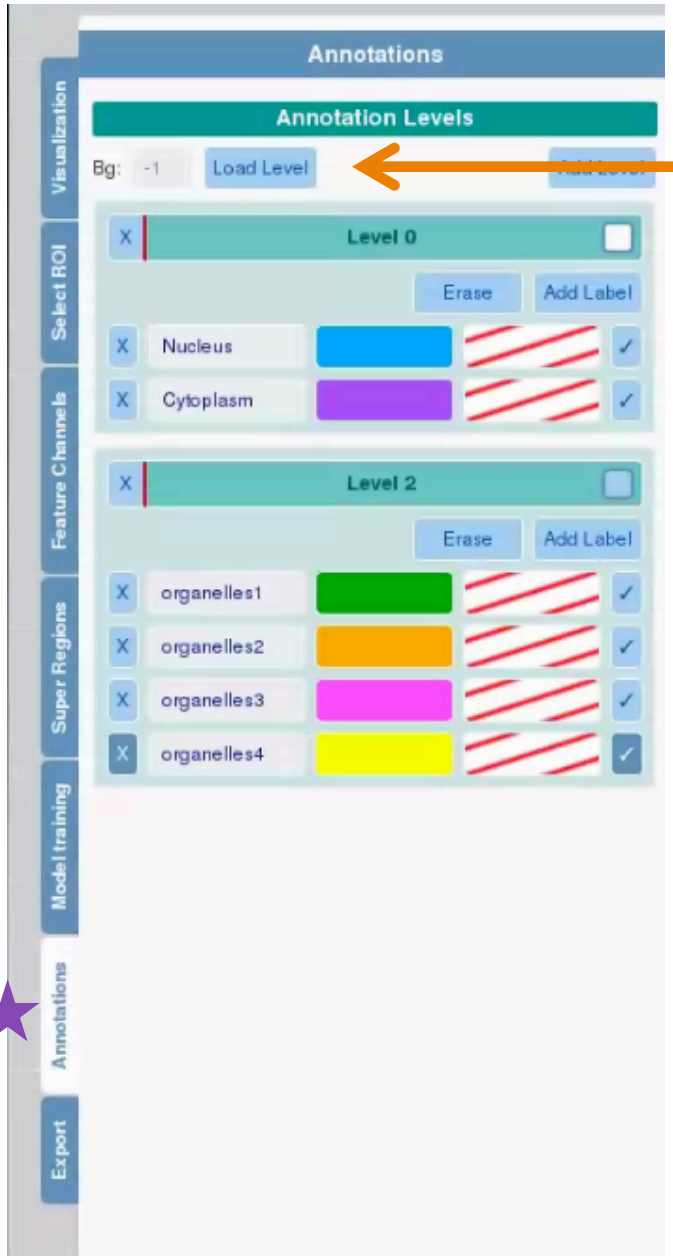
Diamond Light Source
Michele Darrow
Mark Basham
Liz Duke
Corey Hecksel
Alister Burt
Matt Spink
Alan Ashton
Kyle Dent

University of Sussex
Karen Marshall

Also thanks to
Nvidia Corporation for the donation of a Tesla K40.

Time – 5 minutes

Troubleshooting



Can use Load Level to load just the annotation information from a previously segmented project.

If you do this, you will need to load in data as if it is a new project. And if you want to continue segmenting, you will need to recalculate appropriate filters, supervoxels, megavoxels, etc.

Can be used to re-split data or to output plots or object measures.

Installation and Documentation

Website: https://diamondlightsource.github.io/SuRVoS/docs/installation/

Installation

Requirements

- Python Distribution
- NVIDIA GPU with at least 1024 threads
- CUDA SDK already installed and configured.

  If CUDA is not yet installed in the system, follow instructions in:

  https://developer.nvidia.com/cuda-downloads

1. Installing a Python distribution

If there is no Python distribution installed on your system, we recommend installing Anaconda. Otherwise, skip to **step 2**.

1.1 Download Anaconda:

https://www.continuum.io/downloads

1.2 Install Anaconda:

(Windows) Just double click on the installer and follow instructions.

(Linux) Open a terminal and type the following commands:

```
$> cd /path/to/anaconda/
$> chmod a+x Anaconda2-4.0.0-Linux-x86_64.sh
$> ./Anaconda2-4.0.0-Linux-x86_64.sh
        1. press ENTER
        2. press Q
        3. enter "yes"
        4. pres ENTER
        5. enter "yes"
$> source ~/.bashrc
```

NOTE: Replace **2-4.0.0** with your version of Anaconda.

Installation and Documentation Cont.

2. Installing Dependencies

With Anaconda:

```
$> conda update conda pip
$> conda install numpy scipy matplotlib scikit-learn scikit-image cython seaborn networkx pyside
$> pip install scikit-tensor
```

With another python distribution:

```
$> pip install --upgrade pip
$> pip install --upgrade numpy scipy matplotlib scikit-learn scikit-image cython seaborn networkx scil
```

3. Install SuRVoS

3.1 Download SuRVoS

Navigate in a terminal (using `cd`) to a folder where you want to save SuRVoS and type the following commands

```
$> git clone https://github.com/DiamondLightSource/SuRVoS.git
$> cd SuRVoS
```

3.2 Compile SuRVoS features

This step requires CUDA already installed and NVCC compiler in the path (type `which nvcc` to verify it).

```
$> python setup.py build_ext -i
```

4. Run SuRVoS

From the SuRVoS folder:

```
$> ./SuRVoS
```