

Bio-Image Data Science

Robert Haase

GEFÖRDERT VOM



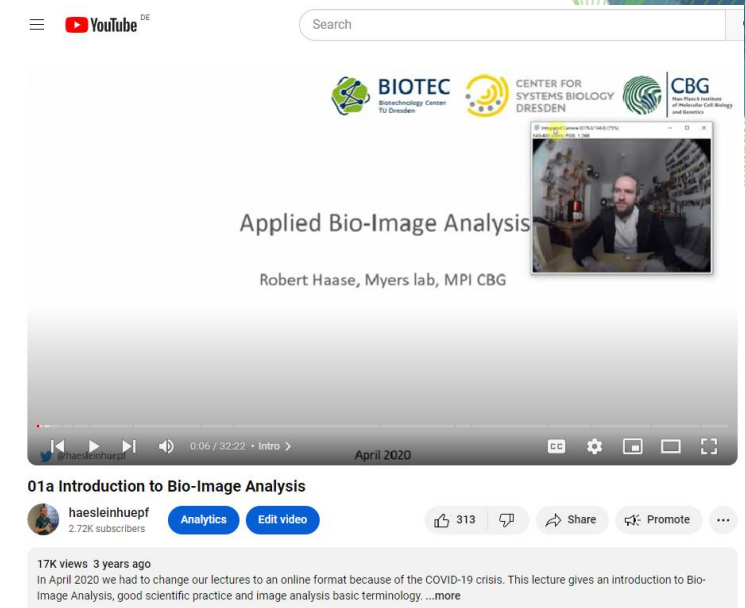
Bundesministerium
für Bildung
und Forschung



Diese Maßnahme wird gefördert durch die Bundesregierung aufgrund eines Beschlusses des Deutschen Bundestages. Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

Hello my name is...

- Robert Haase
- Applied image data scientist, research software engineer
- Trained computer scientist (2010 Dipl-Inf(FH)) with a medical PhD (2016 Dr. rer. medic.)
- Worked 15 years on the biomedical research campus in Dresden-Johannstadt (Hospital, MPI-CBG, TU-Dresden)
- Teaching bio-image analysis to life scientists 2019-2023 @ TU Dresden
- I maintain about 50 Python and Java packages other people use to analyze microscopy image data (GPU-acceleration, machine learning, large language models)
- New at Leipzig University

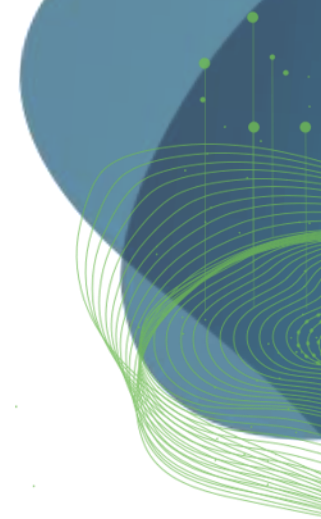


Large parts of this
online lecture are
outdated!



Survey

Think about the FAIR principles for data sharing, which one is wrong?



Findable

Accessible

Interoperable

Reproducible

Survey

Which open-source license might be the least popular in companies?

GPL

BSD

MIT

Apache

Survey

Which topic is typically not covered in a Research Data Management Plan?

Backup

Publishing

Acquisition

Career
development

Survey

Which git command does not exist?

fetch

pull

add

submit

Survey

You typically install Python packages using...

pip

conda

mamba

(other /
not)

Survey

Your favorite Python IDE is...

VS Code

Jupyter

Pycharm

(other /
none)

Survey

What does this Python code spit out?

```
test = "ScaDS.AI"  
print(test[-3:])
```

.AI

AI

ScaDS

(Error
message)

Survey

Which is a background-removal filter?

Laplace

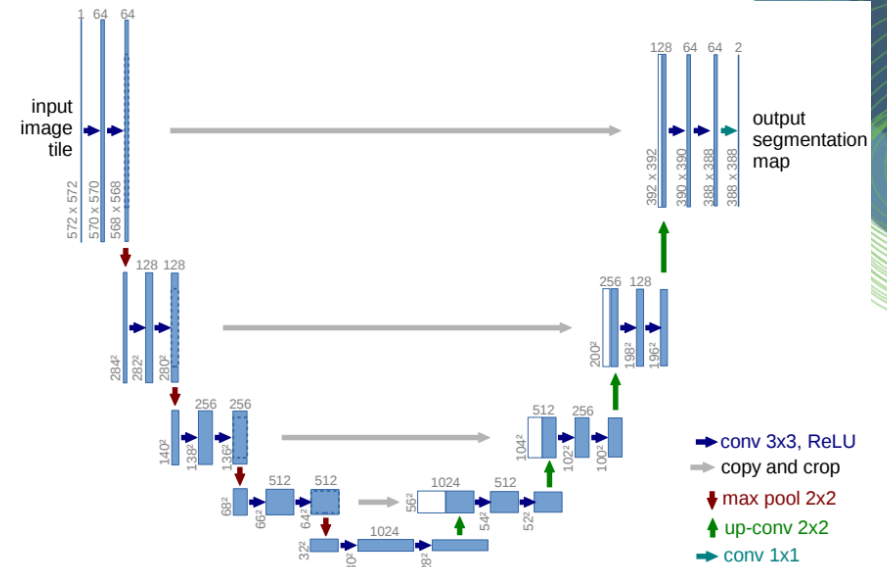
Gaussian

Top-hat

Sobel

Exercise: Survey

How is this neural network architecture called?



Auto-
encoder

U-Net

Alex-Net

(Don't
know)

Survey

Which of the following is no language model?

chatGPT

claude

mistral

llama

Lecture overview

- Every week will follow the same rough scheme
 - 15:15 : 90 min lecture
 - 17:15 : 90 min exercises (programming intense)
 - when you're done, enjoy the sun!
- Exam will cover the semester content accordingly
 - Bio-image Analysis / Microscopy
 - Machine/Deep Learning
 - Generative Artificial Intelligence
 - “closed book exam”

Lecture materials

ScaDS / BIDS-lecture-2024

Training resources for Students at Uni Leipzig who want to dive into bio-image Python. The material will develop between April and July 2024

CC-BY-4.0 license

1 star 0 forks 2 watching 1 Branch 0 Tags Activity

Public repository

main

haesleinhuepf added first exercise

- 01a_setting_up_local_environment added first exercise
- 01b_setting_up_sc_ulei_environment added first exercise
- 01c_testing_environment added first exercise

.gitignore

LICENSE-CC-BY

README.md

Slides commonly available in advance

Bio-image Data Science

This repository contains training resources for Students at Uni Leipzig who want to dive into bio-image data science with Python. The material will develop between April and July 2024 and shared here in this github repository.

Teaching Goal

Students learn the full workflow of common bio-image data science projects to a degree that they can execute a scientific data analysis project in this context on their own. They will be familiar with common bio-image analysis algorithms and workflows, how to choose them according to a scientific goal, and how to measure quality of derived results. Attending the lecture and executing the practicals qualifies the students to work as bio-image data scientist in the pharmaceutical industry or basic biological research.

Course contents

- Introduction to Bio-image Data Science (Apr 2nd 2024)
 - Basics of microscopy
 - Introduction to Bio-image Analysis
 - Exercises:
 - Setting up a local environment
 - Setting up Jupyter Hub at Scientific Computing / Leipzig University
 - Execute the trailer notebook



Preliminary schedule

- April 2nd 2024 – Introduction Microscopy & Bioimage Analysis
- April 9th 2024 – Research Data Management
- April 15th 2024 - Microscopy Image Processing
- April 23rd 2024 – Segmentation of cells and nuclei
- April 30th 2024 - Quality Assurance for image segmentation
- May 7th 2024 – Feature extraction, data visualization
- May 14th 2024 – Big data, parallel processing & distributed computing
- May 21st 2024 – Introduction to Machine Learning for bio-image analysis
- May 28th 2024 – Unsupervised Machine Learning
- June 4th 2024 – Deep Learning for image denoising + segmentation
- June 11th 2024 – Generative Artificial Intelligence (LLMs)
- June 18th 2024 – Image generation + vision models
- June 25th 2024 – Quality assurance
- July 2nd 2024 – Summary, exam preparation

Handout exam pre-requisite
complex exercise

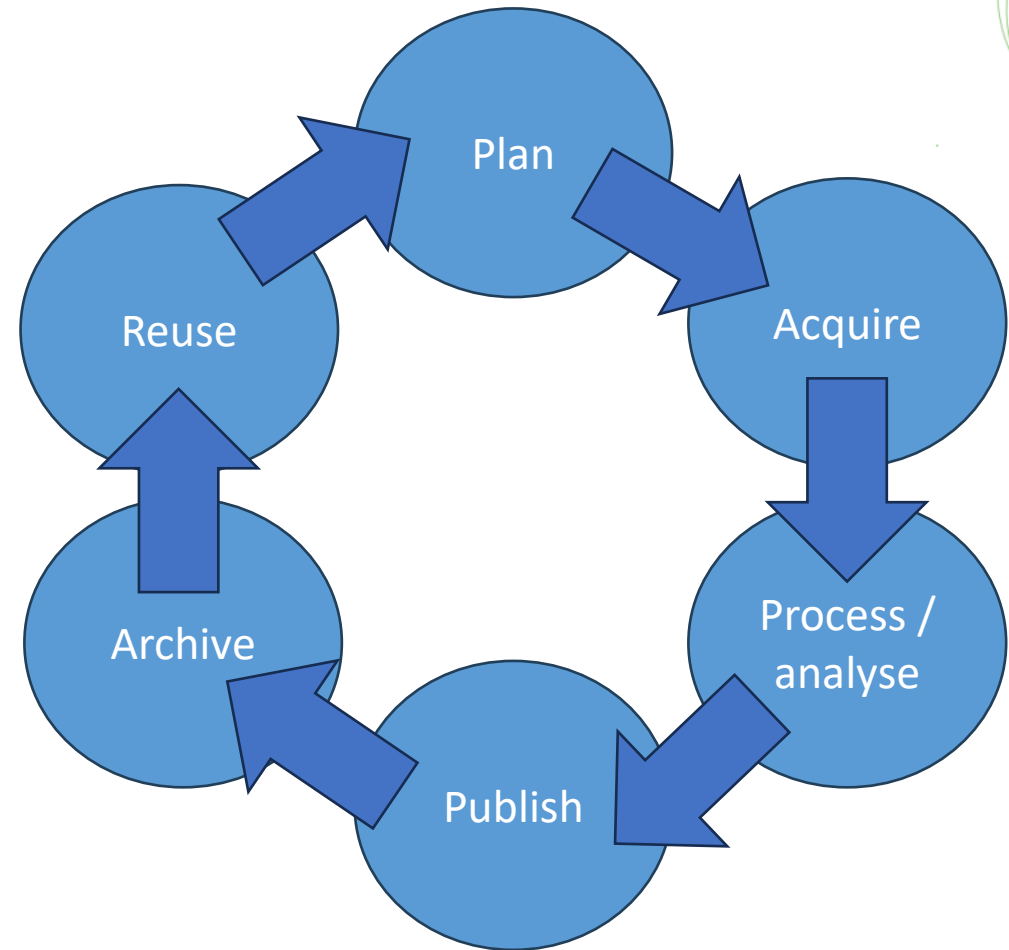
Flexible; I would like
to focus as much as
possible on LLMs

Submission deadline
complex exercise

8 weeks

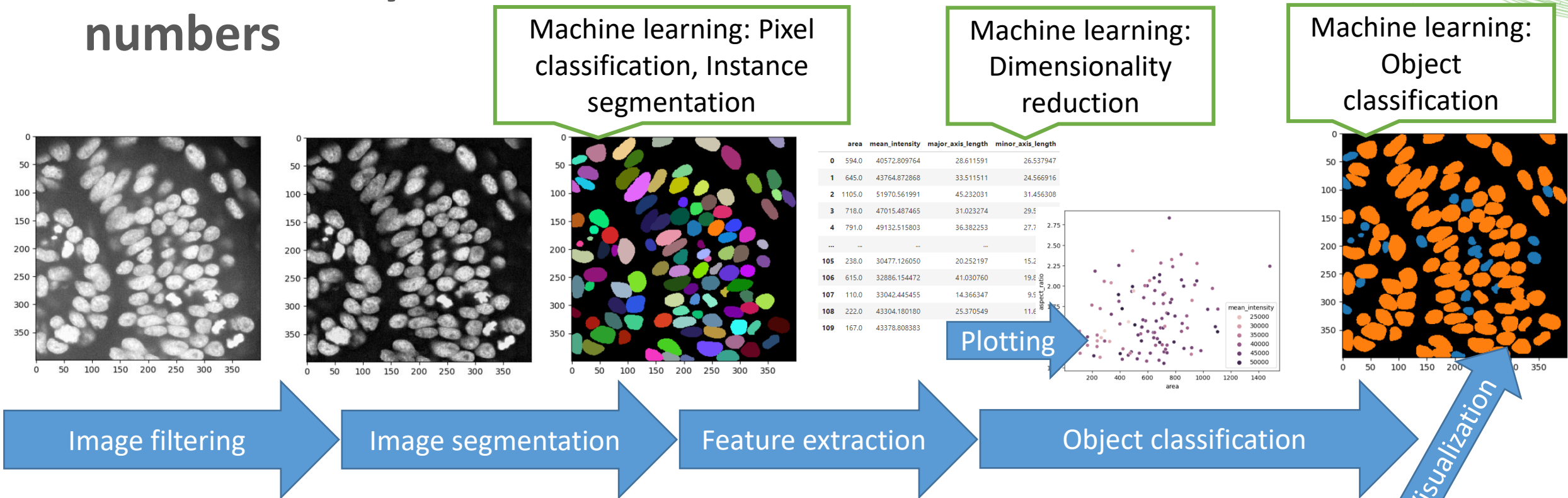
Lecture overview: Research Data Management

- FAIR Principles
- Sharing / licensing
- Open Source code
- Data Management Plans
- Big-Data
- Distributed computing



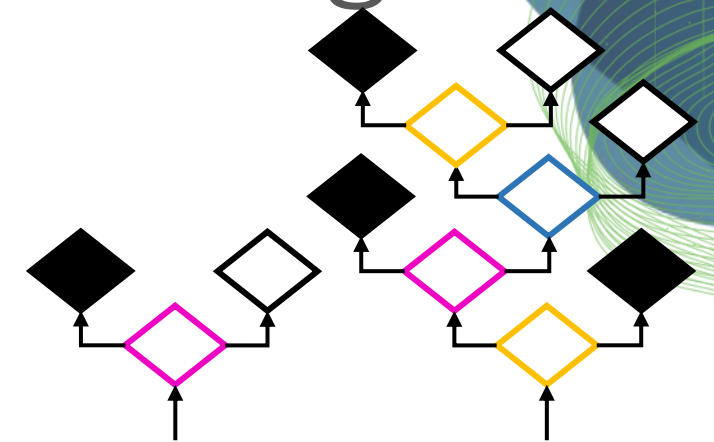
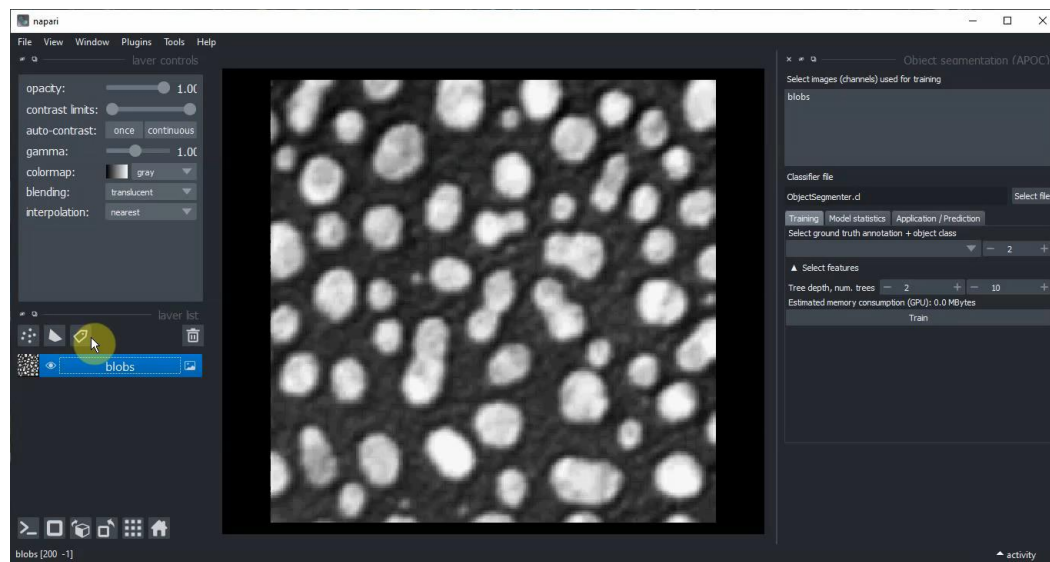
Lecture overview: Bio-image Analysis

- Image Data Analysis workflows
- Goal: Quantify observations, substantiate conclusions with numbers

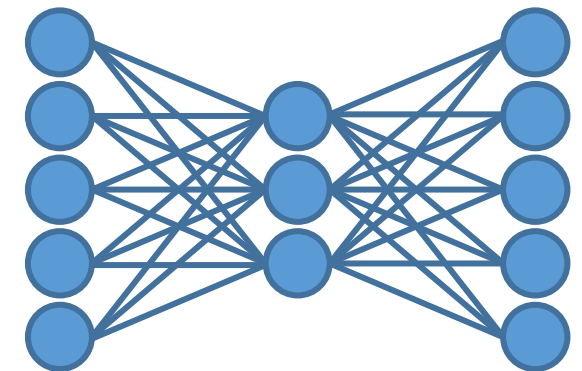


Lecture overview: Machine learning

- Machine learning
 - in the context of bioimage analysis
- Computers can *learn* from data, potentially revealing relationships that are not obvious to a human
- Goal: **Give you an insight into state-of-the-art methods**



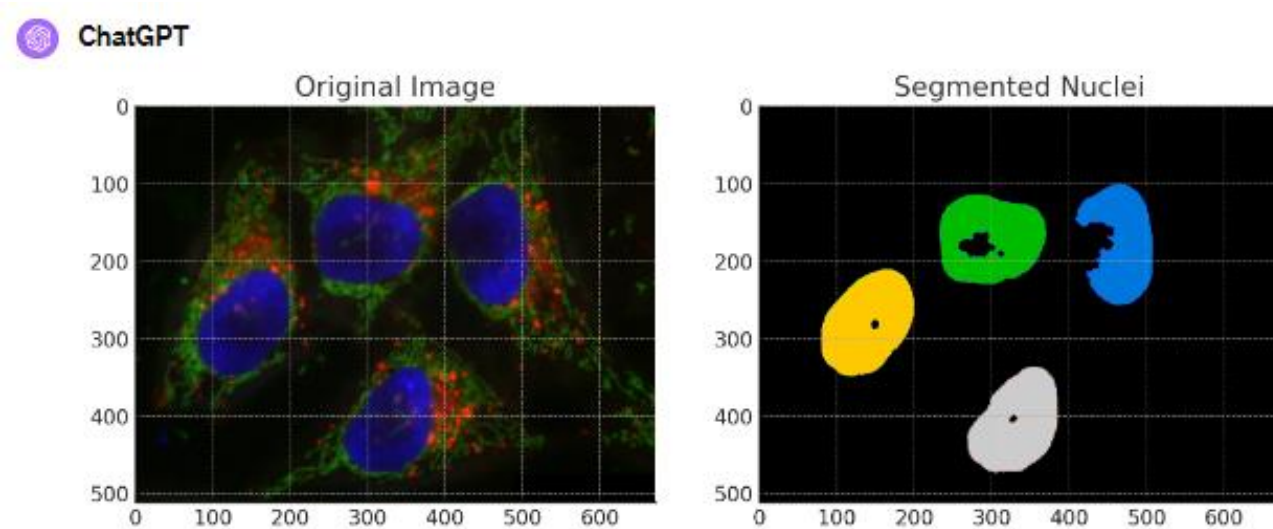
Random forest classifiers



Neural networks

Lecture Overview: Generative AI Multi-modal / Large Language Models

Prompt:
“Please segment
the nuclei in the
blue channel of
the image”



The segmentation of the nuclei has been performed on the blue channel of the image. The right side of the figure shows the segmented nuclei in different colors, each corresponding to a different nucleus in the field of view. Each color represents a unique label assigned to the individual nuclei detected in the image. ↗

```
Analysis
from skimage import io, filters, measure, morphology
import matplotlib.pyplot as plt
import numpy as np

# Load the image
image = io.imread('/mnt/data/hela-cells-8bit.png')

# Since it's a color image, we'll just take the blue channel
blue_channel = image[:, :, 2]

# Threshold the image to get the binary image for nuclei
thresh = filters.threshold_otsu(blue_channel)
binary = blue_channel > thresh

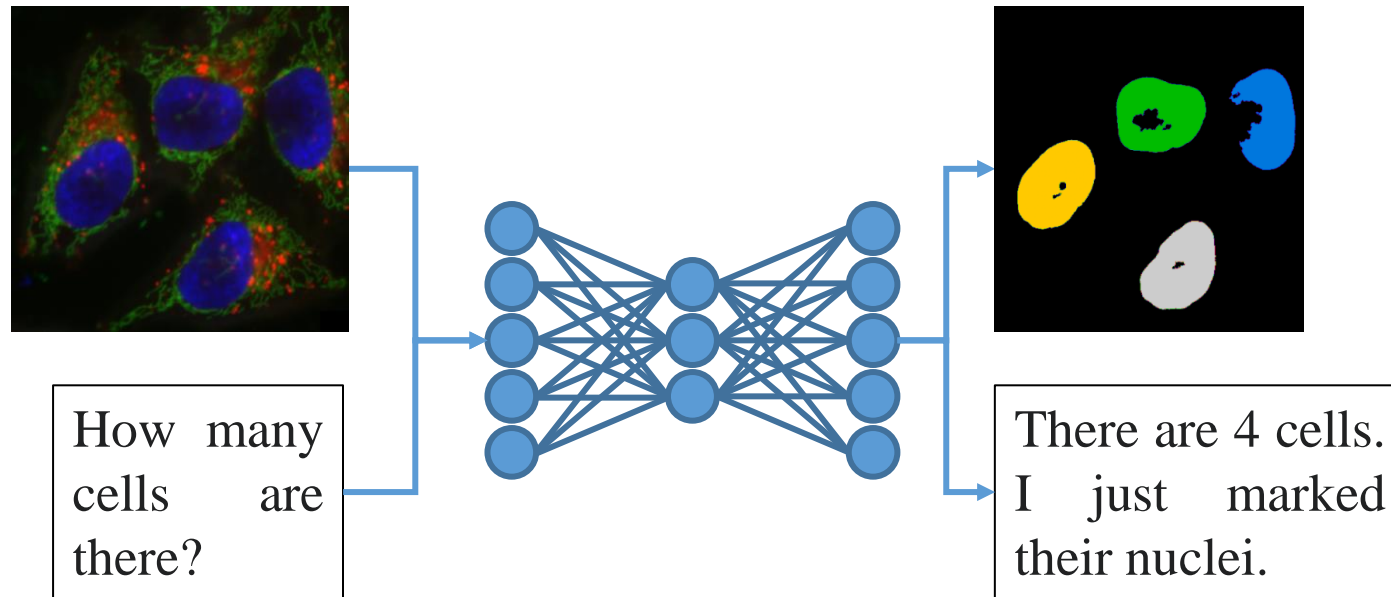
# Remove small objects and perform morphological closing
cleaned = morphology.remove_small_objects(binary, 50)
closed = morphology.closing(cleaned, morphology.disk(3))

# Label the image
label_image = measure.label(closed)

# Plot the original image and the binary image
fig, ax = plt.subplots(1, 2, figsize=(12, 6))
ax[0].imshow(image)
ax[0].set_title('Original Image')
ax[1].imshow(label_image, cmap='nipy_spectral')
ax[1].set_title('Segmented Nuclei')
plt.show()
```

Lecture Overview: Generative AI Multi-modal / Large Language Models

After we learned how analyze images, we will teach an LLM to do it.





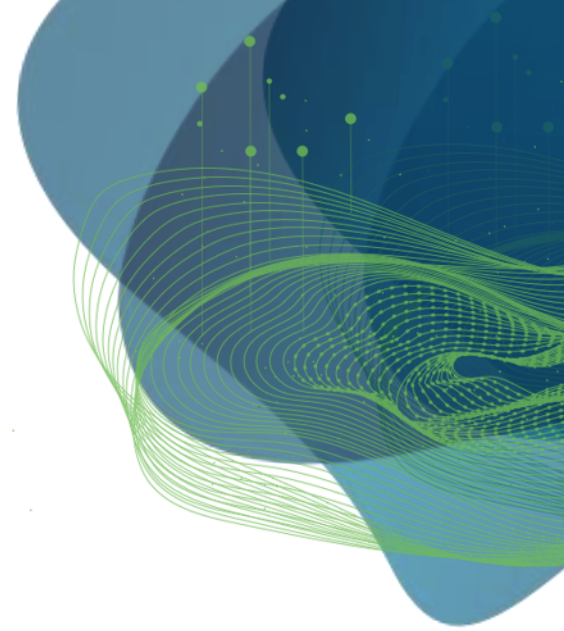
ScaDS.AI

DRESDEN LEIPZIG

CENTER FOR SCALABLE DATA ANALYTICS
AND ARTIFICIAL INTELLIGENCE

Basics of Microscopy

Robert Haase



GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



Diese Maßnahme wird gefördert durch die Bundesregierung aufgrund eines Beschlusses des Deutschen Bundestages. Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

Microscopy

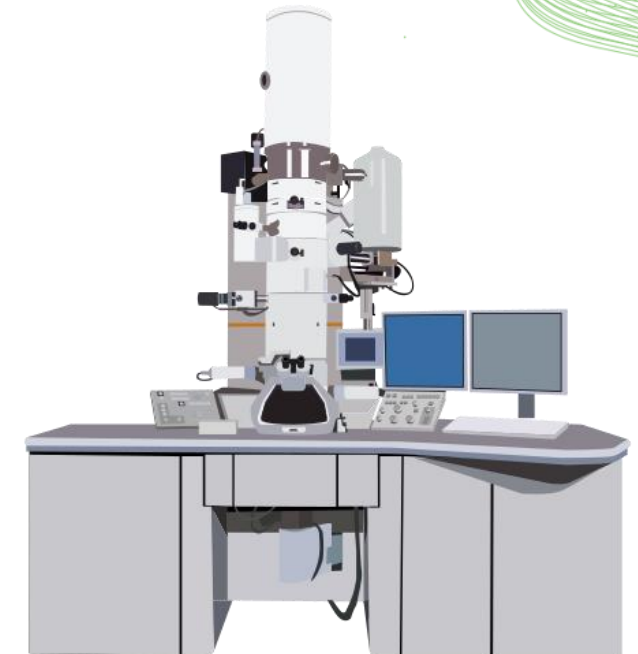
- Common tool to answer biological questions



Transmitted light microscope



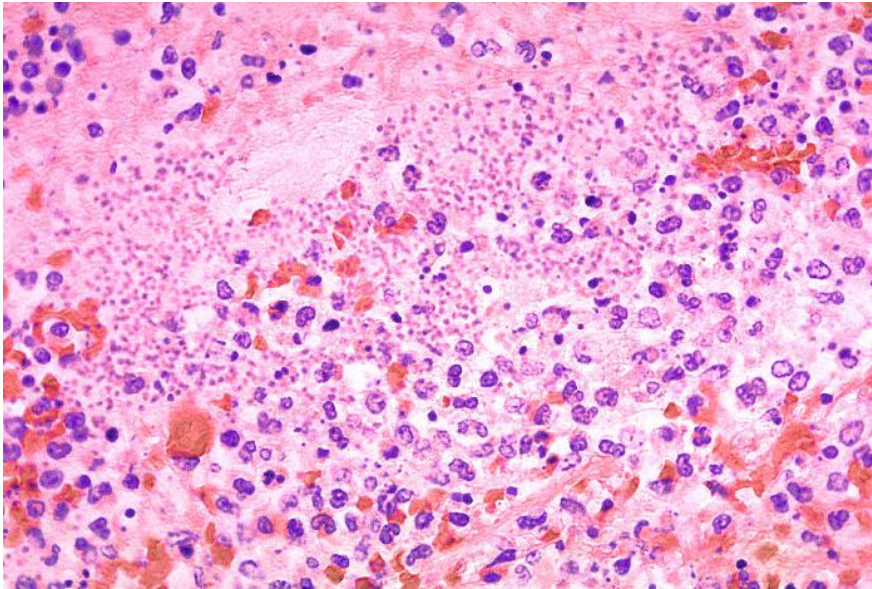
Fluorescence microscope



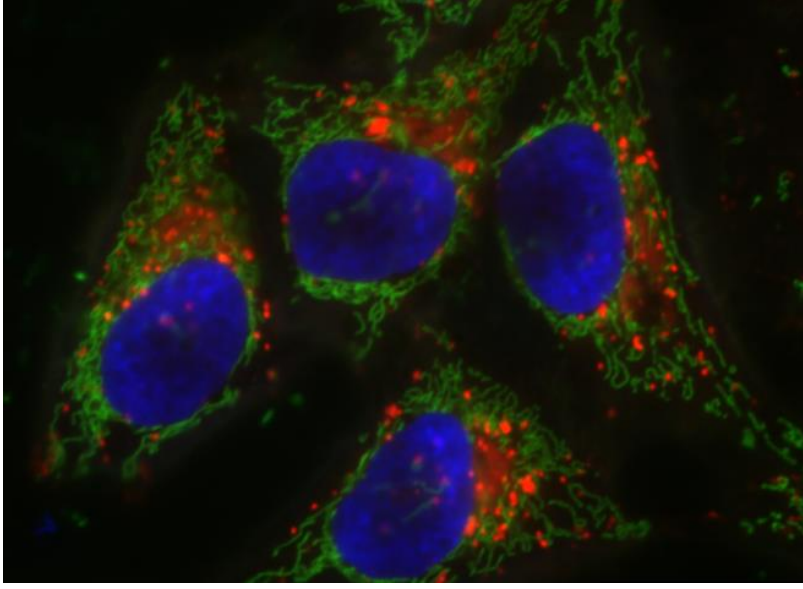
Electron microscope

Microscopy

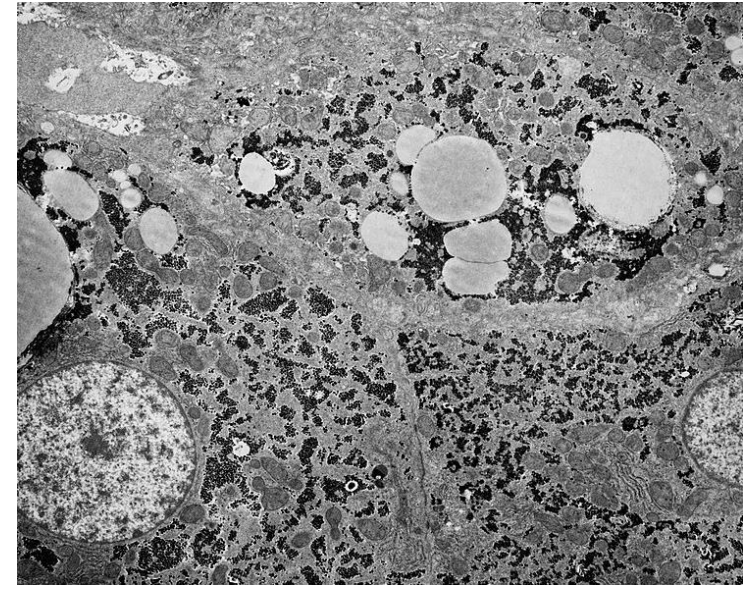
- Common tool to answer biological questions



Transmitted light microscope

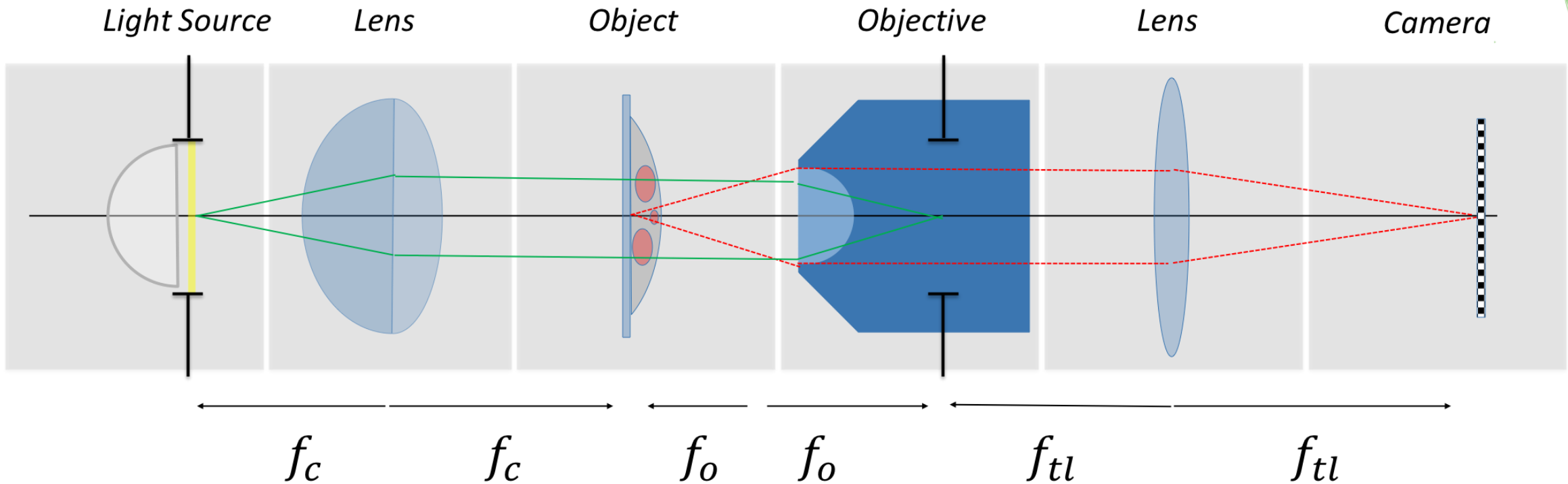


Fluorescence microscope

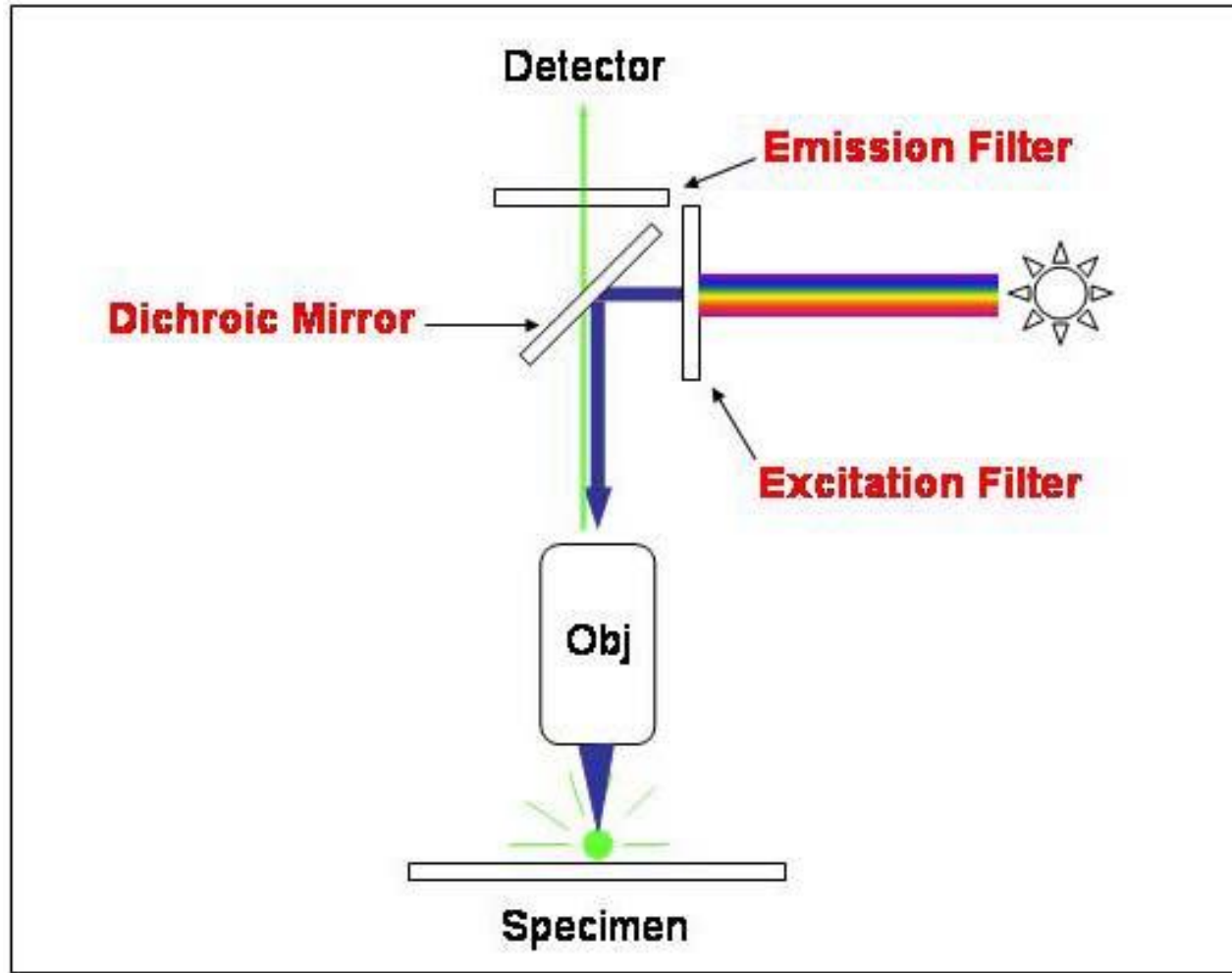


Electron microscope

Transmitted light microscopy



Fluorescence Microscopy



Fluorescence Microscope

- NOT Fluorescent microscope



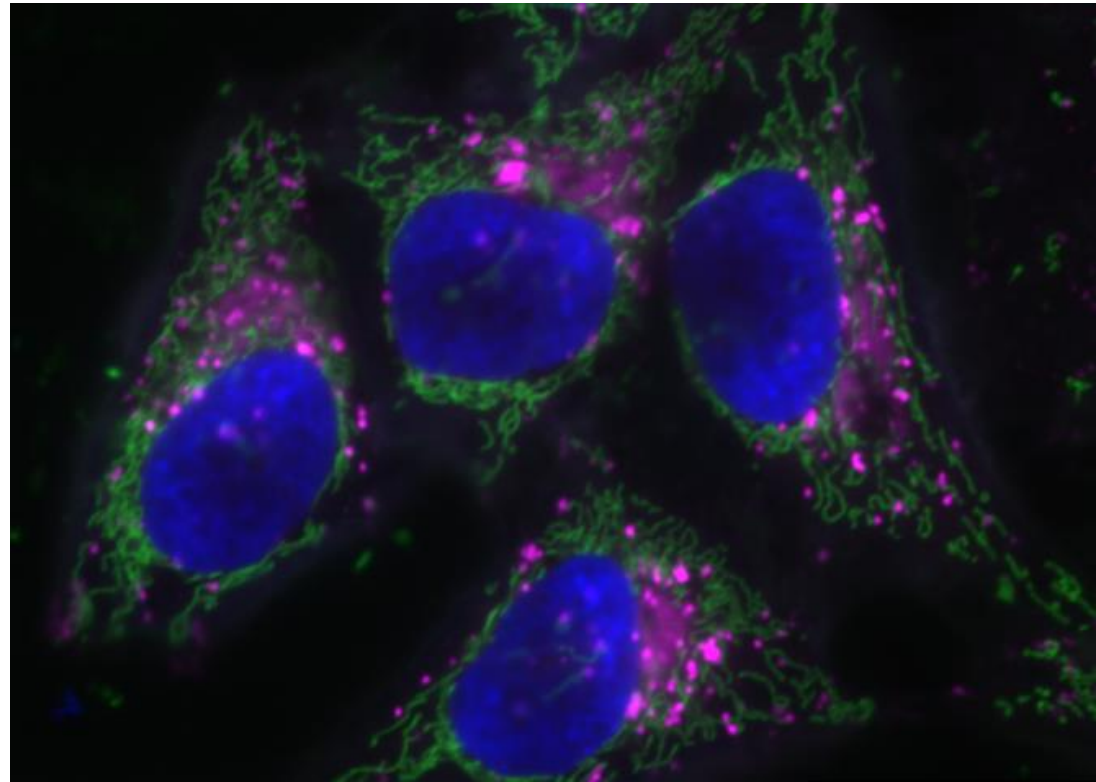
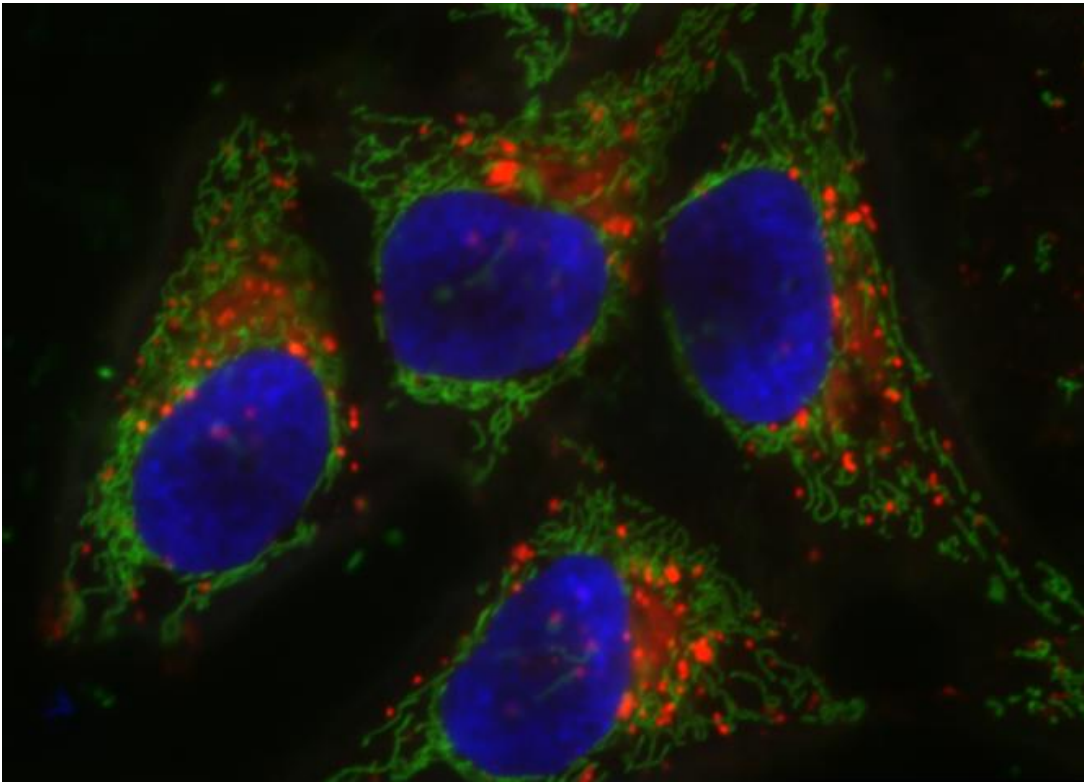
Fluorescence microscope



Fluorescent microscope


Color maps / lookup tables

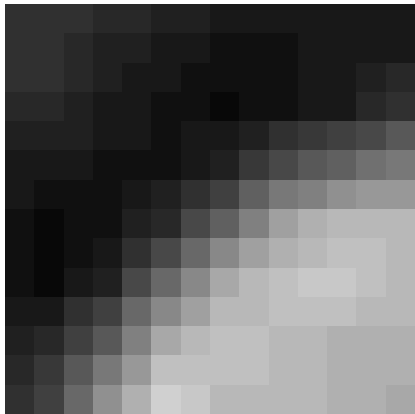
- Just because you see something in red, doesn't mean it was imaged emitting red light.

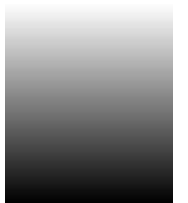


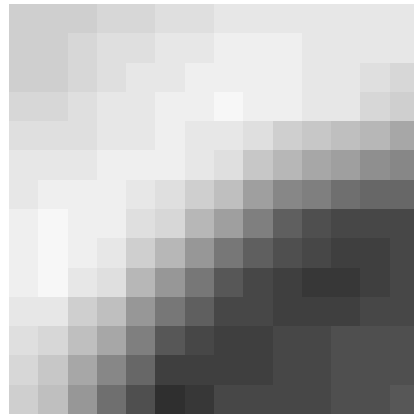
Color maps / lookup tables

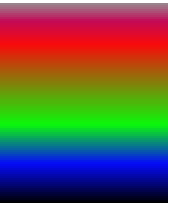
- The lookup table decides how the image is displayed on screen.
- Applying a different lookup table doesn't change the image. All pixel values stay the same, they just appear differently

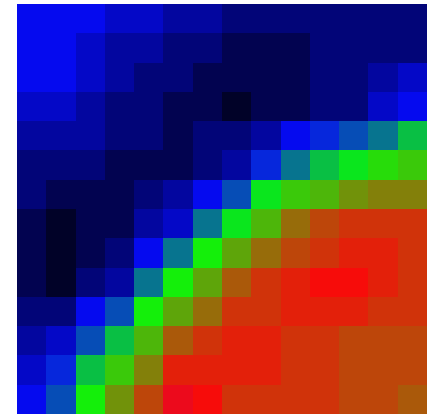
Pixel value	Display color
0	
1	
2	
...	
255	



Pixel value	Display color
0	
1	
2	
...	
255	



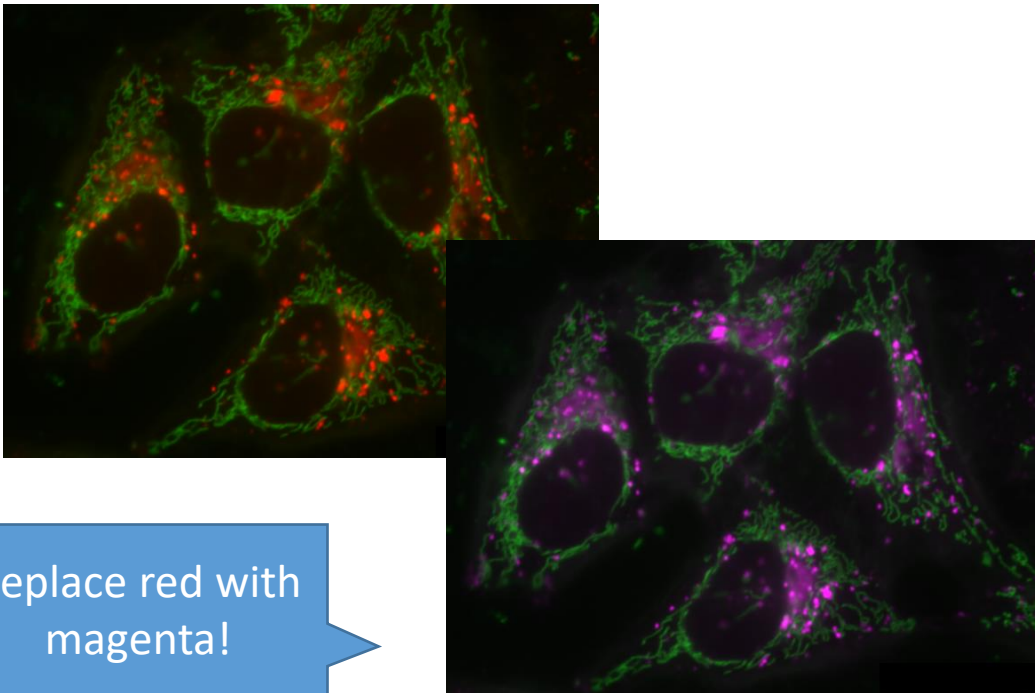
Pixel value	Display color
0	
1	
2	
...	
255	



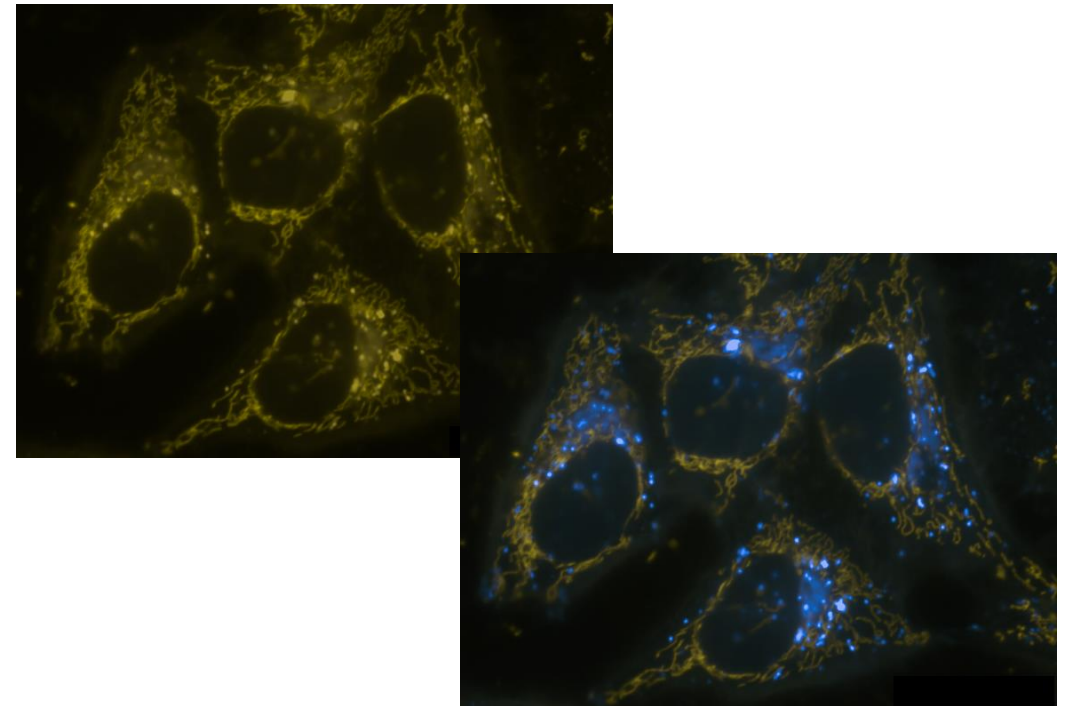
Color maps / lookup tables

- Choose visualization of your color tables wisely!
- Think of people with red/green blindness!

Default view

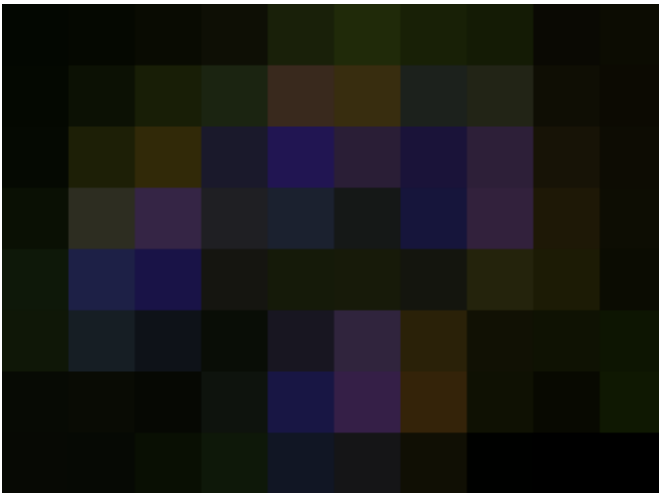


Red/green blind people see it like this

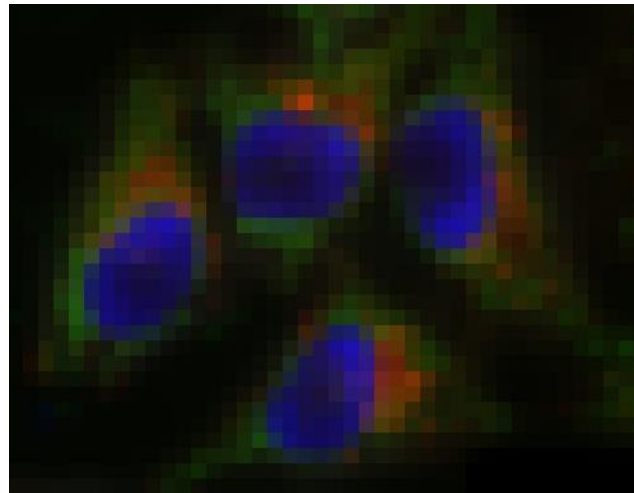


Pixel size versus resolution

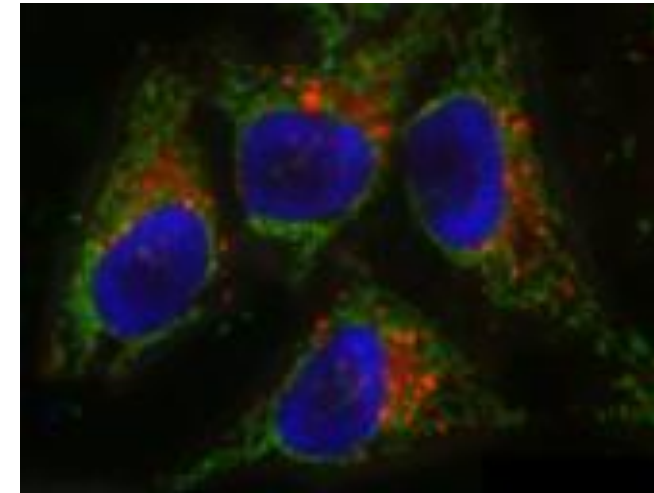
- Pixel size is a property of a digital image.
- You configure it during the imaging session at the microscope.



Pixel size: 3.3 μm



Pixel size: 0.8 μm

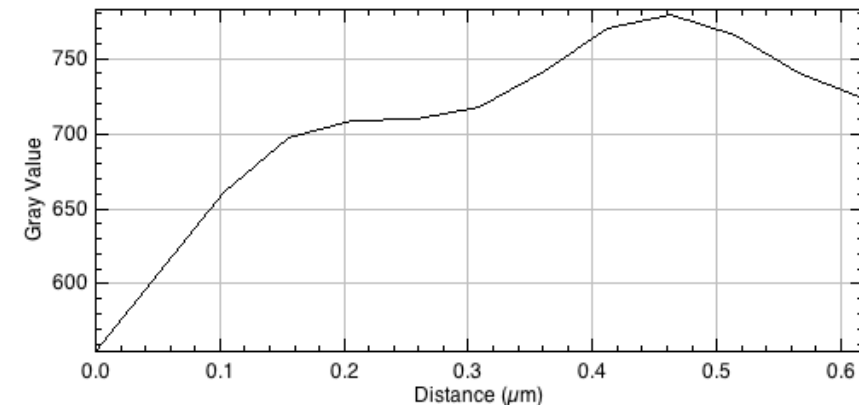
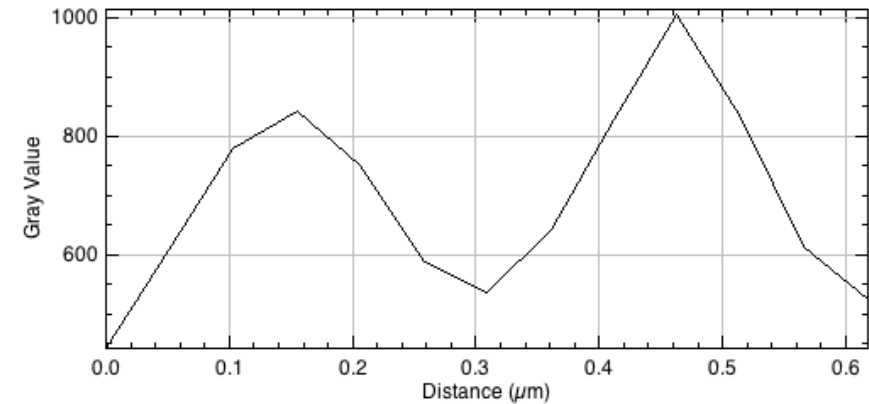
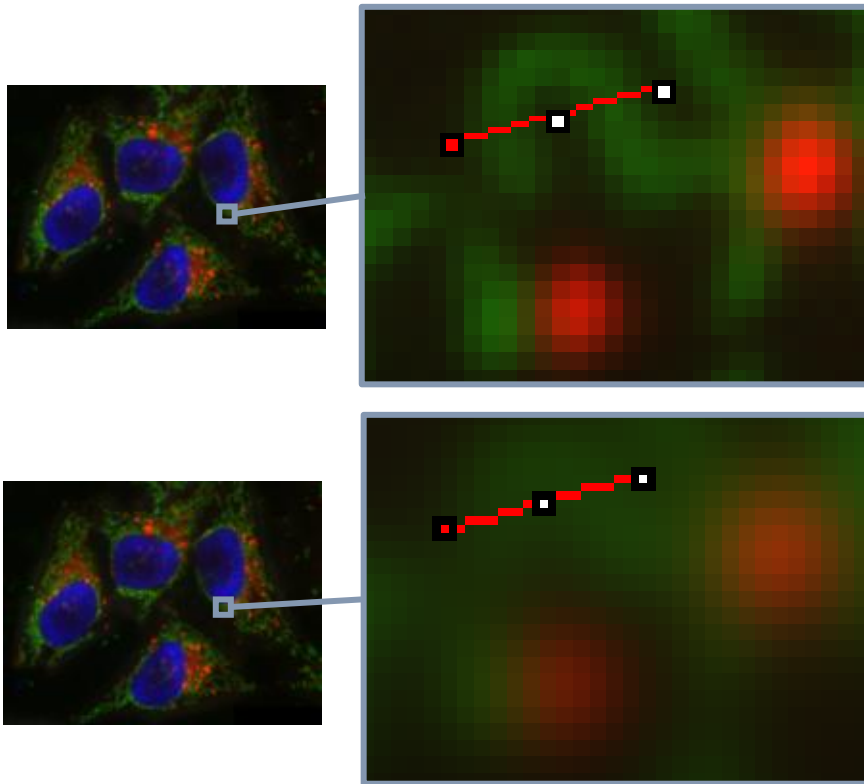


Pixel size: 0.05 μm

- We are not talking about resolution!

Pixel size versus resolution

- Resolution is a property of your imaging system.
- How small can objects be, to be still differentiable?



Quiz

- How can I change the pixel size at a microscope practically?

Move
sample

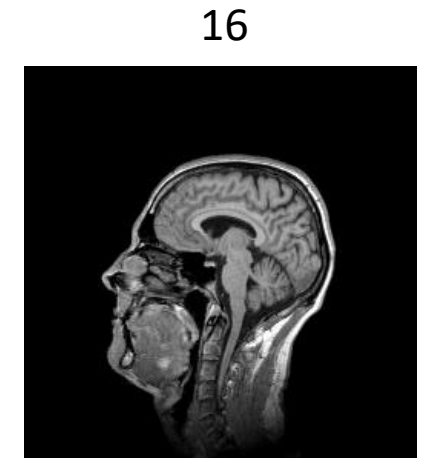
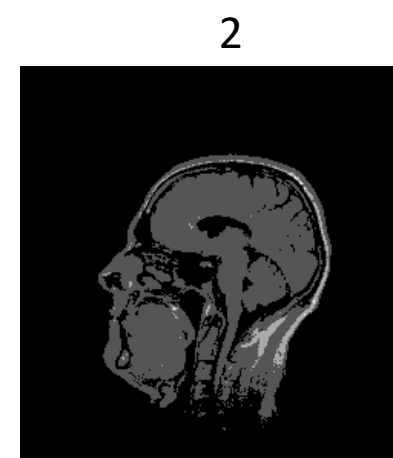
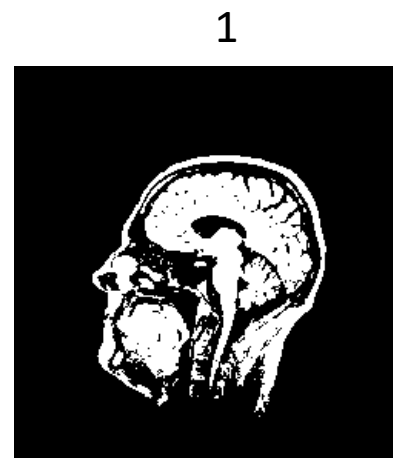
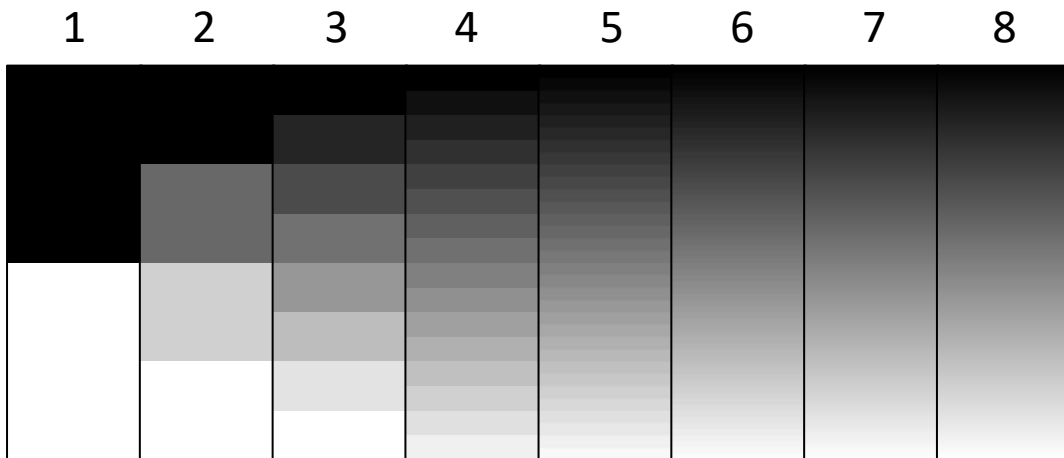
Enter in
software

Change
camera

Change
objective

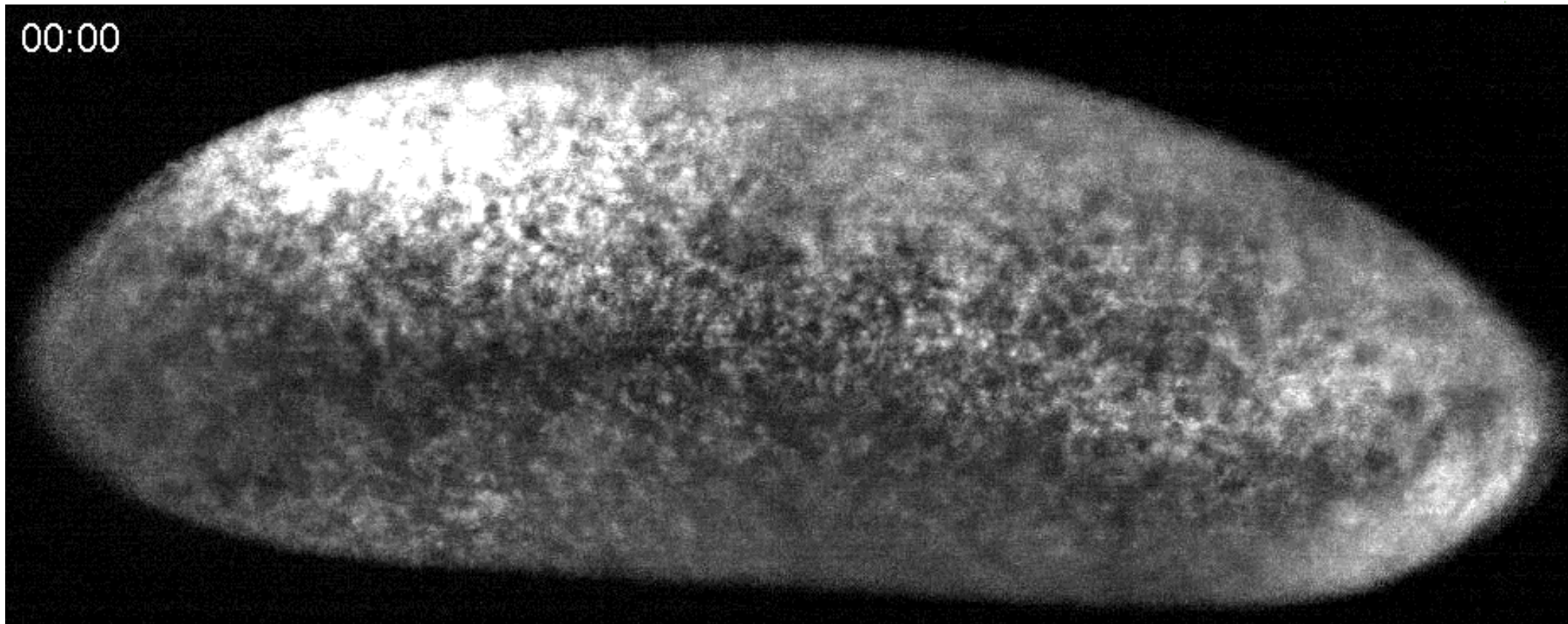
Bit-depth

- A bits is the smallest memory unit in computers, *atomic data*.
- The bit-depth n enumerates how many different intensity values are present in an image:
 - 2^n grey values
- In microscopy, images are usually stored as 8, 12 or 16-bit images.
- In computer vision, 8-bit integer and 32-bit float (range 0...1) is more common



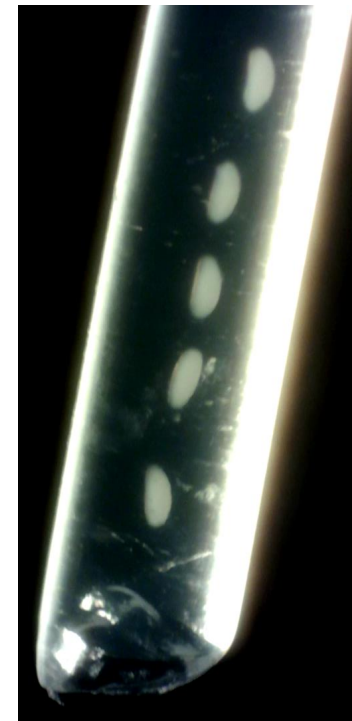
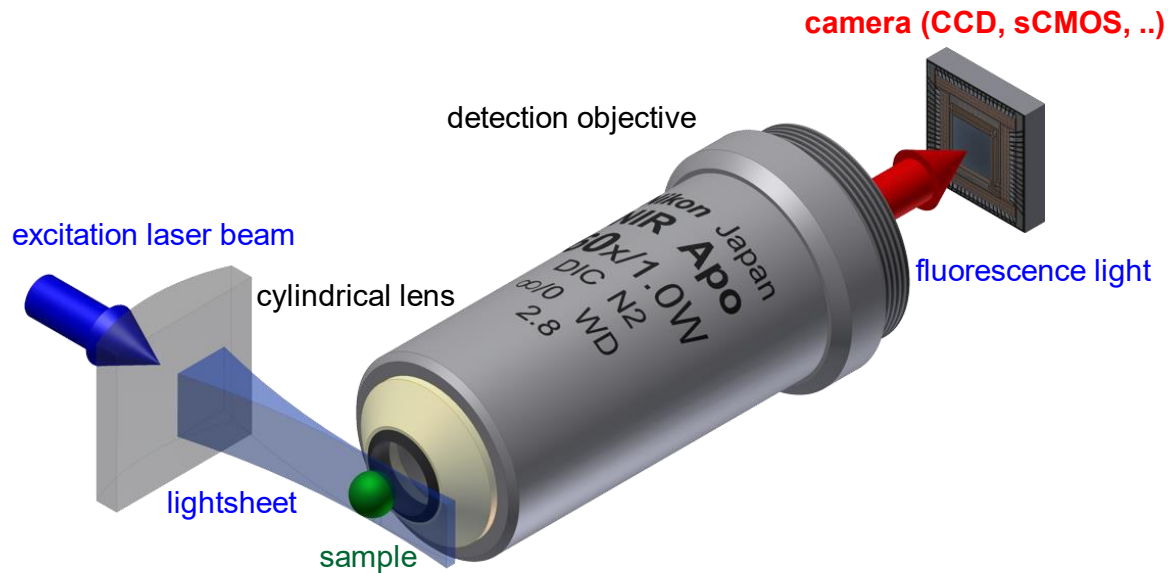
Light-sheet fluorescence microscopy

- Popular tool for large-volume imaging

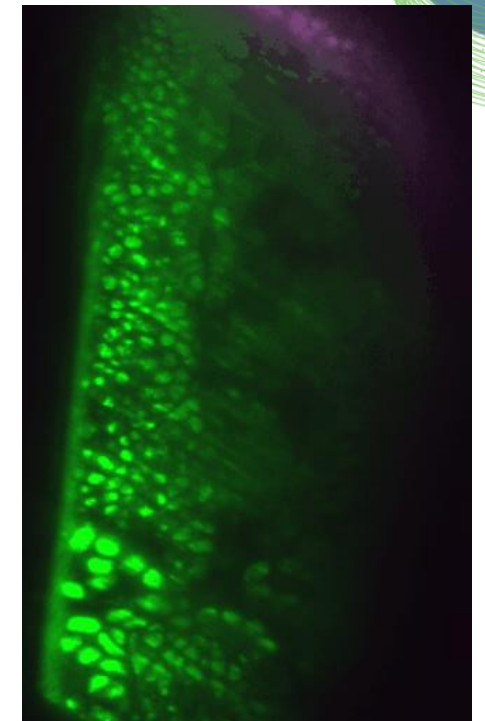


Light-sheet fluorescence microscopy

- Single-plane illumination microscopy (SPIM)



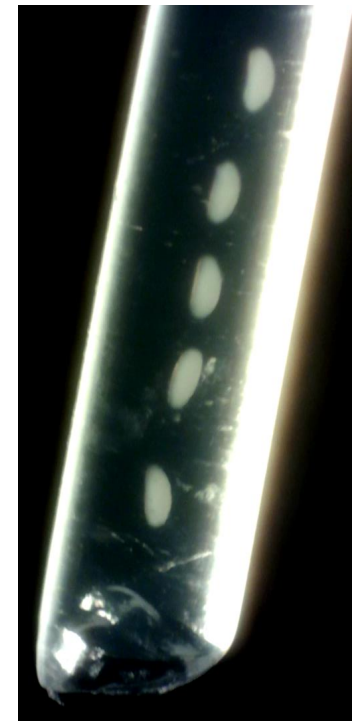
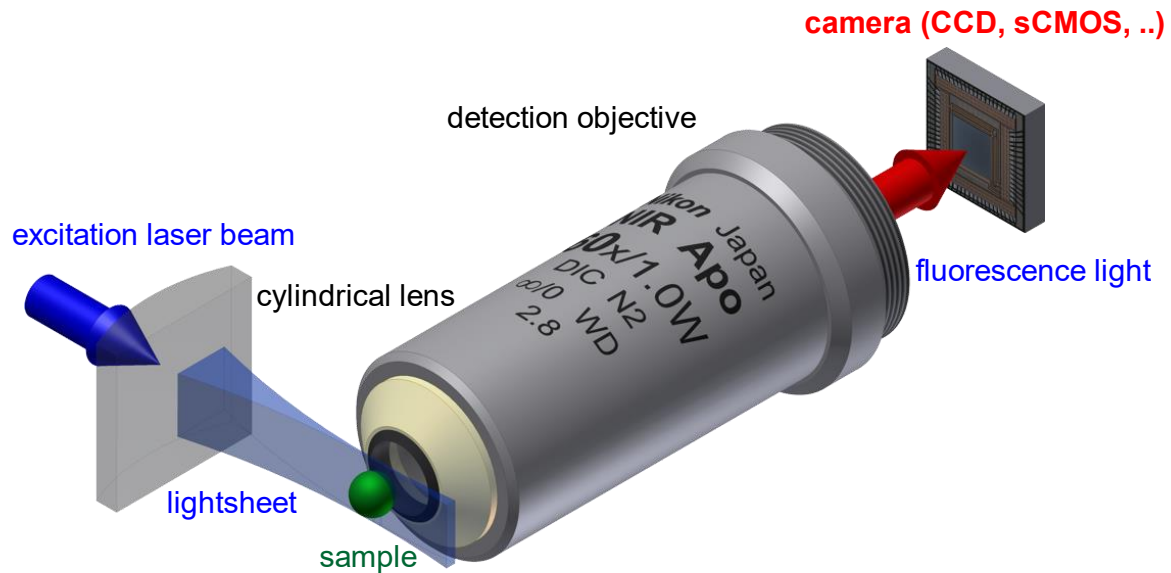
Sample mounting



Laser beam scanning

Light-sheet fluorescence microscopy

- Single-plane illumination microscopy (SPIM)



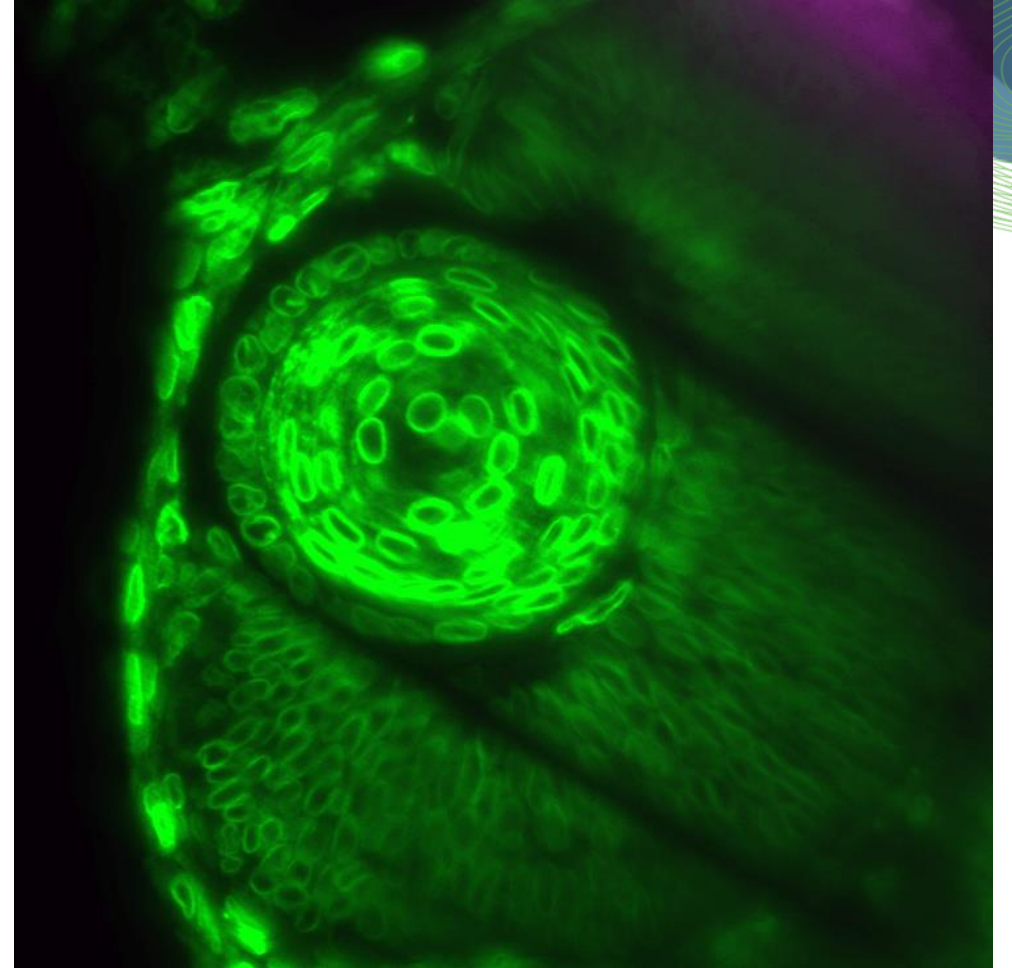
Sample mounting



Plane scanning

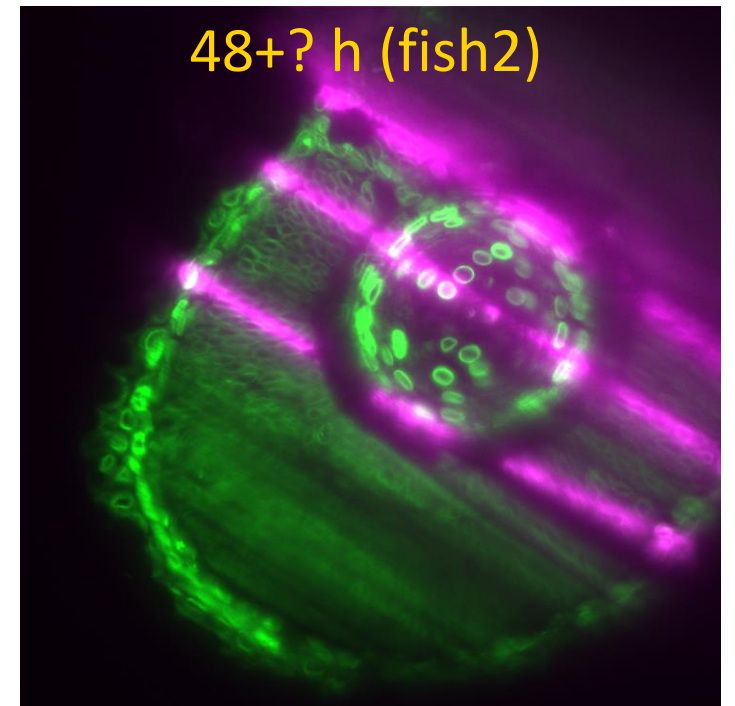
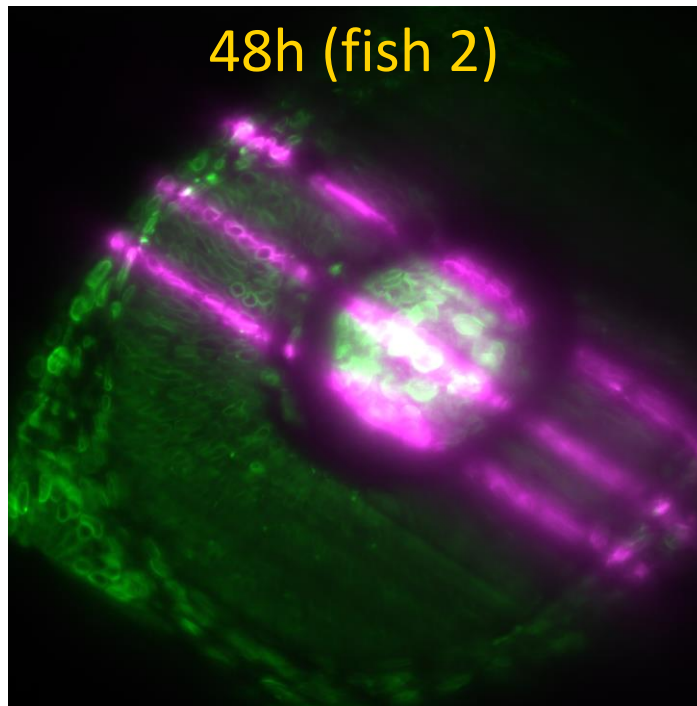
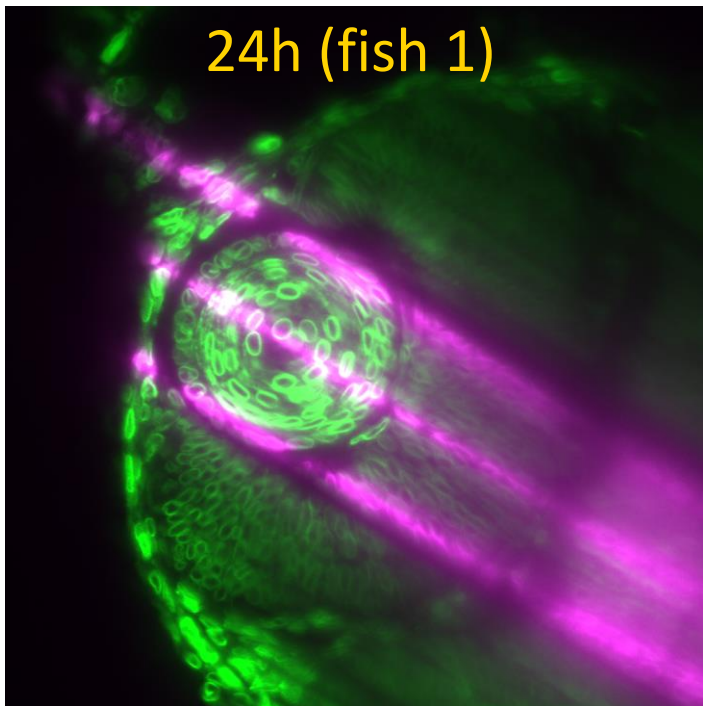
Light-sheet fluorescence microscopy

- The beam interacts with the tissue while passing matter
 - Refraction
 - [Reflection]
 - Scattering
- Challenge for quantitative analysis: image quality varies within the image



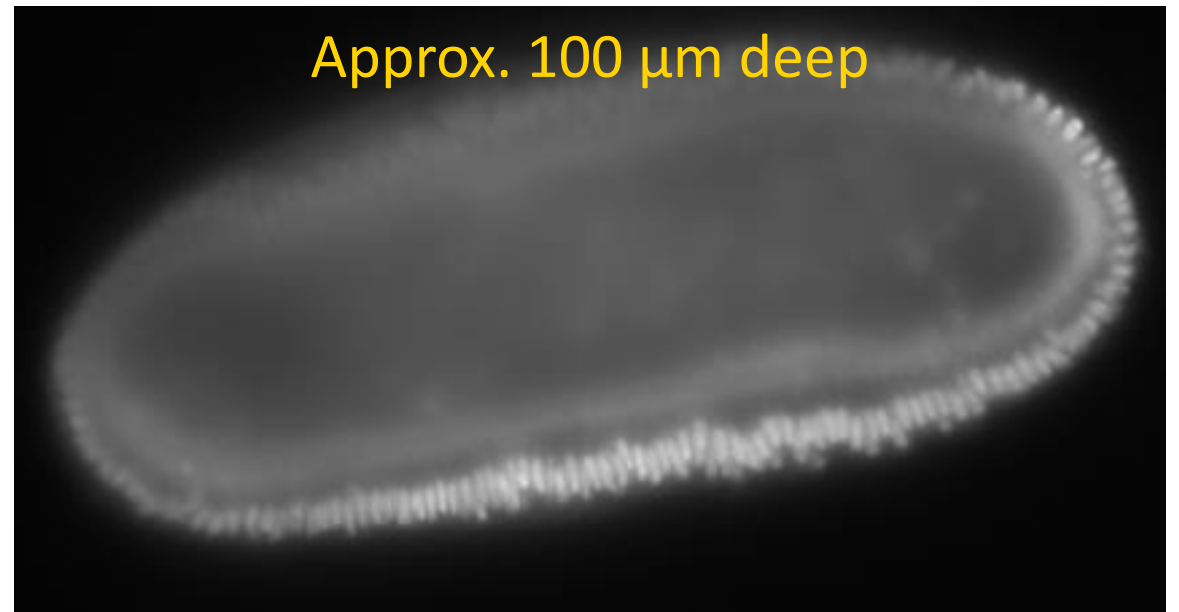
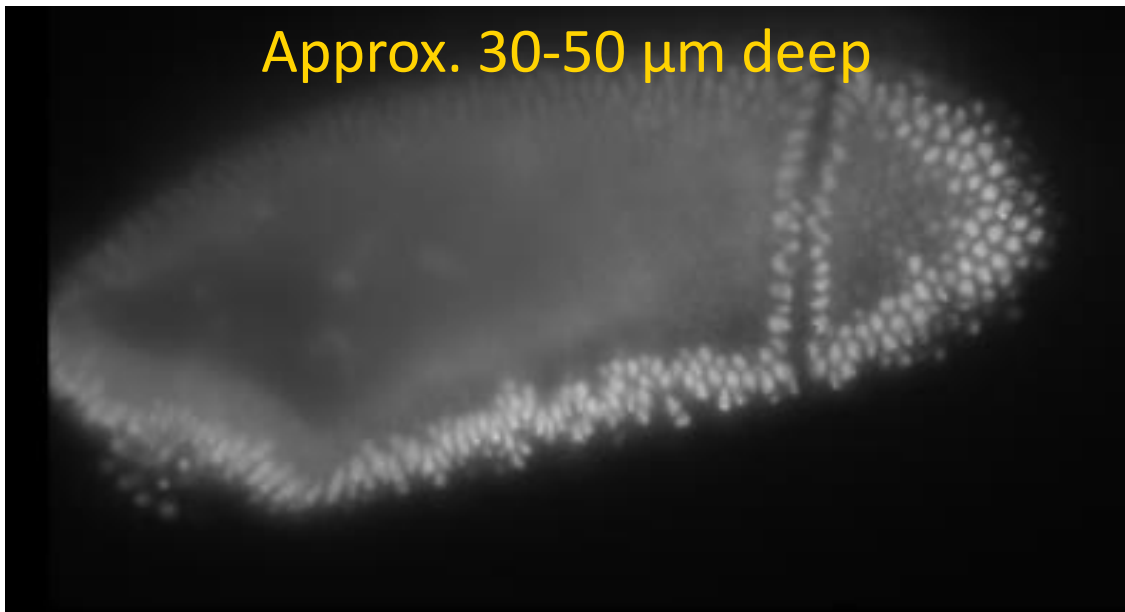
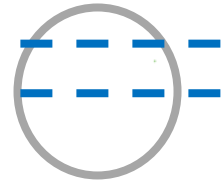
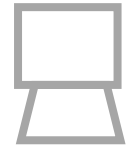
Light-sheet fluorescence microscopy

- These effects depend on the sample
- Big challenge in practice: *Image deep inside*



Light-sheet fluorescence microscopy

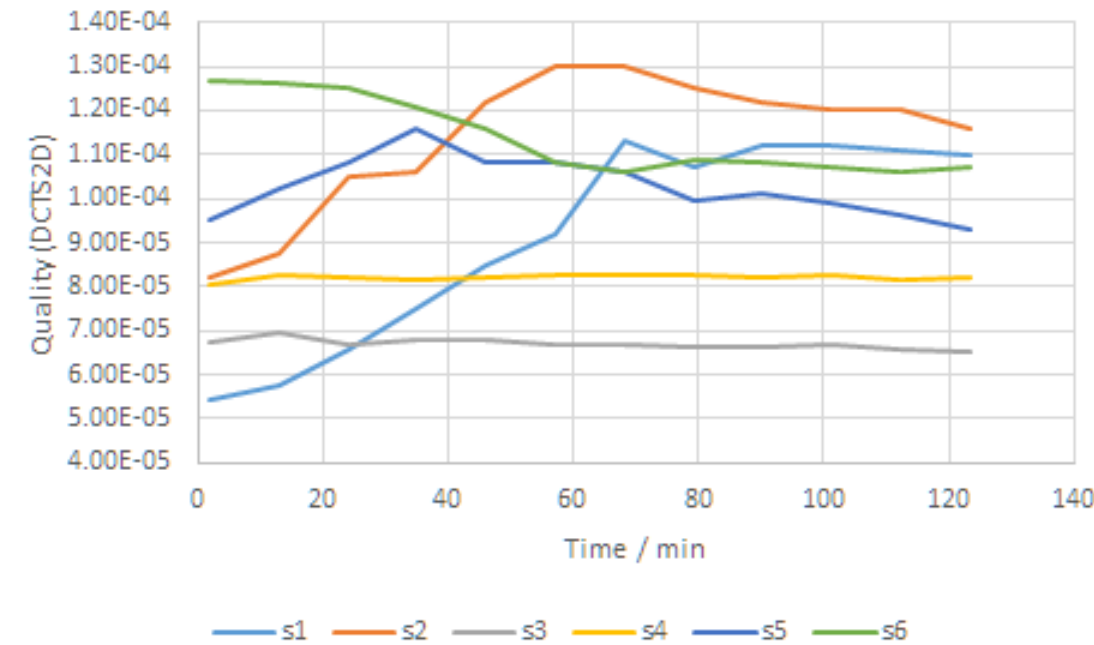
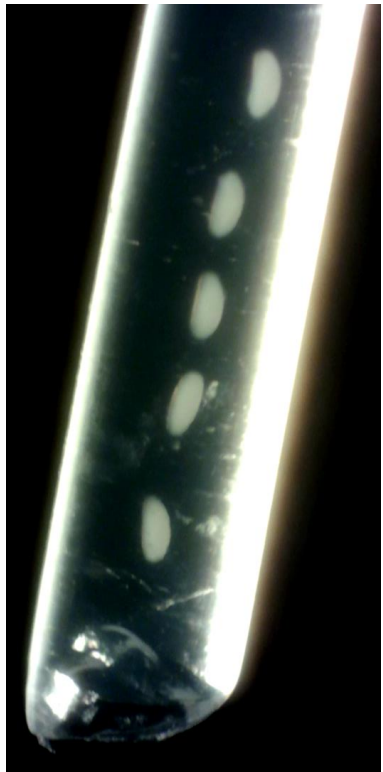
- These effects depend on the sample
- Big challenge in practice: *Image deep inside*



High-throughput imaging

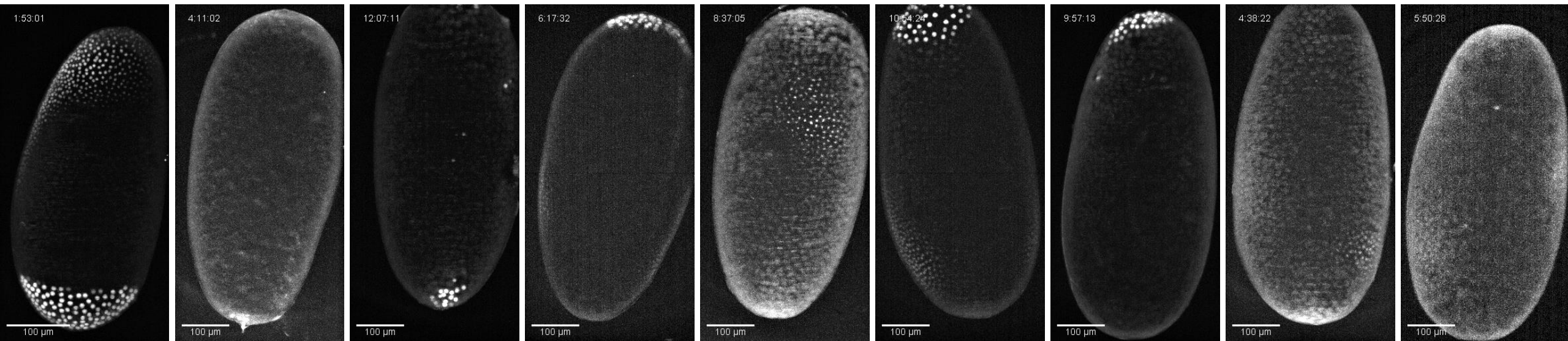
- Challenge: Online decision making while acquisition

-> AI



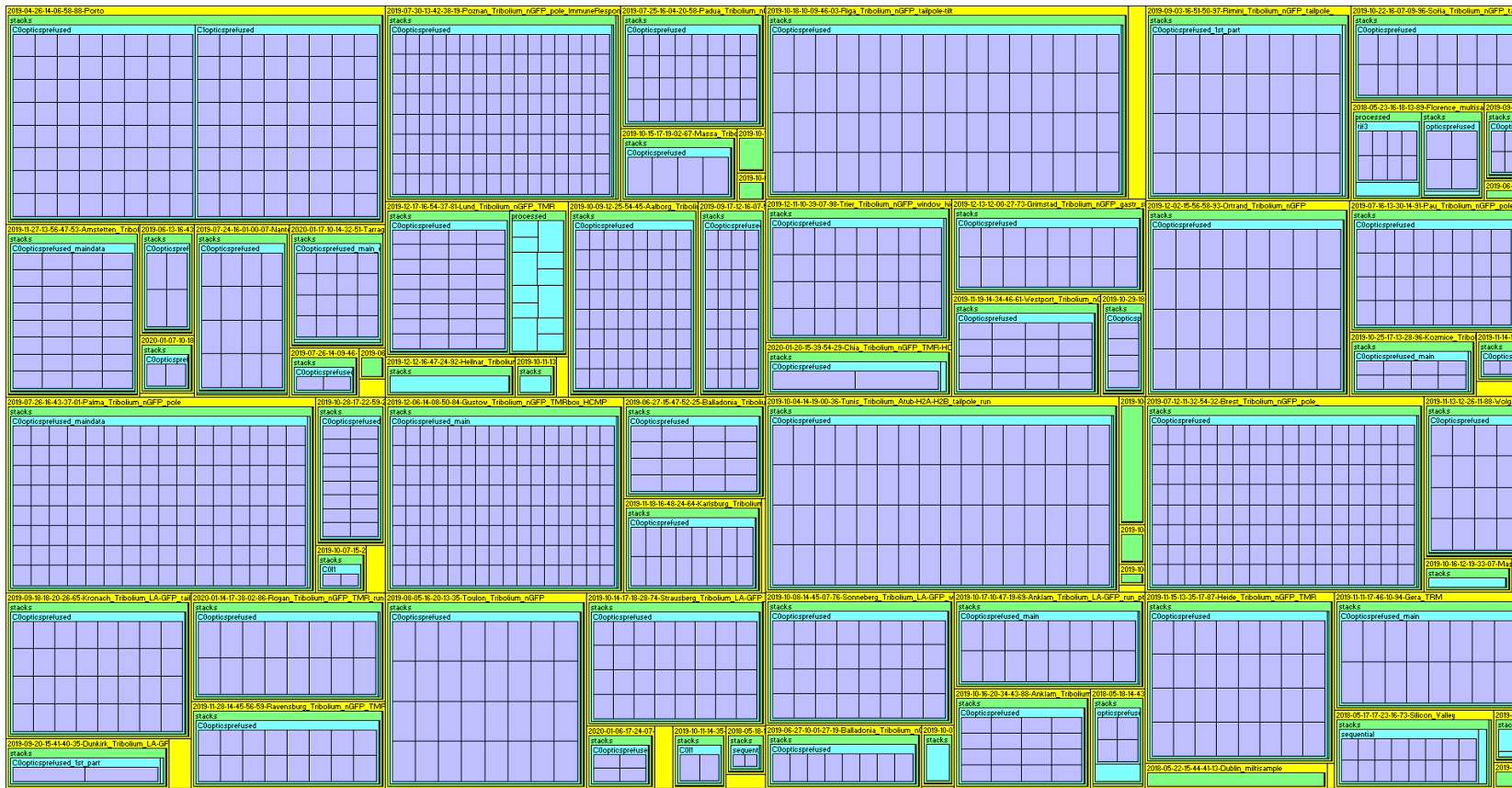
High-throughput imaging

- Quality variation between samples, in space and time [and algorithms which process the data]



High-throughput imaging

- “Big data”



Postdoc 2 years imaging

- 35 TB imaging data



Assuming

- one frame is about 200 MB
- counting nuclei takes 30 sec per frame



Just counting nuclei in all the data would take 2 months.



ScaDS.AI

DRESDEN LEIPZIG

CENTER FOR SCALABLE DATA ANALYTICS
AND ARTIFICIAL INTELLIGENCE



Introduction to Bioimage Analysis

Robert Haase

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



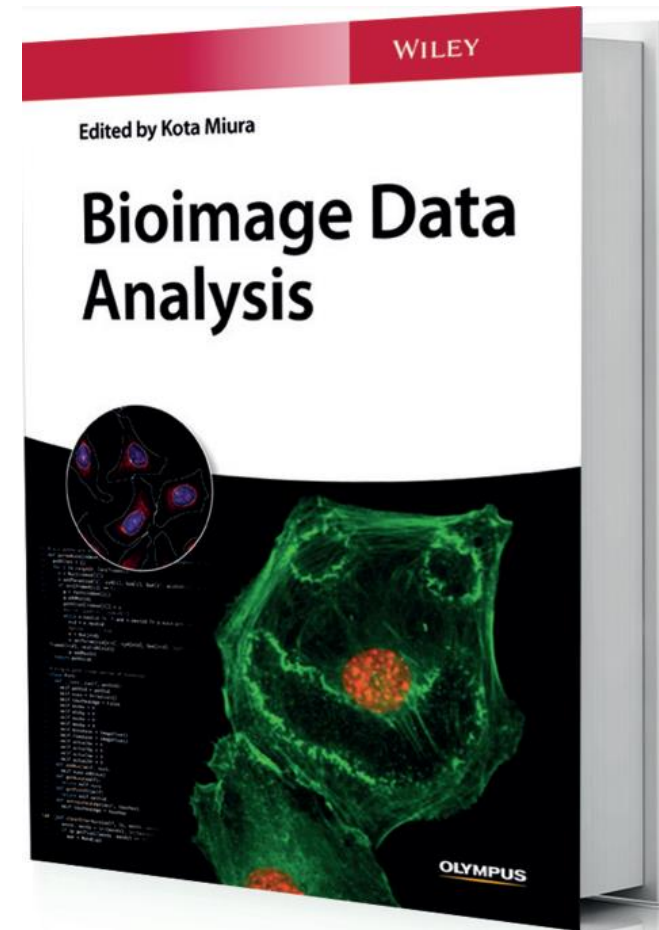
Diese Maßnahme wird gefördert durch die Bundesregierung aufgrund eines Beschlusses des Deutschen Bundestages. Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

Bioimage Analysis

- Kota Miura & Sebastien Tosi 2015:

In the light of this definition, image analysis, which is also called “computer vision,” aims at mimicking the way we see the world and how we identify its visible structures. Image analysis in biology does undeniably also hold this element, but more importantly, its main goal is to *measure* biological structures and phenomena in order to study and understand biological systems in a quantitative way.

To achieve this task, we in fact do not have to be bothered with similarity to the human recognition – we have more emphasis on the objectivity of quantitative measurement, rather than how that computer-based recognition becomes in agreement with human recognition. Therefore, in biology, image analysis is a process of identifying spatial distribution of biological components in images and measuring their characteristics to study their underlying mechanisms in an unbiased way. To underline this difference in the goals of image analysis in the two fields and to distinguish them from each other, we will now on refer to image analysis in biology as *bioimage analysis*.



Quantitative bio-image analysis

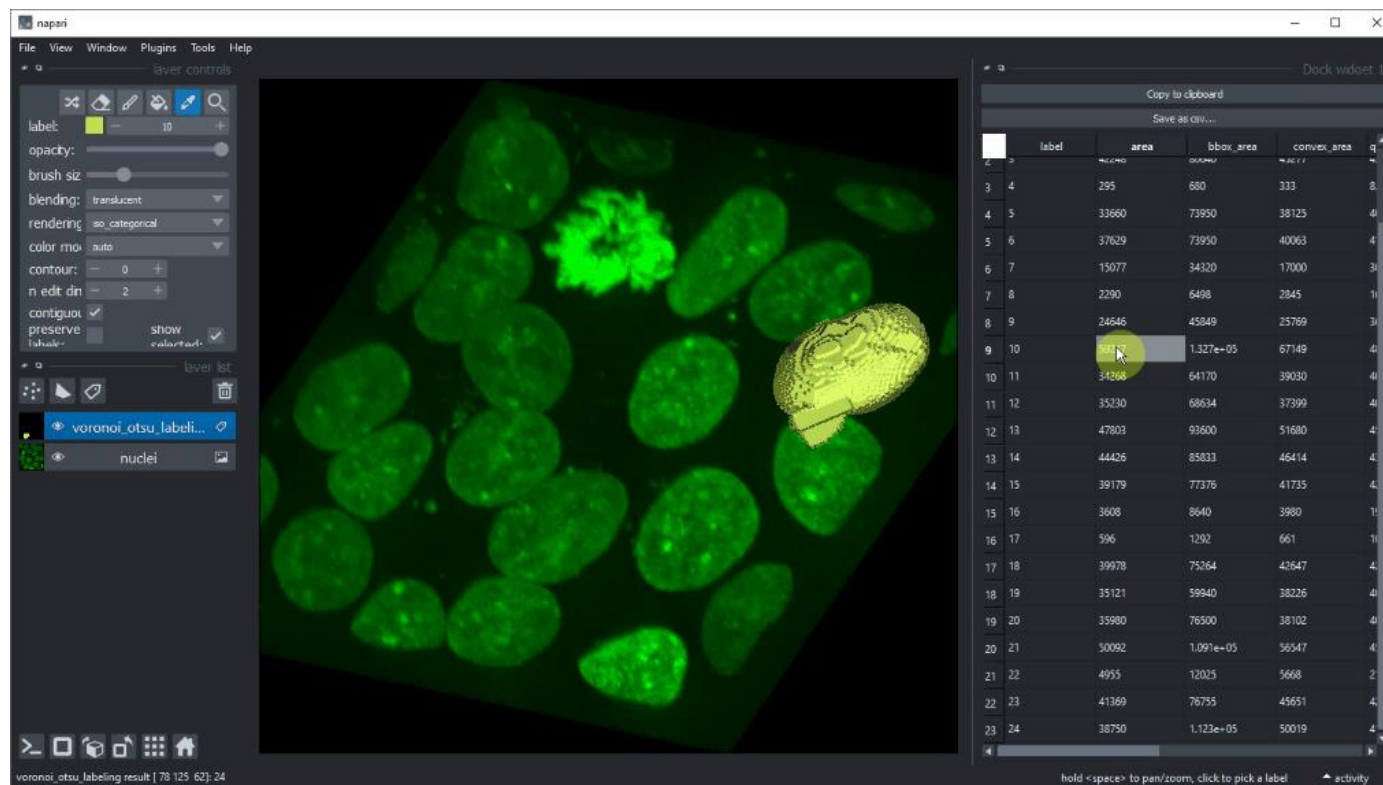
- Deriving quantitative information from images of biological samples taken with microscopes



$$\text{cat height} = \underline{1.5} \times \text{microscope height}$$

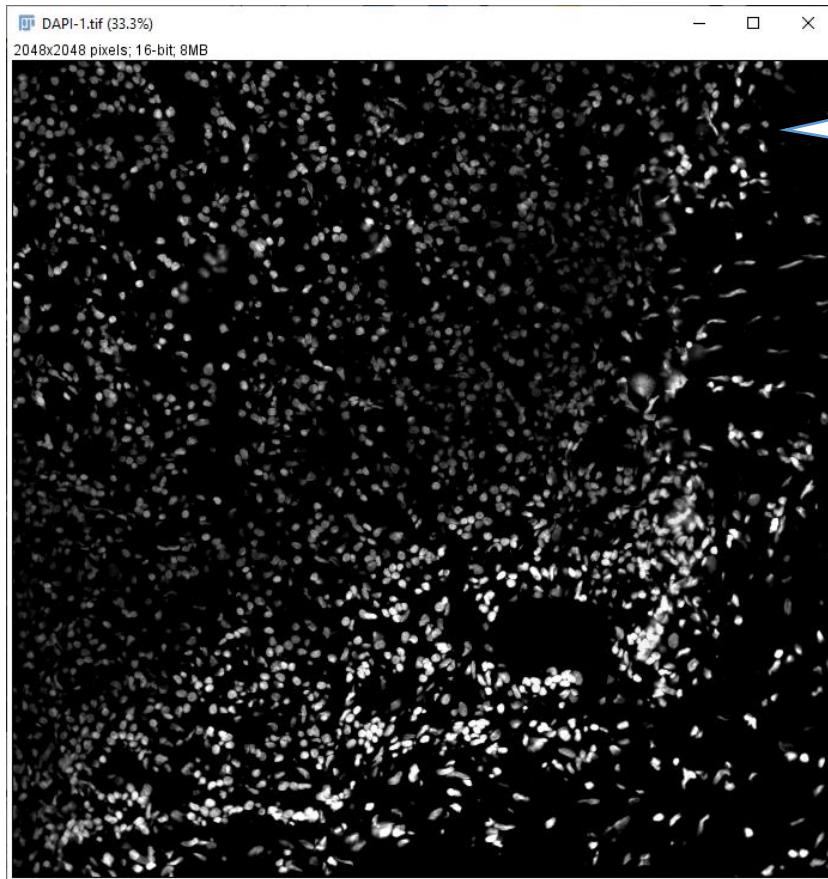
Quantitative bio-image analysis

- Deriving quantitative information from images of biological samples taken with microscopes + visualization



Objective bio-image analysis

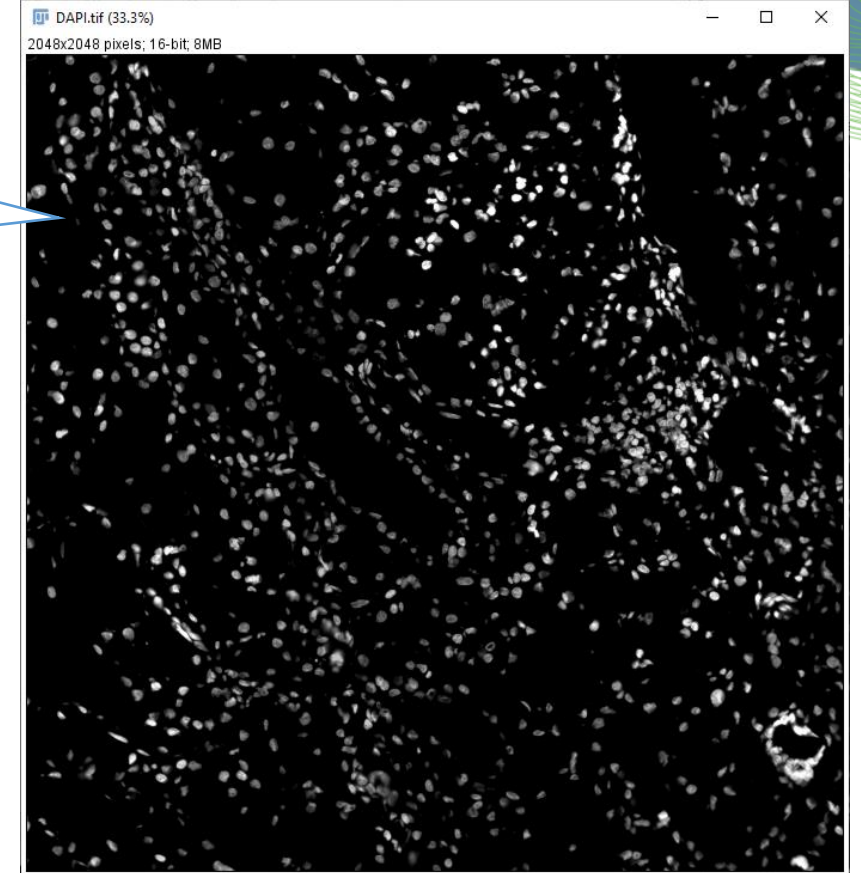
- Measurements should be objective, not influenced by human interpretation



Nuclei in this image are ...

... more dense than in this image.

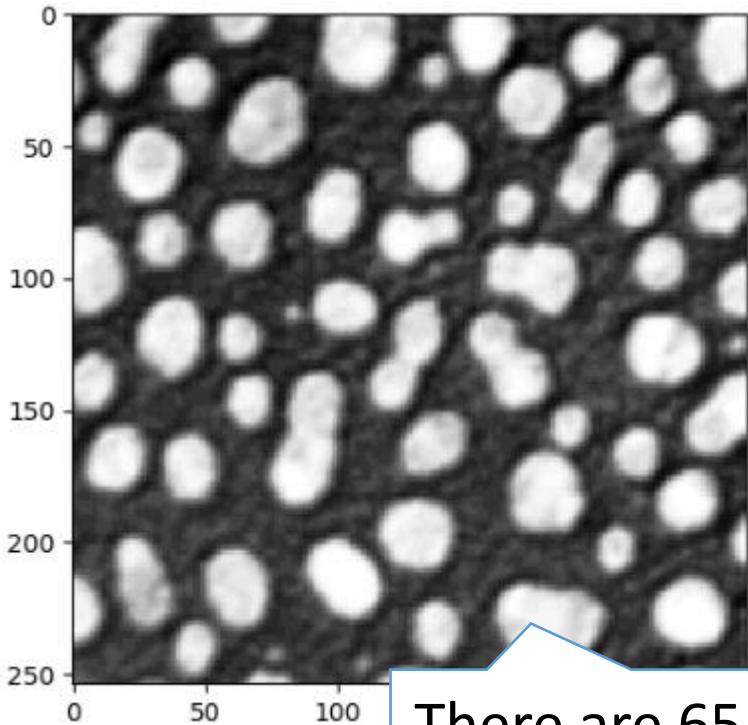
Use automation for less subjective analysis.



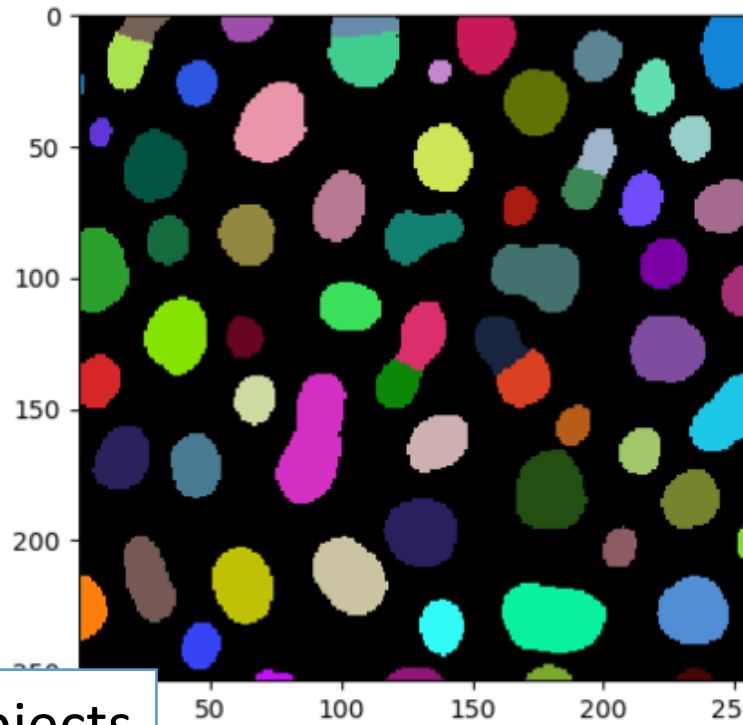
Reliable bio-image analysis

- Algorithms must be reliable (trustworthy).
- Visualization helps gaining trust in automated methods.

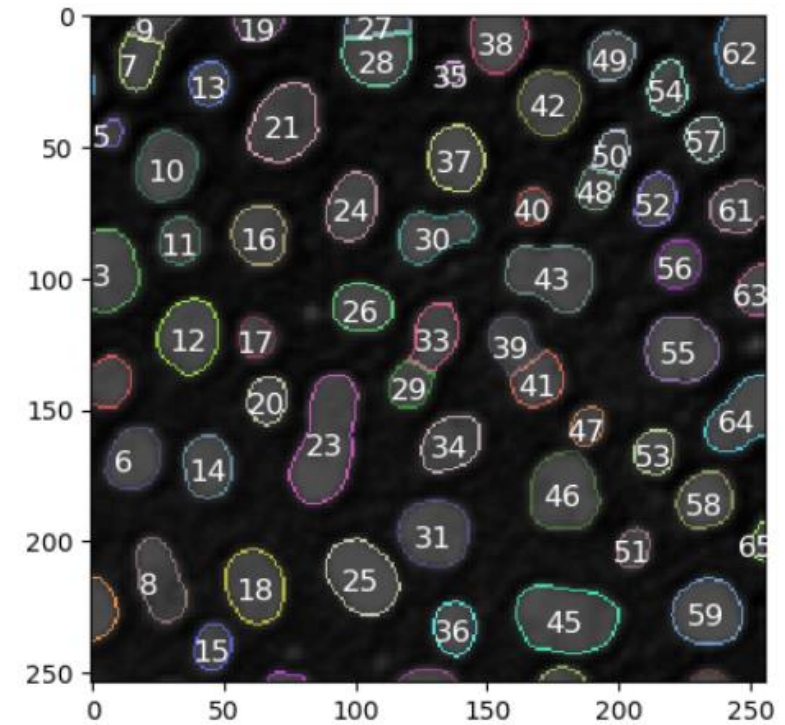
Original image



Label image



Overlay



There are 65 objects
in this image.

Reliable bio-image analysis

- Algorithms must be reliable (validated methods).
- Publicly available benchmark data sets allow to compare algorithms on common data.

Original image data

Human U2OS cells (out of focus)

Broad Bioimage Benchmark Collection

Annotated biological image sets for testing and validation

Introduction Image Sets Benchmarking Contribute

LEGEND: KINDS OF GROUND TRUTH

- C COUNTS
- F FOREGROUND/BACKGROUND
- O OUTLINES OF OBJECTS
- B BIOLOGICAL LABELS
- L LOCATION
- U BOUNDING BOXES

HOME /

Human U2OS cells (out of focus)

Accession number [BBBC006](#) - Version 1

Example images

Ground truth: in focus Out of focus Ground truth: foreground

“Ground truth” label images

Reproducible data analysis

- Allowing others to do your experiment again.
- “The image data was analyzed with Python.”

Can you reproduce what they did?

Reproducible bio-image analysis

- Allowing others to do your experiment again.
- “The image data was analyzed with Python.”

Can you reproduce what they did?

Can you reproduce what they did?

Trailer: Bio-image Analysis with Python

In the following chapters we will dive into image analysis, machine learning and bio-statistics using Python. This first notebook serves as a trailer of what we will be doing.

Python notebooks typically start with the imports of Python libraries which the notebook will use. The reader can check first if all those libraries are installed before going through the whole notebook.

```
import numpy as np
from skimage.io import imread, imshow
import pyclesperanto_prototype as cle
from cellpose import models, io
from skimage import measure
import pandas as pd
import apoc
```

Working with image data

We start with loading the image data of interest. In this example we load an image showing a zebrafish eye, courtesy of Mauricio Rocha Martins, Norden lab, MPI CBG Dresden.

```
# open an image file
multichannel_image = imread("../data/zfish_eye.tif")
print("Image size", multichannel_image.shape)
```

Image size (1024, 1024, 3)

Replicable bio-image analysis

- Others run the same analysis on their data and have consistent results / same conclusions.
- Can only be achieved if data analysis protocol was documented reproducibly.
- See also: *Replication crisis*
 - In Psychology (surveys)
 - In Medicine (clinical trials)
 - In Computer Science (executable code)
 - ...

Open access, freely available online

Essay

Why Most Published Research Findings Are False

John P. A. Ioannidis

Summary

There is increasing concern that most current published research findings are false. The probability that a research claim is true may depend on study power and bias, the number of other studies on the same question, and, importantly, the ratio of true to no relationships among the relationships probed in each scientific field. In this framework, a research finding is less likely to be true when the studies conducted in a field are smaller; when effect sizes are smaller; when there is a greater number and lesser preselection of tested relationships; where there is greater flexibility in designs, definitions, outcomes, and analytical modes; when there is greater financial and other interest and prejudice; and when more teams are involved in a scientific field in chase of statistical significance. Simulations show that for most study designs and settings, it is more likely for

factors that influence this problem and some corollaries thereof.

Modeling the Framework for False Positive Findings

Several methodologists have pointed out [9–11] that the high rate of nonreplication (lack of confirmation) of research discoveries is a consequence of the convenient, yet ill-founded strategy of claiming conclusive research findings solely on the basis of a single study assessed by formal statistical significance, typically for a p -value less than 0.05. Research is not most appropriately represented and summarized by p -values, but, unfortunately, there is a widespread notion that medical research articles

is characteristic of the field and can vary a lot depending on whether the field targets highly likely relationships or searches for only one or a few true relationships among thousands and millions of hypotheses that may be postulated. Let us also consider, for computational simplicity, circumscribed fields where either there is only one true relationship (among many that can be hypothesized) or the power is similar to find any of the several existing true relationships. The pre-study probability of a relationship being true is $R/(R+1)$. The probability of a study finding a true relationship reflects the power $1 - \beta$ (one minus the Type II error rate). The probability of claiming a relationship when none truly exists reflects the Type I error rate, α . Assuming that c relationships are being probed in the field, the expected values of the 2×2 table are given in Table 1. After a research finding has been claimed based on

It can be proven that most claimed research findings are false.

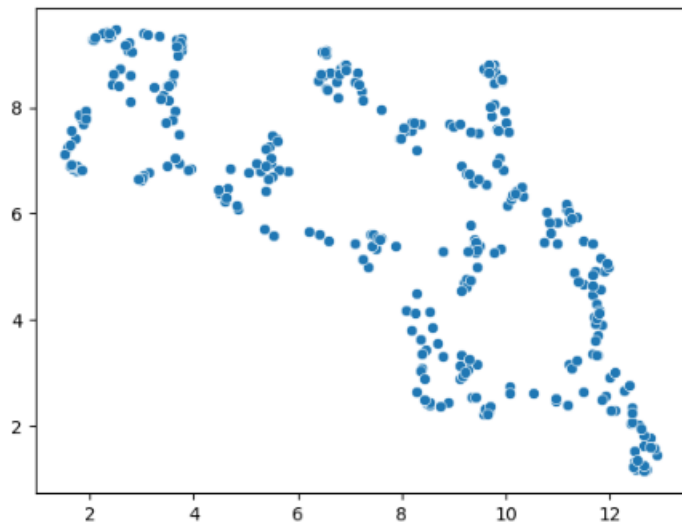
Repeatable data analysis

- In wet-lab experiments, samples may get destroyed while executing the experiment.

- Repeatable
improve

```
[11]: reducer = umap.UMAP()  
      embedding2 = reducer.fit_transform(scaled_statistics)  
  
      seaborn.scatterplot(x=embedding2[:, 0],  
                          y=embedding2[:, 1])
```

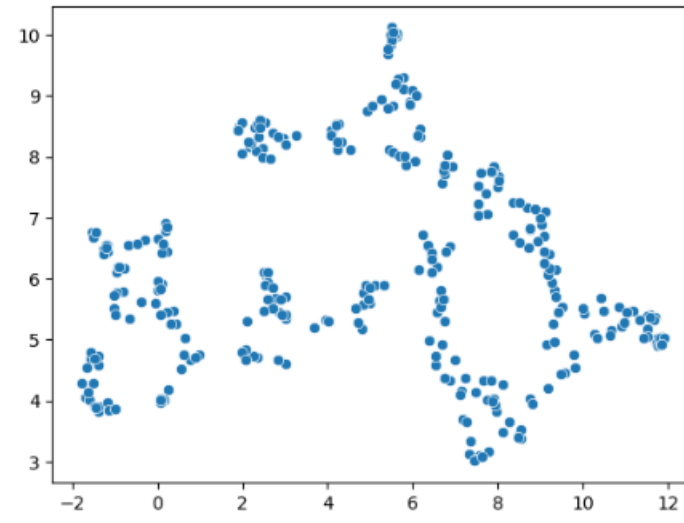
[11]: <AxesSubplot: >



up
ui

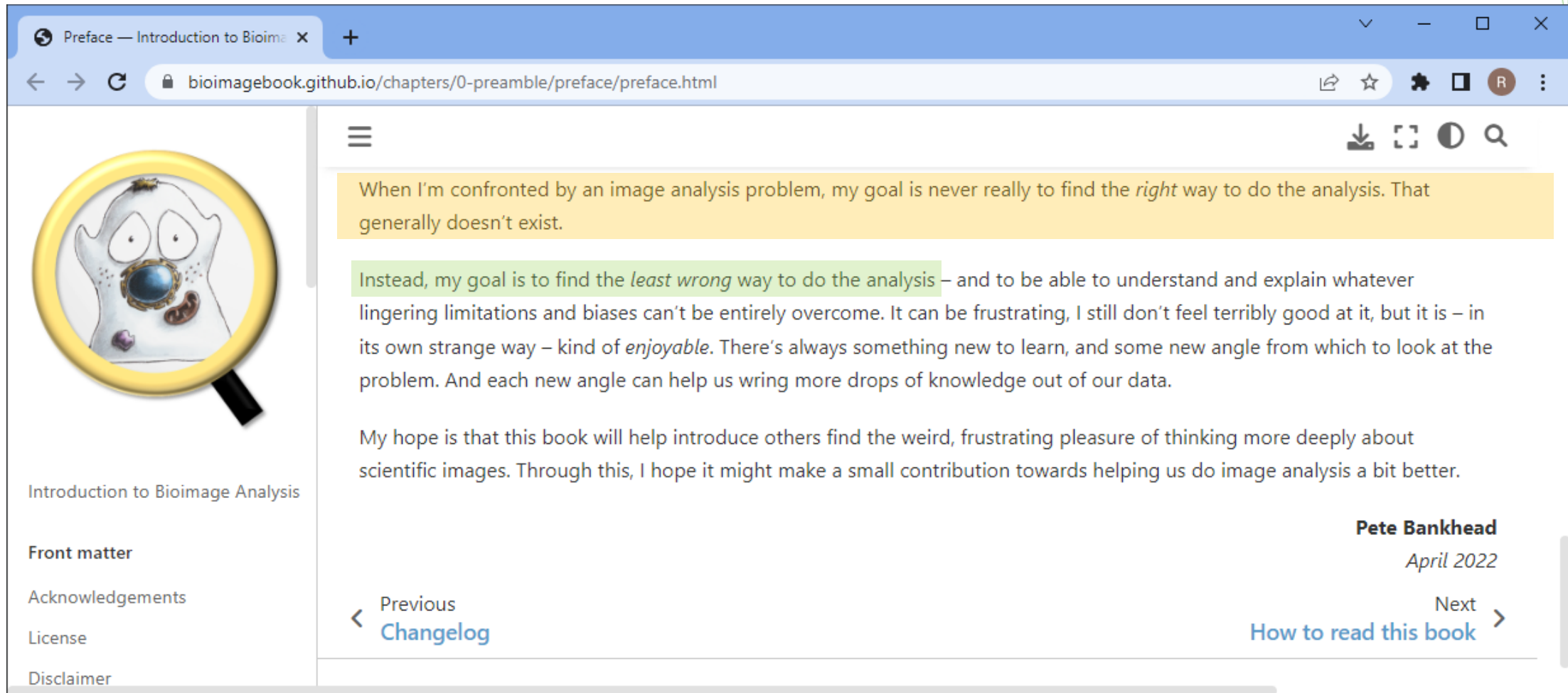
```
[12]: reducer = umap.UMAP()  
      embedding2 = reducer.fit_transform(scaled_statistics)  
  
      seaborn.scatterplot(x=embedding2[:, 0],  
                          y=embedding2[:, 1])
```

[12]: <AxesSubplot: >




annot

Bio-image Analysis: good scientific practice



Preface — Introduction to Bioimage Analysis

bioimagebook.github.io/chapters/0-preamble/preface/preface.html



Introduction to Bioimage Analysis

- Front matter
- Acknowledgements
- License
- Disclaimer

When I'm confronted by an image analysis problem, my goal is never really to find the *right* way to do the analysis. That generally doesn't exist.

Instead, my goal is to find the *least wrong* way to do the analysis – and to be able to understand and explain whatever lingering limitations and biases can't be entirely overcome. It can be frustrating, I still don't feel terribly good at it, but it is – in its own strange way – kind of *enjoyable*. There's always something new to learn, and some new angle from which to look at the problem. And each new angle can help us wring more drops of knowledge out of our data.

My hope is that this book will help introduce others find the weird, frustrating pleasure of thinking more deeply about scientific images. Through this, I hope it might make a small contribution towards helping us do image analysis a bit better.

Pete Bankhead
April 2022

< Previous
Changelog

Next
How to read this book >

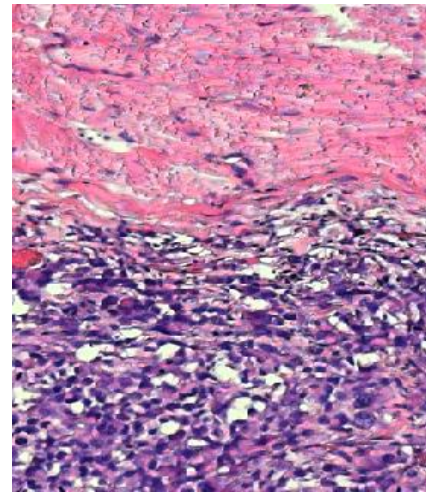
Bio-image analysis is supposed to be

- **Quantitative**
 - We derive numbers from images which describe physical properties of the observed sample.
- **Objective**
 - The derived measurement does not depend on who did the measurement. The measurement is free of interpretation.
- **Reliable (trustworthy / validated)**
 - We are confident that the measurement is describing what it is supposed to describe.
- **Reproducible**
 - Enabling others to re-do the experiment. For this, documentation is crucial!
- **Replicability**
 - Others *do* execute the same analysis, potentially on other data, and see consistent results.
- **Repeatable**
 - We can do the same experiment twice under the *same conditions* and get the same measurements.

Common topics

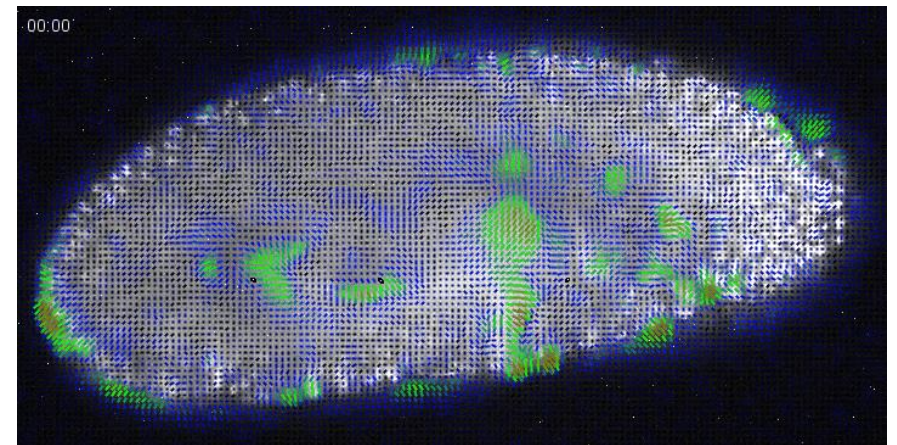
- Typical questions bio-image analysts deal with
 - Is signal intensity different under varying conditions?
 - How many cells are in my image?
 - How high is cell density?
 - Bio-statistics / medicine / disease staging
 - How are different tissues characterized?

- Typical questions bio-image analysts struggle with
 - What force drives the observed processes?
 - What is the lineage tree of one particular cell?
 - Are observation A and observation B related?
 - Are structures observed in different color channels colocalized?



muscle, normal tissue

squamous-cell carcinoma



Hypothesis-driven quantitative biology

- Hypothesis: Cell shape can be influenced by modifying X.
- Null-Hypothesis: Circularity of modified cells is similar to cells in the control group.
 - Sample preparation
 - Imaging
 - Cell segmentation
 - Circularity measurement
 - Statistics

Should we use a different segmentation algorithm?

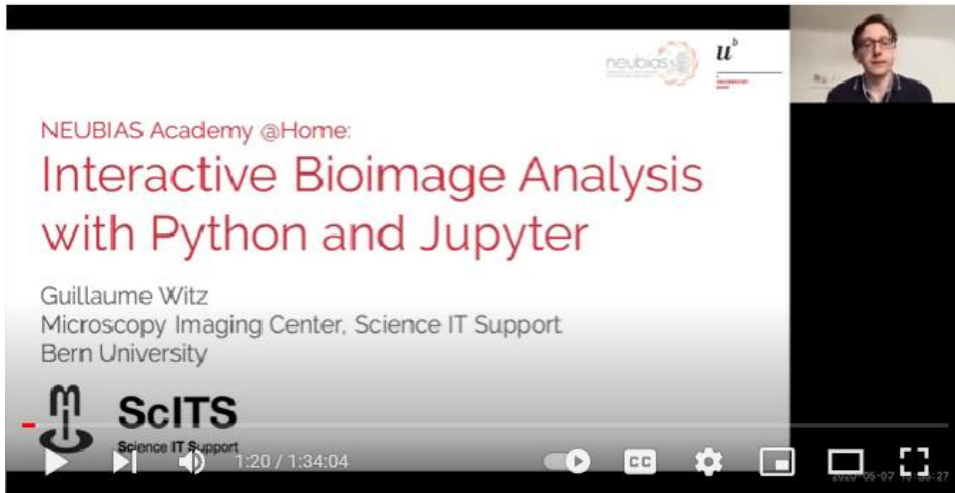
Shall we use a different microscope?

Is circularity the right parameter to measure?

Hypothesis *generating* quantitative biology

- ~~Hypothesis: Cell shape can be influenced by modifying X.~~
 - Question: Which image-derived parameter is influenced when modifying X?
 - Sample preparation
 - Imaging
 - Cell segmentation algorithm A, algorithm B, algorithm C
 - Measurement of circularity, solidity, elongation, extend, texture, intensity, topology ...
 - Statistics
- Which segmentation algorithms allow measurements that show a relationship with X?
- Why?
- Which parameter shows any relationship with X?

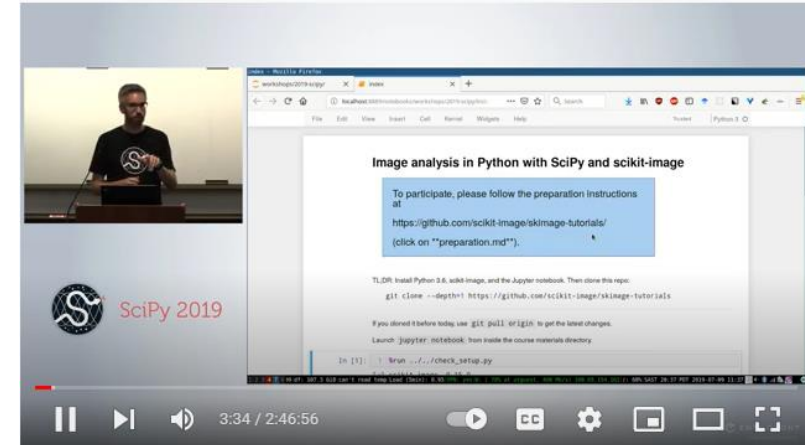
More resources



Guillaume Witz, NEUBIAS Academy 2020

Watch more:

- <https://www.youtube.com/watch?v=2KF8vBrp3Zw>
- <https://www.youtube.com/watch?v=d1CIV9irQAY>
- https://www.youtube.com/watch?v=X_pCiVQ4c4E



Stéfan van der Walt, Juan Nunez-Iglesias, SciPy 2019



Sreenivas Bhattiprolu, Python for Microscopists @Youtube 2019-...

The Image Science Community

- Ask your question online and an expert will likely reply the same day 😊

Install issues with py-clesperanto x +
https://forum.image.sc/t/install-issues-with-py-clesperanto-assistant/54680

image.sc Zulip chat Related Forums

Community Partners

Install issues with py-clesperanto-assistant

Usage & Issues napari, pyclesperanto, clesperanto

William Giang WillGiang Jul 2021

Hi @haesleinhuepf ,

I'm encountering troubles with installing the napari-pyclesperanto-assistant plugin (v0.9.3) for napari 0.4.10 on Windows 10.

Here was my installation process:

First, I have Visual Studio Build Tools 2019 with "Desktop development with C++" as well as

- MSVC v142 - VS 2019 C++ x64/x86 build tools (Latest)
- Windows 10 SDK (10.0.19041.0)
- C++ CMake tools for Windows
- Testing tools core features - Build Tools
- C++ AddressSanitizer

After creating and activating a conda environment with python=3.8, I can pip install "napari[all]" without issue.

Following the [install instructions](#), I downloaded and installed a .whl file with OpenCL 1.2 and Python 3.8. Note that it was pyopenc1-2021.2.3+c112-cp38-cp38-win_amd64.whl instead of the (no longer available) 2020.3.1 version. The log for the napari-pyclesperanto plugin install suggests that it's trying to install 2021.2.1 and also suggests installing pybind11 and mako.

Install issues with py-clesperanto x +
https://forum.image.sc/t/install-issues-with-py-clesperanto-assistant/54680/2

Install issues with py-clesperanto-assistant

Usage & Issues napari, pyclesperanto, clesperanto

Robert Haase haesleinhuepf clij & clesperanto maintainer Jul 2021

Hi @WillGiang ,

thanks for reporting! I recently hit issues with recent pyopenc1 > 2021.2.1 and thus, limited that dependency. Maybe, that was a fault. I'm considering removing this with the next minor release next week.

In the meantime, can you try installing using conda?

```
conda create --name bio11 python==3.8.5
conda activate bio11
conda install -c conda-forge pyopenc1==2021.2.1
pip install napari-pyclesperanto-assistant
pip install napari[all]
```

Let us know if this helps!

Cheers,
Robert

Solution 2 ❤️ 🔗 ✎ ... ↩ Reply

Setup.py for plugin that depends on pyclesperanto 1

William Giang WillGiang Jul 2021

With your instructions to specifically use python 3.8.5 and pyopenc1 v2021.2.1, the install goes through

Quiz

- Enabling others to do your experiment is about ...

Repeatability

Reproducibility

Replicability

Reliability

Quiz

- Reproducibility can be achieved by

Writing
documentation

Writing
code

Providing
example
data

Recording
Video
tutorials

Quiz

- Resolution in imaging describes...

Size of pixels
on a screen

Size of pixels
on a camera
chip

Maximum size
of objects in
relation to the
image

Minimum size
of objects that
can be
differentiated
in an image



Summary

Today, you learned

- Microscopy
 - Fluorescence microscopy
 - Light-sheet microscopy
- Bio-image analysis
 - Quantitative
 - Objective
 - Reproducible
 - Repeatable
 - Reliable

Coming up next

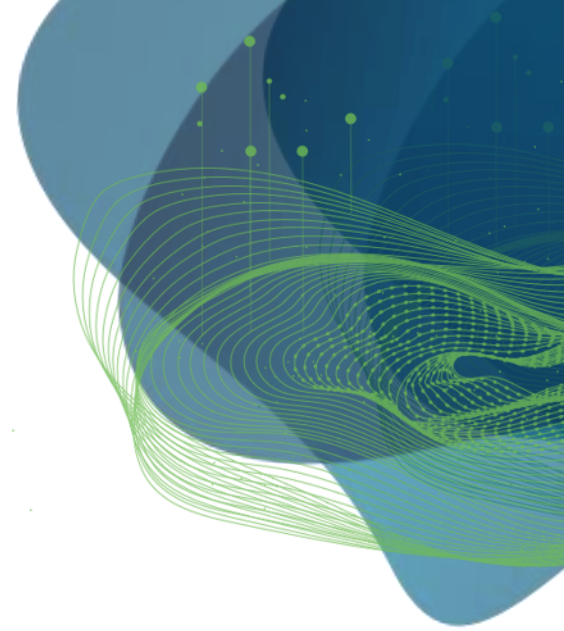
- Exercises: Setting up your environment

Next week(s):

- Research Data Management
 - Image processing for microscopy
- 

Exercises

Robert Haase



GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



Diese Maßnahme wird gefördert durch die Bundesregierung aufgrund eines Beschlusses des Deutschen Bundestages. Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

Exercises

The screenshot displays the GitHub repository page for 'ScaDS/BIDS-lecture-2024'. The repository is a public repository with 1 star, 0 forks, 2 watchers, 1 branch, and 0 tags. The README file is open, showing the following content:

Bio-image Data Science

This repository contains training resources for Students at Uni Leipzig who want to dive into bio-image data science with Python. The material will develop between April and July 2024 and shared here in this github repository.

Teaching Goal

Students learn the full workflow of common bio-image data science projects to a degree that they can execute a scientific data analysis project in this context on their own. They will be familiar with common bio-image analysis algorithms and workflows, how to choose them according to a scientific goal, and how to measure quality of derived results. Attending the lecture and executing the practicals qualifies the students to work as bio-image data scientist in the pharmaceutical industry or basic biological research.

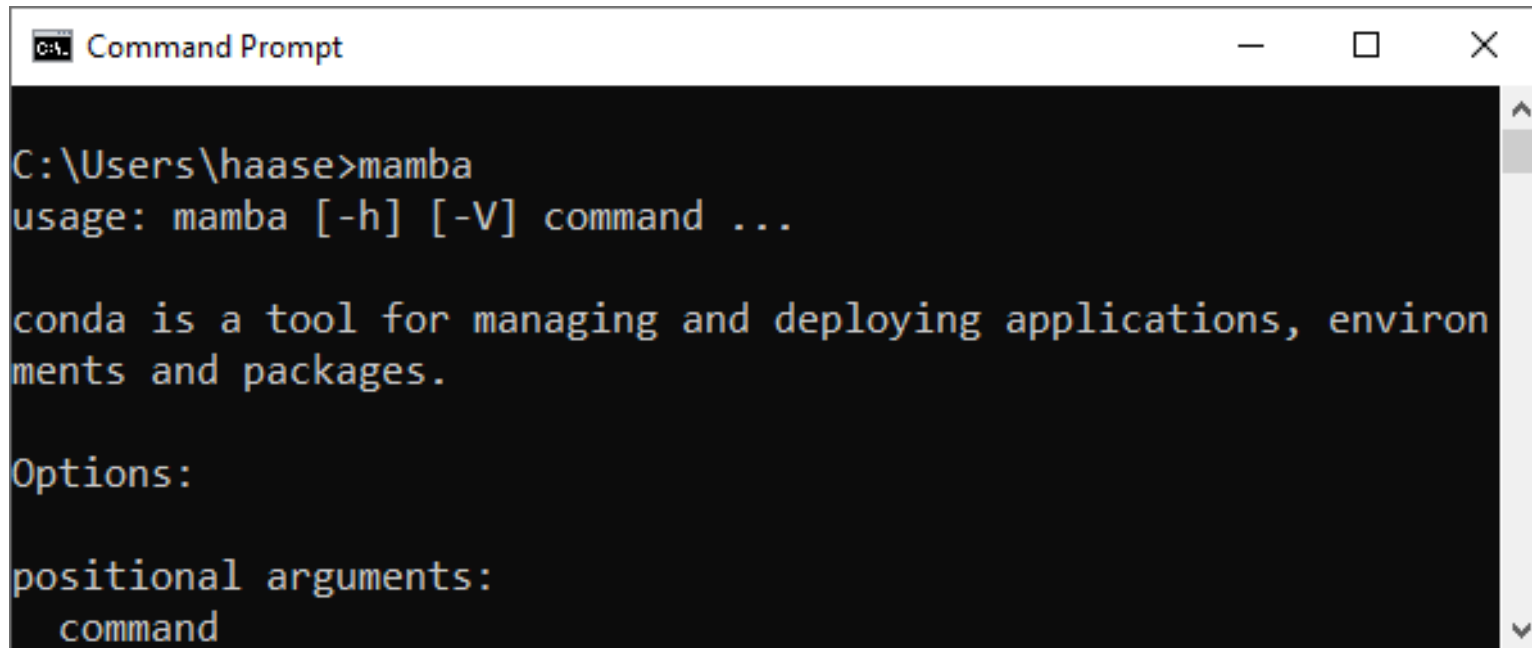
Course contents

- [Introduction to Bio-image Data Science](#) (Apr 2nd 2024)
 - Basics of microscopy
 - Introduction to Bio-image Analysis
 - Exercises:
 - [Setting up a local environment](#)
 - [Setting up Jupyter Hub at Scientific Computing / Leipzig University](#)
 - [Execute the trailer notebook](#)



Installation

- You can skip the first local installation steps if you already use mamba



```
Command Prompt
C:\Users\haase>mamba
usage: mamba [-h] [-V] command ...

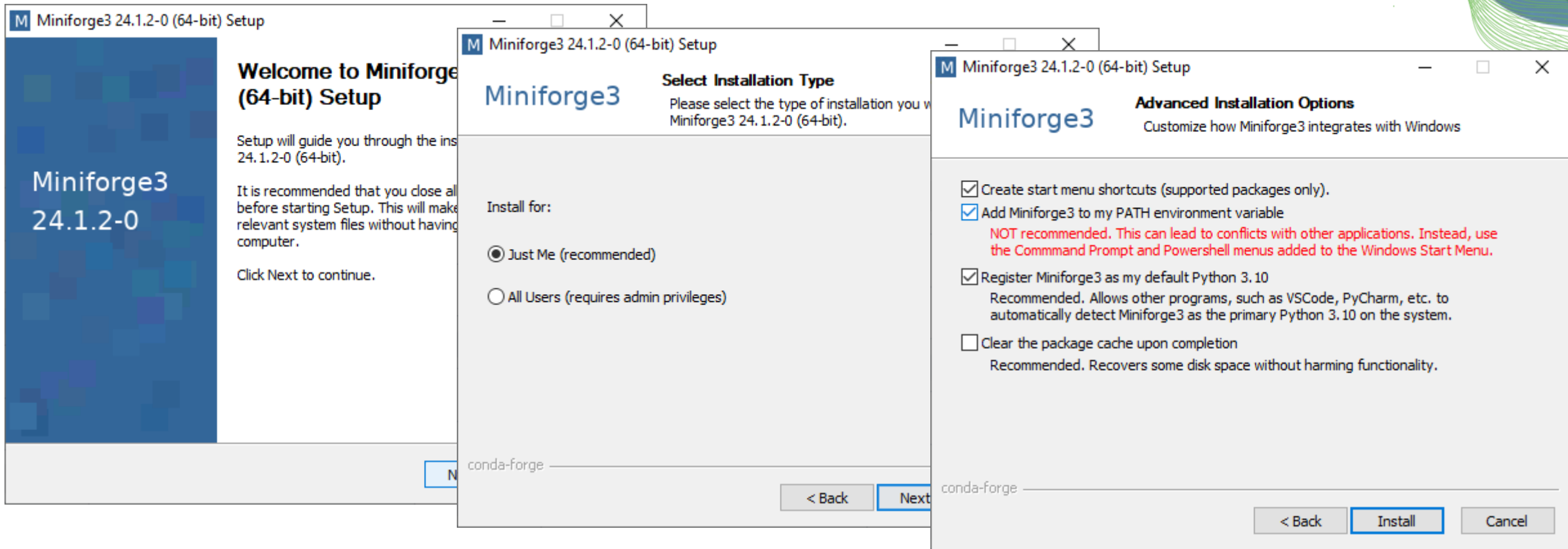
conda is a tool for managing and deploying applications, environ
ments and packages.

Options:

positional arguments:
  command
```

Installation

- Install mini-forge



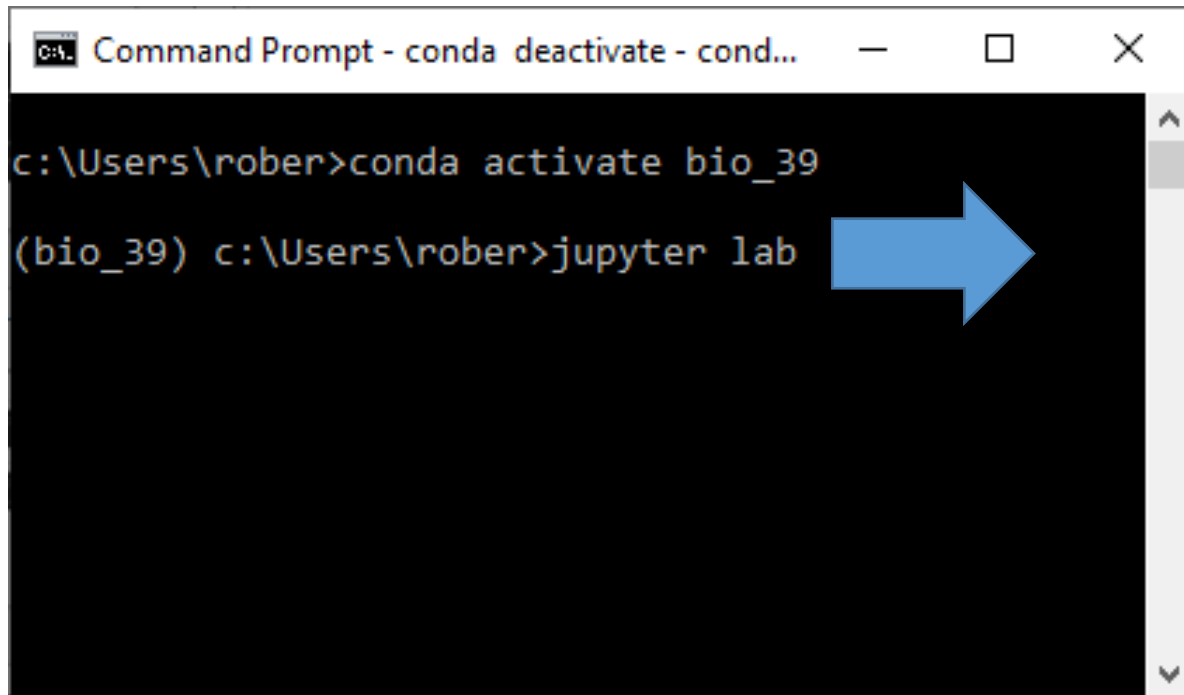
Setting up an environment

- Create a conda/mamba environment

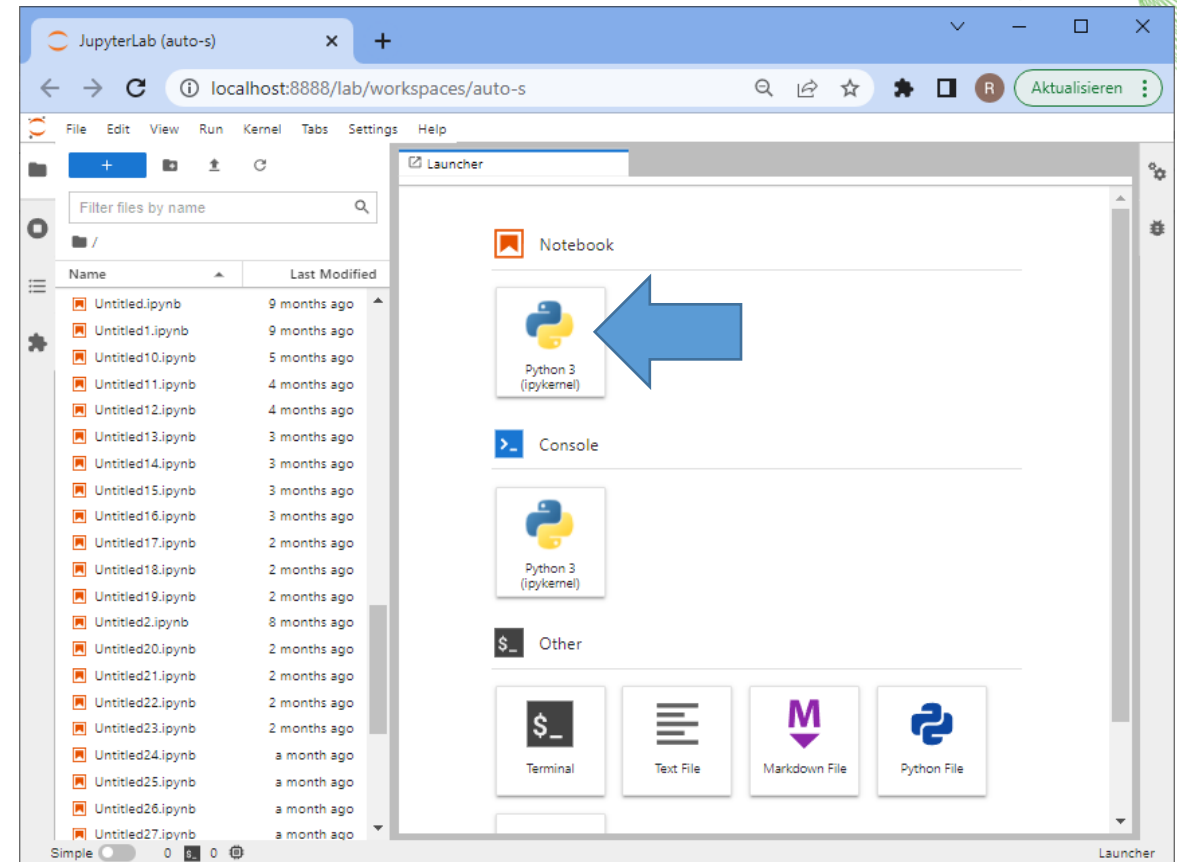
```
mamba create --name my_fist_env  
python=3.9 devbio-napari pyqt -c  
conda-forge
```


Jupyter lab

- Start Jupyter lab from the folder you want to work in
- Create a new notebook

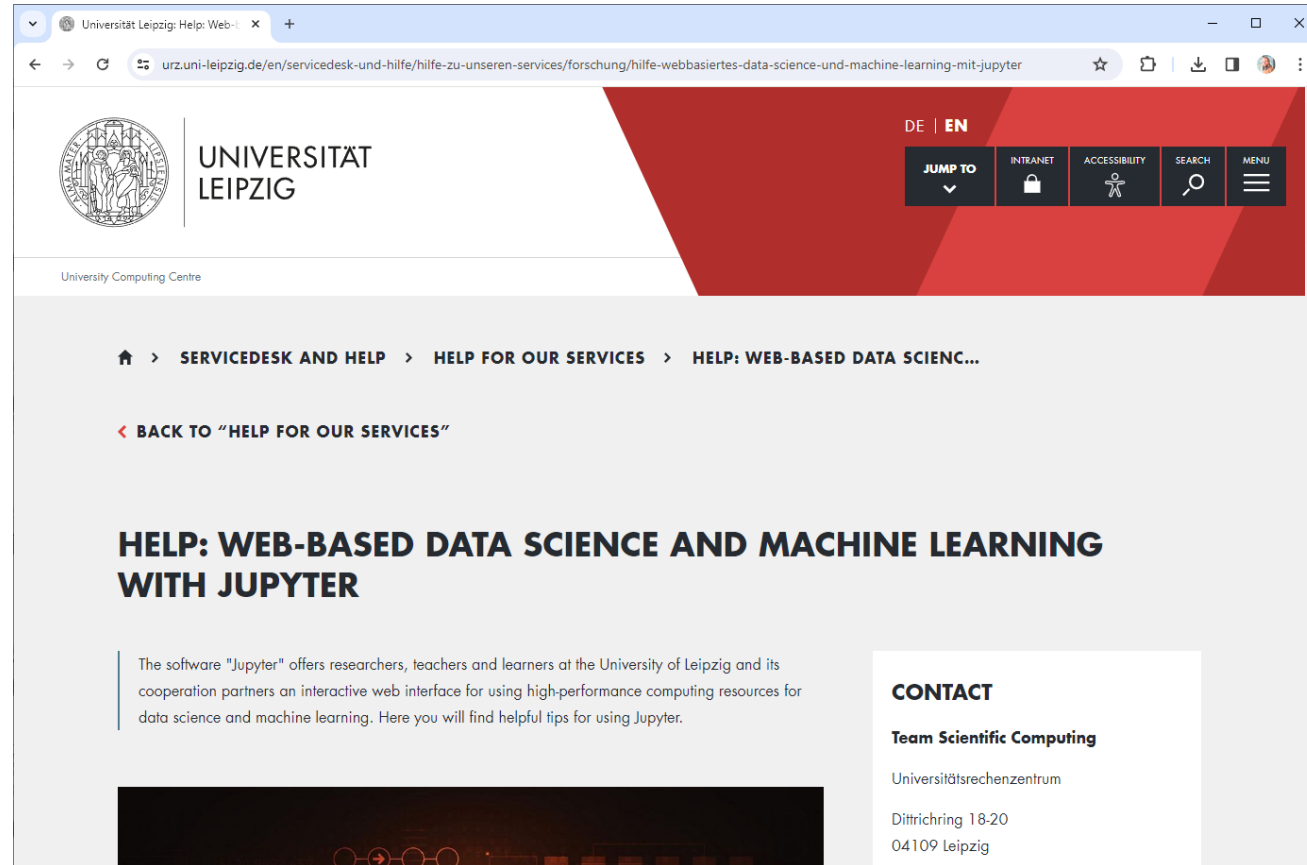


```
C:\Users\rober>conda activate bio_39
(bio_39) c:\Users\rober>jupyter lab
```



Setting up the JupyterHub @ Scientific Computing ULEI

- Alternatively: Register @ Scientific Computing / Uni Leipzig



The screenshot shows a web browser window displaying the University of Leipzig website. The URL in the address bar is <https://www.urz.uni-leipzig.de/en/servicedesk-und-hilfe/hilfe-zu-unseren-services/forschung/hilfe-webbasiertes-data-science-und-machine-learning-mit-jupyter>. The page features the University of Leipzig logo and name, along with navigation links for DE and EN. The main content area is titled "HELP: WEB-BASED DATA SCIENCE AND MACHINE LEARNING WITH JUPYTER". Below the title, there is a brief description of the software and a "CONTACT" section for the Team Scientific Computing, located at the University Computing Centre, Ditttrichring 18-20, 04109 Leipzig.

Setting up the JupyterHub @ Scientific Computing ULEI

The image displays three overlapping browser windows illustrating the JupyterHub setup process:

- Left Window:** The JupyterHub home page. It features the university logo and a "Sign in" button. Below the button are input fields for "Username:" and "Password:", followed by another "Sign in" button.
- Middle Window:** The "Resource selection" page. It shows configuration options for a new server:
 - Memory:** 16 GB
 - Number of CPUs:** 4
 - Partition:** clara
 - GPU:** RTX 2080TiA red "Start" button is located at the bottom right of this window.
- Right Window:** A confirmation page with the text: "Your server is starting up. You will be redirected automatically when it's ready for you." Below this text is an "Event log" section.

Setting up the JupyterHub @ Scientific Computing ULEI

- Detailed instructions



Installation instructions for devbio-napari on clara

These instructions are derived/modified from [this page](#).

Request a [Scientific Computing Account at ULEI](#).

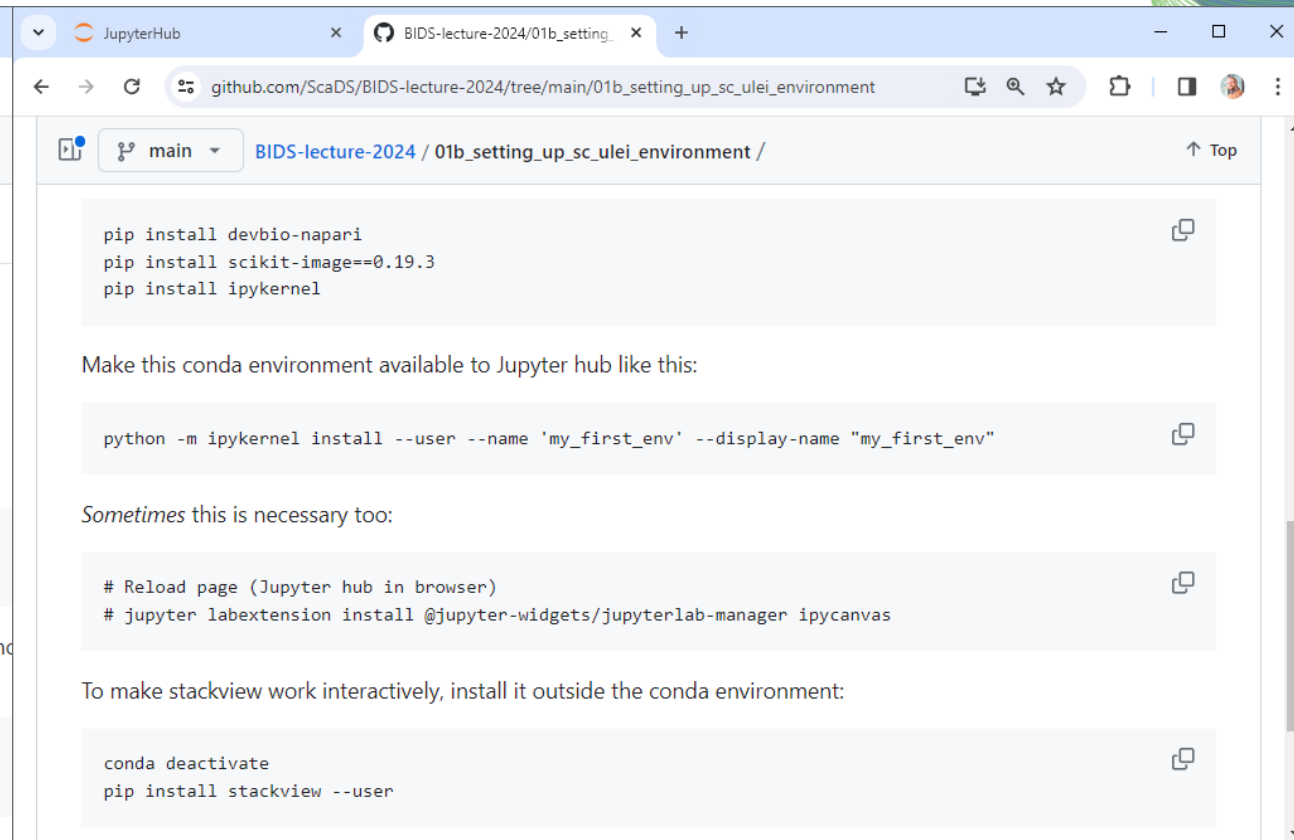
Login to VPN and [Jupyter Hub](#) and open a terminal.

Install [ana]conda

```
module load Anaconda3
conda init bash
```

At this point, we need to reopen the terminal. Afterwards, create a conda environment with one specific python version only.

```
conda create --name my_first_env python=3.9
conda activate my_first_env
```



```
pip install devbio-napari
pip install scikit-image==0.19.3
pip install ipykernel
```

Make this conda environment available to Jupyter hub like this:

```
python -m ipykernel install --user --name 'my_first_env' --display-name "my_first_env"
```

Sometimes this is necessary too:

```
# Reload page (Jupyter hub in browser)
# jupyter labextension install @jupyter-widgets/jupyterlab-manager ipycanvas
```

To make stackview work interactively, install it outside the conda environment:

```
conda deactivate
pip install stackview --user
```


Exercise: Test the environment

- Download and test the trailer notebook

The image displays three overlapping screenshots of a JupyterLab environment, illustrating a workflow for bio-image data science. Each screenshot shows a different notebook page with code and visual outputs.

Trailer: Bio-image Data Science

In this semester, we will analyze microscopy images. A very basic workflow is demonstrated here. We will go through all the details later on.

```
[1]: from skimage.io import imread
from skimage.filters import threshold_otsu
from skimage.measure import label, regionprops_table
import stackview
import pandas as pd
import numpy as np
```

Working with images

First we load an image and display it to get some first insights.

```
[2]: image = imread('https://samples.fiji.sc/blobs.png')[0]
stackview.insight(image)
```

```
[2]:
```

shape (254, 256)
dtype uint8
size 63.5 kB

Image segmentation

To analyze the individual objects, we need to segment them. A basic algorithm doing this involves

- Otsu's method
- Connected component labeling

```
[3]: binary = image > threshold_otsu(image)
labels = label(binary)
stackview.insight(labels)
```

```
[3]:
```

shape (254, 256)
dtype uint8
size 63.5 kB
min 0
max 255

Correlation analysis

We can also study how the measured parameters are correlated

```
[7]: corr = table.corr()
def colorize(styler):
    styler.background_gradient(axis=None, cmap="seismic")
    return styler
corr.style.pipe(colorize)
```

```
[7]:
```

	area	perimeter	mean_intensity	minor_axis_length	major_axis_length
area	1.000000	0.961579	0.622818	0.898116	
perimeter	0.961579	1.000000	0.707842	0.873781	
mean_intensity	0.622818	0.707842	1.000000	0.743692	
minor_axis_length	0.898116	0.873781	0.743692	1.000000	
major_axis_length	0.894802	0.962880	0.617727	0.713343	