Electron microscopy (3D) – image processing and analysis

František Kitzberger

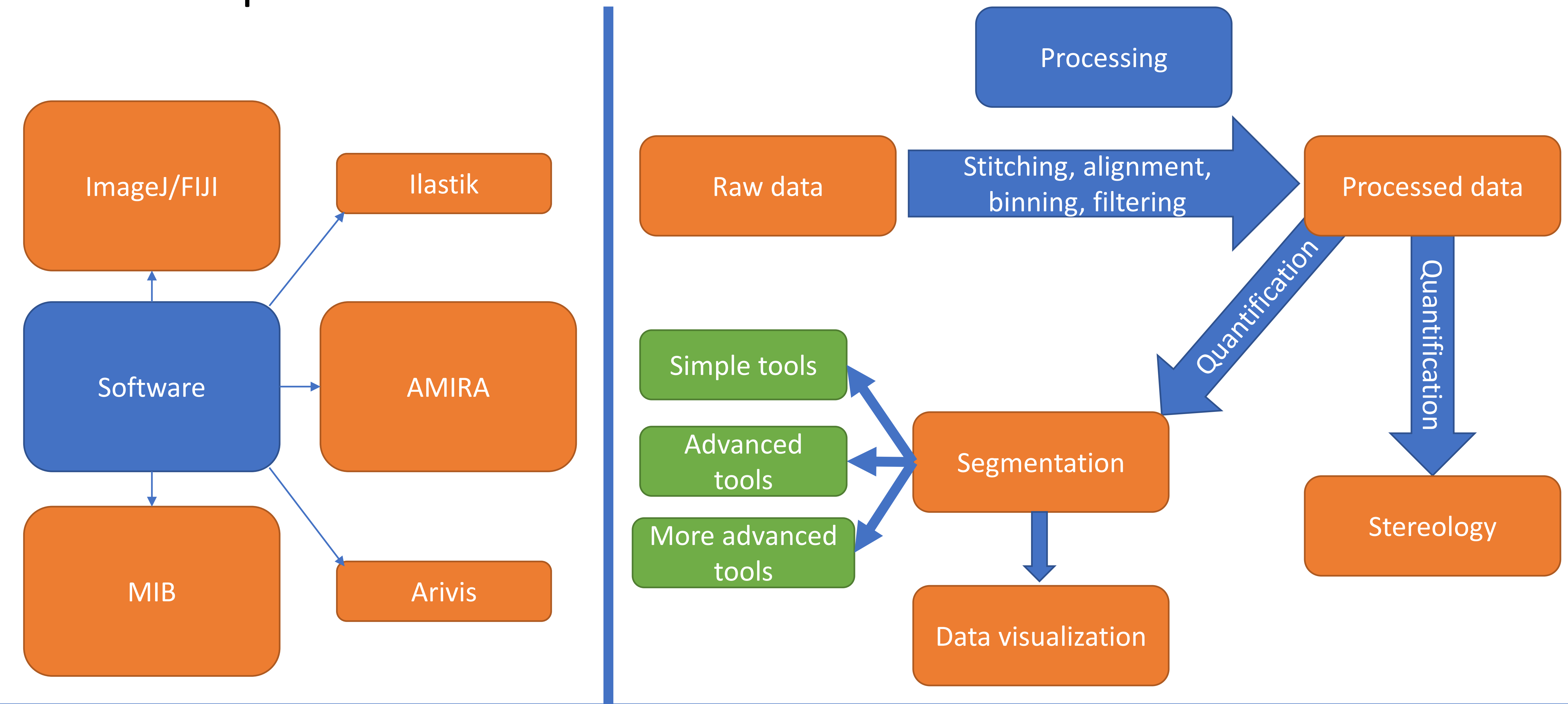
Biology Centre CAS, České Budějovice

Faculty of Science, Charles University in Prague

frantisek.kitzberger@paru.cas.cz

PŘÍRODOVĚDECKÁ
FAKULTA
Univerzita Karlova

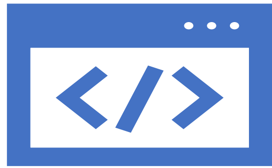
How to process data from the EM? – Lecture outline



What if the dataset is way too big for manuAl segmentation?

FIJI – Fiji is just ImageJ

Software



ImageJ

Java-based image processing software

Open architecture – lot of plugins to instal to do what you need

- Did not find the plugin? Learn to code and write one yourself – you can even share it

Recordable macros – even in the human legible protocol (not only the code but it explains you what you did, when you recorded the macro long time ago)

Image filtering, bit-depth transformation, Stitching, alignment, segmentation

Too many choices – you need to know what do you need

FIJI – ImageJ with batteries included (lot of plugins are already installed)

MIB – Microscopy Image Browser

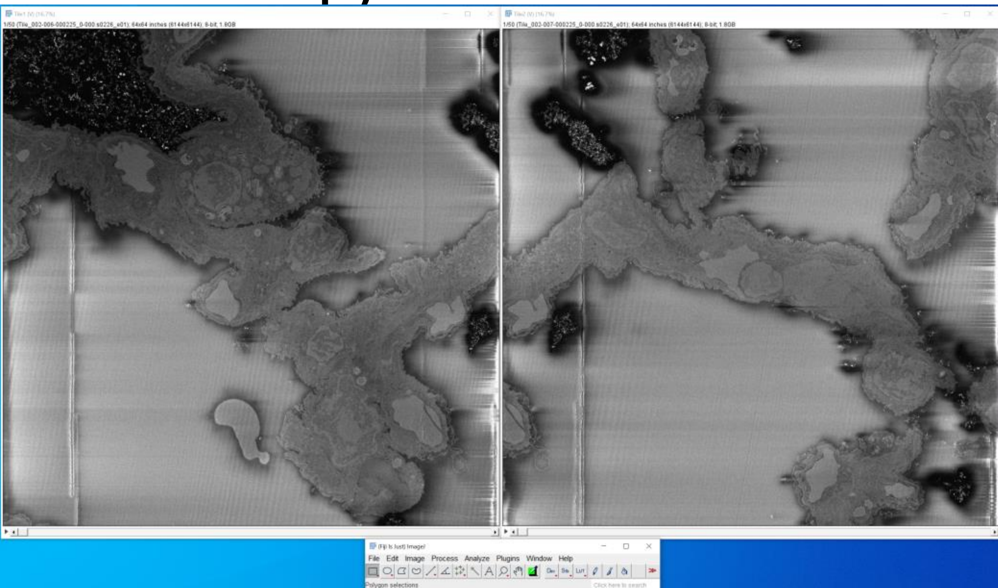
Software

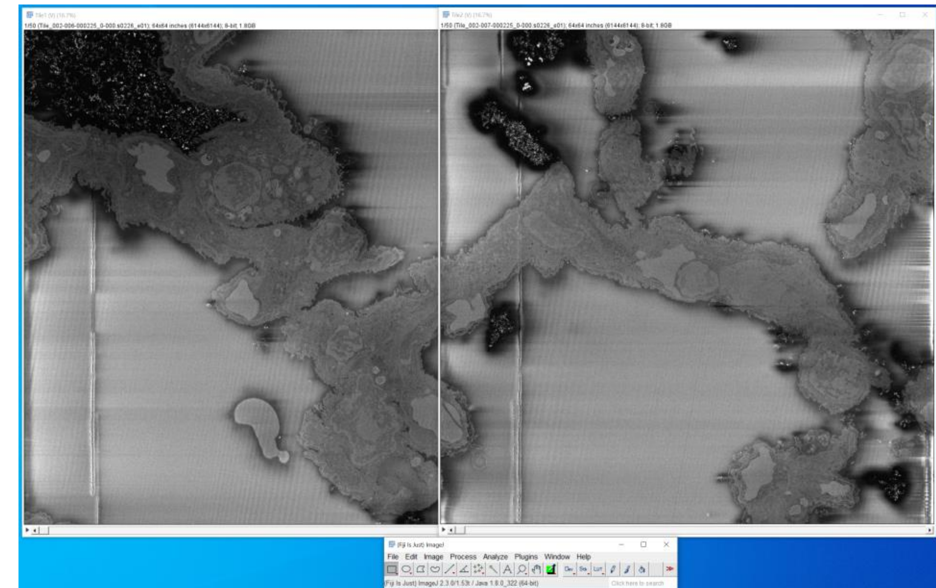
- Matlab based software developed by Ilya Belevich (Laboratory of Eija Jokitalo, EMBI Helsinki)
  - Matlab version/standalone version (does not require a Matlab installed)
  - **Freeware**, constantly developed and updated
- Functions for data processing
  - Alignment→filtering→segmentation→deep learning→basic model rendering
- Most processes are dependent on RAM – usually 2.5x RAM than is your dataset
  - For data from 3D-SEM (tens to hundreds of Gb) you need powerfull workstation or a cluster

Raw data/dataset

- 8bit or 16bit image, usually .tif
  - Raw data formats:

- From 3D-SEM – images, images to be stitched or stitched images (big map)

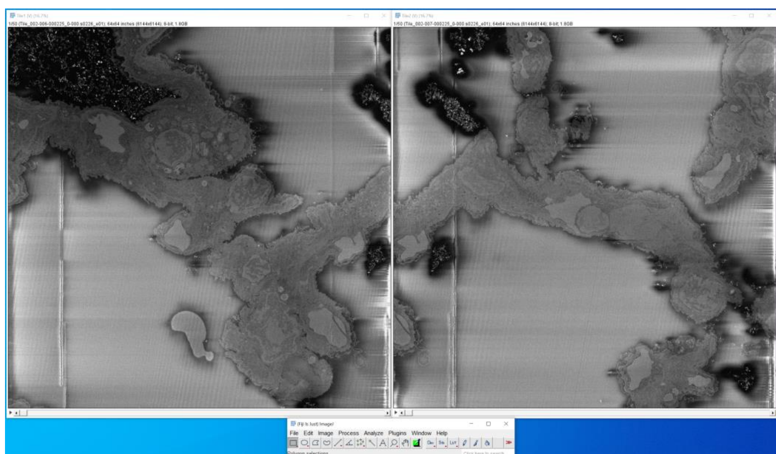




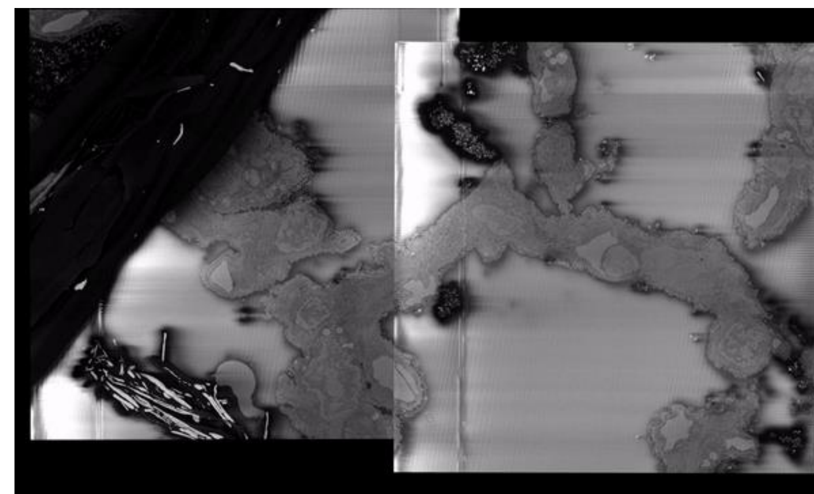
Stitching

Processing

Algorithm based combining multiple images into a big high-resolution image

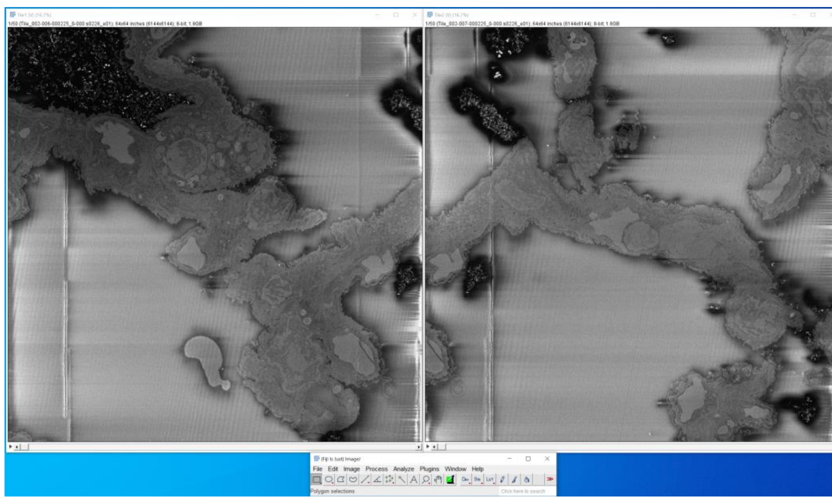


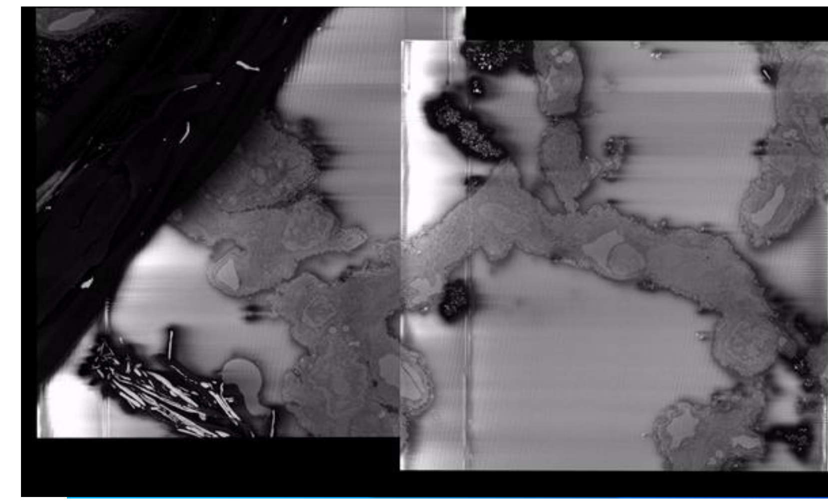
MAGIC



Neighboring tiles are ''sewn'' together and they are aligned within the stack

Stitching

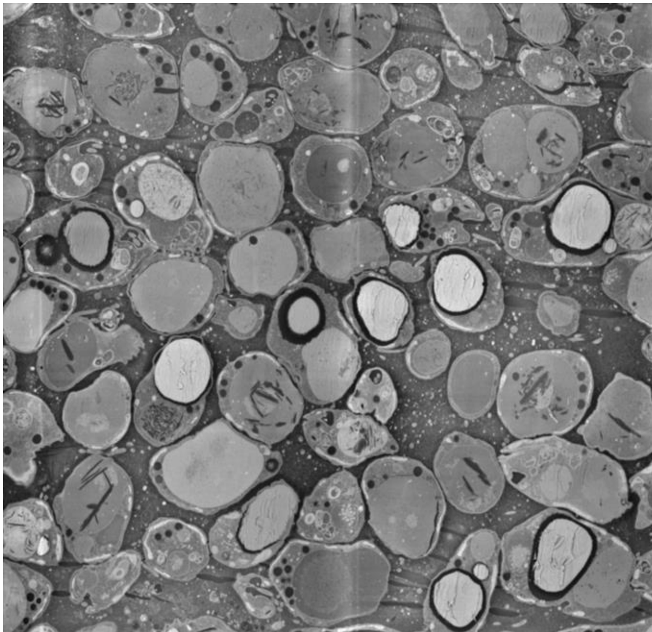


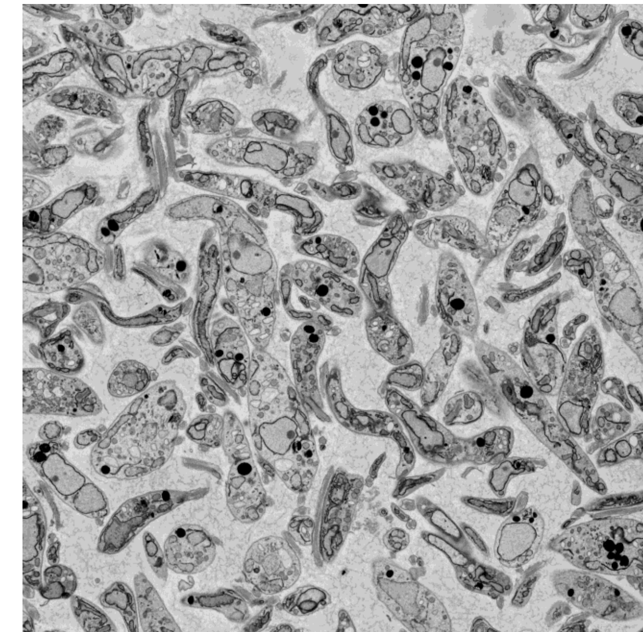


Imaging software (e.g. MAPS) provide some sort of own stitching

- Works well on good samples (brain or nervous tissues, or samples with a lot of structures)

Diplonemids

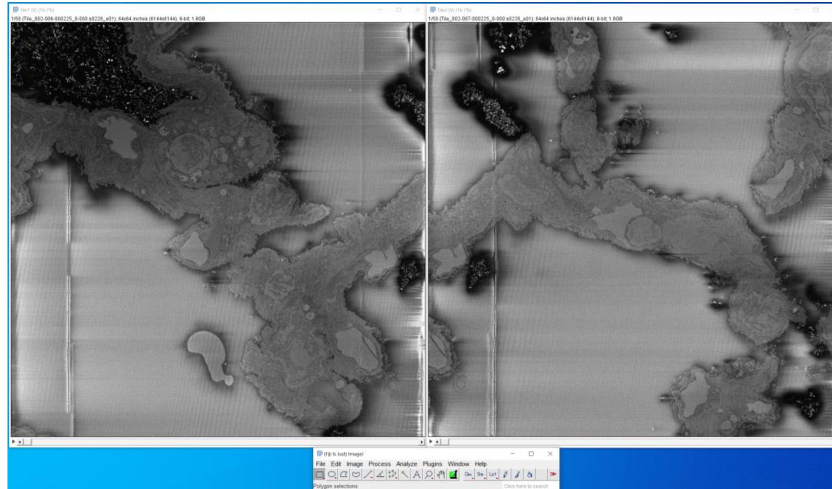


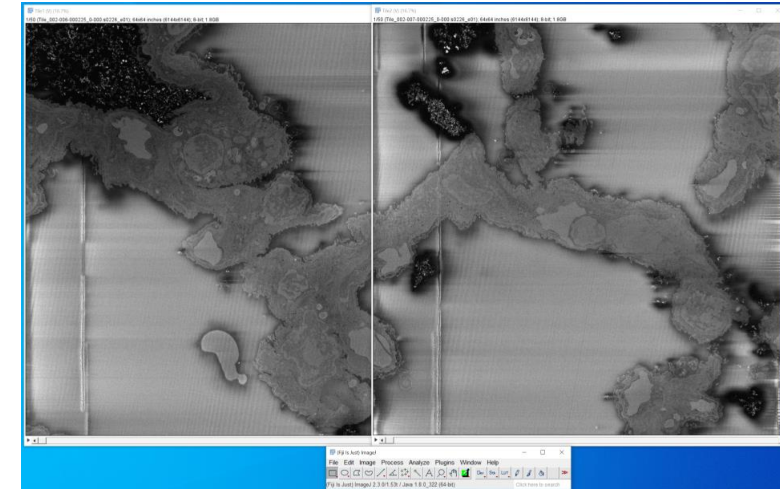


Trypanosomas

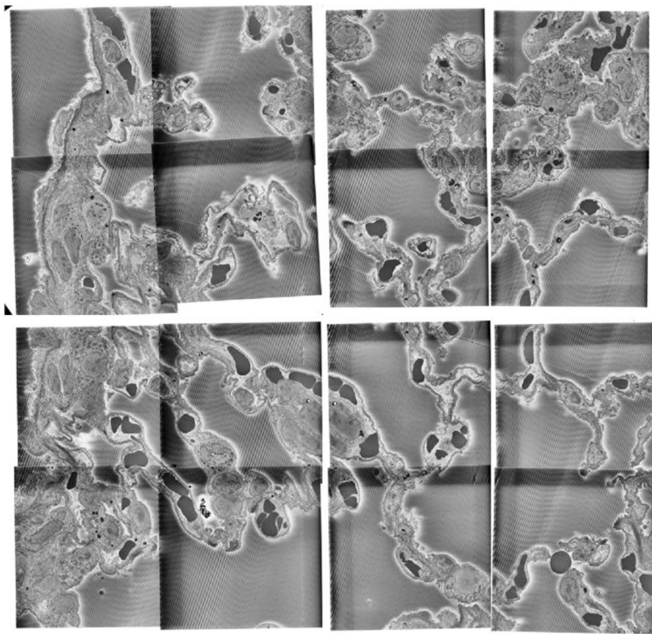
MANY STRUCTURES = GOOD AUTOMATIC STITCHING

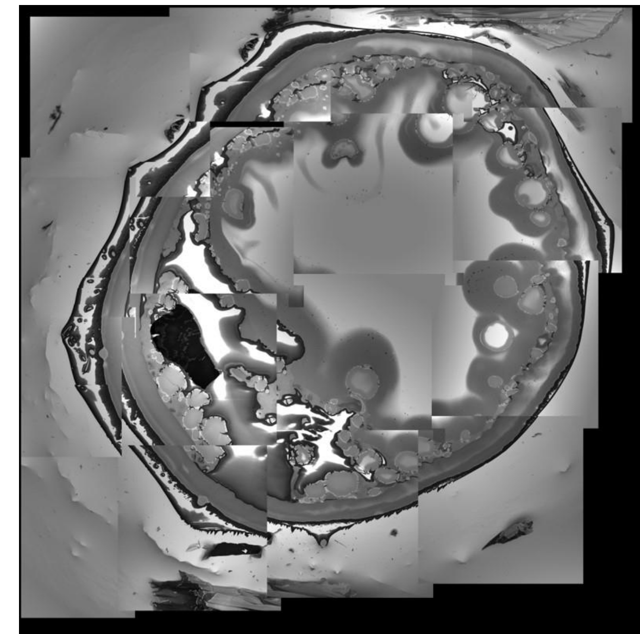
Stitching

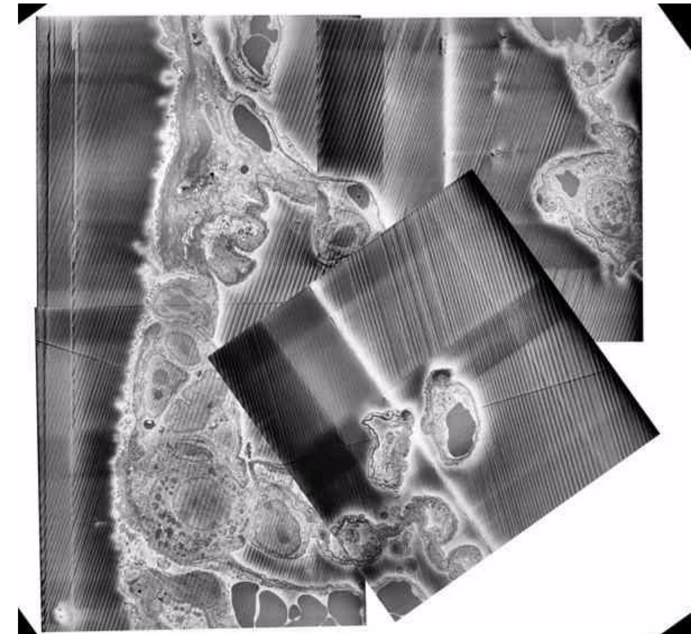




Empty resin = problems for the automatic stitching





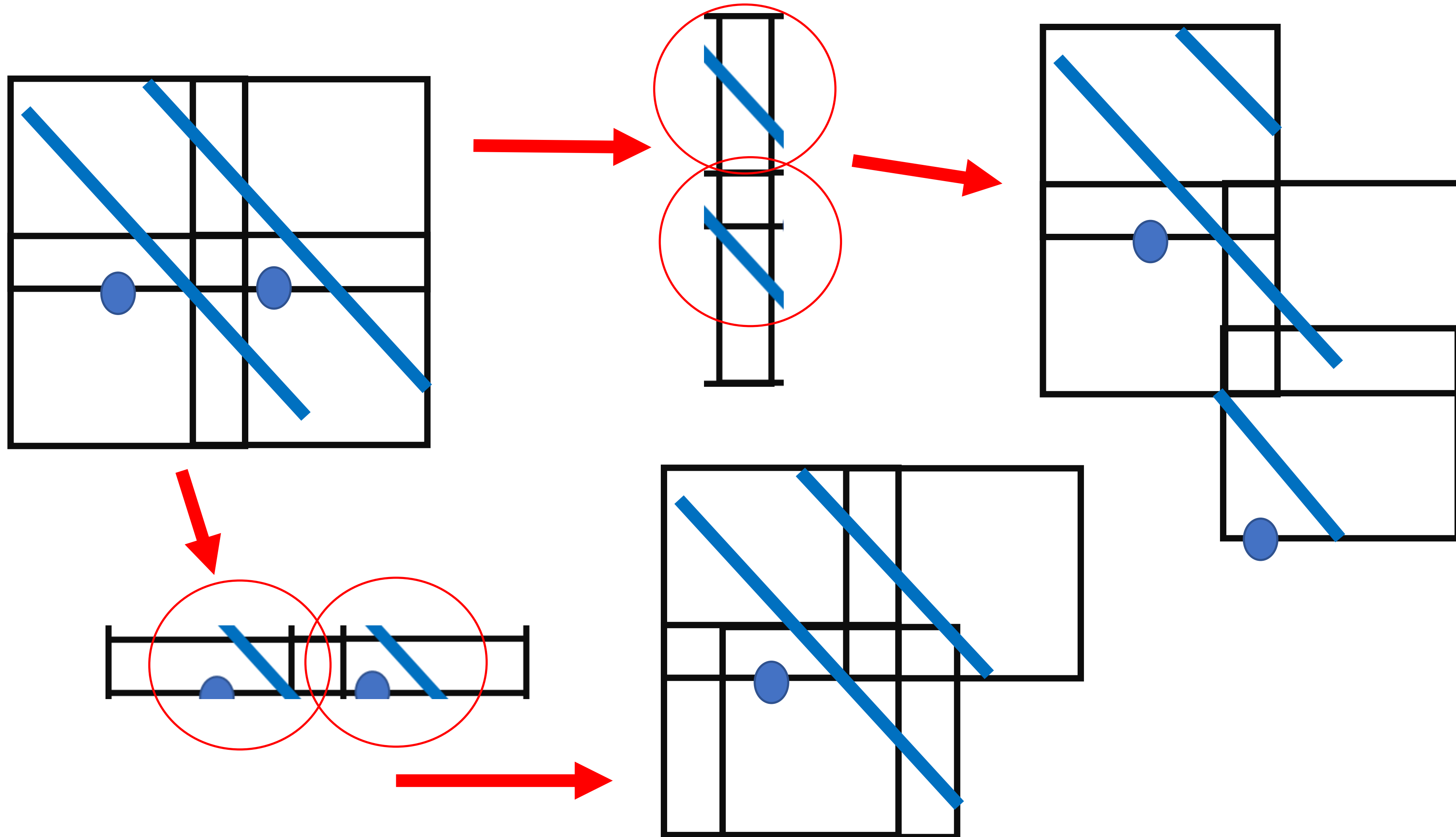


Software works differently than a human, we are looking for a trends, software matches points in the overlapping area.

Stitching – how does it work?

Processing

Software works differently than a human, we are looking for a trends, software matches points in the overlapping area.

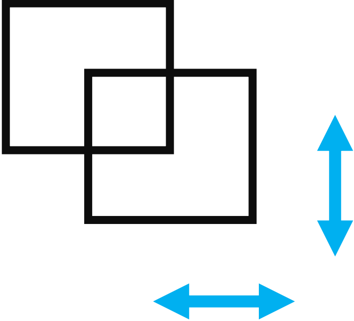


Processing

Software steps – Dataset

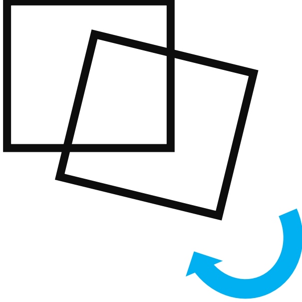
How it works (in Image J – TrakEM)

A) Translation



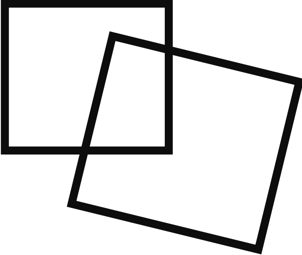
only displacements in X,Y.

B) Rigid



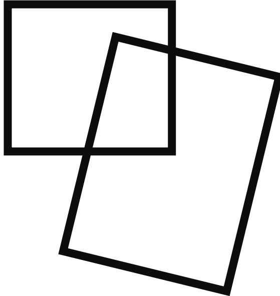
translations plus rotation

C) simillarity

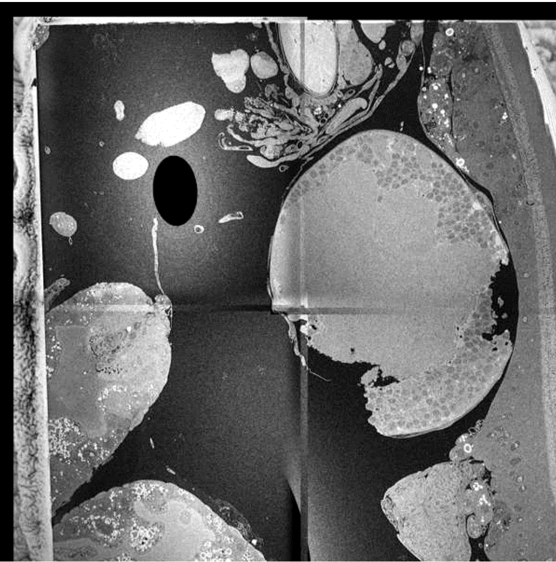


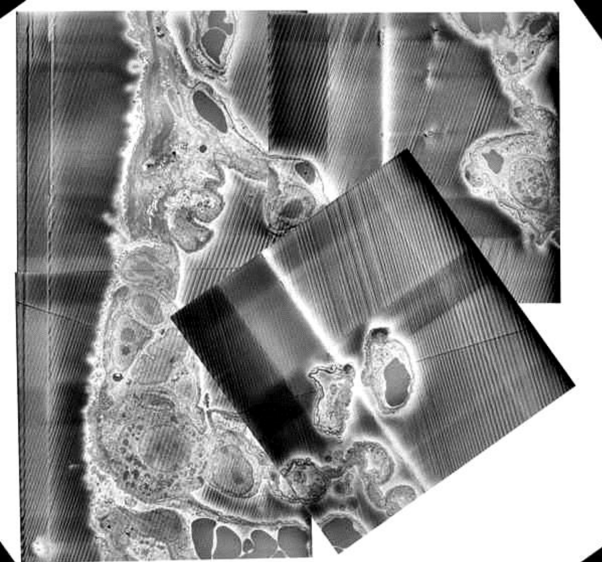
translation, rotation and isotropic scaling (that is, it preserves image aspect ratio.)

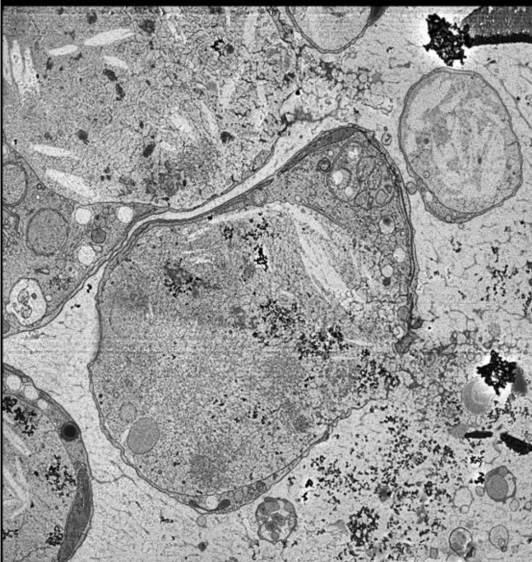
D) Affine



free affine transform, which amounts to translation, rotation, scaling, and shear

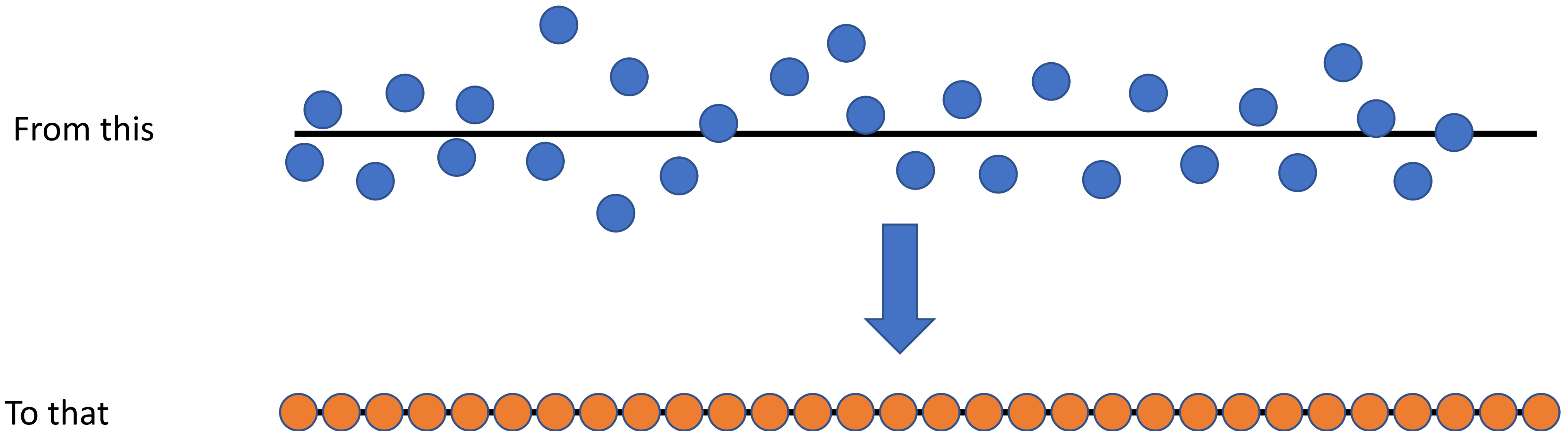






Processing

Alignment – put the disordered stacked (stitched) images to aligned state



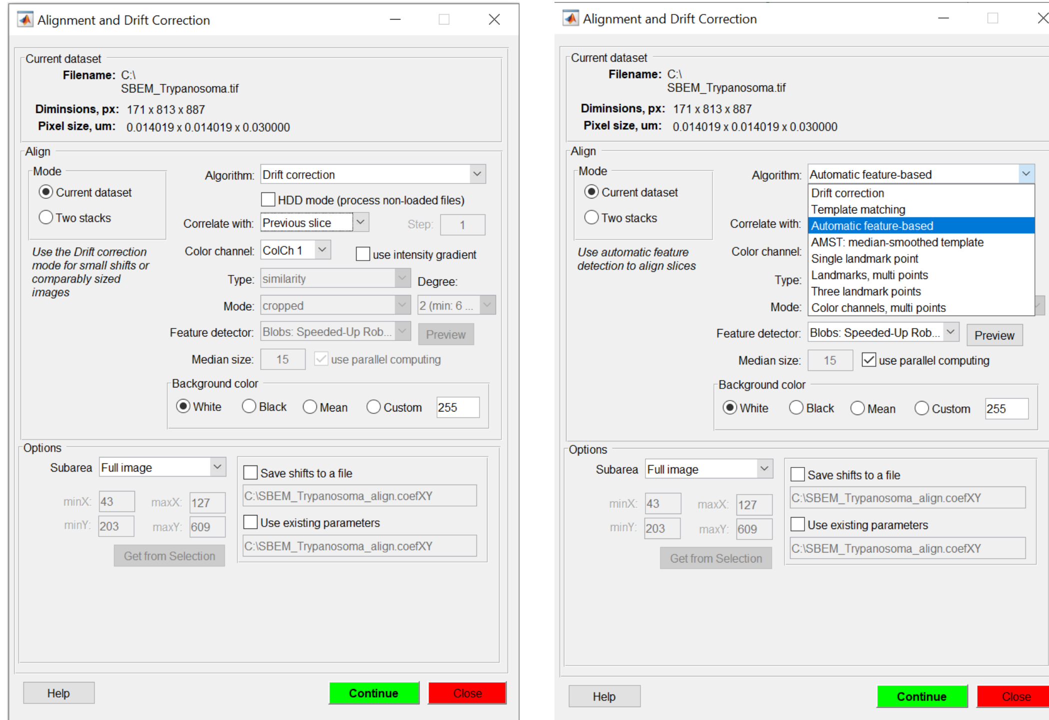
Stack of EM images that comes from the microscope is never aligned –Beam jumps etc.

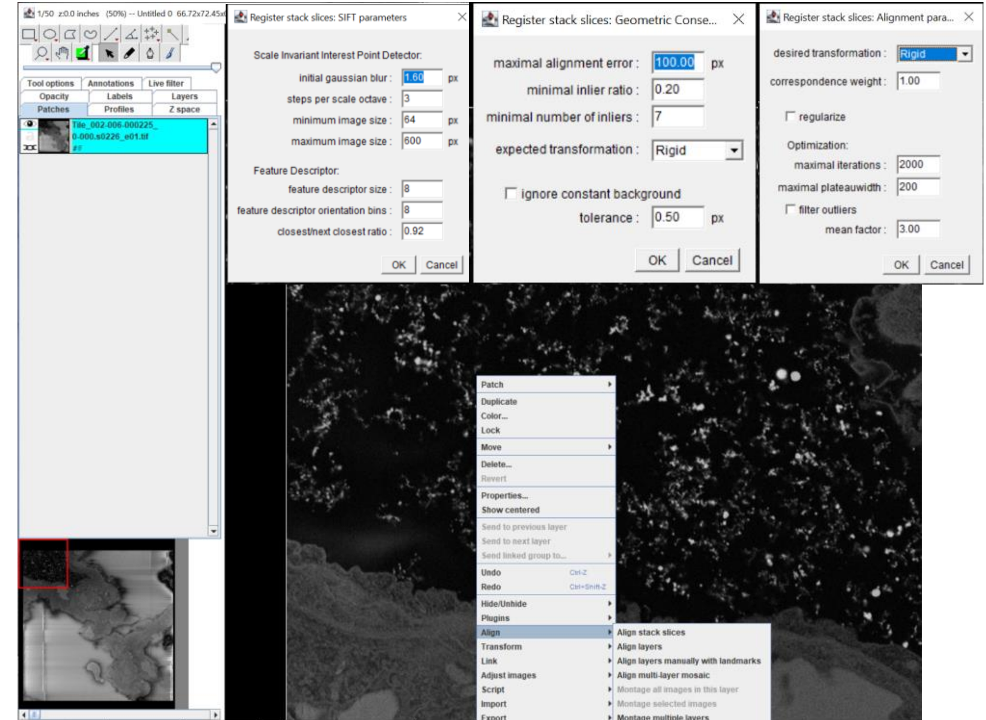
Alignment – put the disordered stacked (stitched) images to aligned state

Processing

Various algorithms to align the images – each has specific use (MIB)

Straightforward and robust stack alignment
More individual parameters to set (if you know what they do…)





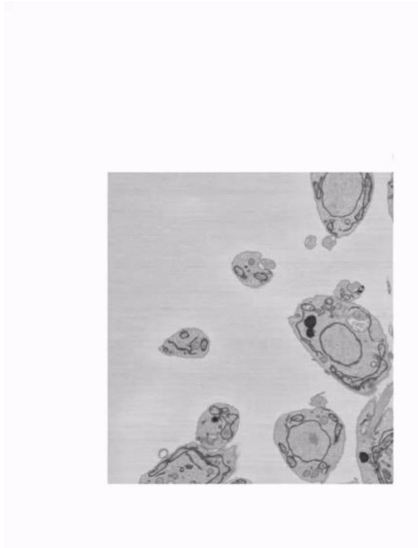
Cannot do the stitching

Can do in parallel stitching of multiple tiles

Processing

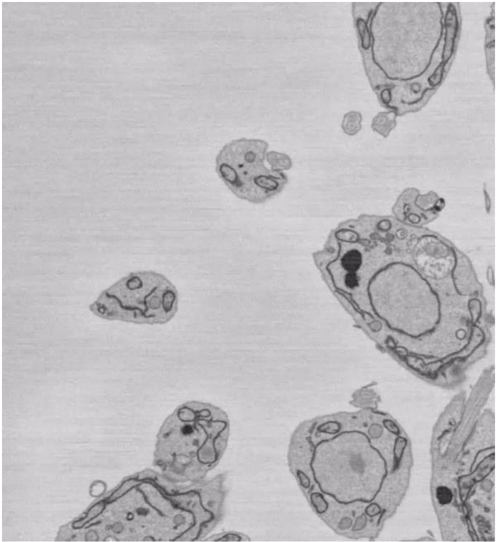
Alignment – put the disordered stacked (stitched) images to aligned state

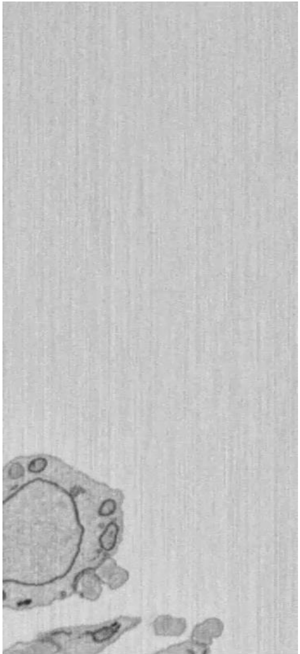
From this:

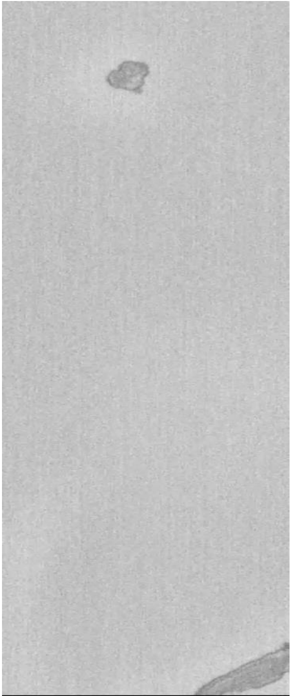


XY YZ XZ

To that:







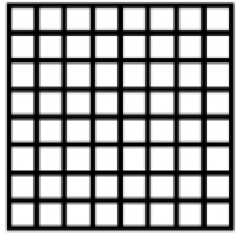
XY YZ XZ

Pixel binning

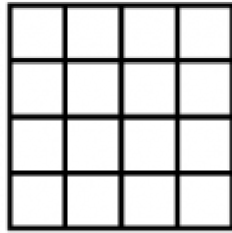
Processing

- Process of combining adjacent pixels in the image by summing or averaging their values into single pixels.
  - Binning 2x2 merge array of 4 pixels into one

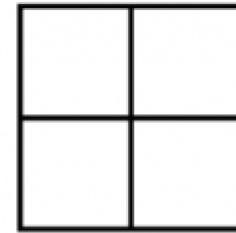
Normal

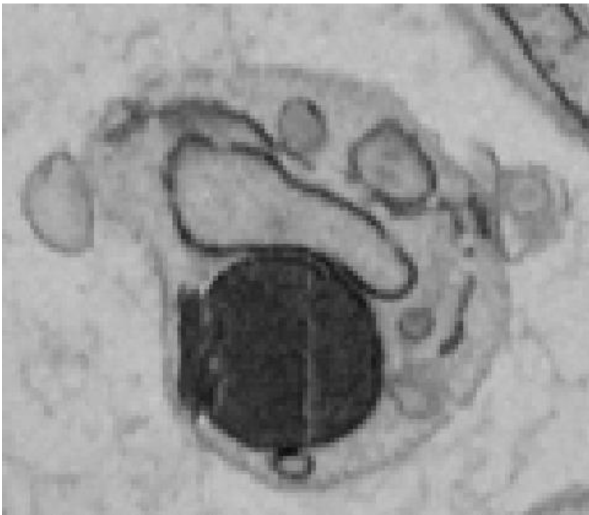


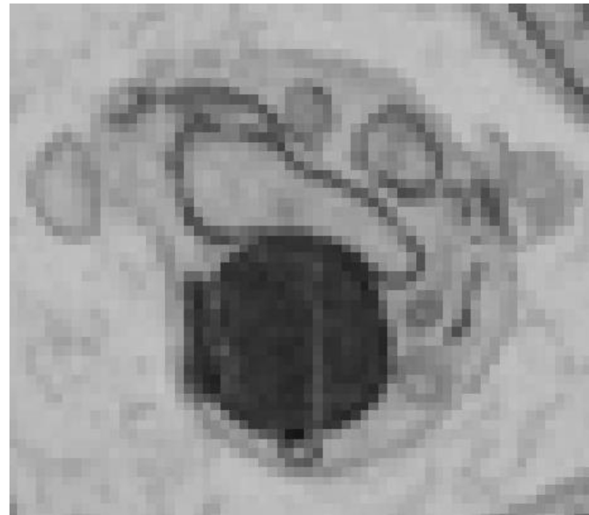
2x2 binning

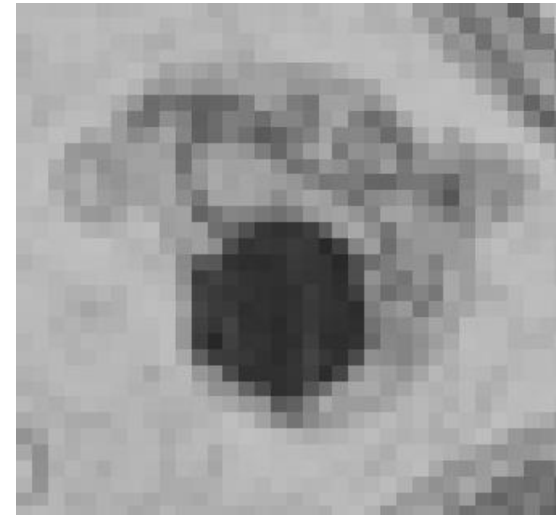


4x4 binning



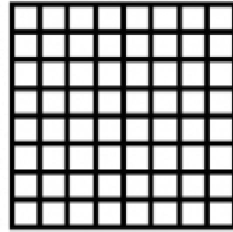




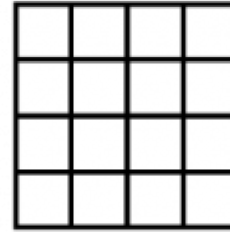


Pixel binning

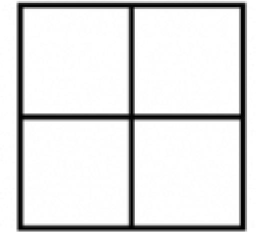
Normal

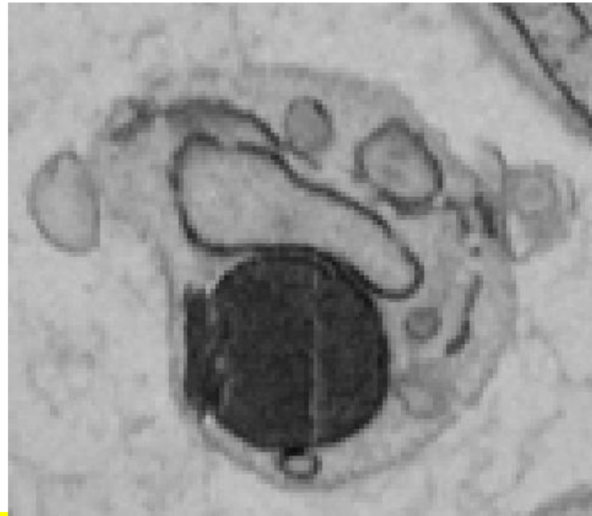


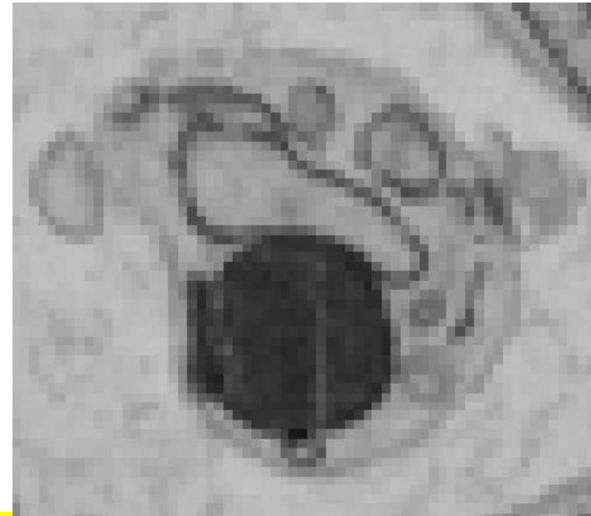
2x2 binning

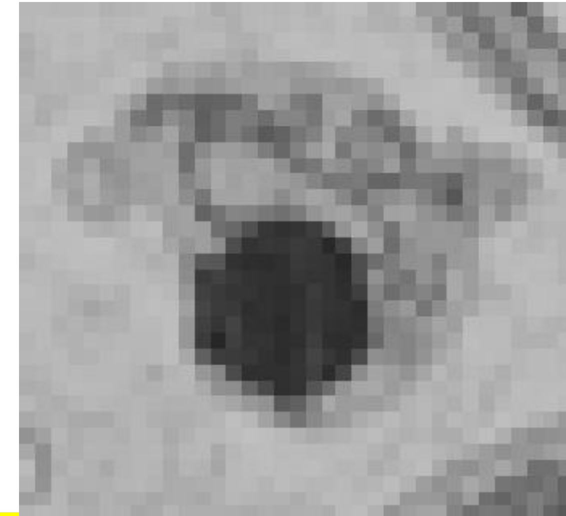


4x4 binning





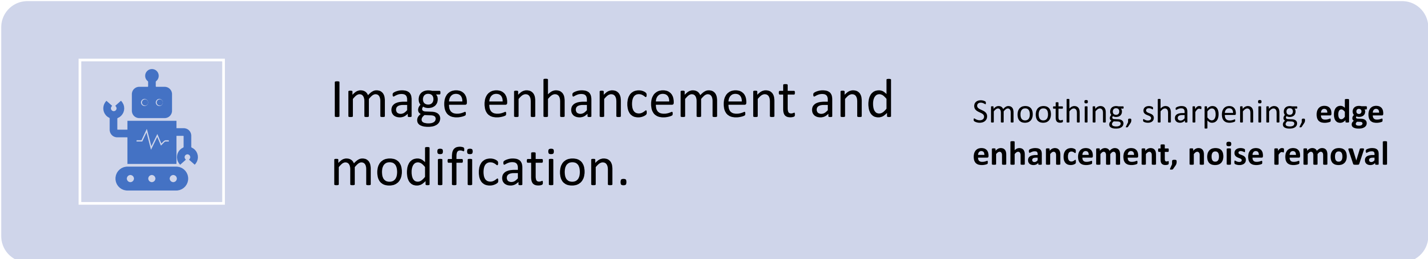


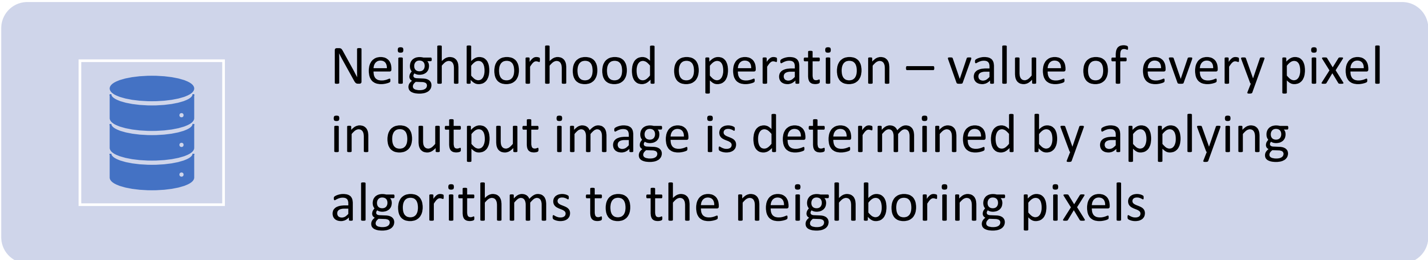


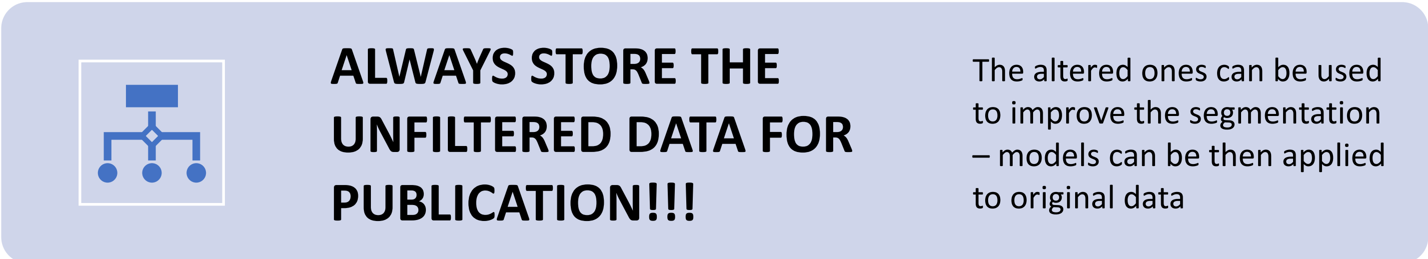
- **The binning decreases number of pixels in the image – thus decreases the image size, decreases noise and increases contrast ☺**
- **BUT also – decreases image resolution – we lose some image information ☹**
- **For huge datasets it is inevitable as too big data are difficult to work with (long calculation time, not enough RAM/GPU…)**

Processing

Image filtering







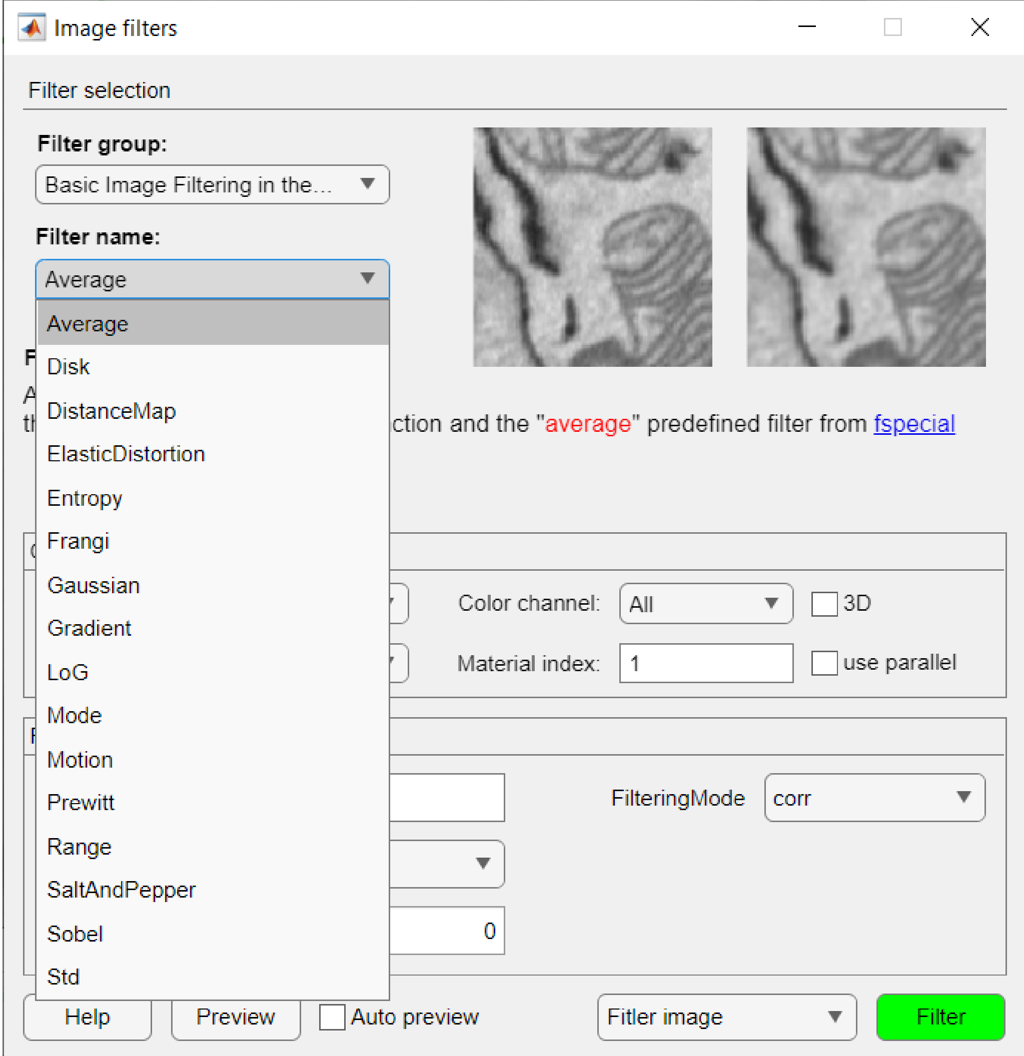
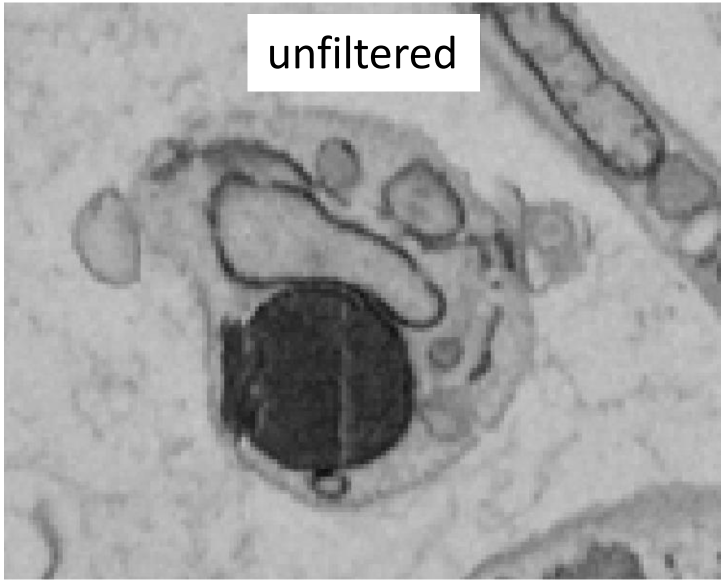
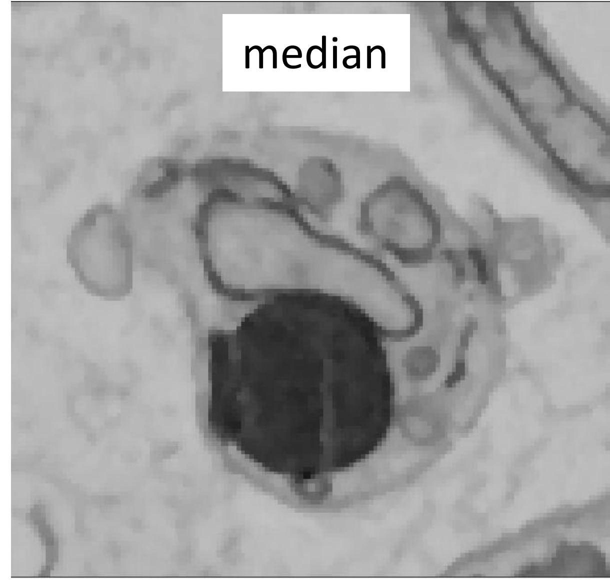
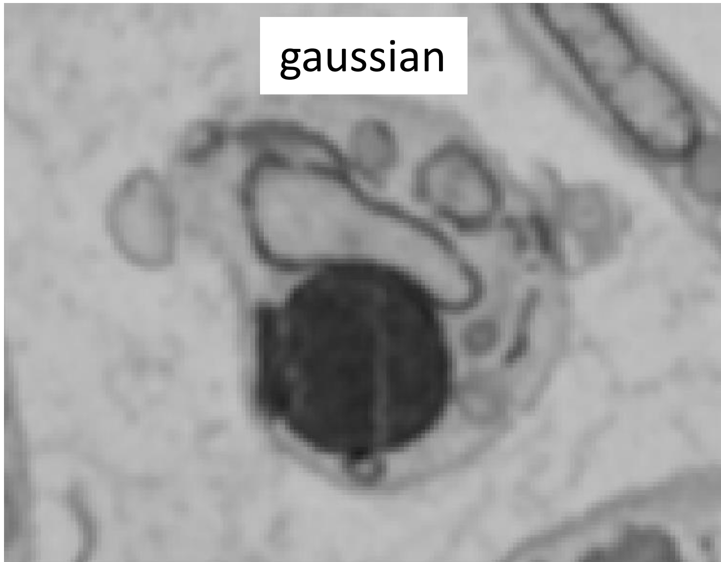


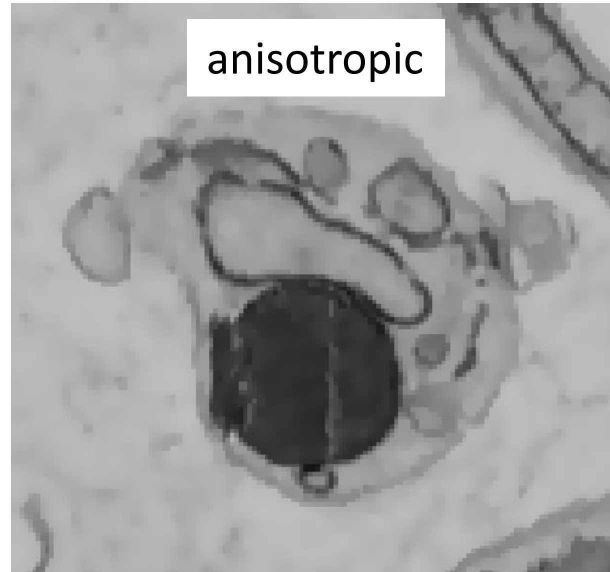
Image filtering – examples

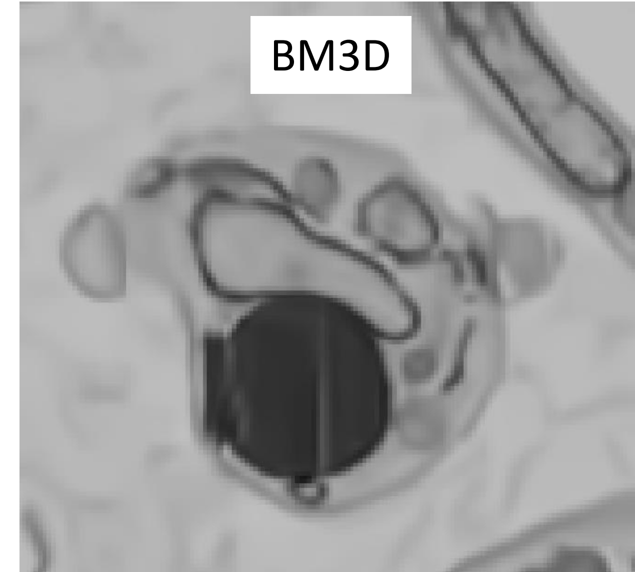
Processing









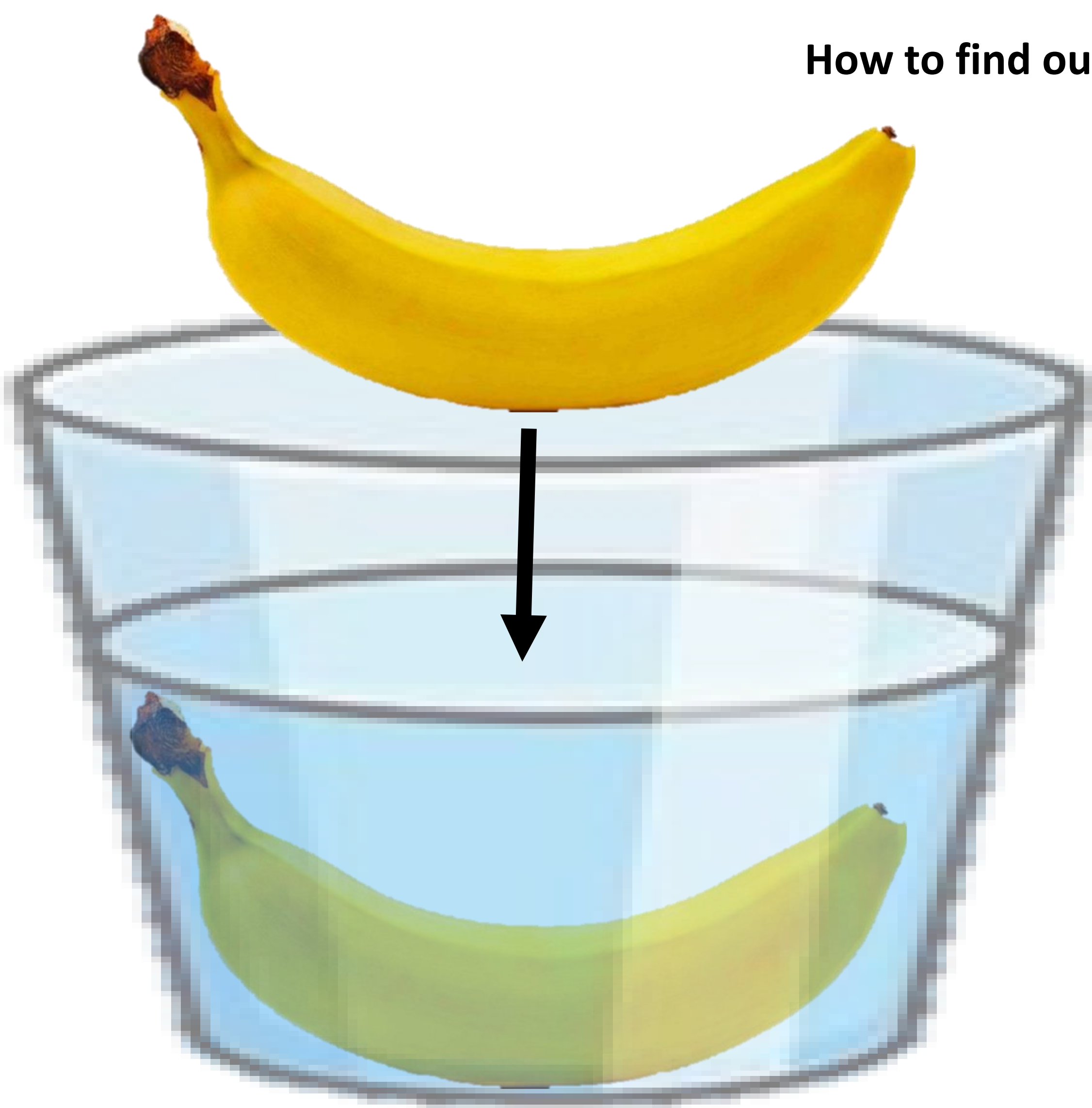


“Fast(est)“ quantification for big datasets

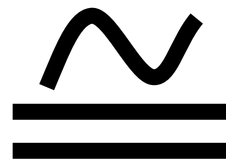
Stereology

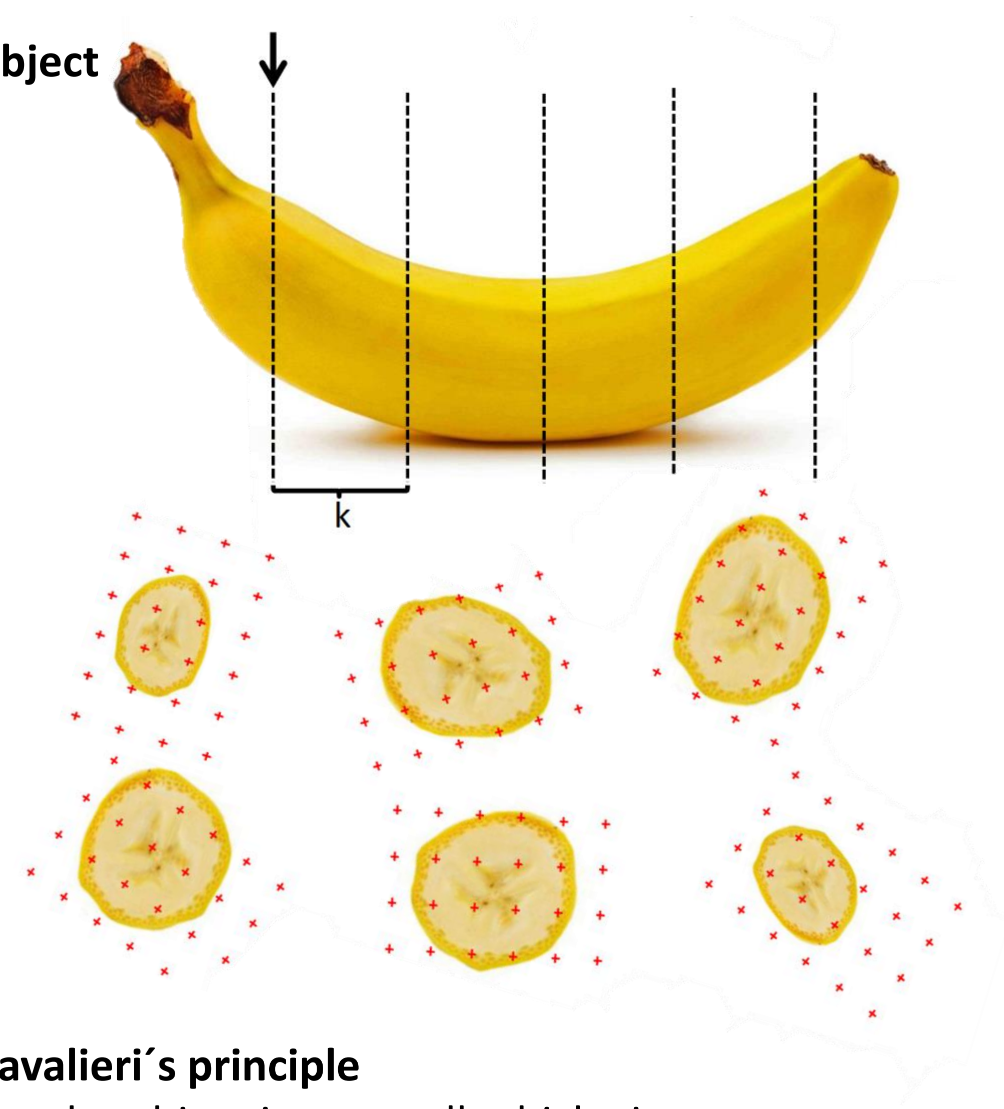
- Stereology – Random systematic sampling method for obtaining quantitative information about 3D material from measurements done on 2D planar sections
  - Grid based
  - We use simplified version (Cavalieri´s principle) just for the statistical purposes
    - True stereology is much more complicated

How to find out the volume of the object



Sink it in the water and find out how much the water level rised.

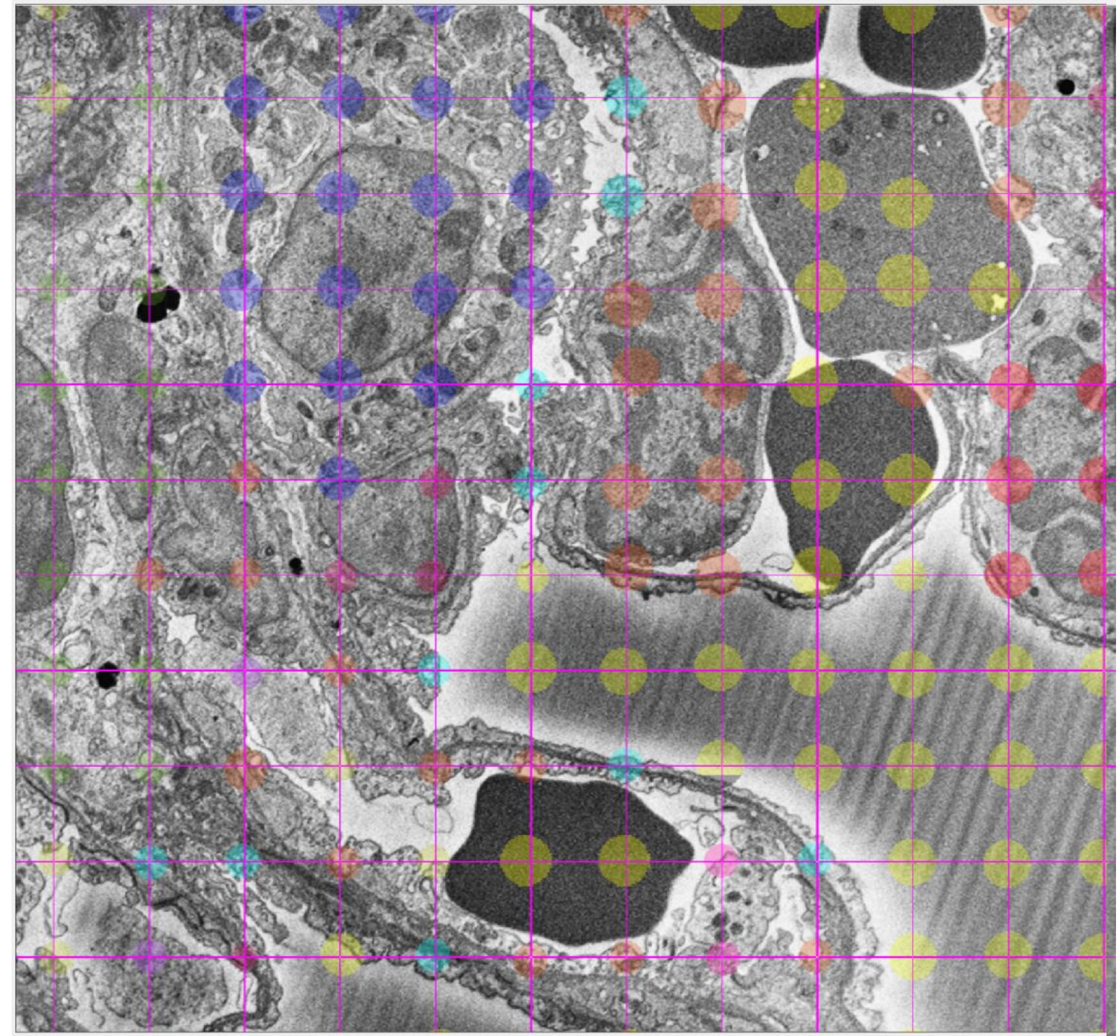




Cavalieri´s principle
Cut the object into equally thick pieces
Put a grid of known size on the cuts
Count the points (we know the area represented by one point – grid constant)
Multiply the point count with the grid constant and with the cut thickness

“Fast(est)“ quantification for big datasets

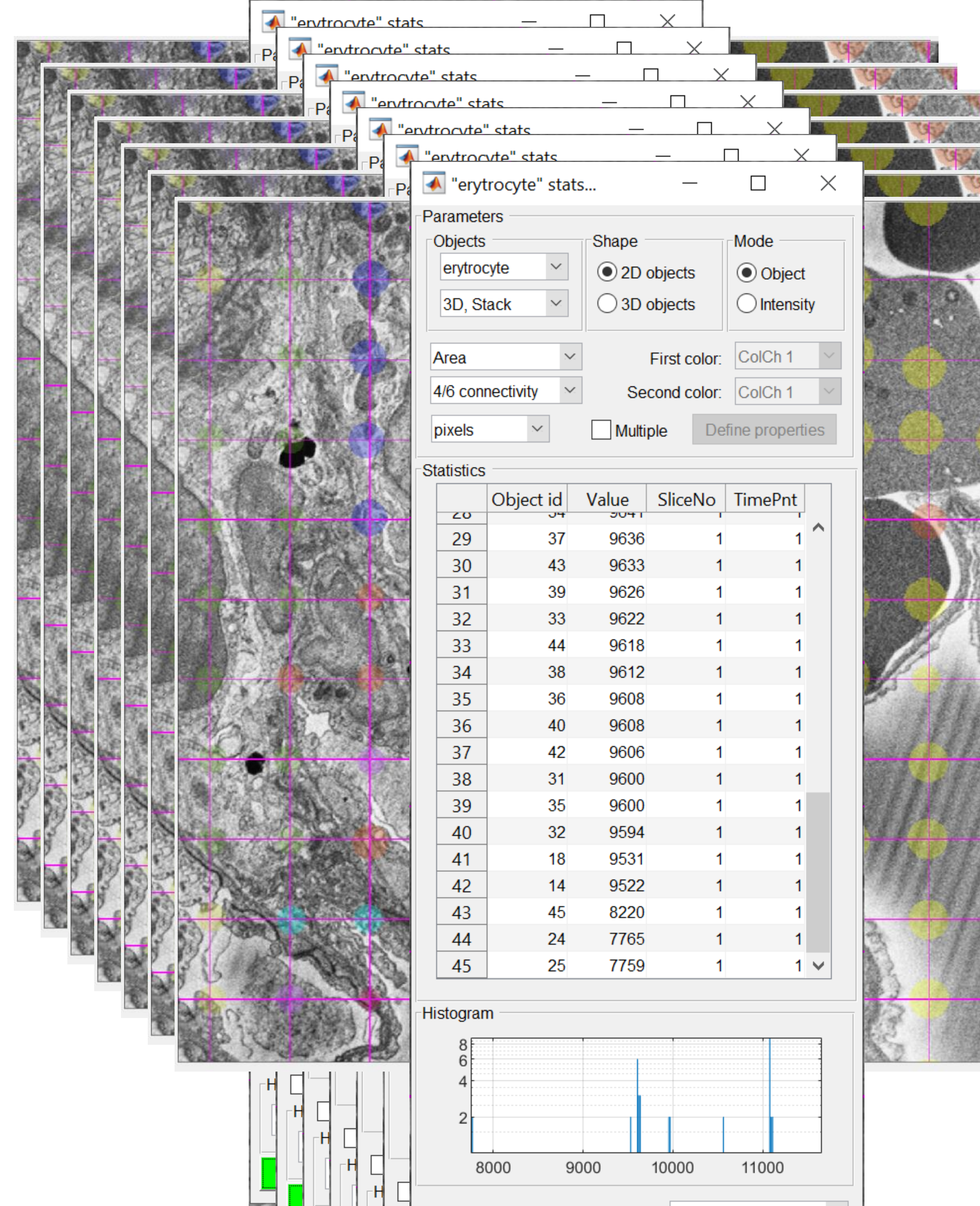
- Start with an image of known pixel size (defined at the acquisition)
- Apply a grid with known proportions
- Assign each cross-section to a „material“



Stereology

"Fast(est)" quantification for big datasets

- Start with an image of known pixel size (defined at the acquisition)
- Apply a grid with known proportions
- Assign each cross-section to a „material"
- Count numer of dots of each material in the image
- Repeat for all images in the stack (or every Nth)



	Object id	Value	SliceNo	TimePnt
29	37	9636	1	1
30	43	9633	1	1
31	39	9626	1	1
32	33	9622	1	1
33	44	9618	1	1
34	38	9612	1	1
35	36	9608	1	1
36	40	9608	1	1
37	42	9606	1	1
38	31	9600	1	1
39	35	9600	1	1
40	32	9594	1	1
41	18	9531	1	1
42	14	9522	1	1
43	45	8220	1	1
44	24	7765	1	1
45	25	7759	1	1

“Fast(est)“ quantification for big datasets

Stereology

- Start with an image of known pixel size
- Apply a grid with known proportions
- Assign each cross-section to a „material“
- Count numer of dots of each material in the image
- Repeat for all images in the stack (or every Nth)
- Calculate „comparisonal“ stereology

					2 DPI									5 DPI								
	Mutation	NC			951			LP			Brit			951			LP			Brit		
	1	points	%		points	%		points	%	%	points	%	%	points	%	%	points	%	%	points	%	%
Pneumocyt	Pneu I	521	8	16	596	11,2	19	1018	9,9	19,9	594	7,3	17,4	317	6,0	15,5	405	8,0	10,4	1582	7,9	19,6
	Pneu I dam	55	1	2	14	0,3	0	28	0,3	0,5	32	0,4	0,9	109	2,1	5,3	71	1,4	1,8	16	0,1	0,2
	Pneu II	372	6	12	227	4,3	7	408	4,0	8,0	412	5,1	12,1	231	4,4	11,3	521	10,3	13,4	1108	5,5	13,7
	Pneu II dam	0	0	0	16	0,3	1		0,0	0,0	27	0,3	0,8	0	0,0	0,0	84	1,7	2,2	0		0,0
Capillary	Endoth. cell	1052	16	33	744	14,0	24	1267	12,4	24,7	655	8,1	19,2	439	8,4	21,4	522	10,3	13,4	1807	9,0	22,4
	Endoth. cell dam	18	0	1	154	2,9	5	96	0,9	1,9	123	1,5	3,6	42	0,8	2,1	86	1,7	2,2	62	0,3	0,8
	Ery	271	4	9	203	3,8	7	254	2,5	5,0	131	1,6	3,8	201	3,8	9,8	130	2,6	3,3	534	2,7	6,6
	Immature ery	0	0	0	8	0,2	0		0,0		0	0,0		0	0,0	0,0				0		
	Thrombocyte	39	1	1	55	1,0	2	65	0,6	1,3	10	0,1	0,3	19	0,4	0,9	21	0,4	0,5	73	0,4	0,9
	Immune cells	13	0	0	147	2,8	5	14	0,1	0,3	134	1,6	3,9	46	0,9	2,2	173	3,4	4,4	194	1,0	2,4
	Neutrophil	0	0	0	23	0,4	1	18	0,2	0,4	0	0,0	0,0	1	0,0	0,0	89	1,8	2,3	38	0,2	0,5
	Eosinophil	3	0	0			0		0,0	0,0	0	0,0	0,0		0,0	0,0	0	0,0	0,0	11	0,1	0,1
	Capillary lumen	162	2	5	53	1,0	2	260	2,5	5,1	69	0,8	2,0	58	1,1	2,8	31	0,6	0,8	181	0,9	2,2
Fibro-ECM	Lipofibroblasts	555	8	18	313	5,9	10	897	8,8	17,5	706	8,7	20,7	395	7,5	19,3	465	9,2	11,9	1287	6,4	16,0
	Lipofibroblasts dam	0	0	0	37	0,7	1	0	0,0	0,0	2	0,0	0,1	7	0,1	0,3	0	0,0	0,0	0		0,0
	Fibroblasts	0	0	0	0	0	0	0	0,0		0	0,0	0,0	0	0,0	0	0	0	0	27	0,1	0,3
	LD	0	0	0	0	0,0	0	0	0,0	0,0	8	0,1	0,2	0	0,0	0,0	27	0,5	0,7	52	0,3	0,6
	ECM	27	0	1	122	2,3	4	77	0,8	1,5	64	0,8	1,9	51	1,0	2,5	90	1,8	2,3	240	1,2	3,0
	Dense Fibers	7	0	0	49	0,9	2	61	0,6	1,2	17	0,2	0,5	10	0,2	0,5	43	0,9	1,1	68	0,3	0,8
	Collagen Fibers	0	0	0	80	1,5	3	77	0,8	1,5	27	0,3	0,8	22	0,4	1,1	77	1,5	2,0	213	1,1	2,6
Alveolar space	Macrophage	39	1	1	43	0,8	1	430	4,2	8,4	141	1,7	4,1	10	0,2	0,5	212	4,2	5,4	375	1,9	4,7
	Macrophage cryst.	0	0	0	0	0,0	0	0	0,0	0,0	0	0,0	0,0	0	0,0	0,0	467	9,3	12,0	0		0,0
	Neutrophil free	0	0	0	44	0,8	1	0	0,0	0,0	17	0,2	0,5	0	0,0	0,0	0	0,0	0,0	0	0,0	0,0
	Trombo free	0	0	0	8	0,2	0	0			0	0,0	0,0	0	0,0	0,0	0	0,0	0,0	0	0,0	
	Necrotic unknown	22	0	1	60	1,1	2	95	0,9	1,9	220	2,7	6,4	38	0,7	1,9	314	6,2	8,1	99	0,5	1,2
	Erythrocyte	2	0	0	22	0,0	1	0	0,0	0,0	0	0,0	0,0	0	0,0	0,0	0	0,0	0,0	15	0,1	0,2
	Empty	3509	53		2269	42,5		5115	50,0		4719	58,0		3198	61,0		1154	22,9		11952	59,7	
	Unknown	4	0	0	46	0,9	2	55	0,5	1,1	27	0,3	0,8	52	1,0	2,5	65	1,3	1,7	79	0,4	1,0
	Sample dam	0	0	0			0		0,0		0	0,0								0		
	total points	**6671**			**5333**			**10235**			**8135**			**5246**			**5047**			**20013**		
	total bez empty	**3162**	**100**	**###**	**3064**	**99,6**	**###**	**5120**	**100,0**	**####**	**3416**	**#####**	**100,0**	**2048**	**100,0**	**####**	**3893**	**####**	**100,0**	**8061**	**100,0**	**#####**
Ld/total fibroblas		0			0			0			1,13			0			5,806451613			3,96		

voxelsize 2x2x5um

		NC		2 DPI		5 DPI	
		points	%	point	%	points	%
Pneumocytes	Pneu I ok	521	16	2208	19,0	2304	16,5
	Pneu I dam	55	2	74	0,6	196	1,4
	PneuII ok	372	12	1047	9,0	1860	13,3
	PneuII dam	0	0	43	0,4	84	0,6
Capillary	Endoth. Cell	1052	33	2666	23,0	2768	19,8
	Endoth. cell dam	18	1	373	3,2	190	1,4
	Ery	271	9	588	5,1	865	6,2
	Immature ery	0	0	8	0,1	0	0,0
	Thrombocyte	39	1	130	1,1	113	0,8
	Immune cells	13	0	295	2,5	413	2,9
	Neutrophil	0	0	41	0,4	128	0,9
	Eosinophil	3	0	0	0,0	11	0,1
	Capillary lumen	162	5	382	3,3	270	1,9
Fibro-ECM	Lipofibroblasts	555	18	1916	16,5	2147	15,3
	Lipofibroblasts da	0	0	39	0,3	7	0,0
	Fibroblasts	0	0	0	0,0	27	0,2
	LD	0	0	8	0,1	79	0,6
	ECM	27	1	263	2,3	381	2,7
	Dense Fibers	7	0	127	1,1	121	0,9
	Collagen Fibers	0	0	184	1,6	312	2,2
Alveolar space	Macrophage	39	1	614	5,3	597	4,3
	Macrophage cryst.	0	0	0	0,0	467	3,3
	Neutrophil free	0	0	61	0,5	0	0,0
	Trombo free	0	0	8	0,1	0	0,0
	Necrotic unknown	22	1	375	3,2	451	3,2
	Erythrocyte	2	0	22	0,2	15	0,1
	Empty	3509		######		16304	
	Unknown	4	0	128	1,1	196	1,4
	Sample dam	0	0	0	0,0	0	0,0
	total points	**6671**		######		30306	
	total points without empty	**3162**					

$$\textbf{Relative area of interest}(\%) = \frac{\textbf{number of poins of material}}{\textbf{total number of points}} * \textbf{100\%}$$

$$\textbf{Absolute area of interest} = \textbf{relative area of interest} * \textbf{image size} * \textbf{pixel size}$$

“Fast(est)“ quantification for big datasets

Stereology

- Start with an image of known pixel size
- Apply a grid with known proportions
- Assign each cross-section to a „material“
- Count numer of dots of each material in the image
- Repeat for all images in the stack (or every Nth)
- Calculate „comparisonal“ stereology
- Extrapolate the data…

Segmentation – what is a segmentation?

- Process of partitioning of digital image into multiple segments/regions/objects – **giving every pixel a label**
  - **Pixels with certain characteristics share a label**
  - Goal: Simplify the image in order to facilitate reconstruction and analysis

https://upload.wikimedia.org/wikipedia/commons/d/d4/Image_segmentation.png

Segmentation

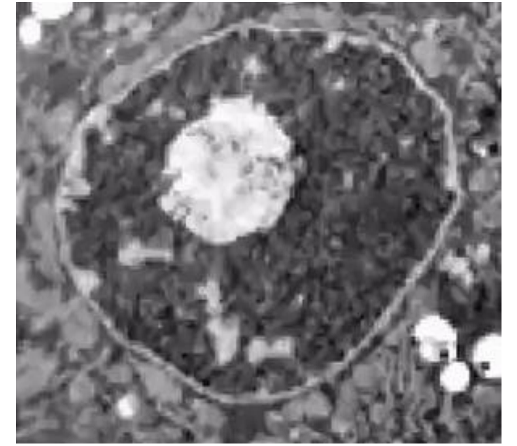
Brief explanation of how the segmentation works

- **In layers (similar to Photoshop):**
  - Image – **Base layer** – original data which we use to selet objects of interest on
  - Model – **Second Layer** – digital representation of object of interest in the image (different objects – different materials)
  - Mask – **Third Layer** – separate layer for selecting region of interest

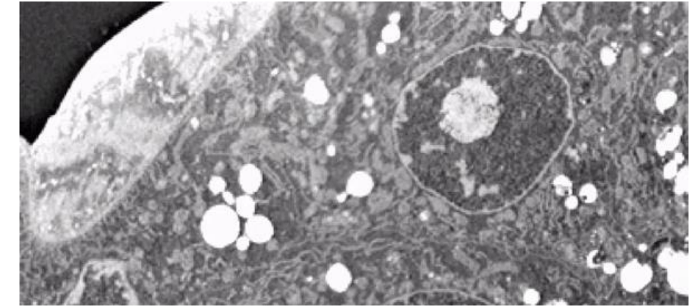
Segmentation

Most basic/manual segmentation tools

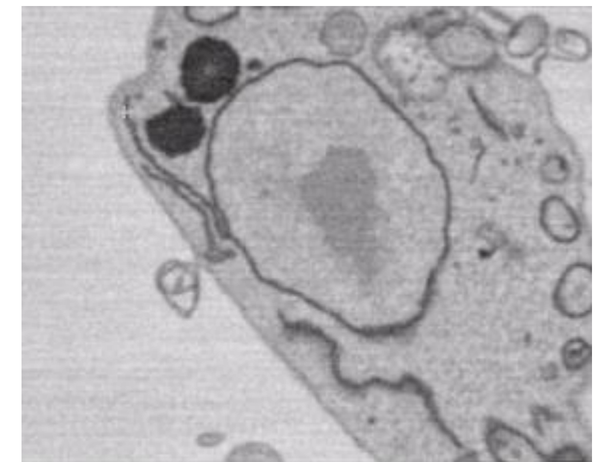
- **Brush** – Manual painting of regions of interest – similar to Microsoft Paint



- **B/W threshold** – selects pixels of corresponding parameters – in whole image/stack or in the selected area (mask/material)

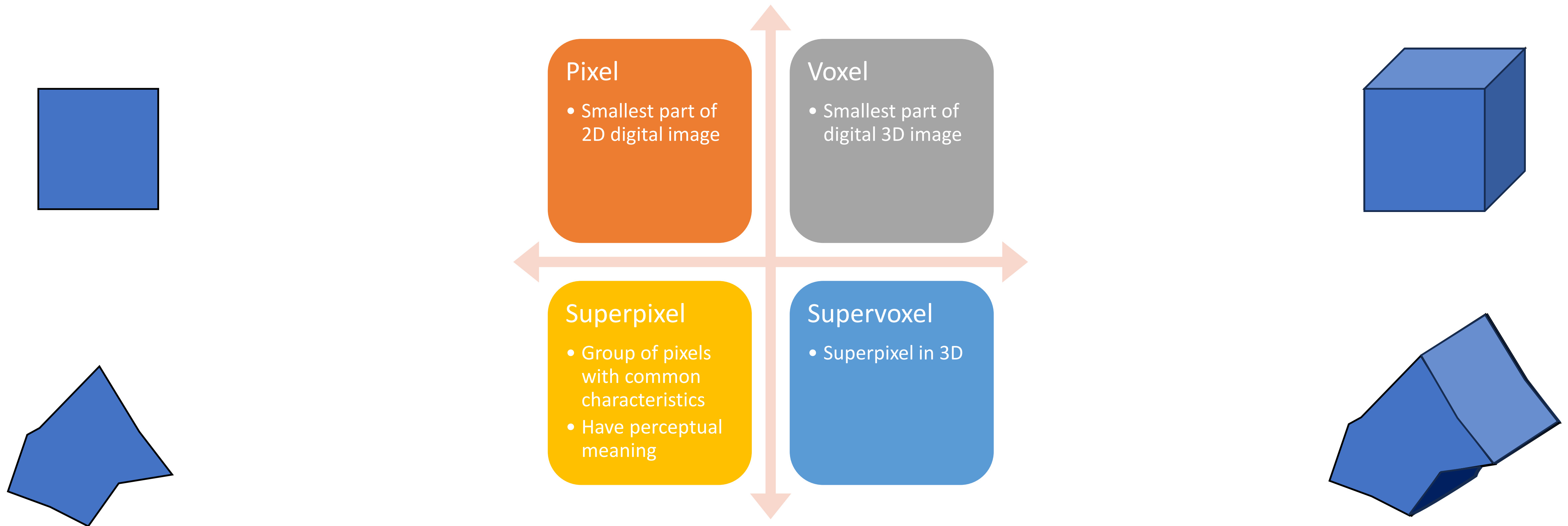


- **Membrane ClickTracker** – tool for following lines along the same or very similar pixel intensity



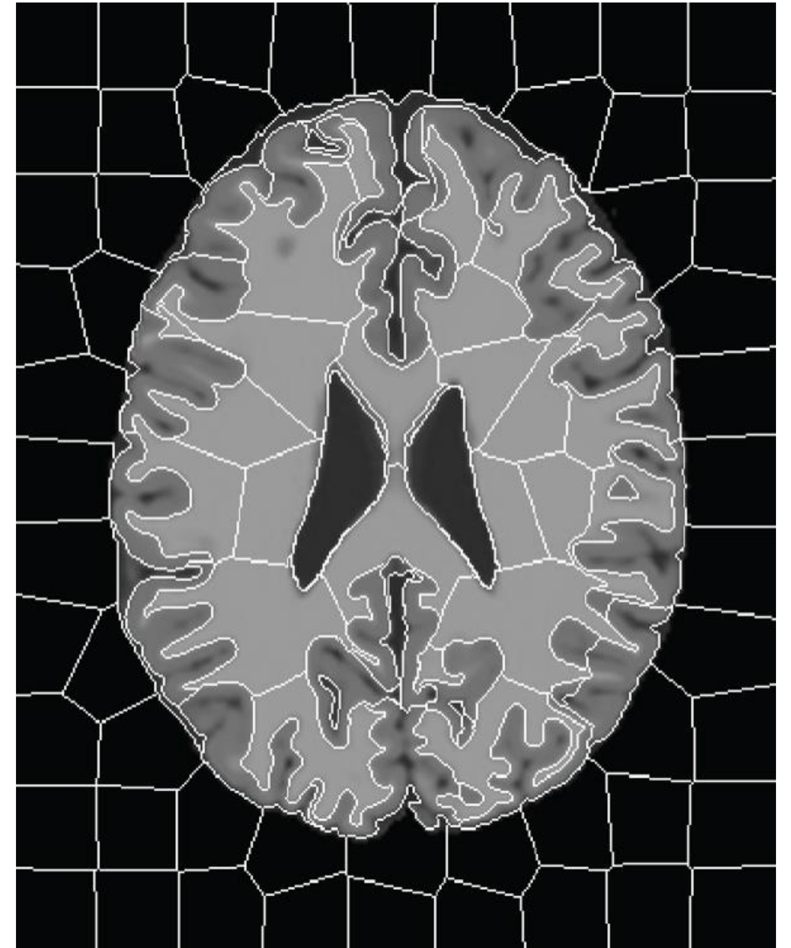
They still have their uses but for big datasets are slow – we need something more powerful!

Superpixel clustering used for semiautomatic segmentation



Superpixels

- Compact and coherent regions obtained by grouping pixels together based on their similarities in color, texture, and spatial proximity.
- Higher-level representation of an image by reducing the number of pixels while preserving important image structures.
  - Each superpixel represents a region of similar characteristics.
  - Less pixels→less datapoints→faster segmentation
  - Boundary adherence →more precision than pixel-based segmentation



Zhen Yu, et al 2021

Superpixel clustering algorithms

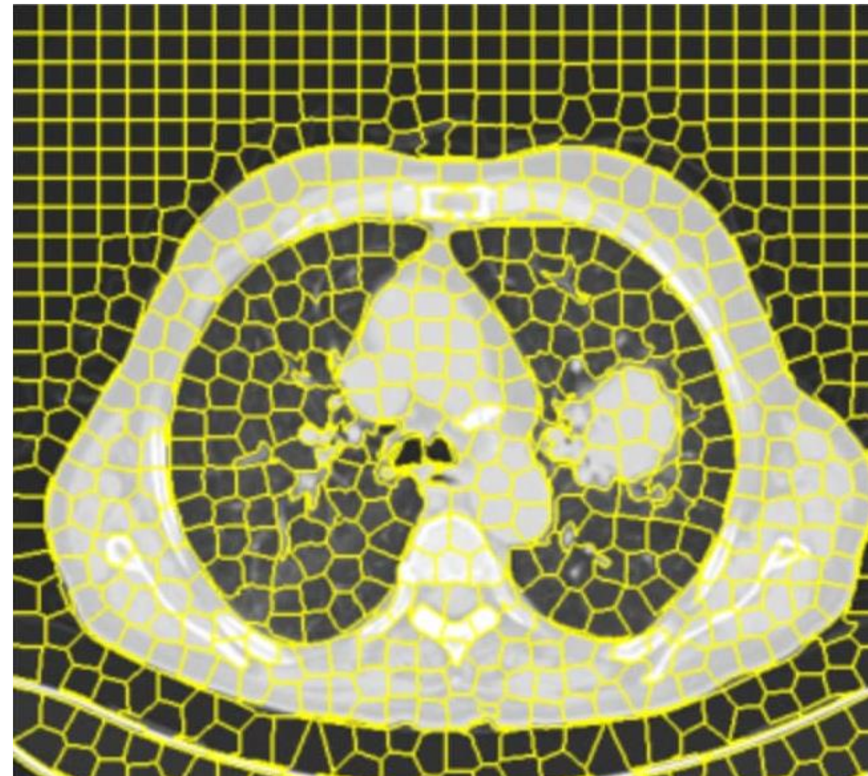
Felzenszwalb´s Algorithm

Quickshift

Watershed

Simple Linear Iterative Clustering (SLIC)

- Clustering of pixels in color-space and spatial domain
  - Similar pixel intesities that are close together
  - Regular shaped superpixels
- **Useful for superresolution and electron microscopy**



https://forum.image.sc/t/need-to-find-out-the-order-in-which-individual-superpixels-are-numbered-on-the-image/66535

Superpixel clustering algorithms

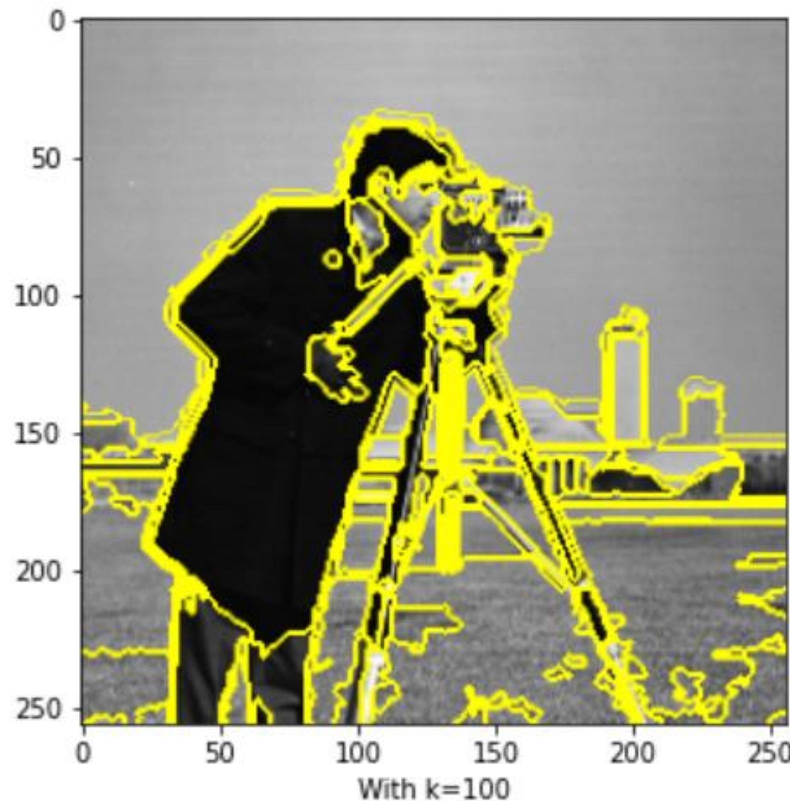
Simple Linear Iterative Clustering (SLIC)

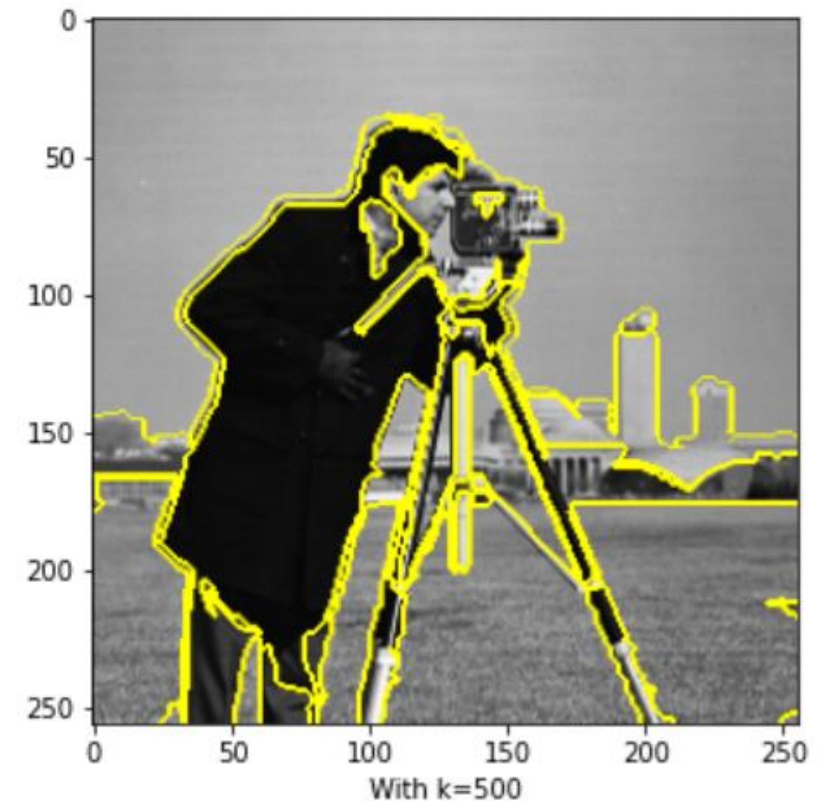
Quickshift

Watershed

Felzenszwalb´s Algorithm

- Clustering according to criterion of similarity
  - Irregularly shaped superpixels
  - Not so useful EM and superresolution microscopy





https://www.analyticsvidhya.com/blog/2021/05/image-segmentation-with-felzenszwalbs-algorithm/

Superpixel clustering algorithms

Simple Linear Iterative Clustering (SLIC)

Felzenszwalb´s Algorithm

Watershed

Quickshift

- Clustering according to color-space and feature-space
  - Pixel size varies
  - Not so useful for EM and superresolution microscopy

(a)

(b)

https://www.researchgate.net/figure/Superpixel-segmentations-by-quick-shift-a-shows-raw-image-b-shows-quick-shift_fig5_339754839

Superpixel clustering algorithms

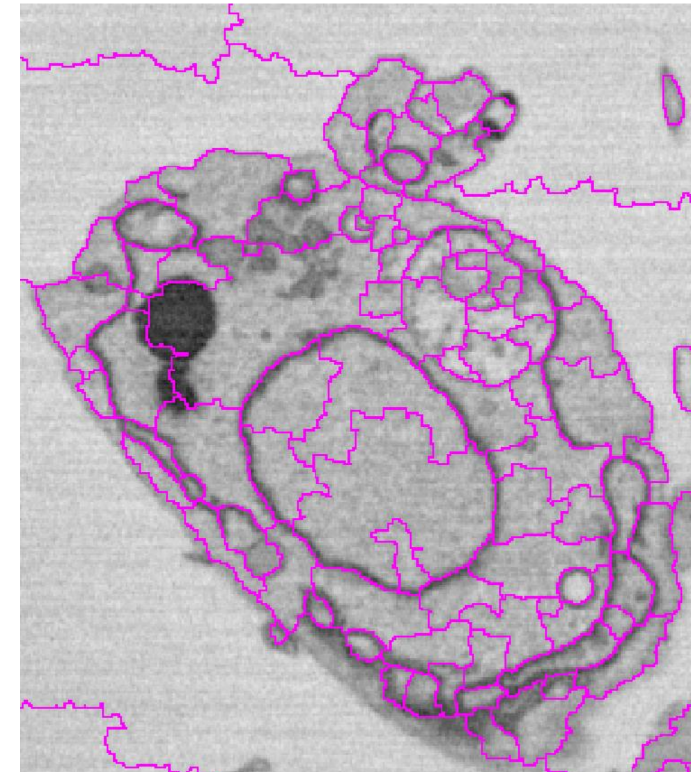
Simple Linear Iterative Clustering (SLIC)

Felzenszwalb´s Algorithm

Quickshift

Watershed

- Treats image as a topgraphic surface
- Clustering according to the pixel intensity gradient (flooding valleys)
    - More contrast, the higher the hils – the more water is necessary to overcome them
- **Useful for membrane bound organelles**



Superpixel clustering algorithms

Simple Linear Iterative Clustering (SLIC)

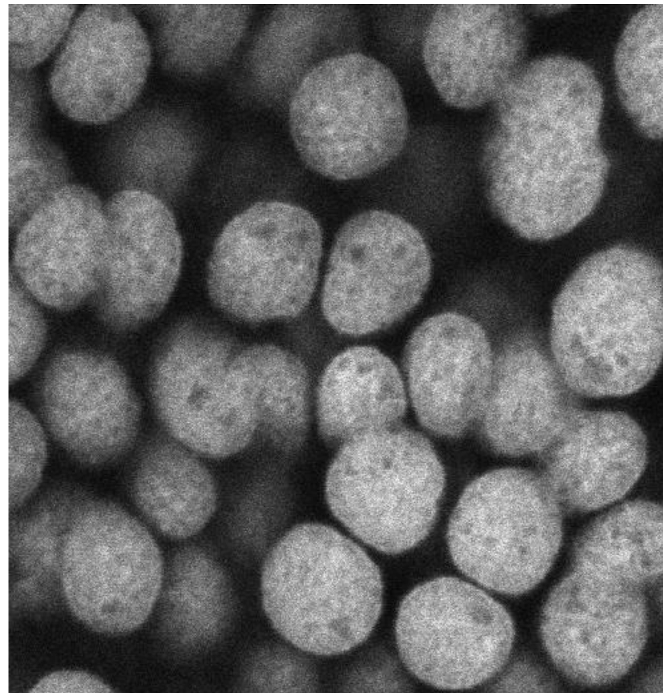
Felzenszwalb´s Algorithm

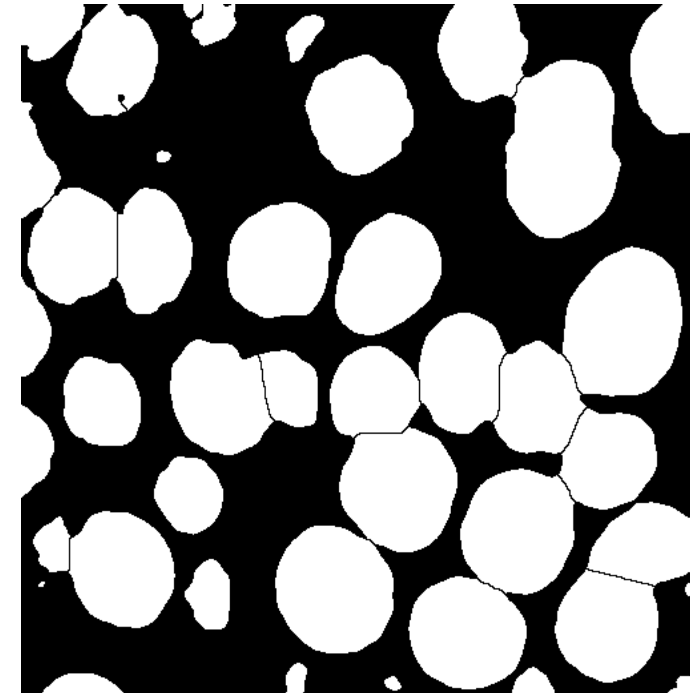
Quickshift

Watershed

- Another use is the separation of touching objects:

1. Convert image to Binary
2. Finds centers of objects by eroding the mask
3. Starts filling the objects with imaginary water
   a) When 2 watersheds meet – the separation is created





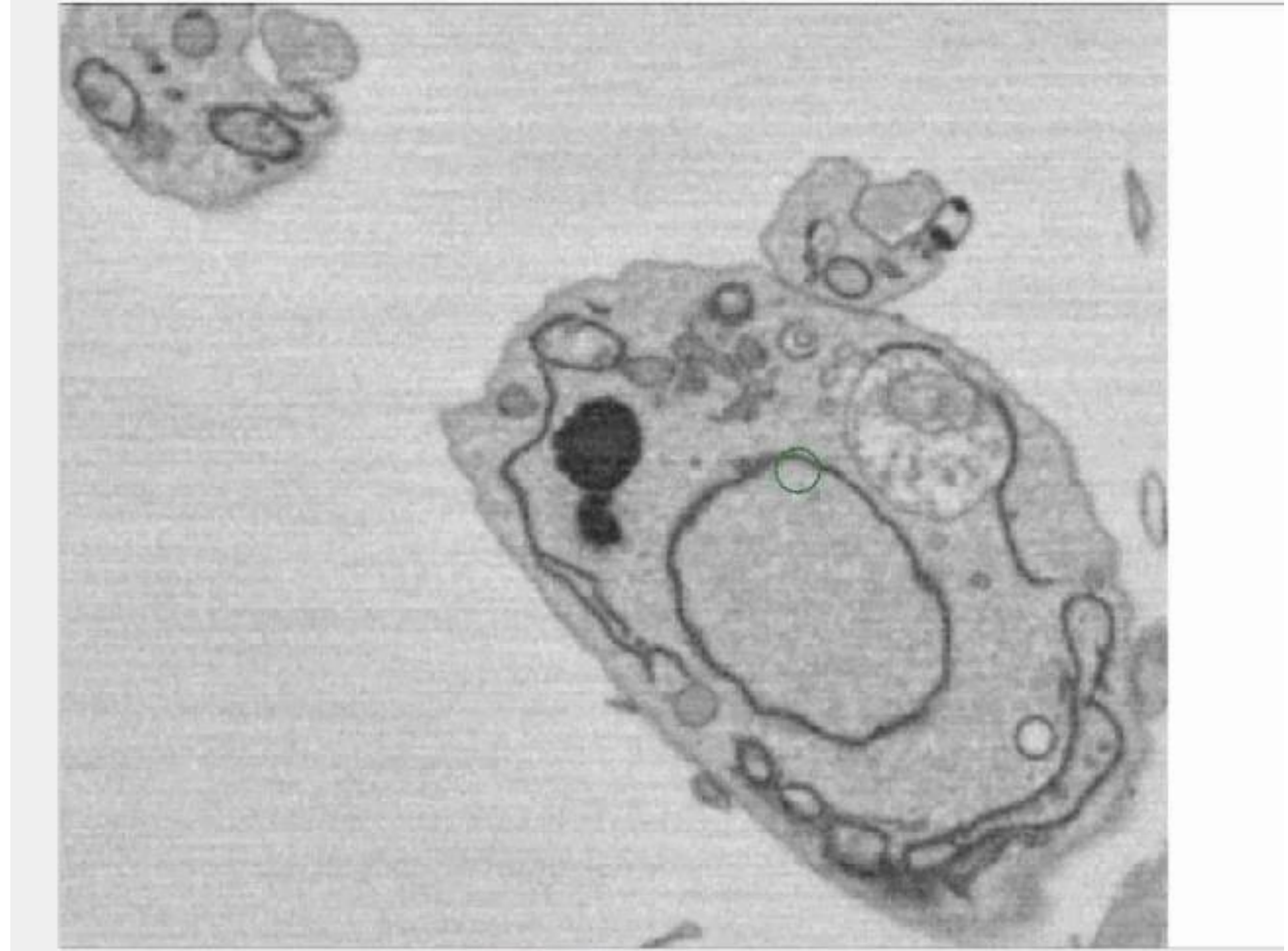
https://imagej.net/imaging/watershed

What to do with the superpixels?
- Manual and semi automatic segmentation

Manual pixel brush vs superpixel brush

Manual brush segmentation
 - lengthy, tedious, imprecise

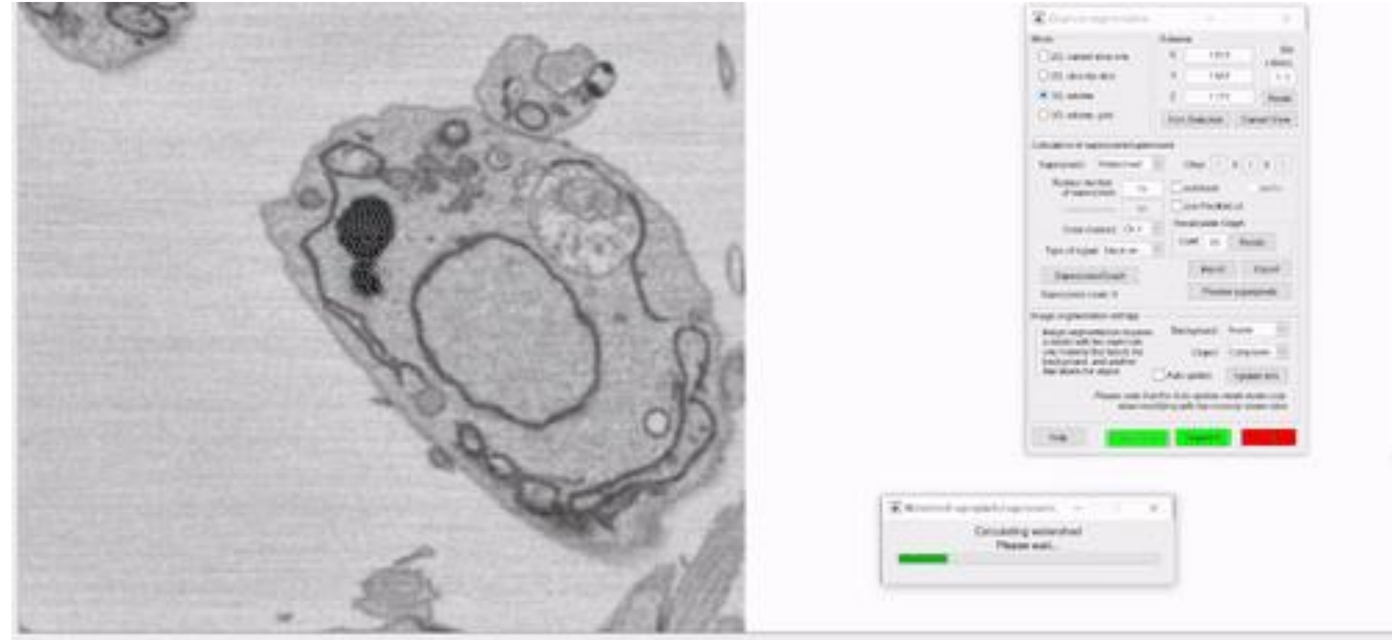
Superpixel brush segmentation
 - with propper clustering
 - fast, precise
 - there is no sigle optimal clustering that works perfectly for everything



What to do with the superpixels?
- Manual and semi automatic segmentation

Graphcut semiautomatic segmentation

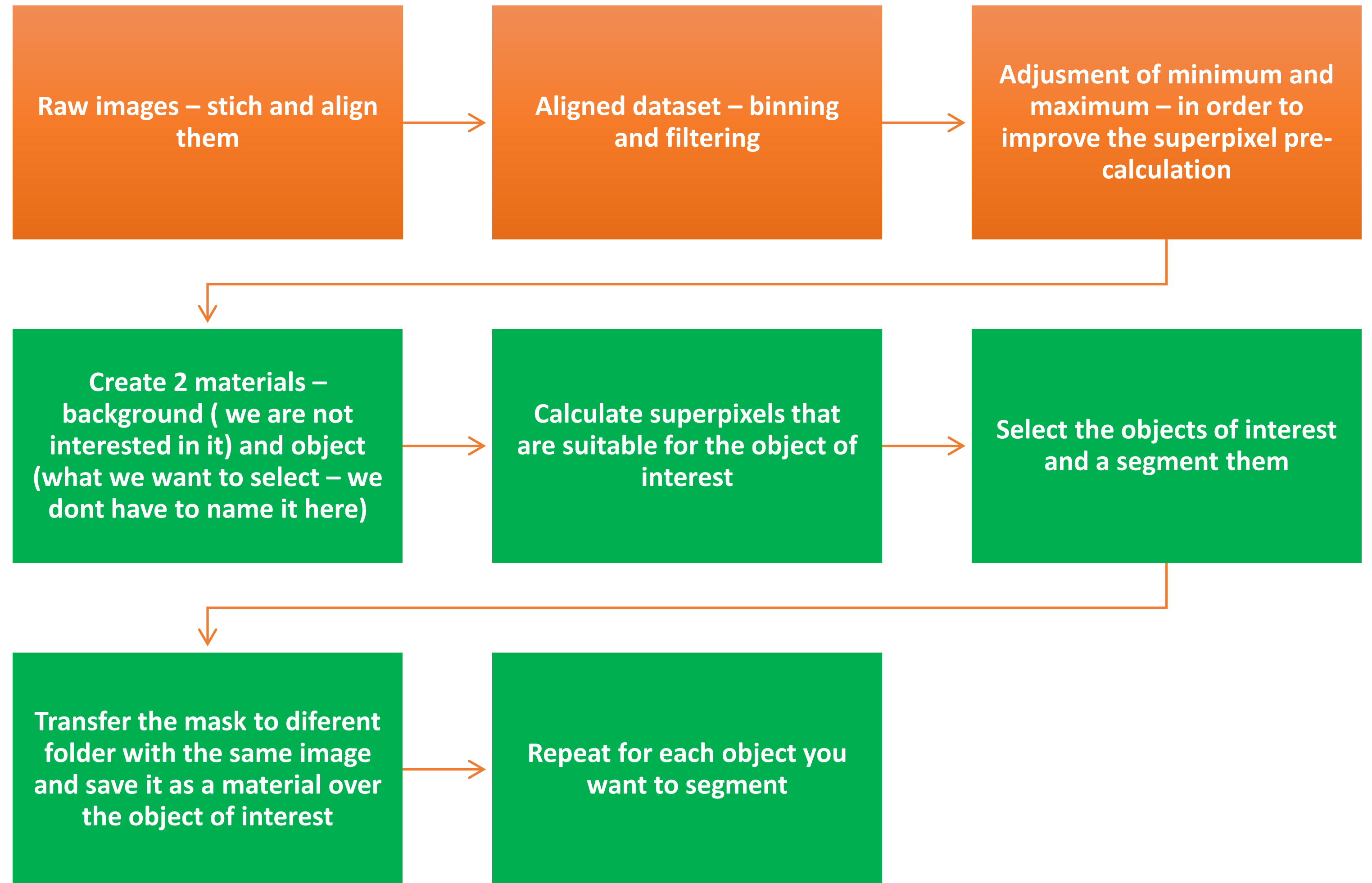
- Pre-calculates the superpixels/supervoxels in whole image/stack – oversegmentation
- Graph representation of an image is created
  - Superpixel is represented as node in the graph, coneced by edges to another superpixel – node
    - The weight of each edge is dependent on the dissimilarity of neighboring superpixels – more dissimilarity – less weight
- Cuts the edges in order to separate image into background and desired object (segments)
  - The cut tries to find optimal path in with minimal energy cost and/maximizing desired criteria, such as boundary adherence or pixel intensity coherence

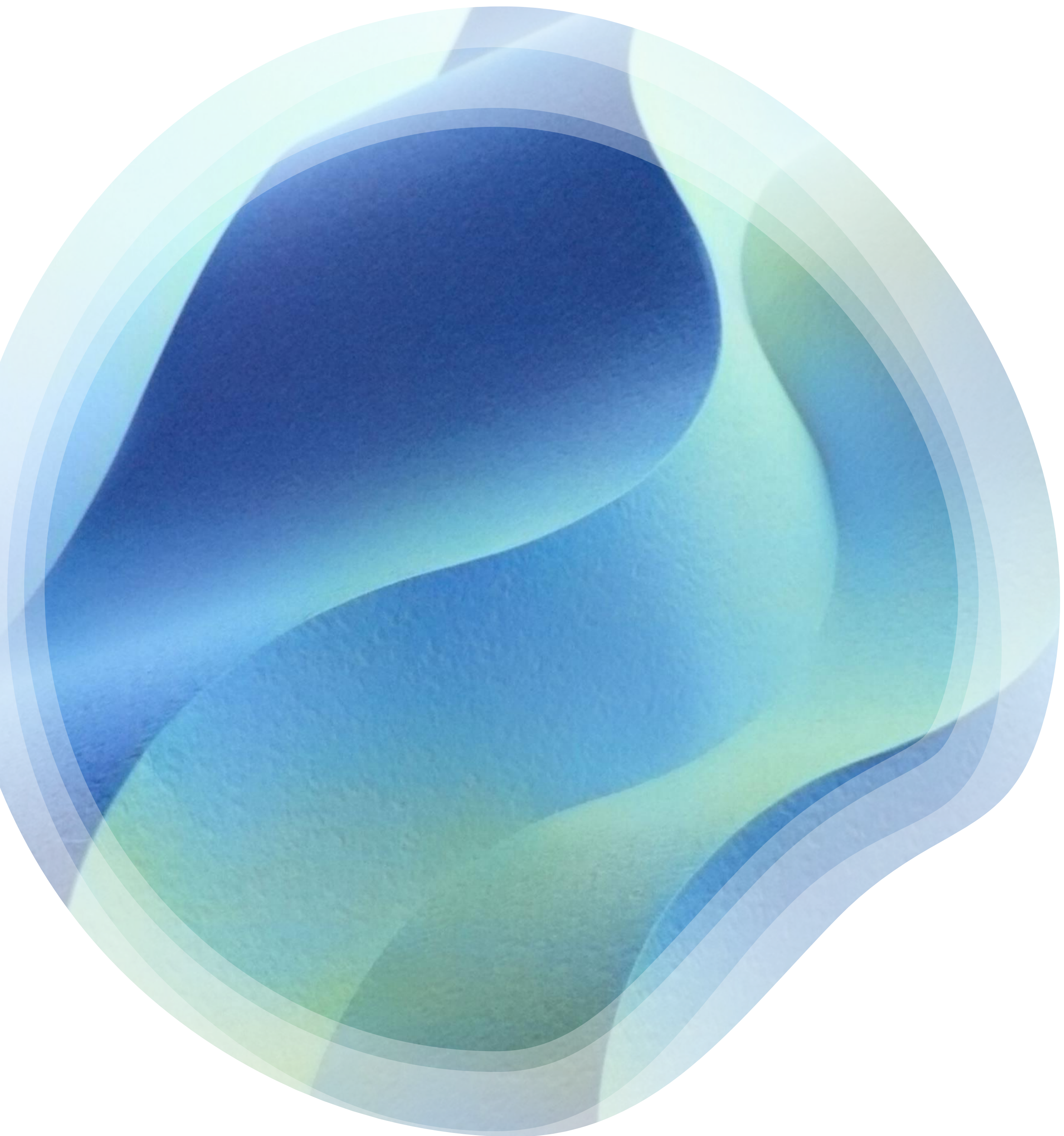


With a good data, you can have segmentation of individual organelles/image parts in couple of clicks
Most commonly works in binary

Segmentation

Segmentation pipeline with MIB Graphcut





Data visualization

What to do next?

Statstics

- Lengths, areas, surfces, volumes, counts
- Not visual
- Scientifically important data

Model rendering

- Nice visual representation of your laborious work
  - And to have better image what actually goes on
- Good to show to the public
- Not so scientifically important

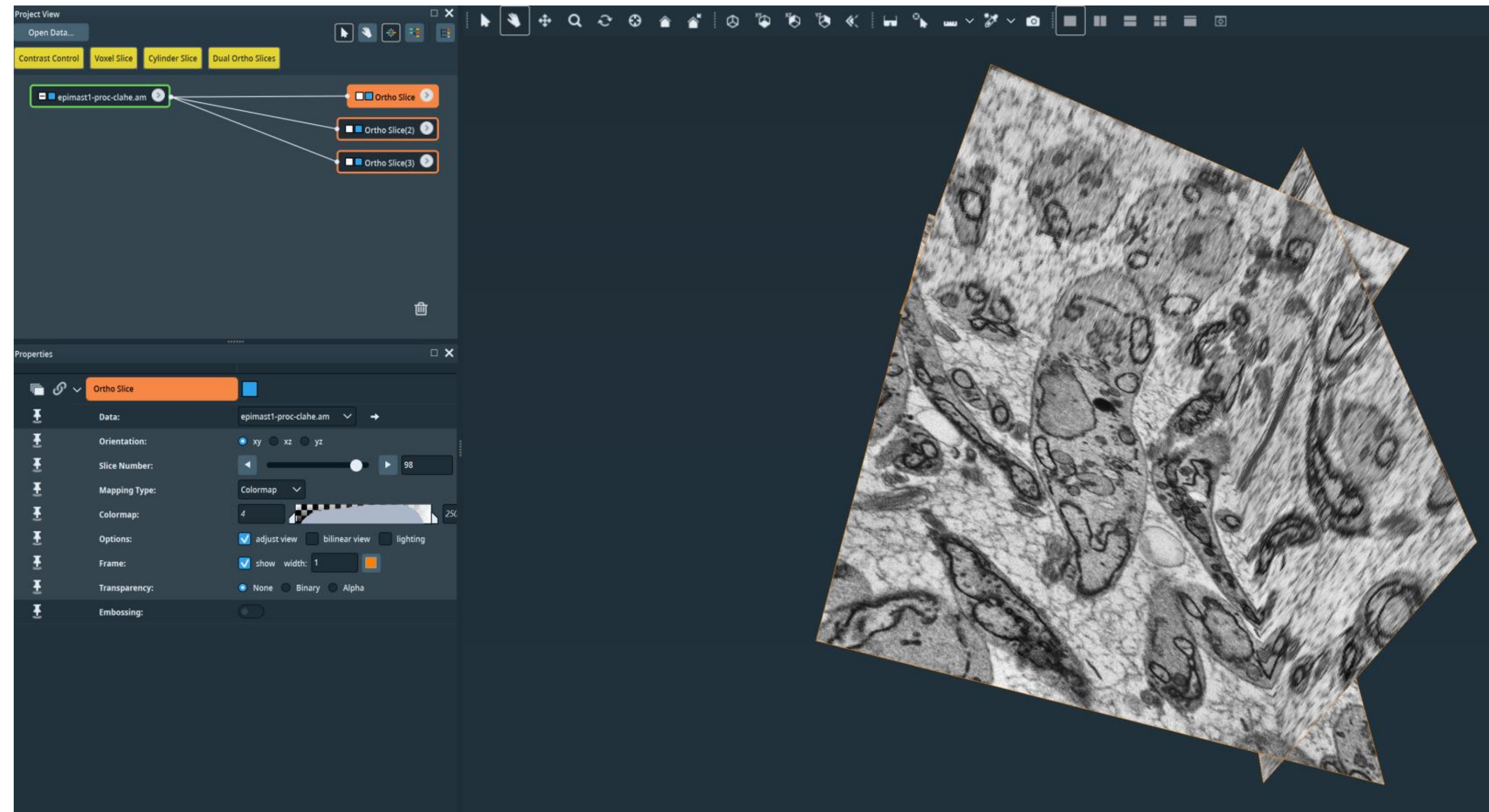
Model Rendering

- The segmenting software usually have way to visualize what you segmented or you can use some other software
  - MIB – VolumeViewer, Imaris viewer (free), ImageVis3D, **AMIRA Avizo**

Data visualization

Object visualization in Amira - orthoslice

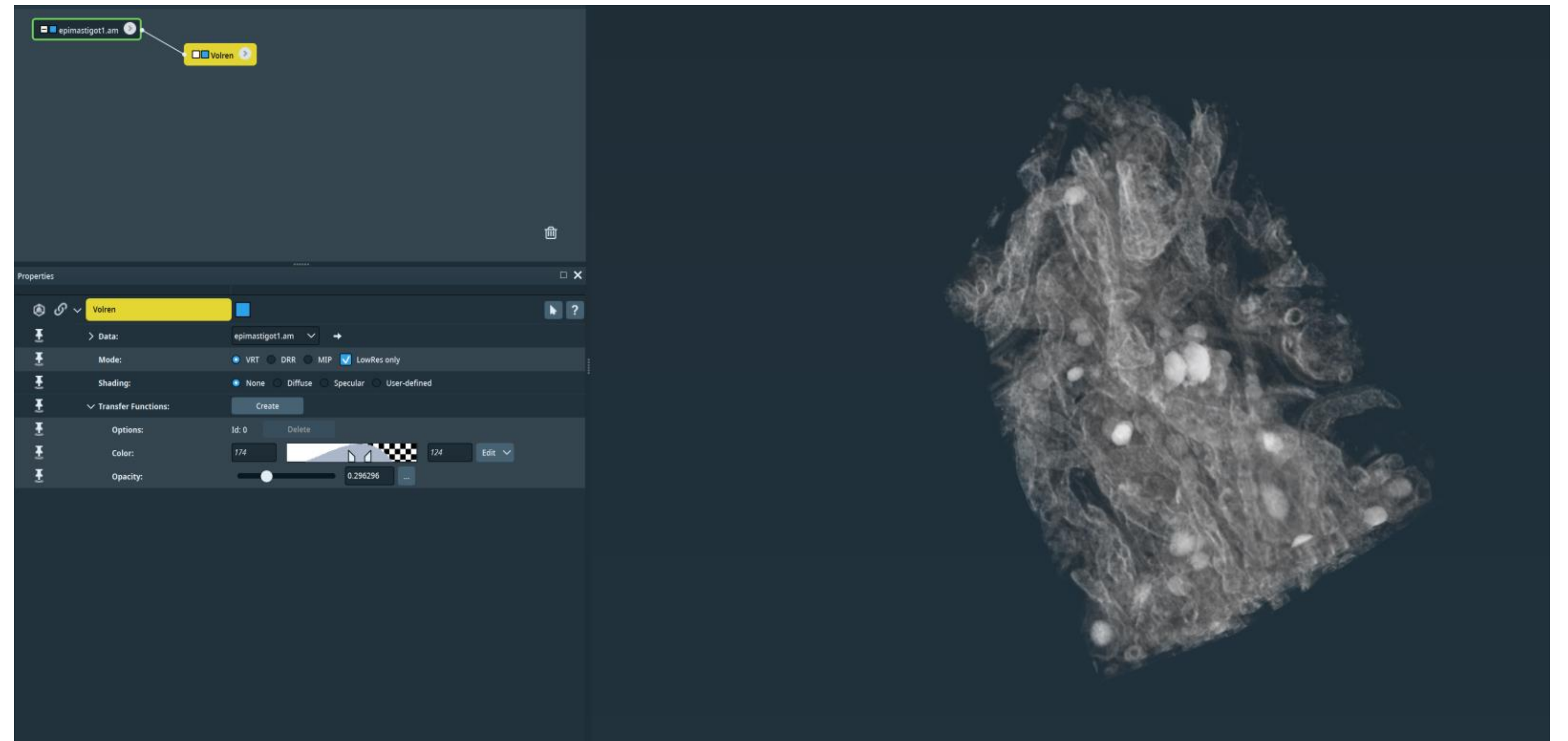
- No segmentation is necessary
- Simply look through the stack on all 3 axes



Data visualization

Model Rendering in Amira - volren

- Simplest and fastest visualization, if the dataset has good contrast
- No segmentation is necessary

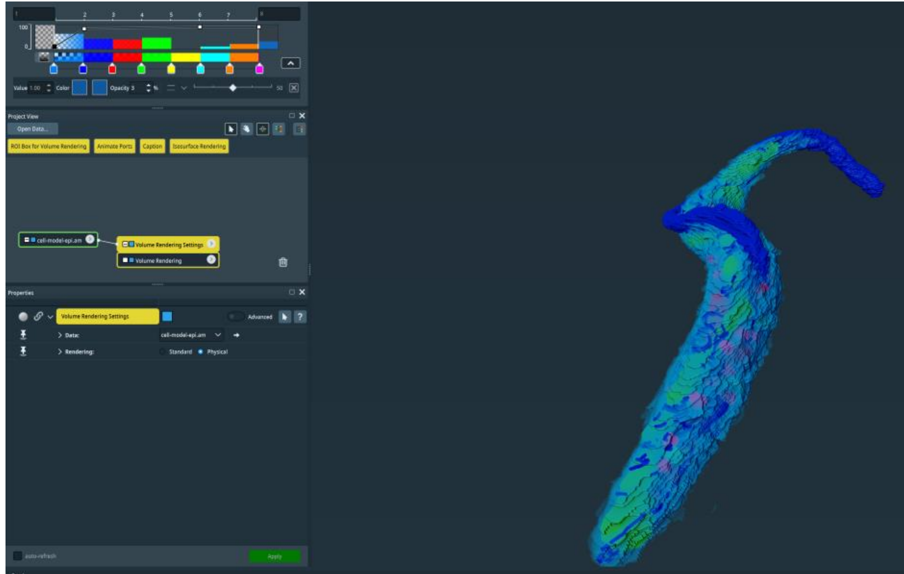


Set the minimal values, maximal values, and opacity to visualize objects of interest

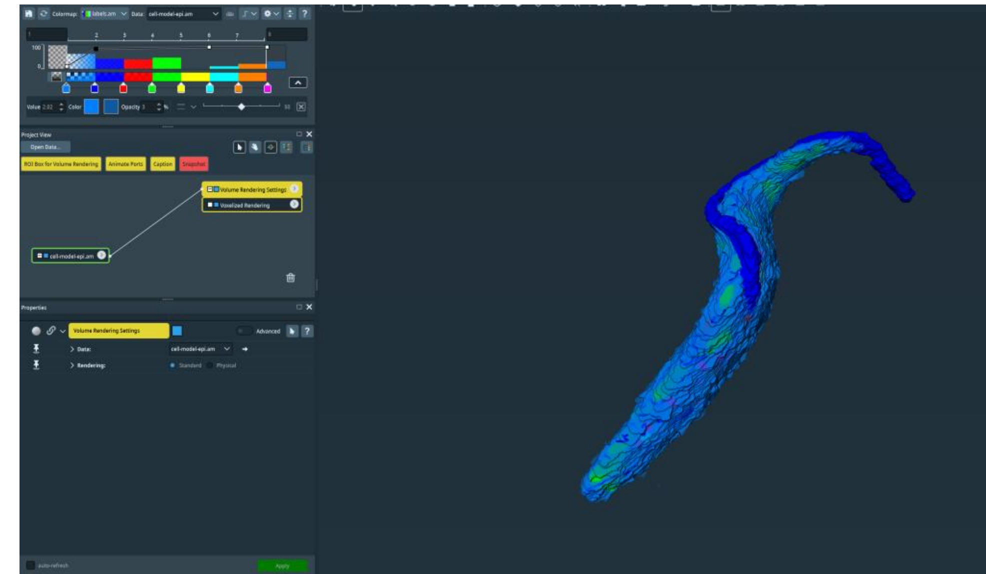
Data visualization

Model Rendering in Amira – volume rendering/ voxelized rendering

- Already requires segmented sample



Volume rendering

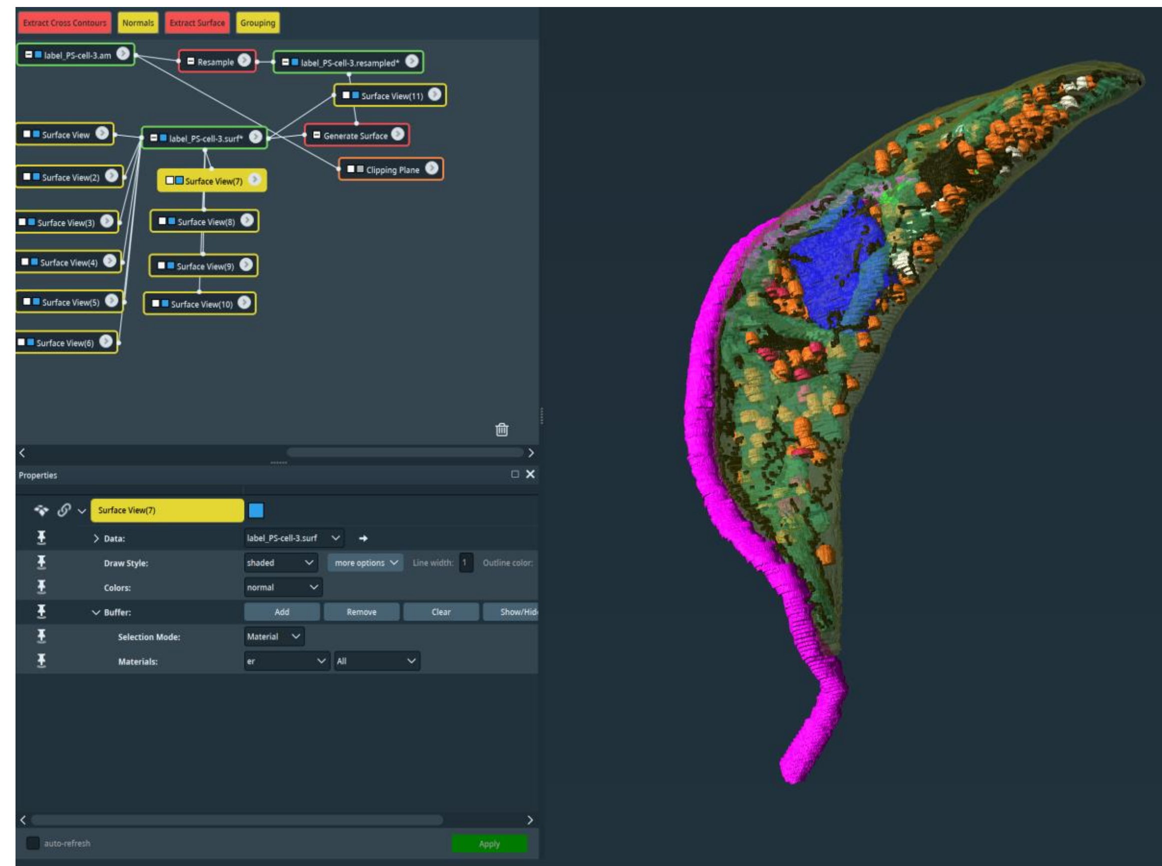


Voxelized rendering

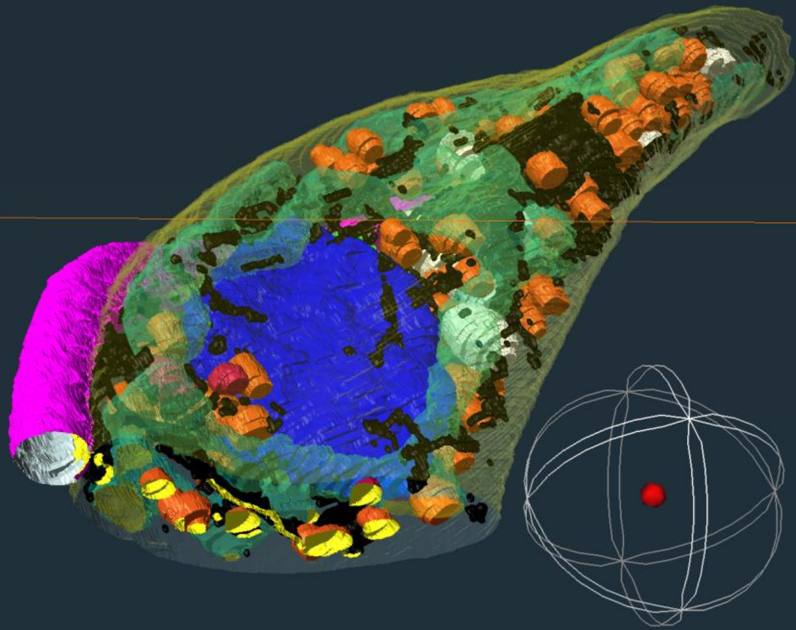
Data visualization

Model Rendering in Amira – Surface generation

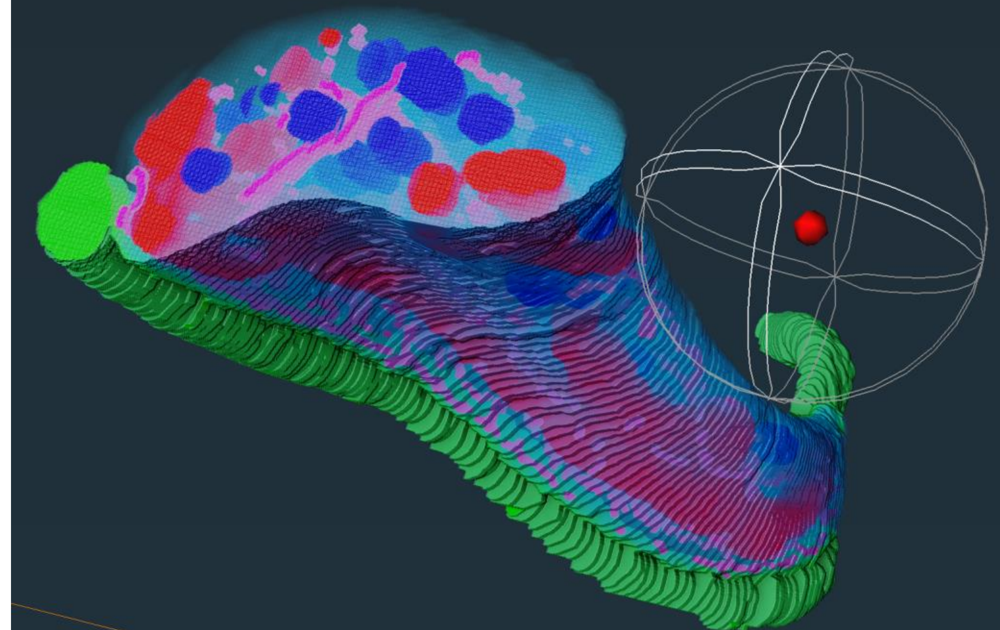
- Already requires segmented sample
- Bit more complex, but more variable
  - Allows smoothing
  - Each material can be visualized individually – much more responsive transparency
  - More laborious – each material is set individually



Model Rendering in Amira – Surface generation difference between volume and surface



Surface – cut

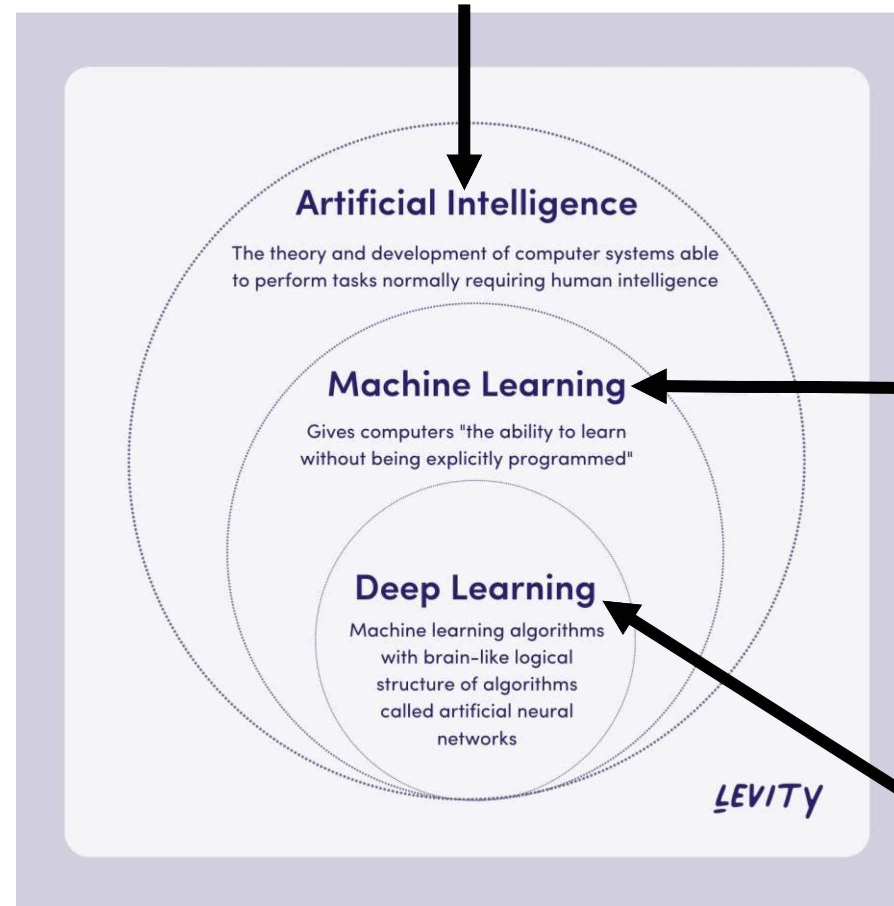


Volume – cut

Data visualization

What if the dataset is much bigger?

Segment Anything Model (META)



Ilastik, Arivis

Deep MIB, Apeer, Amira

Artigficial Inteligence

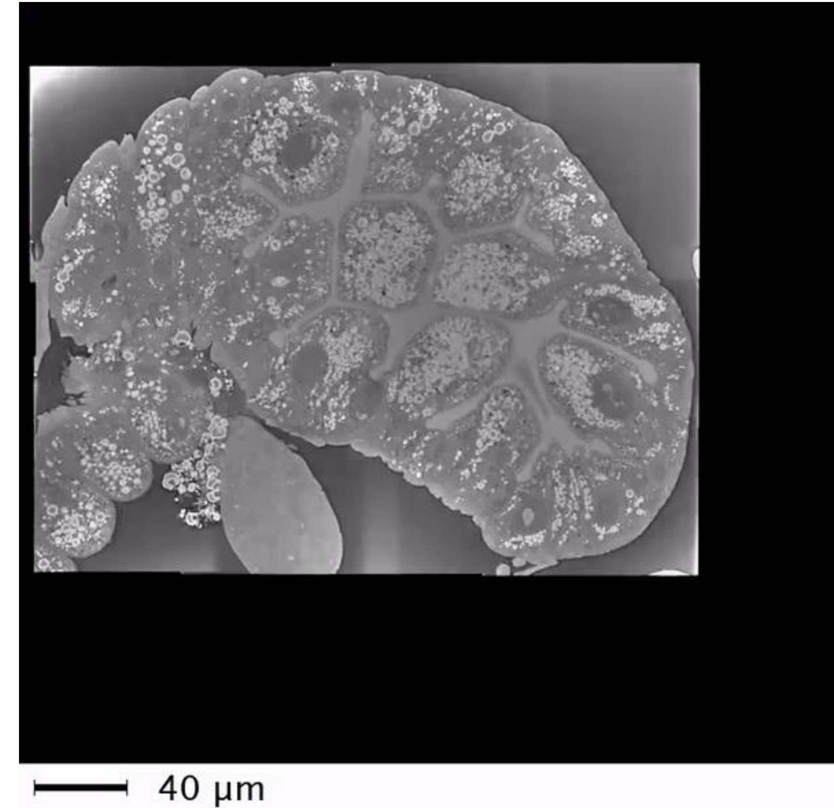
DeepMIB – actual neural network segmentation – GPU Based

- **Manually segmented part of a dataset serves a textbook**
  - **The network learns about particular type of data and can be used on similar ones**
  - The images of the stack are split with the corresponding labels into 2 files
    - Training
    - Validation
  - The software then trains on the training data and checks how it is doing on the validation data
    - From tens of minutes to days (depending on the complexity of segmentation, size of the dataset, GPU
- When the training is finished the learning is tested on „unseen“ part of the dataset
- If the predictions are good – predict the rest of the dataset

DeepMIB - neural network segmentaion – tick gut

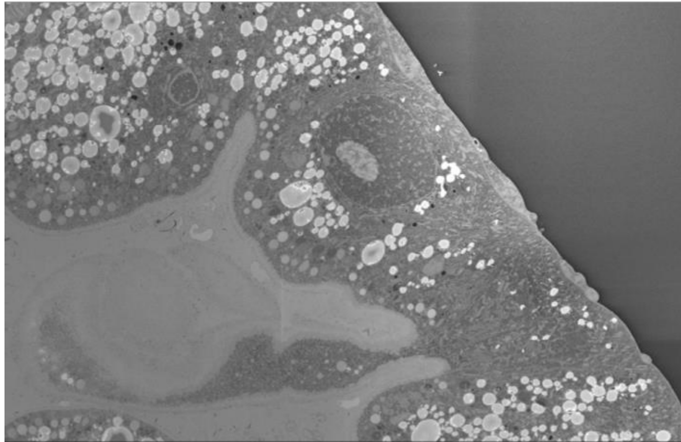
- 23450px x 21750px x 558sections
- Resolution – 15nmx15nm*120nm
- (351.8μm x 326.3μm x 156.2μm=0.018mm^3)
- 265 Gb

- Cooperation with scientific group (BC CAS): J. Perner

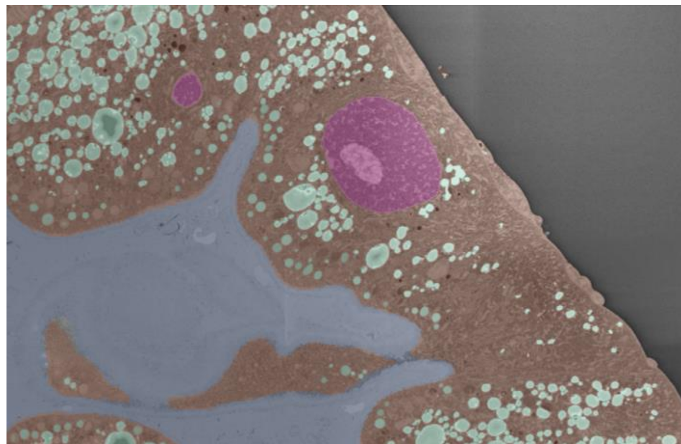


DeepMIB actual neural network segmentaion

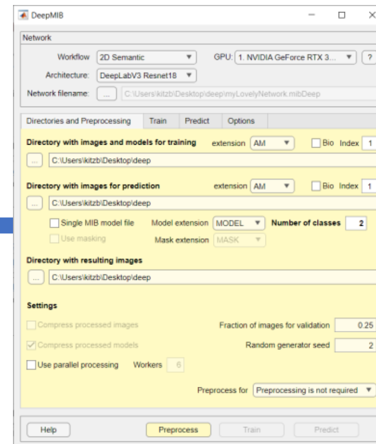
Part of the dataset to be segmented:
6750px x 4300px x 100 sections

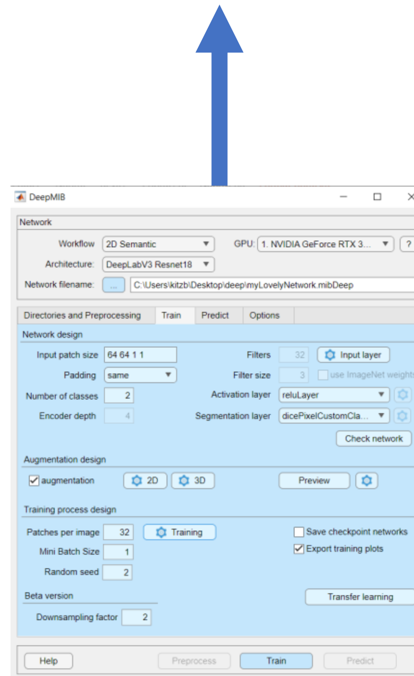


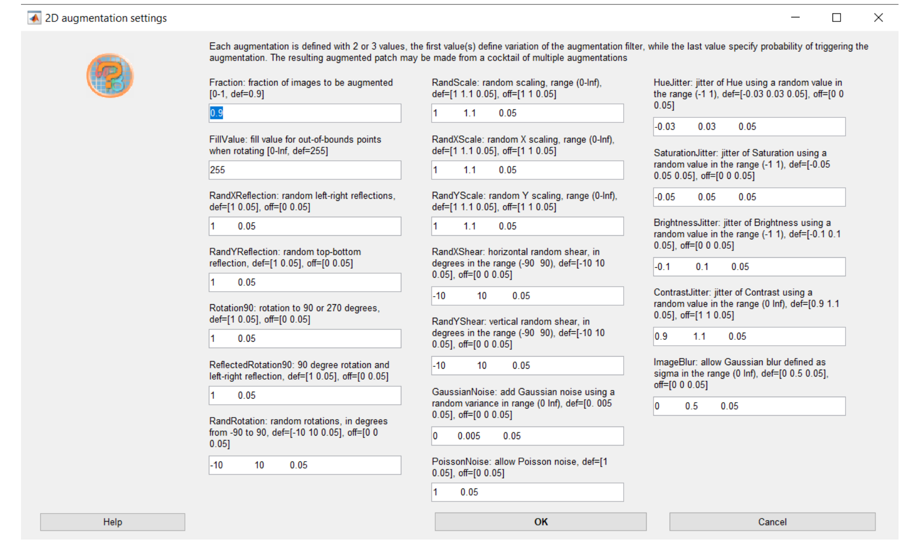
Facilitated segmentation with semi automatic tools using the superpixel clustering, thresholding...

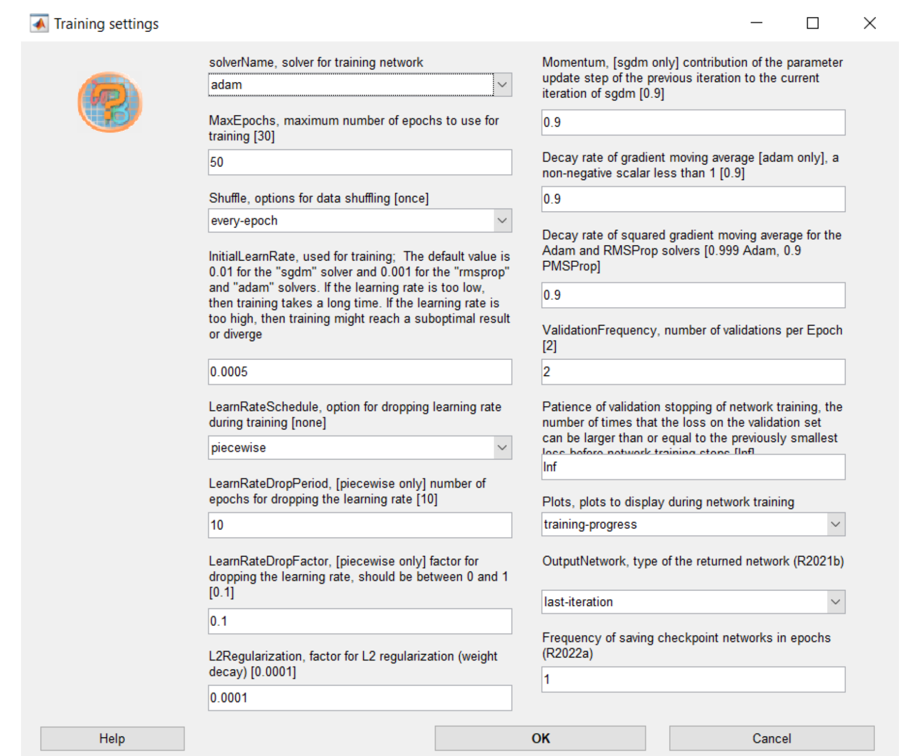


Roughly 10 days of work



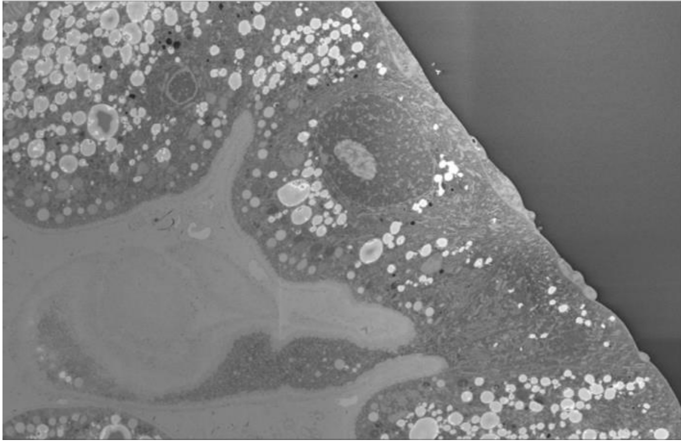




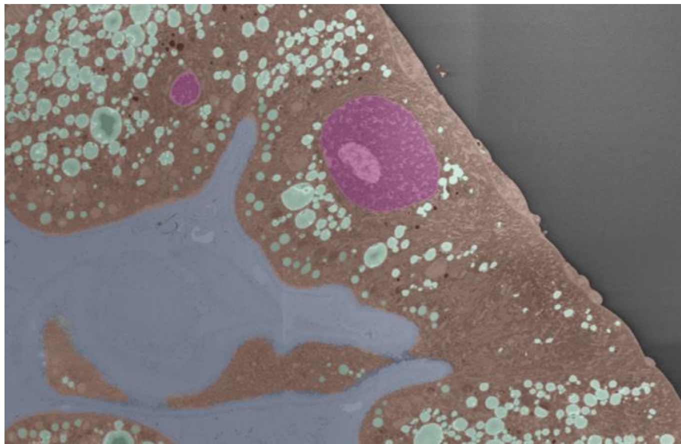


DeepMIB actual neural network segmentaion

Part of the dataset to be segmented:
6750px x 4300px x 100 sections



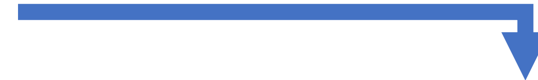
Facilitated segmentation with semi automatic tools using the superpixel clustering, thresholding...

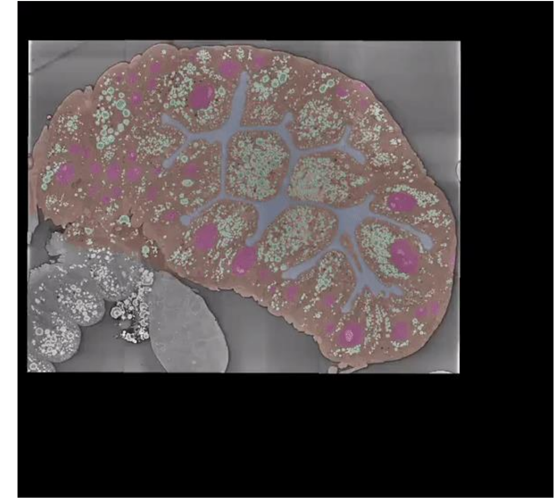


Roughly 10 days of work

Roughly 4 days of network learning
and couple of hour of prediction

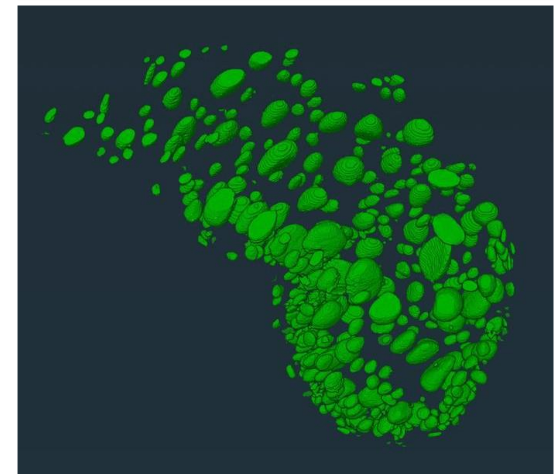
Okay, some more time for correction of prediction (2-3days)





40 µm

DeepMIB segmented dataset



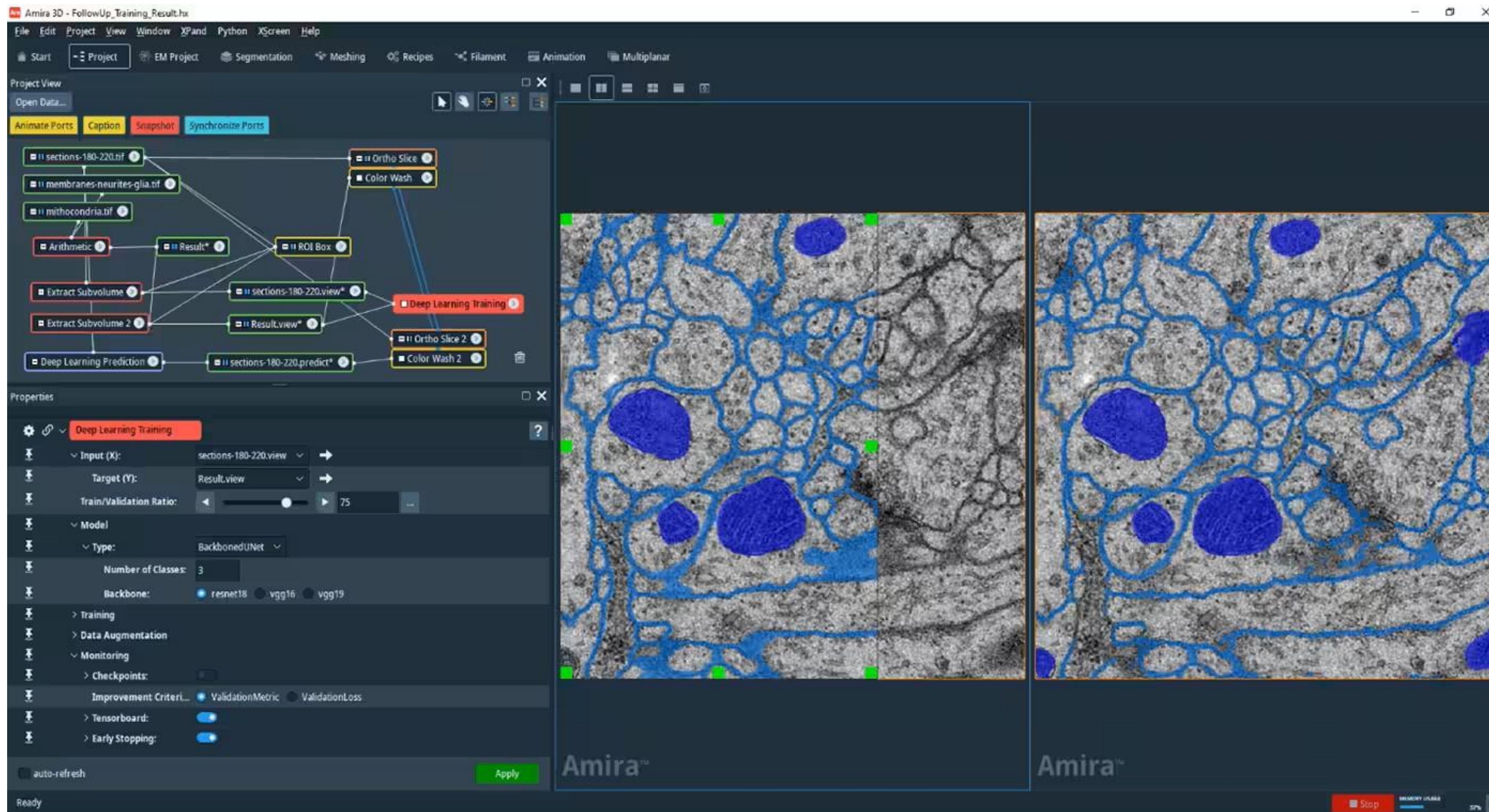
AMIRA rendered model from the DeepMIB segmented dataset

TAKEAWAY MESSAGE: The segmentation of dataset of this size just by basic tools would take years. With help of semiautomatic tools and DeepLearning this was done in a few weeks.

DATASET size: 23450px x 21750px x 558sections
351.8µm x 326.3µm x 156.2µm

Deep learning module in AMIRA

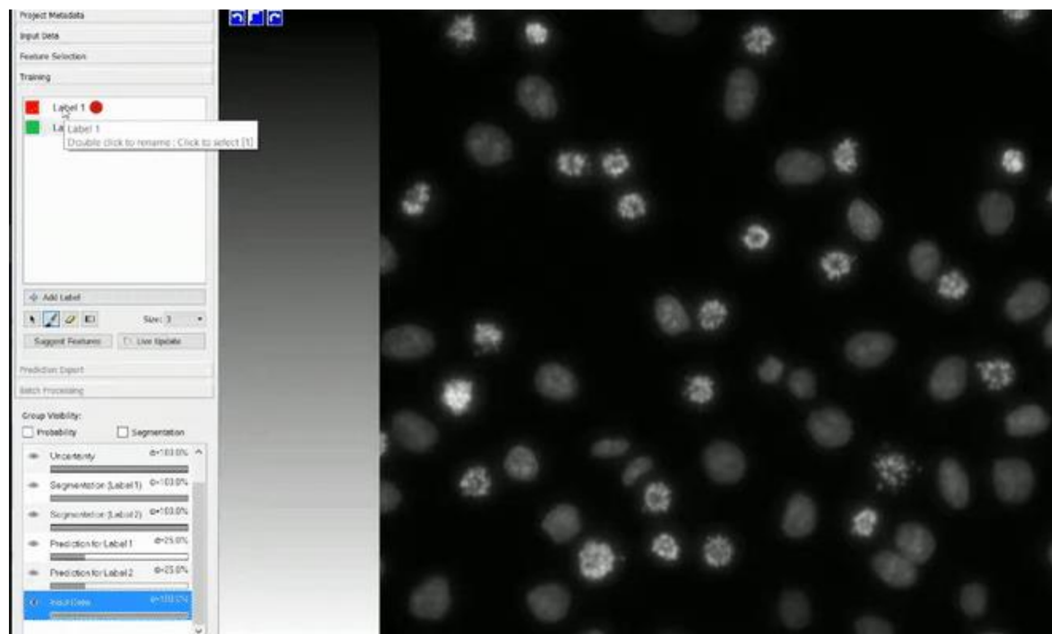
- Straightforward – but requires to buy an expensive package to even more expensive software

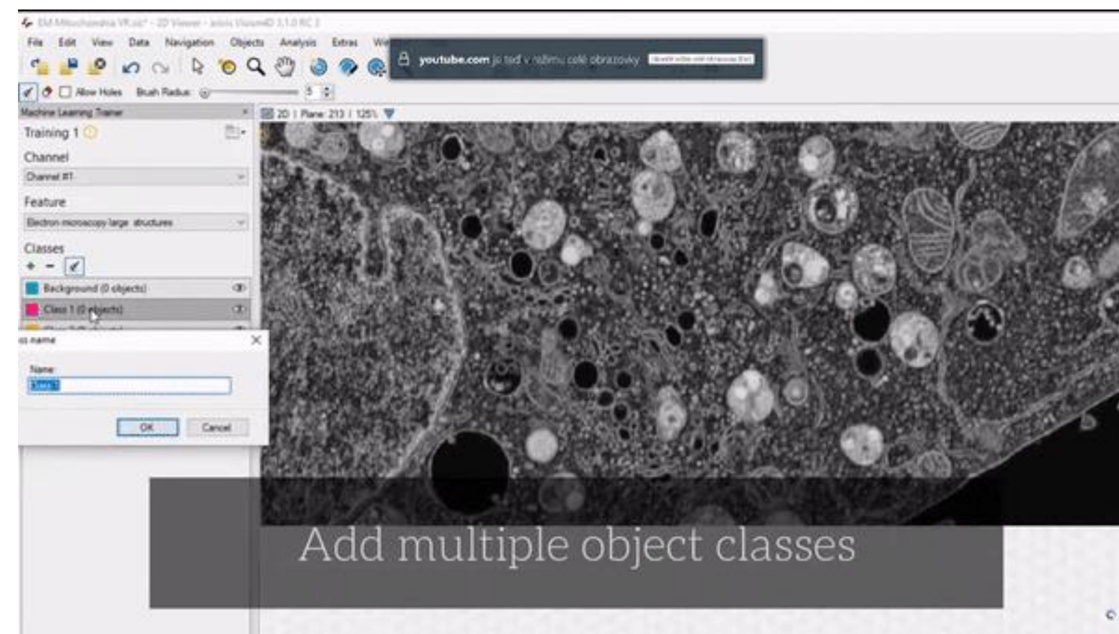


Artigficial Inteligence

Machine learning – Ilastik and Arivis

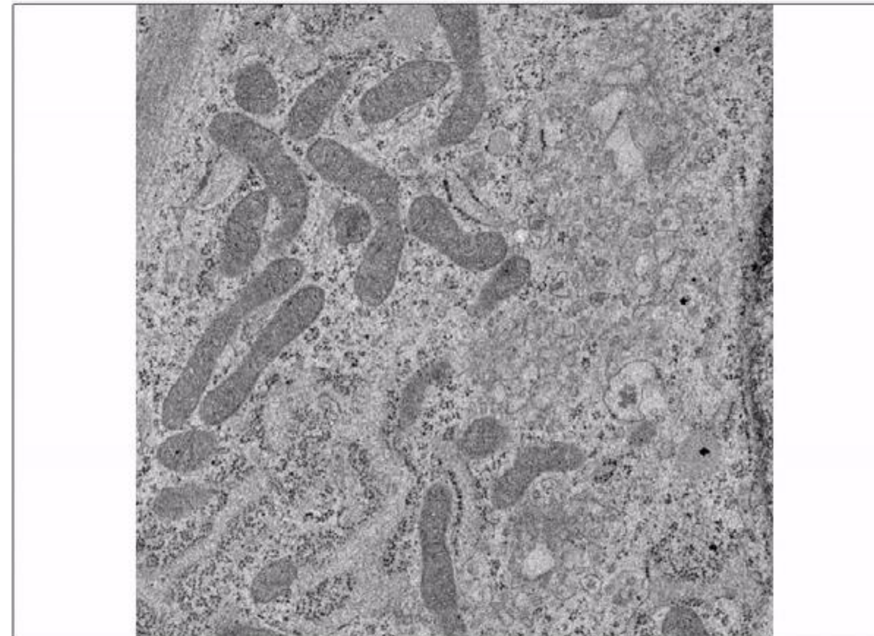
- **„Manual segmentation“ for machine learning can be more like an interactive lecture than a textbook**
  - You are selecting pixels that are of structures of interest and the program tries to reproduce what you do in the rest of image
    - And you correct it

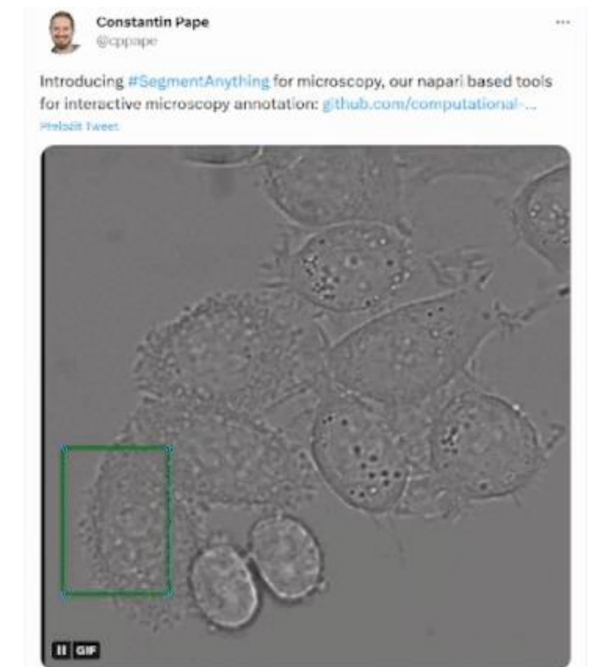




Future look of segmentation Using AI

- **Implemetation of artificial inteligence into segmentation processes**
  - **SAM from META**





Still with me? Good Job!

Thank you for your attention!

Laboratory of Electron Microscopy
Institute of Parasitology, BC CAS

Marie Vancová
Jana Nebesářová
Petra Masařová
Martina Tesařová
Jana Kopecká
Jan Langhans
Jiří Vaněček
Tomáš Bílý
Zdeno Gardian
Eva Ďurinová
Andrea Hatmanstorfer
Valentina Hawlicek
Ana-Maria Coroianu
And least I forget:
Jiří Týč

Laboratory of Molecular Biology of Ticks

Institute of Parasitology, BC CAS
Jan Perner
Veronika Urbanová

BIOLOGY CENTRE ASCR
INSTITUTE OF PARASITOLOGY
Biology Centre, ASCR
University of South Bohemia in České Budějovice
Faculty of Science
JKU JOHANNES KEPLER UNIVERSITÄT LINZ
CZECH BIOIMAGING
Imaging principles of life
EURO BIOIMAGING
Image life, discover the future