

Introduction to image analysis using ImageJ/Fiji

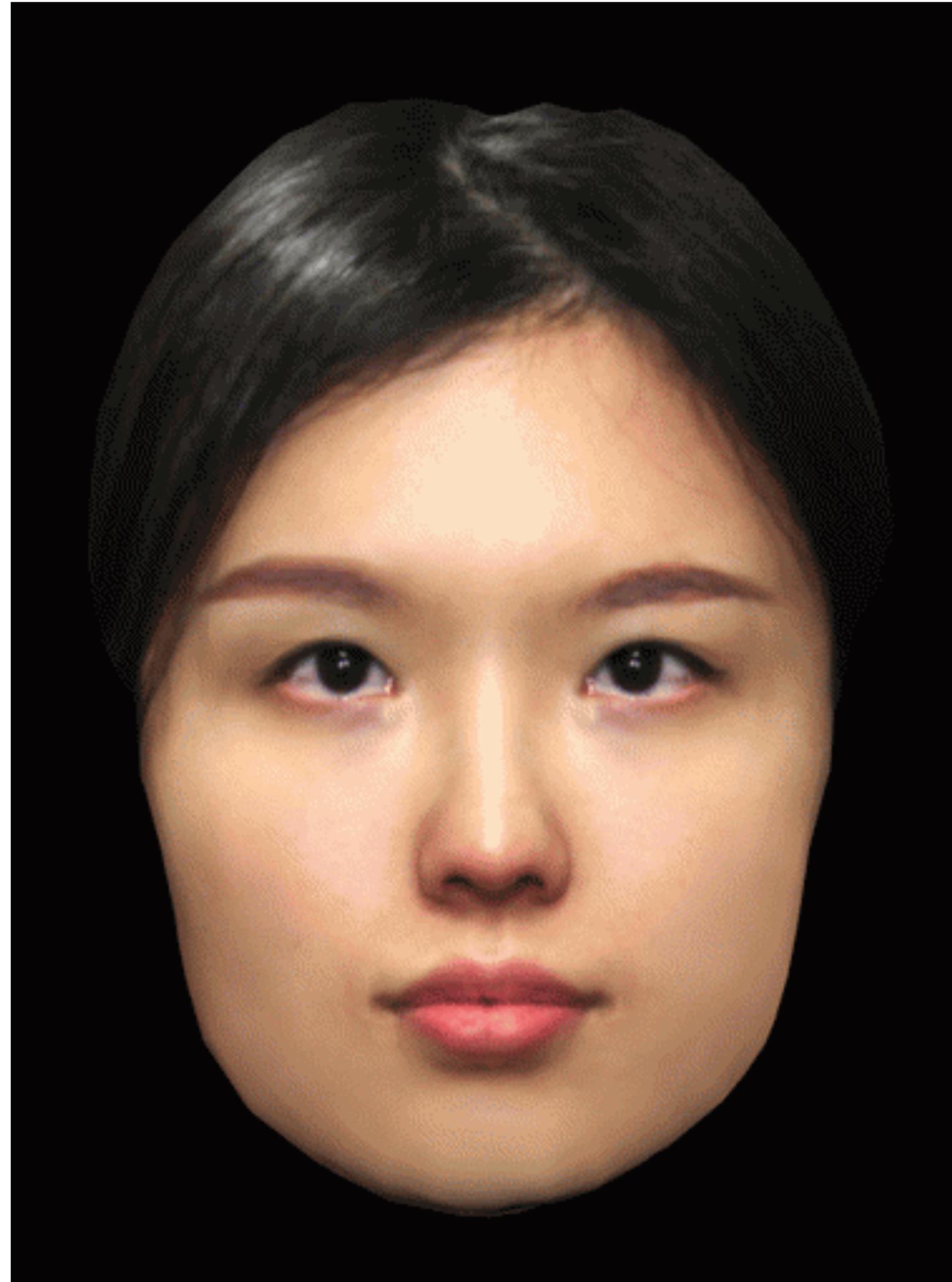
Federico Gasparoli & Simon F. Nørrelykke

Get the course materials

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

One-stop resource for everything we'll cover today

Problem: Seeing is Believing



Mask rotates continuously in the same direction
(nose passing right to left on screen)

Can you see the inside of the mask?

Learning objectives

1. **Motivate** the use of algorithms in image analysis
2. **Introduce** some image-analysis nomenclature
3. **Learn** to use Fiji effectively and reproducibly
4. **Touch** on deep learning

Reasons for Learning About Image Processing

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

Reasons for Not Learning About Image Processing

- ...

none

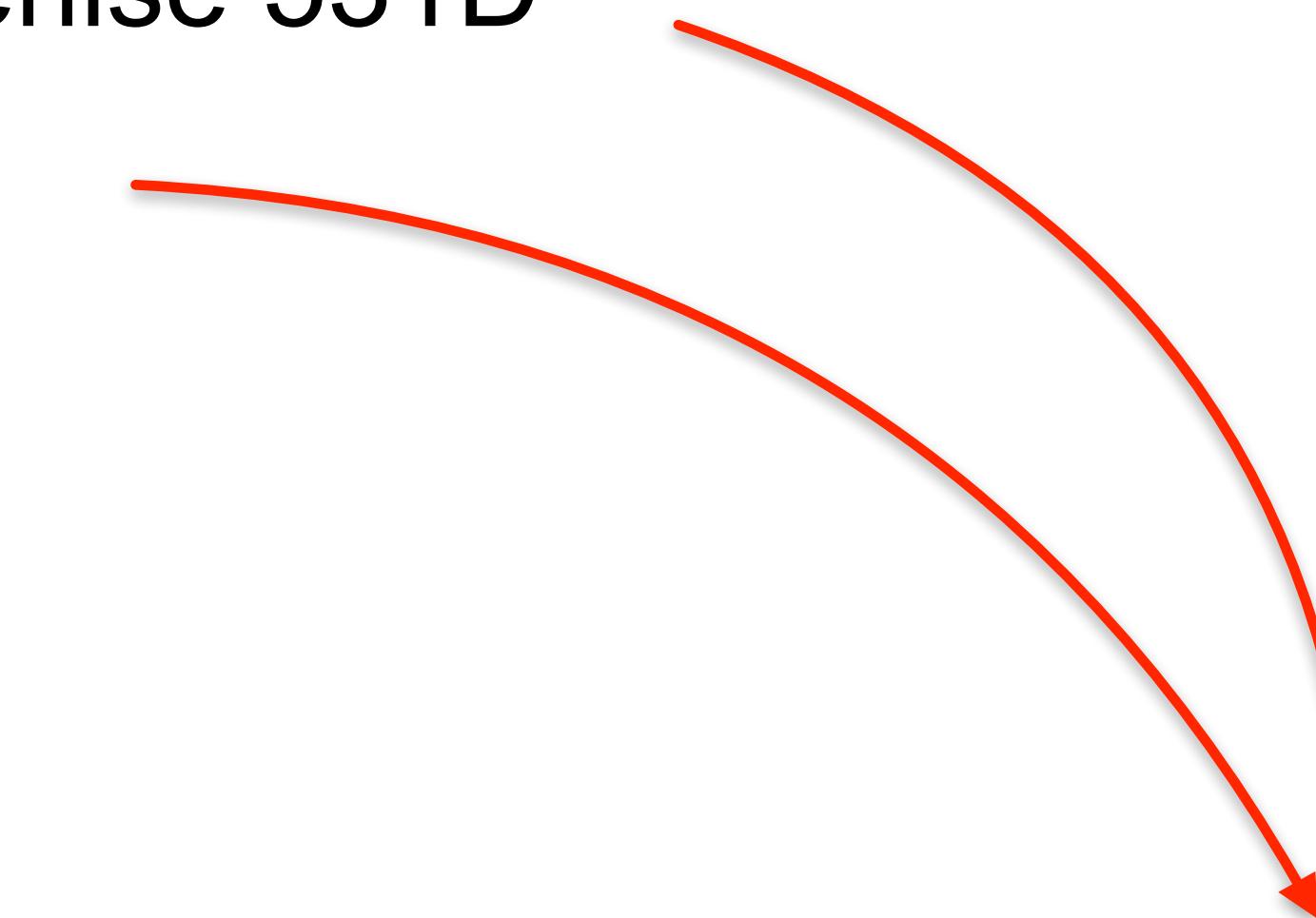
Acknowledgements

- Szymon Stoma, ETH Zurich, Switzerland
 - slides and workflows
- Peter Bankhead, Edinburgh University, Scotland
 - slides and graphics

I.A.C. (Image Analysis Collaboratory)

@ HMS since Sep '22

- **Support and collaboration:** for image and data analysis projects
- **Teaching:** at HMS and internationally
- **IT:** access to relevant solutions, open- and closed-source
- **R&D:** centered on methods and tools
- **Locations:** LHRRB 105 & Armenise 531D
- **Consults:** Contact Simon
- **simon@hms.harvard.edu**



<https://iac.hms.harvard.edu/>

Who we are, currently



Antoine A. Ruzette

Associate



Ranit Karmakar

Specialist Postdoc



Maria Theiss

Specialist Postdoc



Federico Gasparoli

Research Associate



Simon F. Nørrelykke

Director, Lecturer

Backgrounds in

Physics

Biology

Microscopy

Bioengineering

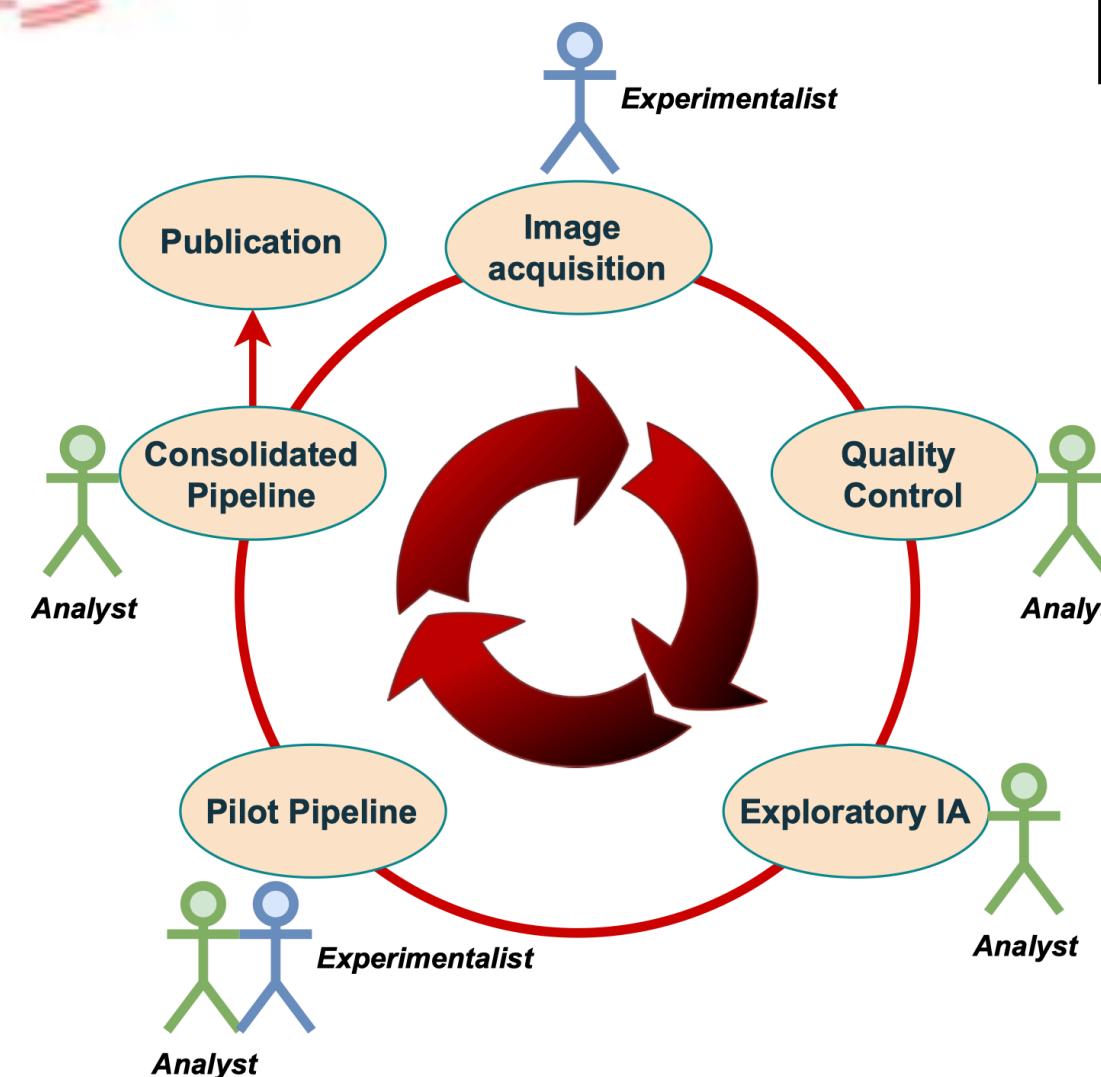
Computer Engineering

How we operate



Consultations
“Image Clinics”

Teaching and Training
Courses in image analysis
Postdoc training program



Collaborations

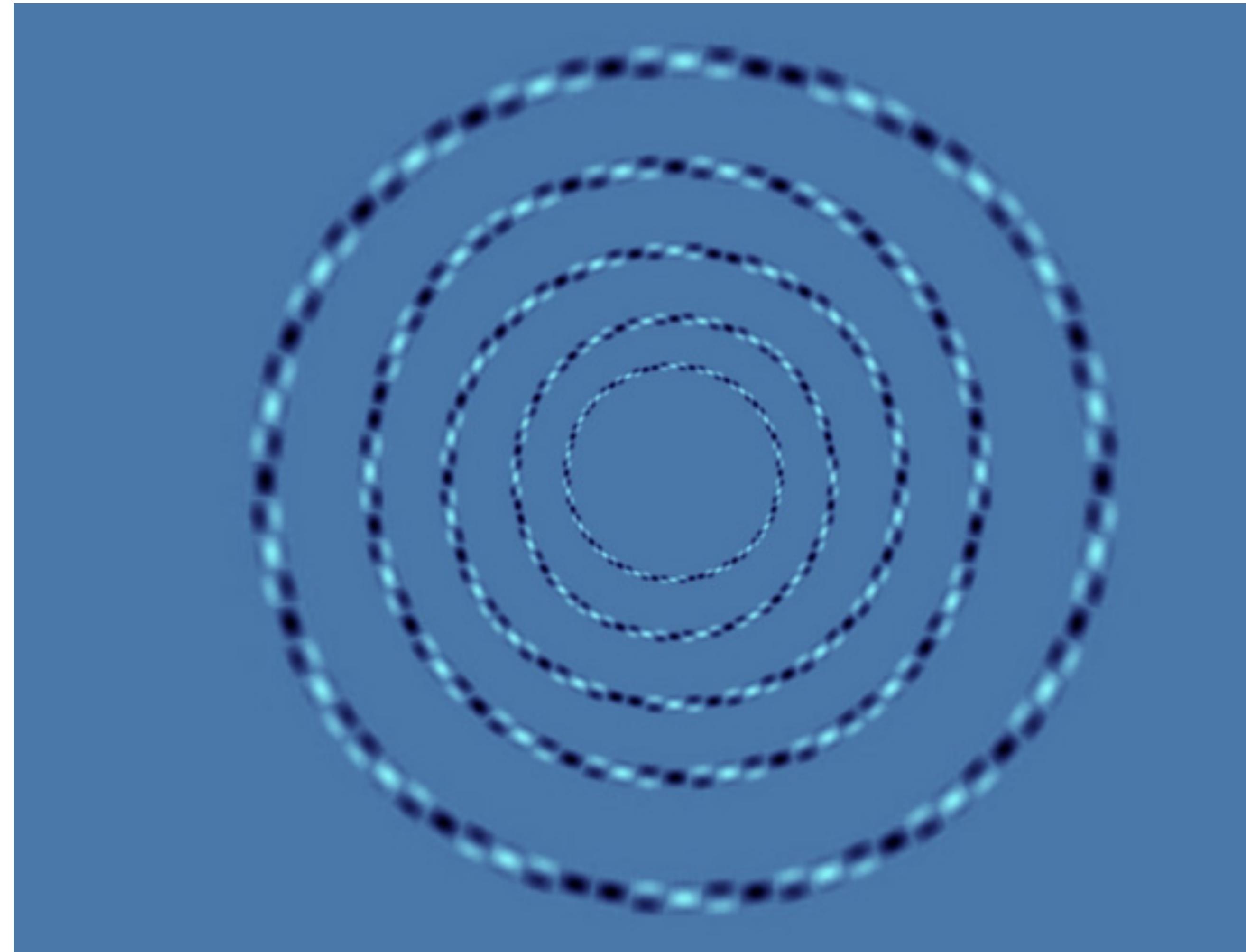
Self-introductions

1. My **name** is *Donald Duck*
2. My **position** is as a *10th year postdoc*
3. My **lab** is *Mad City Labs*
4. My model **system** is *blue whales*
5. I have **data** from *confocal microscopy*
6. A fun **fact** about me is *I pipette at the speed of light*

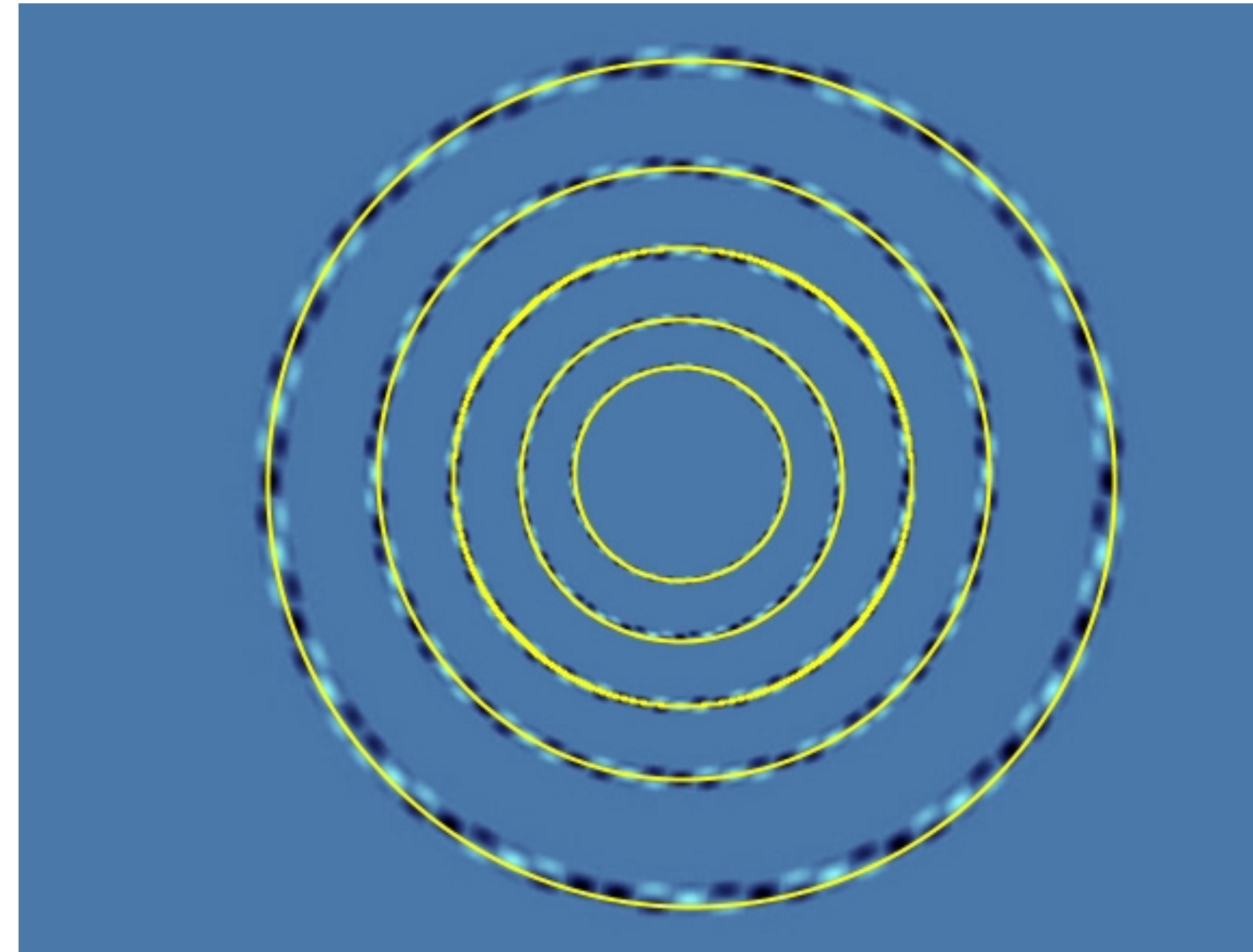
Optical Illusions

Why should you analyze images
with computers

**Color perception and pattern recognition is individual –
science less so**

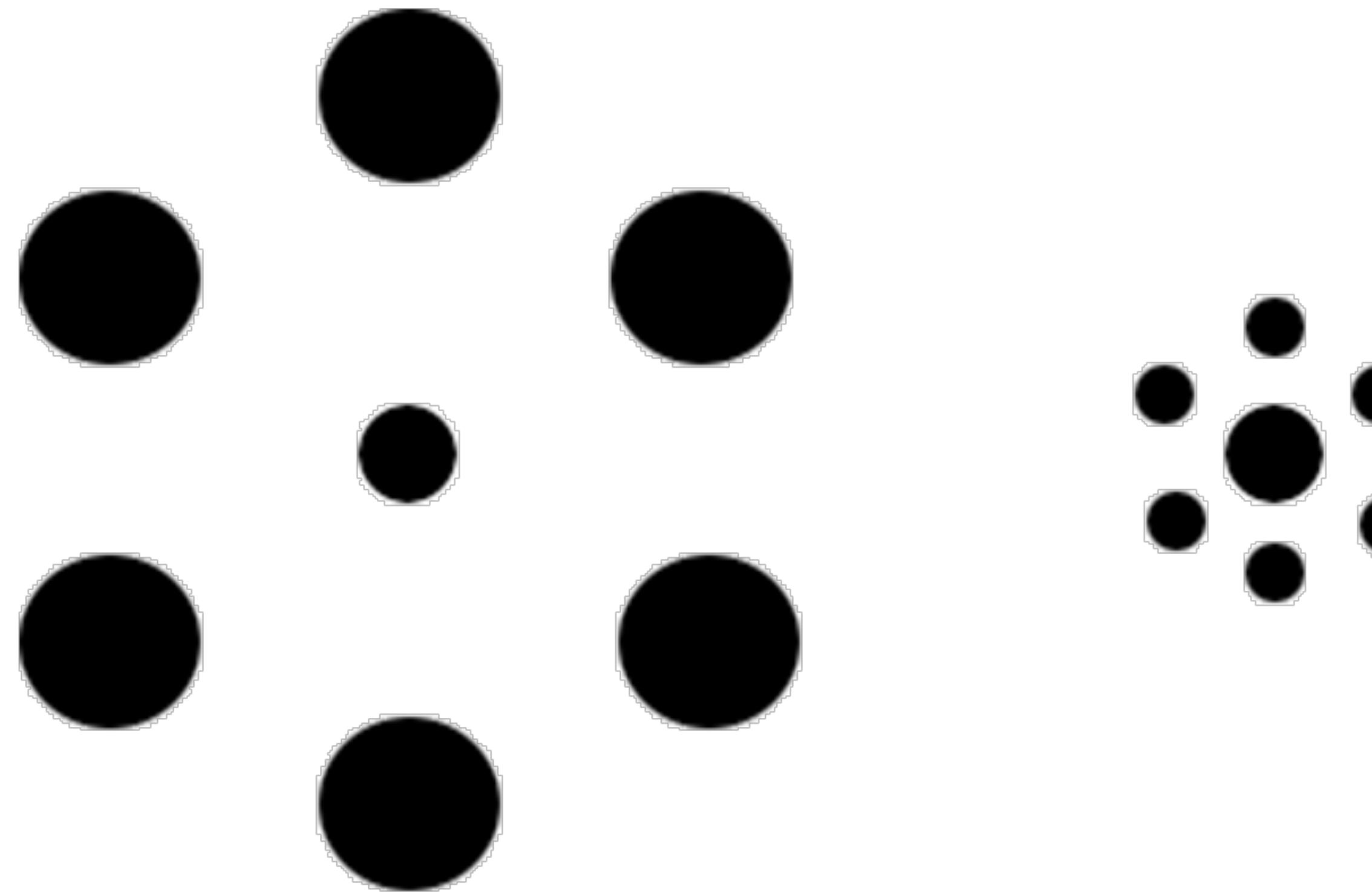


Concentric circles !?



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

Identical central discs?



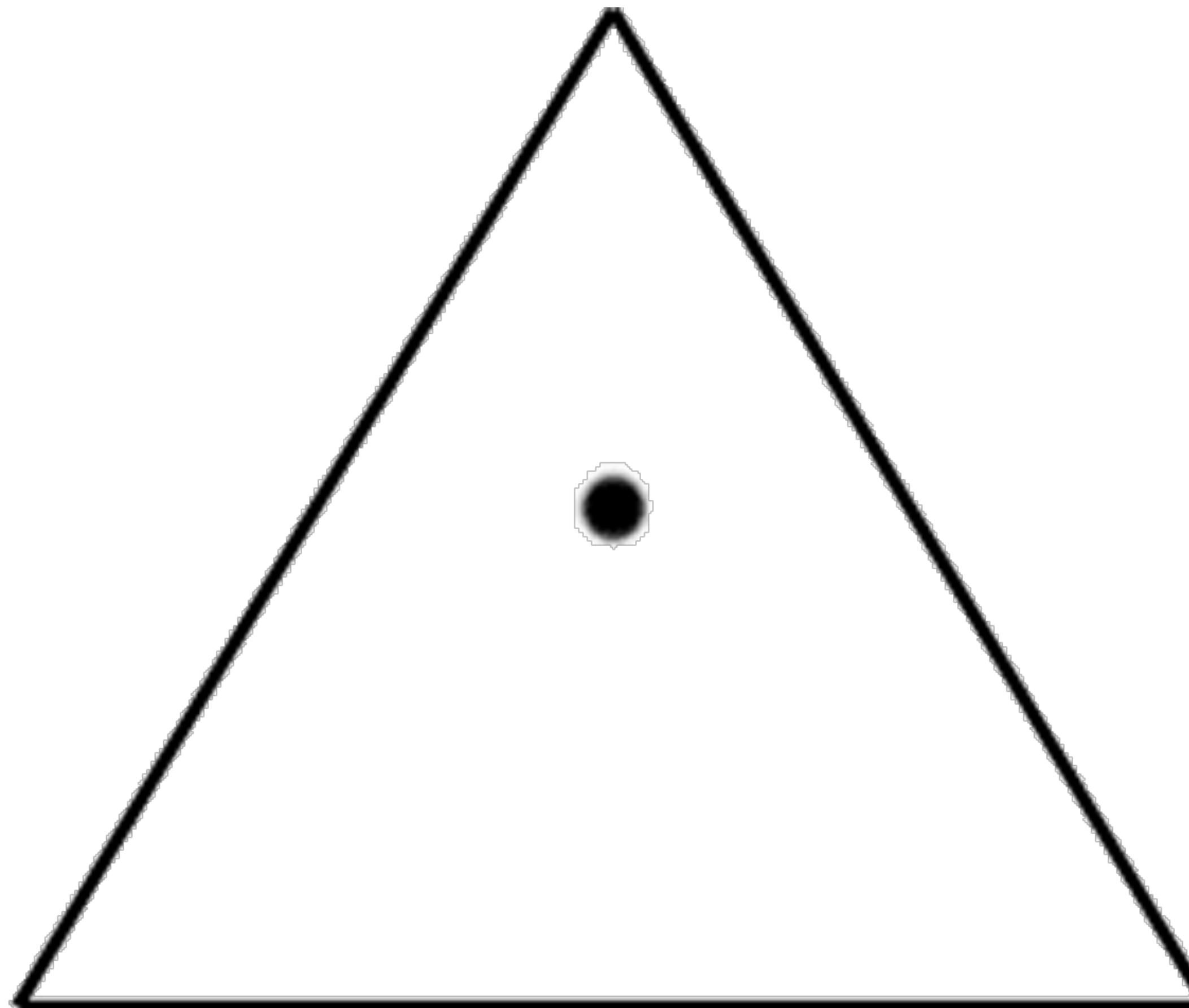
Our size estimate is strongly influenced by the local neighbourhood

Identical central discs?



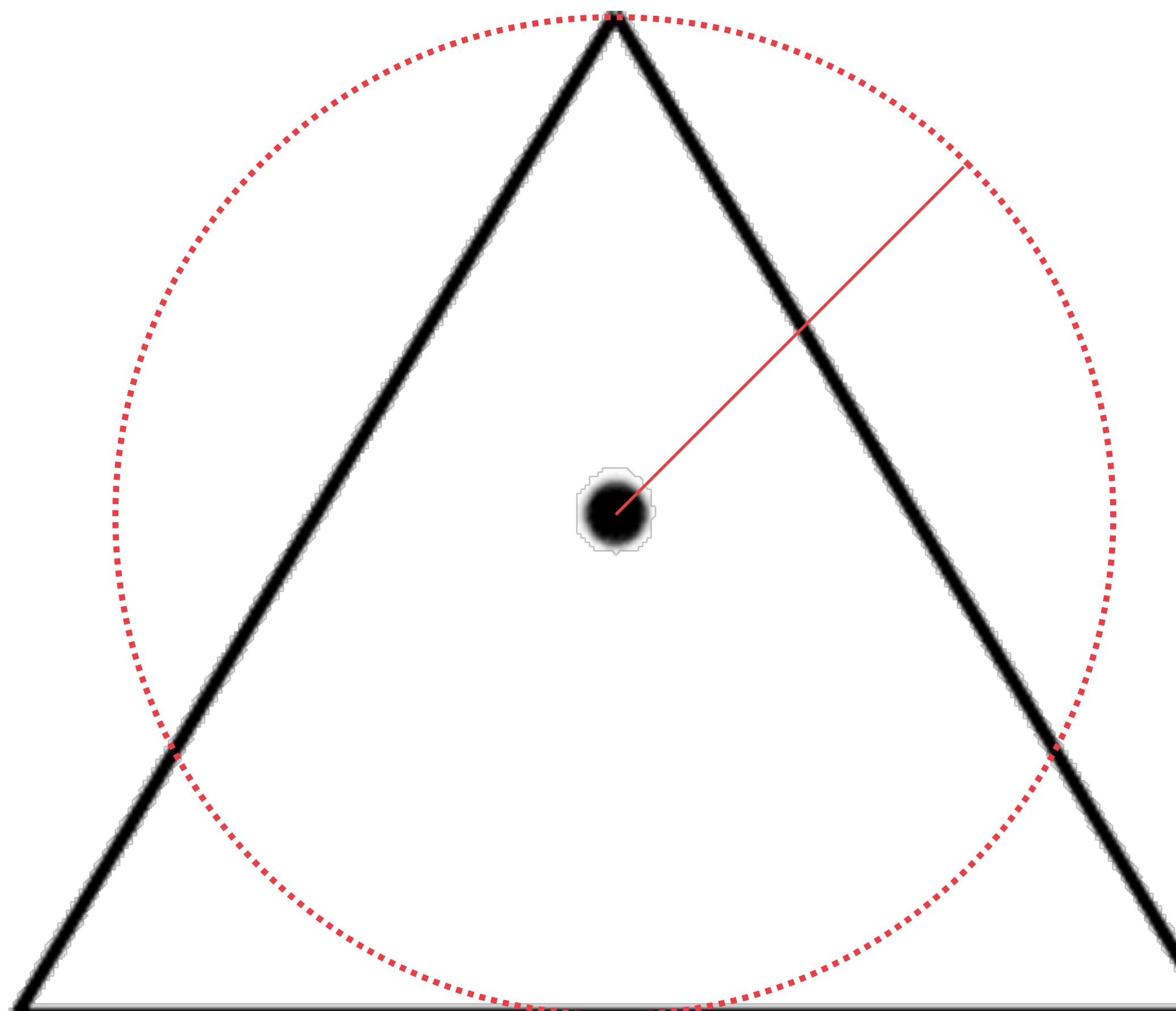
Yes, the discs are identical

Is the dot half-way up?



Our sense of distance depends on neighbourhood

Is the dot half-way up?



Yes it is!

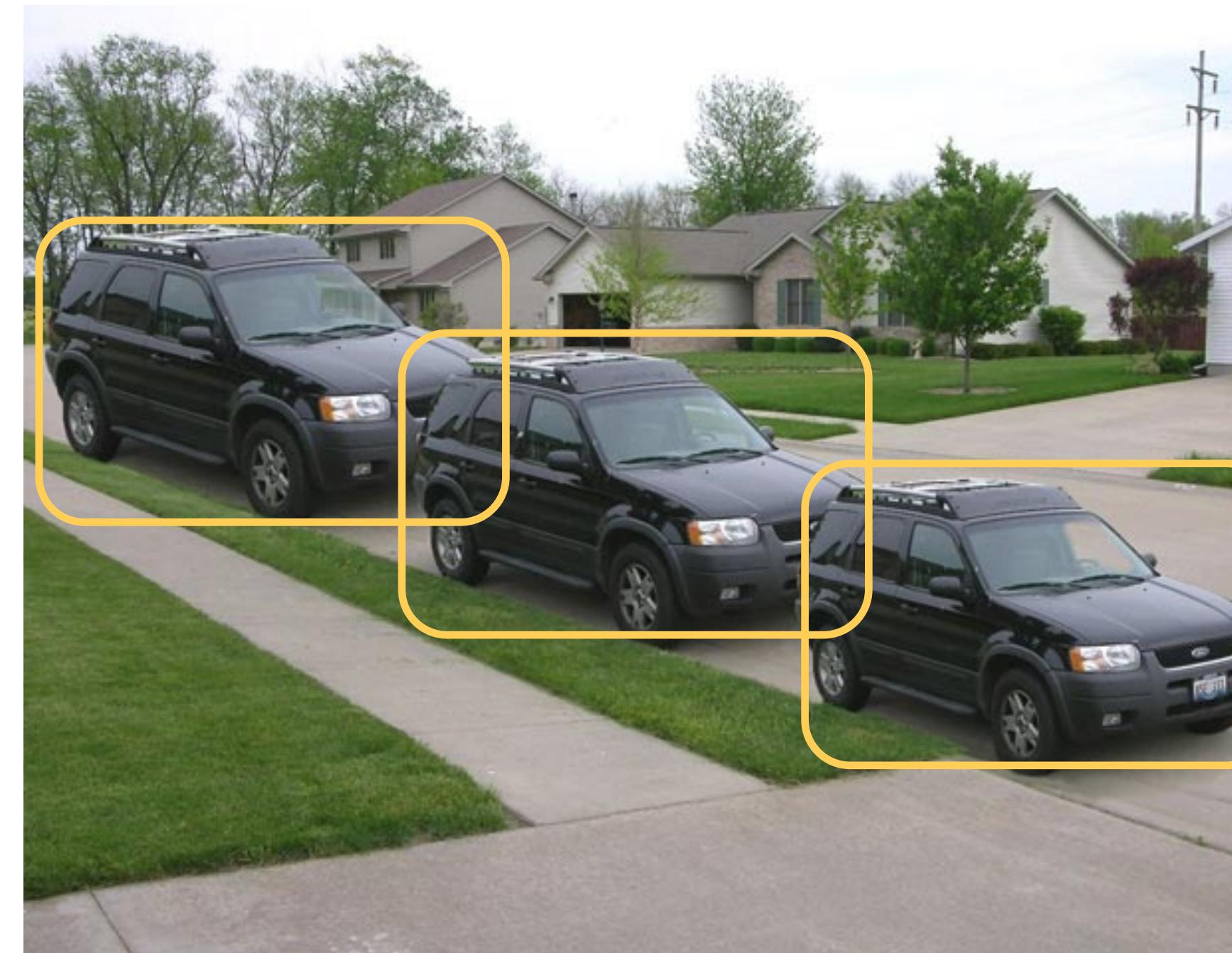
Which car is bigger?



Which car is bigger?

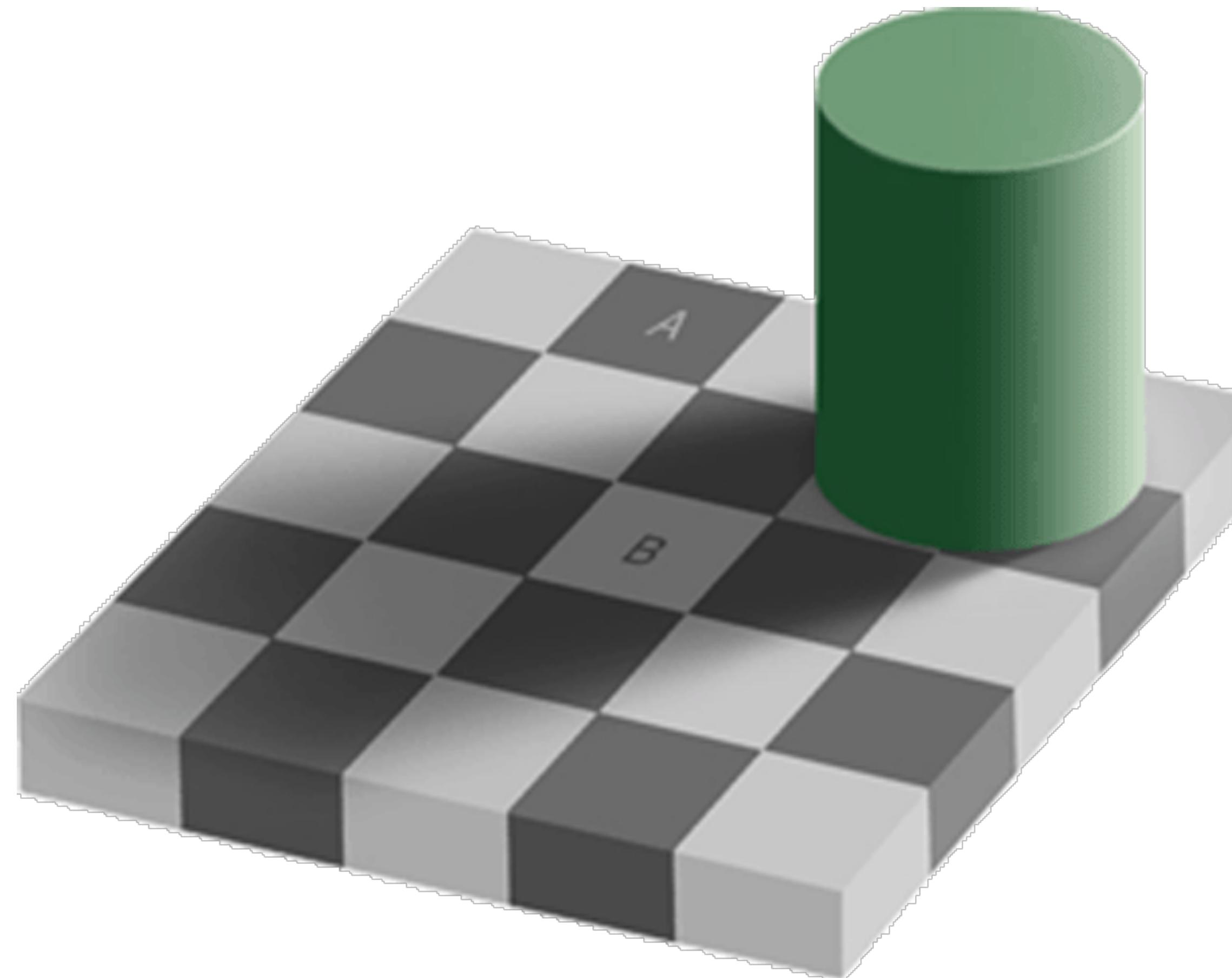


Which car is bigger?



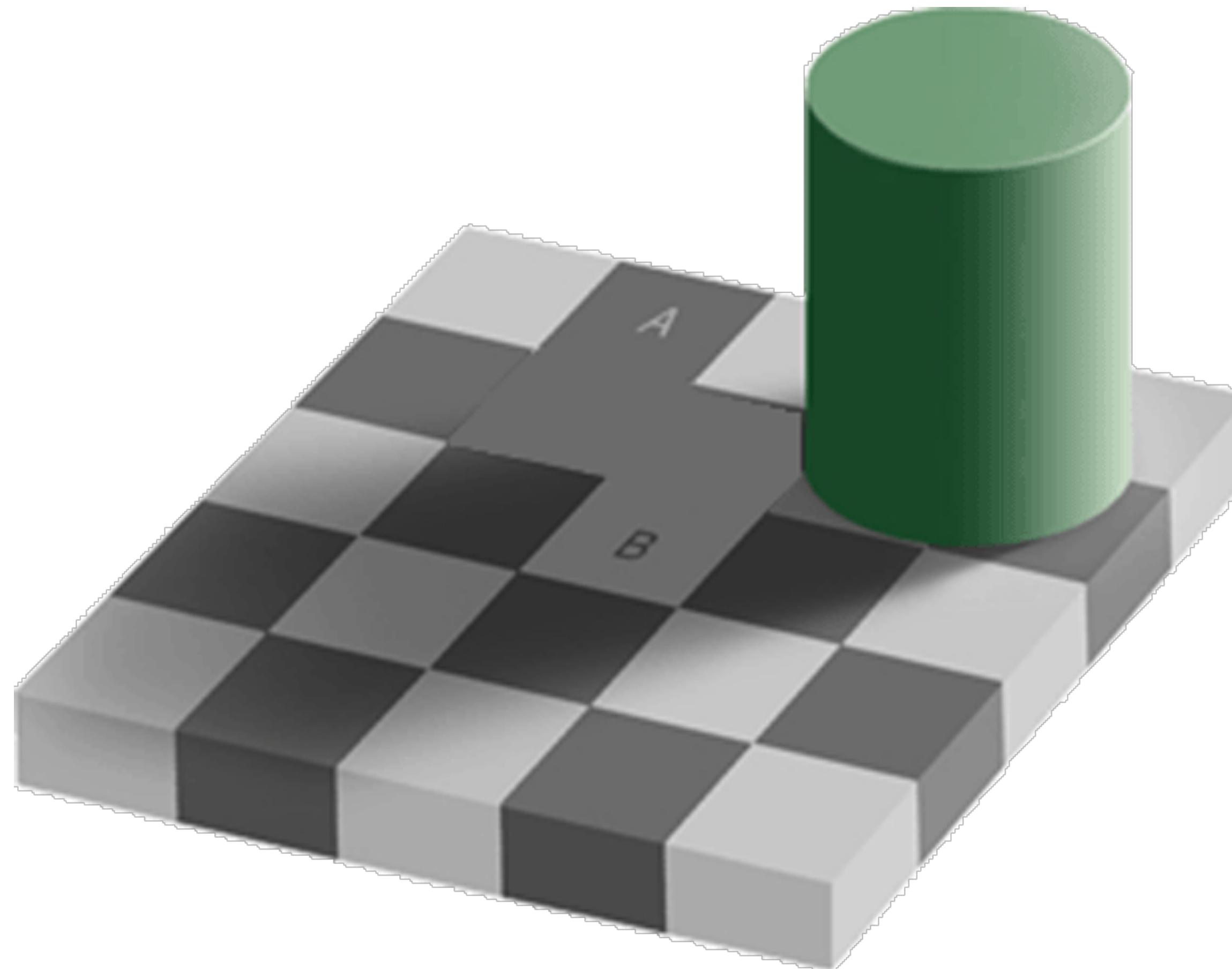
All cars are same size
(Erroneous, application of subjective/perceptual constancy algorithm by your brain)

Are A and B equally grey?



Intensity perception depends strongly on neighbourhood

Are A and B equally grey?



Yes they are!

Spot the animal

Why you should still use your brain
(a pre-trained neural network)

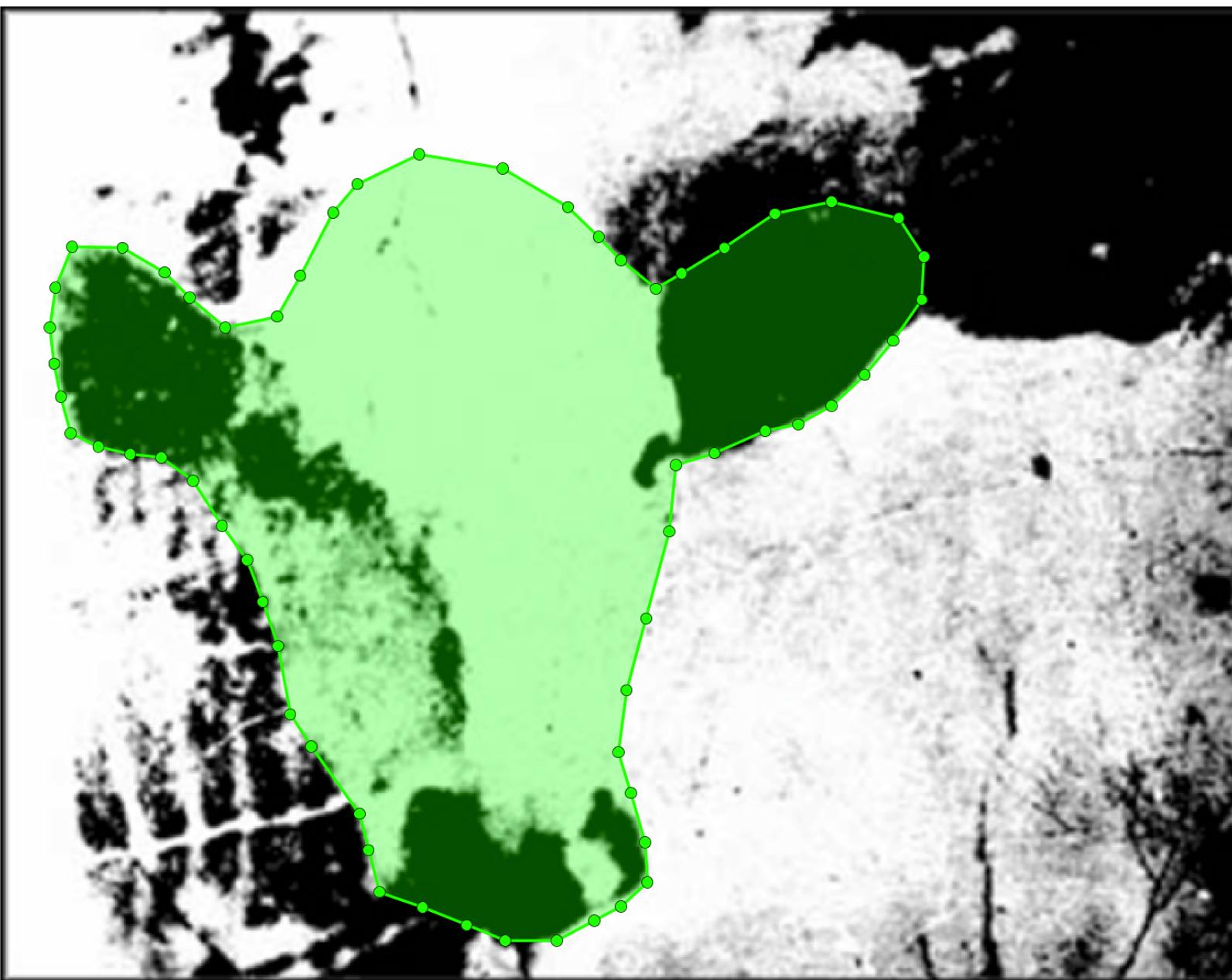
Pattern Recognition



Yes, that is a cow!

<http://www.brainbashers.com>

Pattern Recognition



Yes, that is a cow!

<http://www.brainbashers.com>

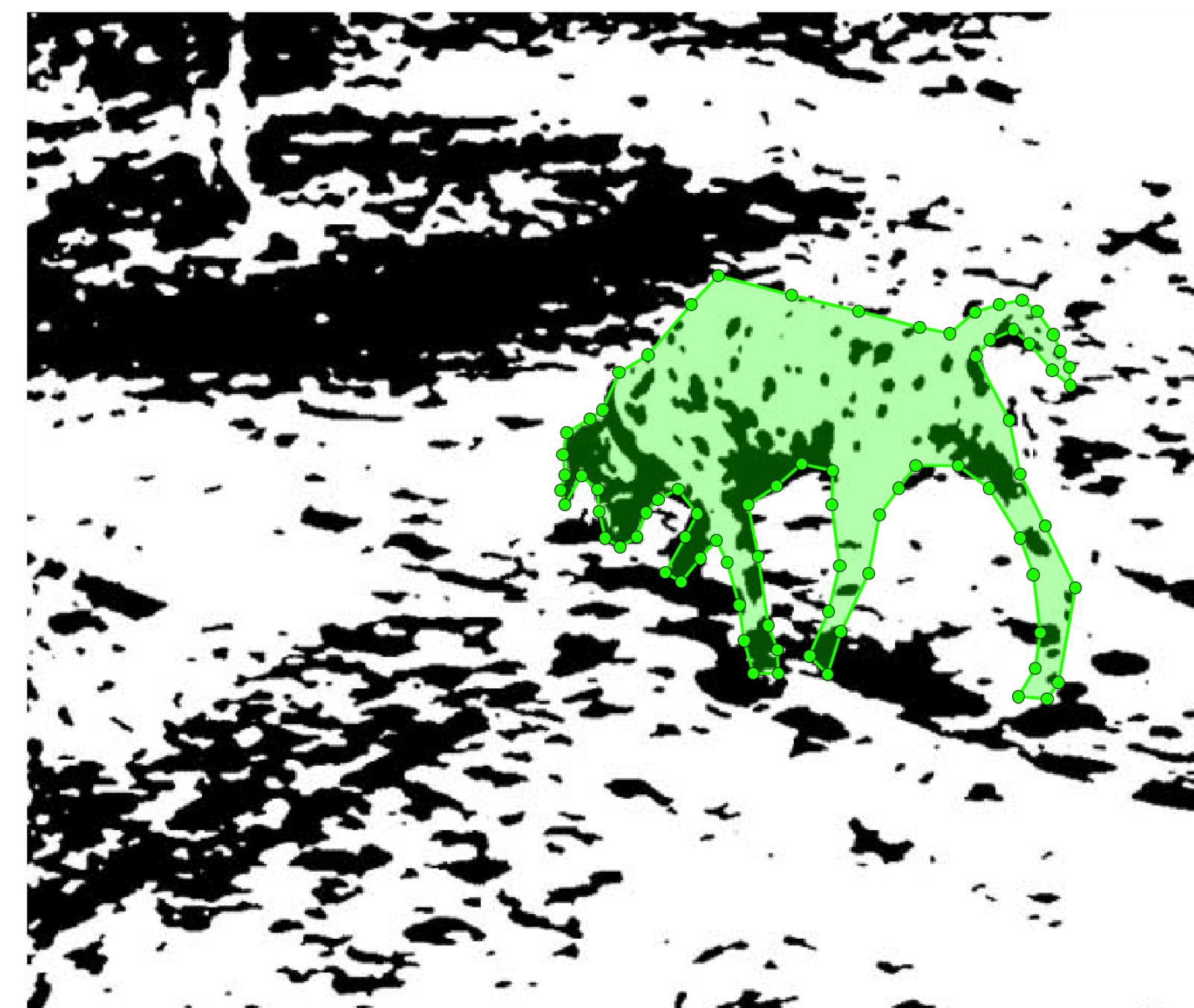
Using Prior Knowledge



Keyword: Dalmatian dog!

<http://www.brainbashers.com>

Using Prior Knowledge



Keyword: Dalmatian dog!

A brief history of image analysis

ImageJ centric

Bioimage analysis 37 years ago (1987)

- **NIH Image** is the only game in town (below \$150,000)
 - Written in Pascal for the Macintosh (Mac II) by Wayne S. Rasband
 - First released in 1987, shared on floppy disks
 - Supporting 8-bit 256 gray “colors”
 - If you wanted it, buy a Macintosh, or
- Windows supported through buggy closed-source C port *Scion Image*
 - Scion built frame-grabbers and wanted to expand to Windows
- Self-driving off-road in day and night time conditions demonstrated
- 1988: Yann LeCun publishes convolutional neural network [paper](#)
 - This is a technology, not a tool, few people notice, paper is fun to read today
 - Middle of second [AI winter](#) (1987-1993)



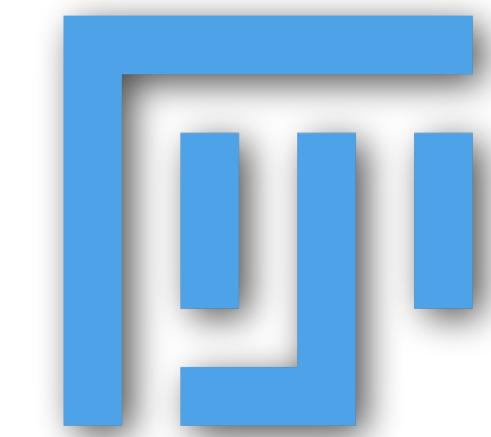
Bioimage analysis 27 years ago (1997)

- ***ImageJ*** (J for Java)---still the main game in town
 - Released in 1997 as rewrite of *NIH Image*
 - Apple in decline, the PC on the rise
 - 1995: Sun Microsystems created the Java programming language as an operating system–agnostic programming platform
 - Rasband wanted to support Windows, but not two sets of code
 - “*Instead of ‘write once, run everywhere’, Rasband found himself writing once and debugging everywhere.*”
- First autonomous coast-to-coast drive of the United States two years earlier



Bioimage analysis 12 years ago (2012)

- [**AlexNet**](#) is published (2012)
 - Wins the *ImageNet Large Scale Visual Recognition Challenge*
 - Error of 15.3%, more than 10.8 percentage points better than #2
 - Possible due to GPUs
- [**CellProfiler**](#) has been around for 6 years (2006)
- [**Fiji**](#) (Fiji is just ImageJ) has been around for 5 years (2007)
- [**scikit-image**](#) released 3 years ago (2009)
- Bitcoin grows from \$5 to \$13 (remember Silk Road?) (2012)
- U-Net, GANs, and [**Jupyter**](#) will appear in 2-3 years (2014/15)
- [**AlphaGo**](#) will beat Lee Sodol in 4 years (2016)
- [**QuPath**](#) is still 4 years in the future (2016)
- [**Vision Transformers**](#) are still 8 years in the future (2020)



Bioimage analysis now (2024)

- Increased integration of DL technologies into existing bioimage tools
 - CARE, Stardist, CellPose, [DeepCell](#) (MESMER) as plugins, modules, notebooks for
 - Fiji, CellProfiler, [Napari](#), ZeroCostDL4Mic, etc
- Most major Microscopy producers offer AI
 - Implemented in-house or through acquisition
 - Zeiss: Arivis (bought)
 - Leica: Aivia (bought)
 - Nikon: NIS.ai suite (in-house implementation of open-source technologies)
- “Smart microscopy” includes DL for event detection
 - <https://www.nature.com/articles/s41592-022-01589-x>
- Fully autonomous cars still in the future
 - Image analysis is hard if you don't understand what you are looking at



GPT-3.5

GPT-4

ChatGPT

Create a workout plan
for resistance training

Design a database schema
for an online merch store

Explain why popcorn pops
to a kid who loves watching it in the microwave

Write an email
to request a quote from local plumbers

Please write an ImageJ macro that uses StarDist to segment an image from fluorescence microscopy



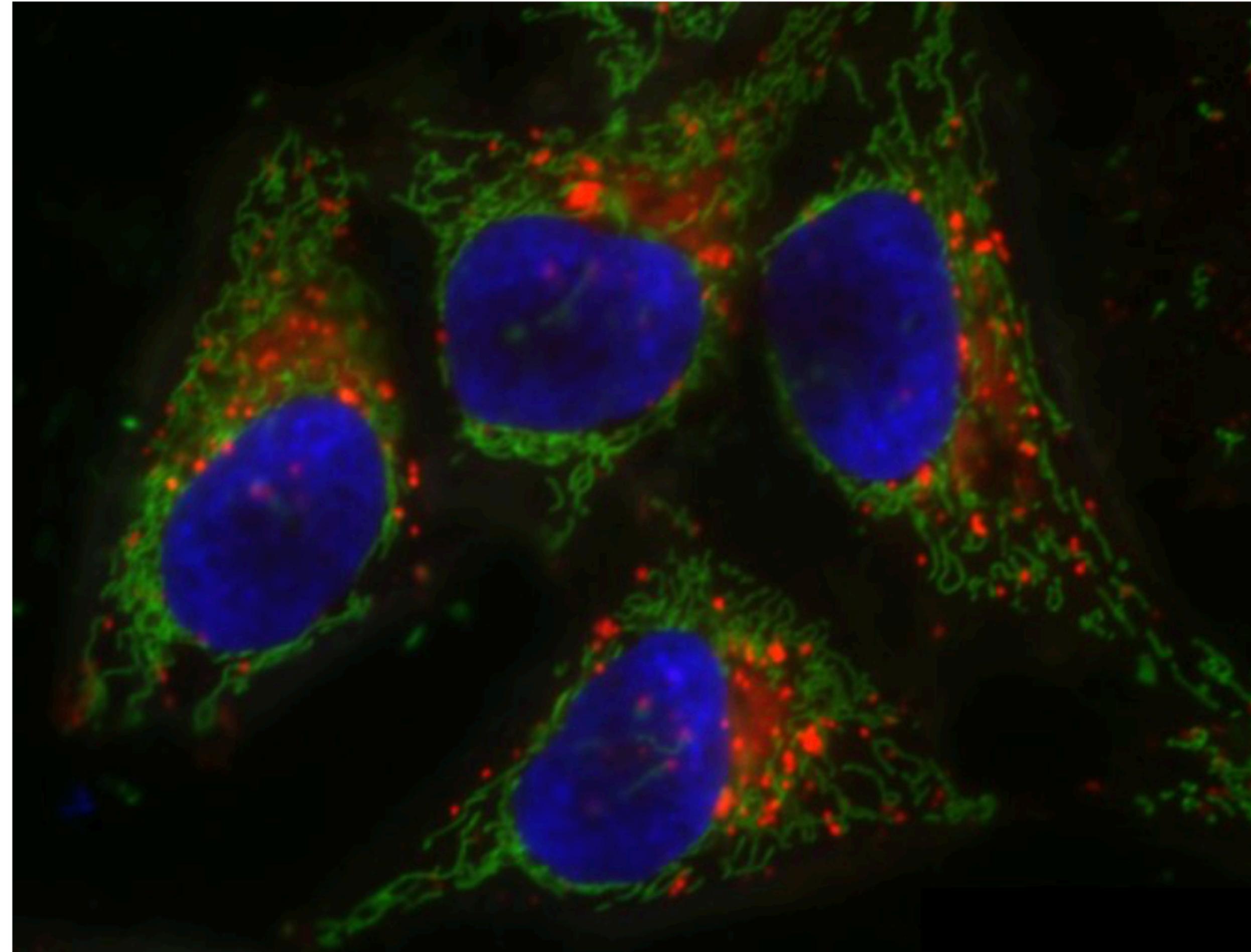
?

Free Research Preview. ChatGPT may produce inaccurate information about people, places, or facts. [ChatGPT September 25 Version](#)

Using ChatGPT for Image Analysis

SI

You

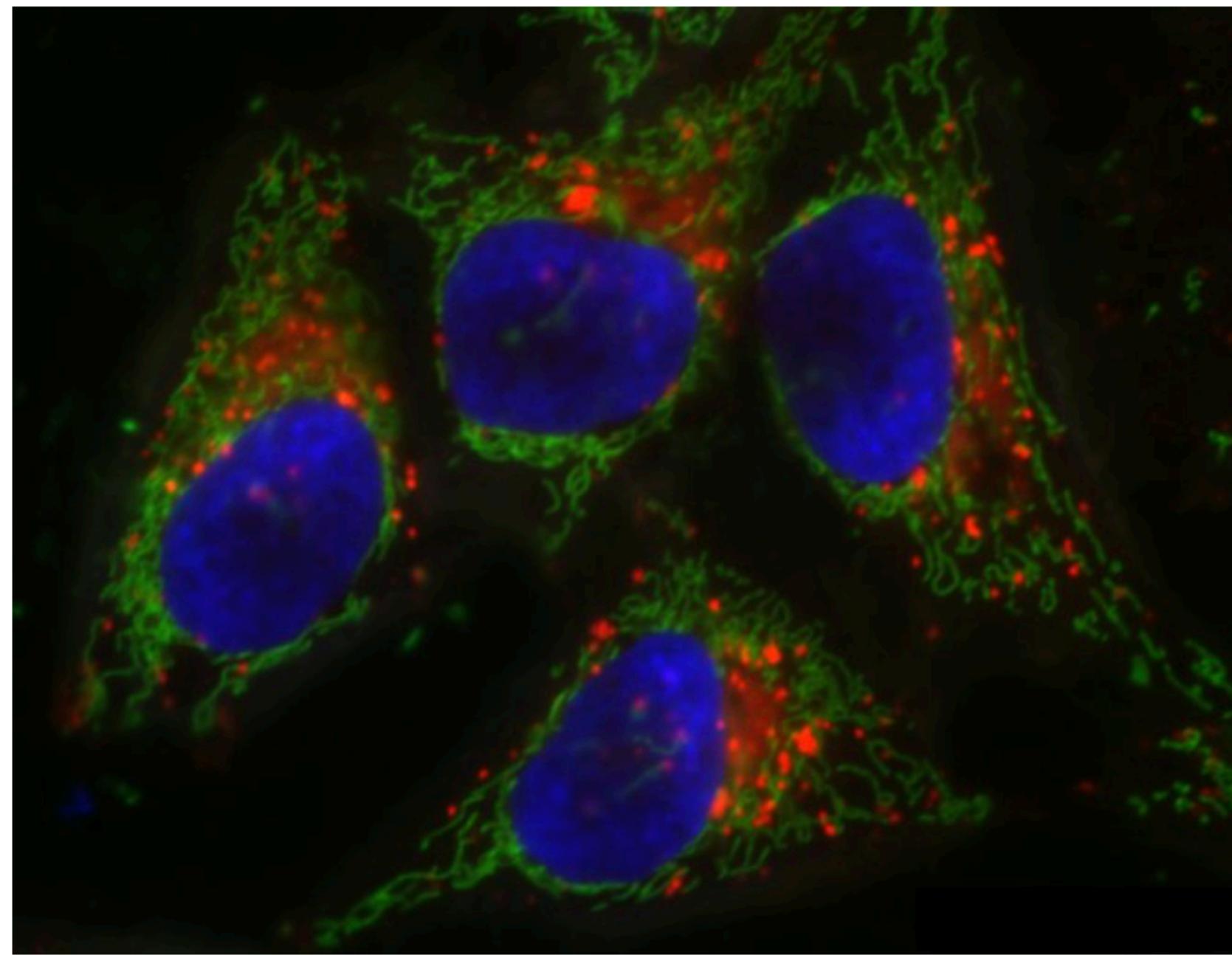


please segment the nuclei and show me the relevant code you used

Using ChatGPT for Image Analysis

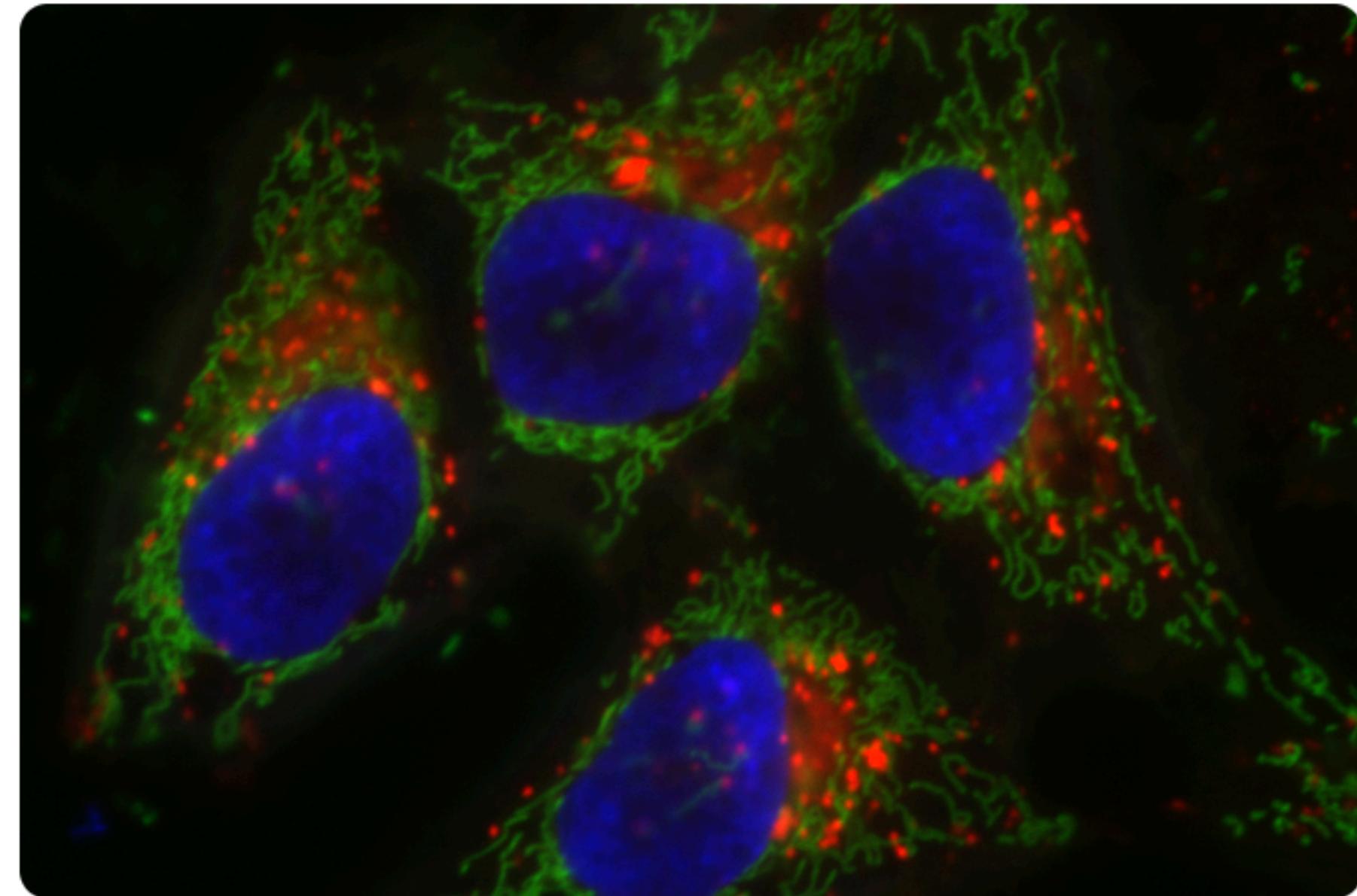
GPT-4, May 2024

You



please segment the nuclei and show me the relevant code you used

GPT-4o, Oct. 2024



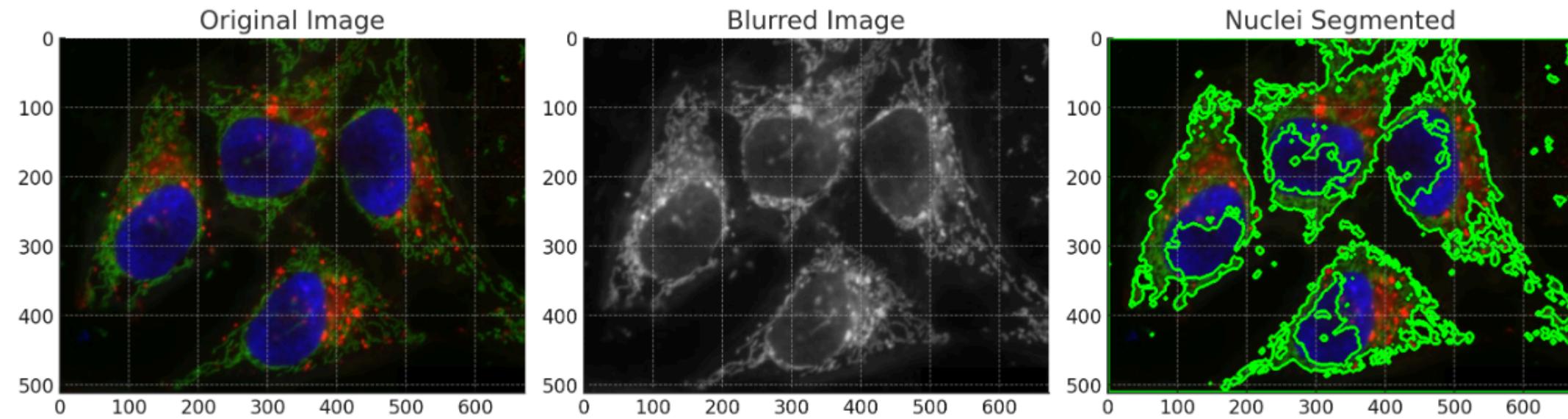
please segment the nuclei and show me the relevant code you used

Using ChatGPT for Image Analysis

GPT-4, May 2024



ChatGPT

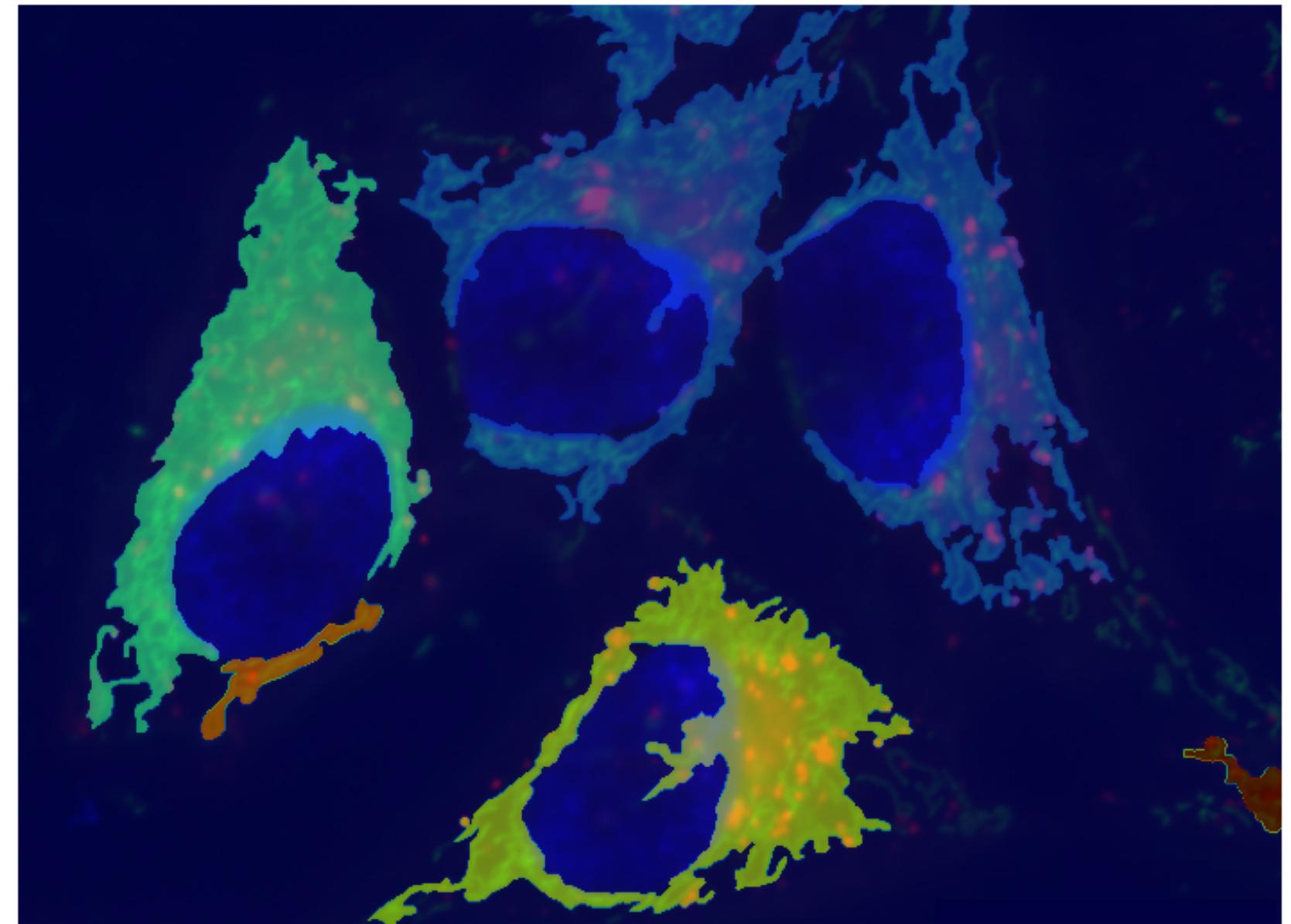


The nuclei in the image have been segmented using the following steps:

- The image was converted to grayscale to simplify the analysis.
- Gaussian Blur was applied to reduce noise and detail in the image, which aids in thresholding and contour detection.
- Otsu's thresholding was performed to convert the grayscale image into a binary image, which helps in distinguishing the nuclei from the background.
- Contours were found in the binary image, which correspond to the boundaries of the nuclei.
- These contours were then drawn onto the original image to visually segment the nuclei.

GPT-4o, Oct. 2024

Segmented Nuclei



HARVARD
MEDICAL SCHOOL

Using ChatGPT for Image Analysis

GPT-4, May 2024

```
python
Copy code

import cv2
import numpy as np
from matplotlib import pyplot as plt

# Load the image
image_path = '/mnt/data/hela-cells.jpg'
image = cv2.imread(image_path)

# Convert the image to gray scale
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

# Apply GaussianBlur, which is good for edge detection
blurred = cv2.GaussianBlur(gray, (5, 5), 0)

# Perform Otsu's thresholding
_, thresh = cv2.threshold(blurred, 0, 255, cv2.THRESH_BINARY_INV + cv2.THRESH_OTSU)

# Find contours which will give us the nuclei boundaries
contours, _ = cv2.findContours(thresh, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)

# Draw the contours on the original image
contoured_image = image.copy()
cv2.drawContours(contoured_image, contours, -1, (0, 255, 0), 3)

# Save the output image with segmented nuclei
output_path = '/mnt/data/hela-cells-segmented.jpg'
cv2.imwrite(output_path, contoured_image)
```



GPT-4o, Oct. 2024

```
python
Copy code

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

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

# Convert to grayscale
gray_image = color.rgb2gray(image)

# Apply Gaussian filter to smooth the image
smoothed = filters.gaussian(gray_image, sigma=1)

# Use Otsu's threshold to create a binary mask of the nuclei
threshold_value = filters.threshold_otsu(smoothed)
binary_mask = smoothed > threshold_value

# Remove small objects and fill holes to improve segmentation
cleaned_mask = morphology.remove_small_objects(binary_mask, min_size=500)
cleaned_mask = morphology.remove_small_holes(cleaned_mask, area_threshold=500)

# Label the segmented regions
labeled_nuclei = measure.label(cleaned_mask)
```

Using ChatGPT for Image Analysis

GPT-4, May 2024



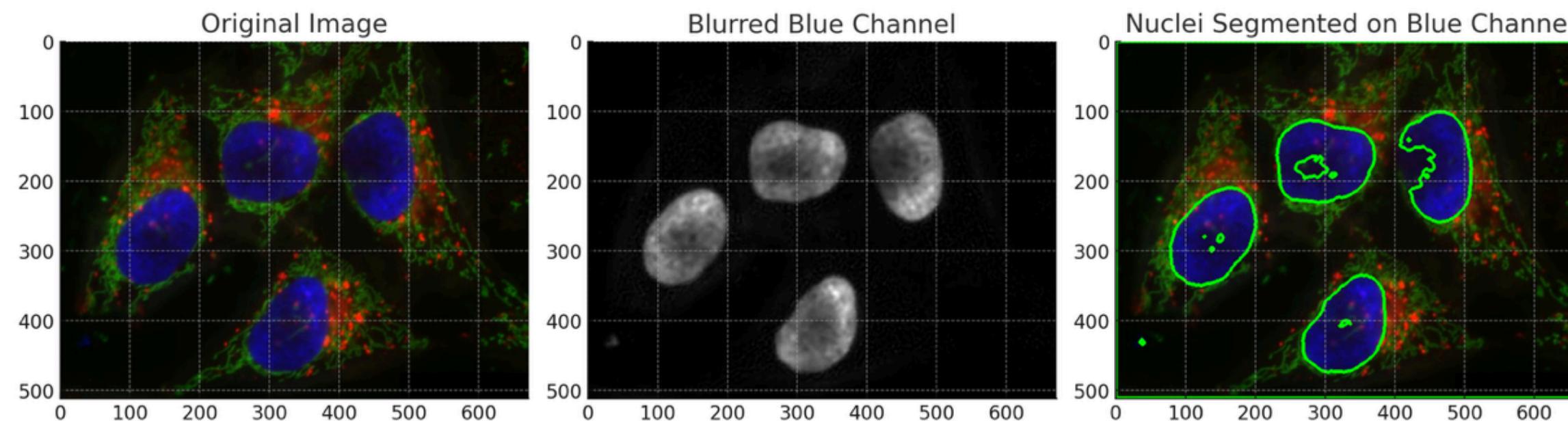
SI

You

please use the blue channel for segmenting the nuclei



ChatGPT

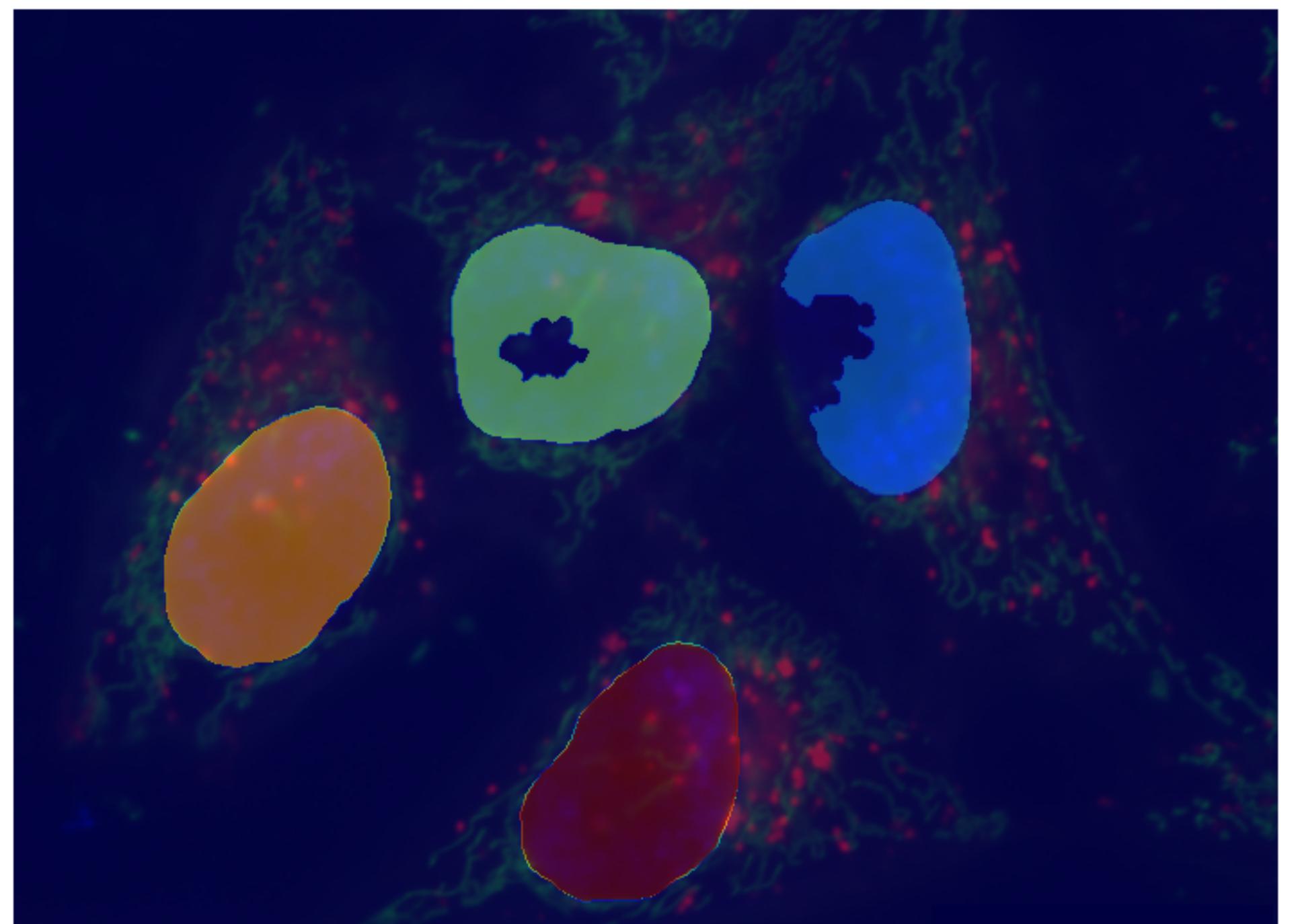


The nuclei segmentation has been performed using the blue channel of the image with the following steps:

- The blue channel was extracted from the original image since the nuclei appear to be stained blue.
- Gaussian Blur was applied to the blue channel to smooth the image, which helps in reducing noise for thresholding.
- Otsu's thresholding was then used on the blurred blue channel to create a binary image where the nuclei are distinguished from the background.

GPT-4o, Oct. 2024

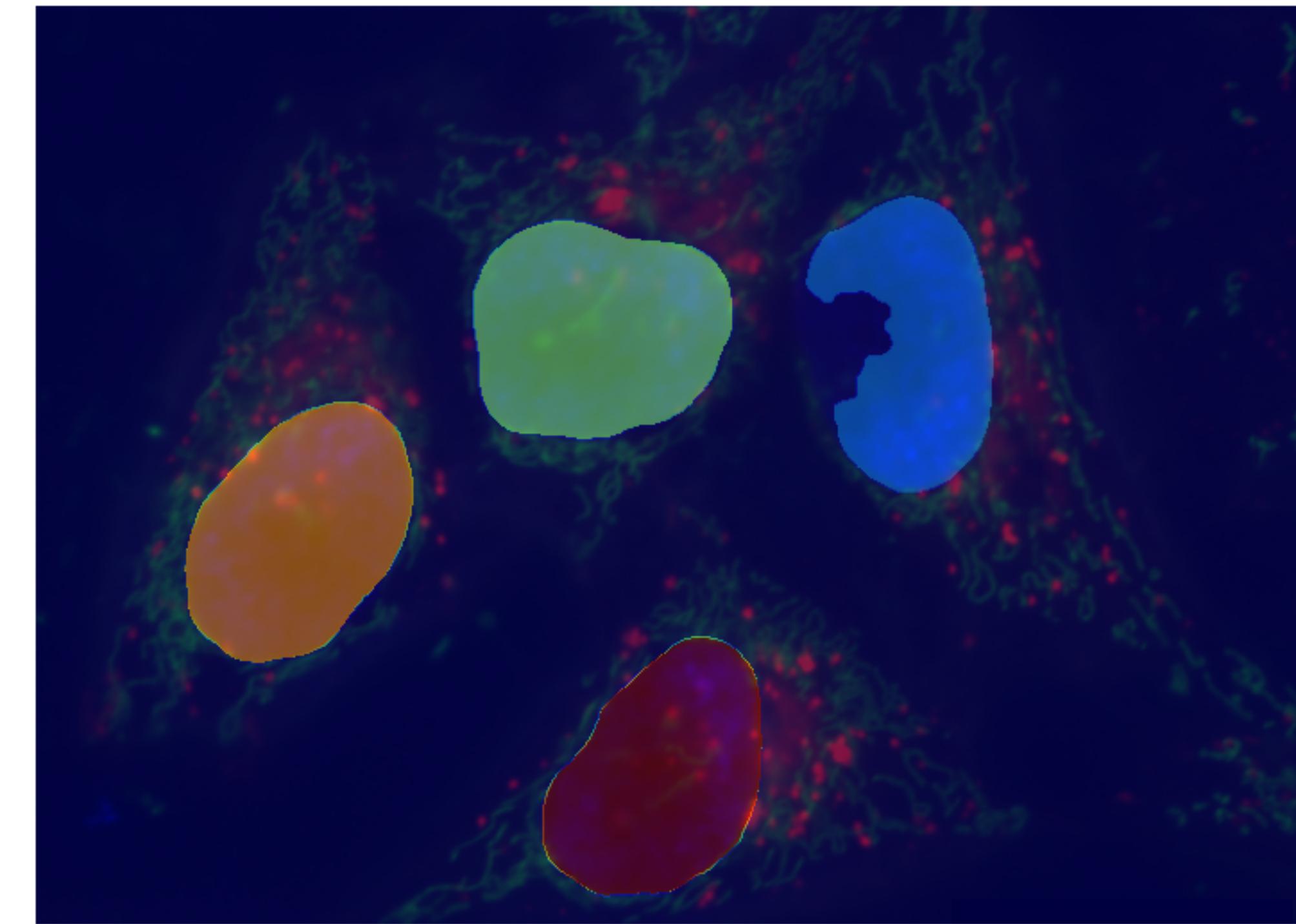
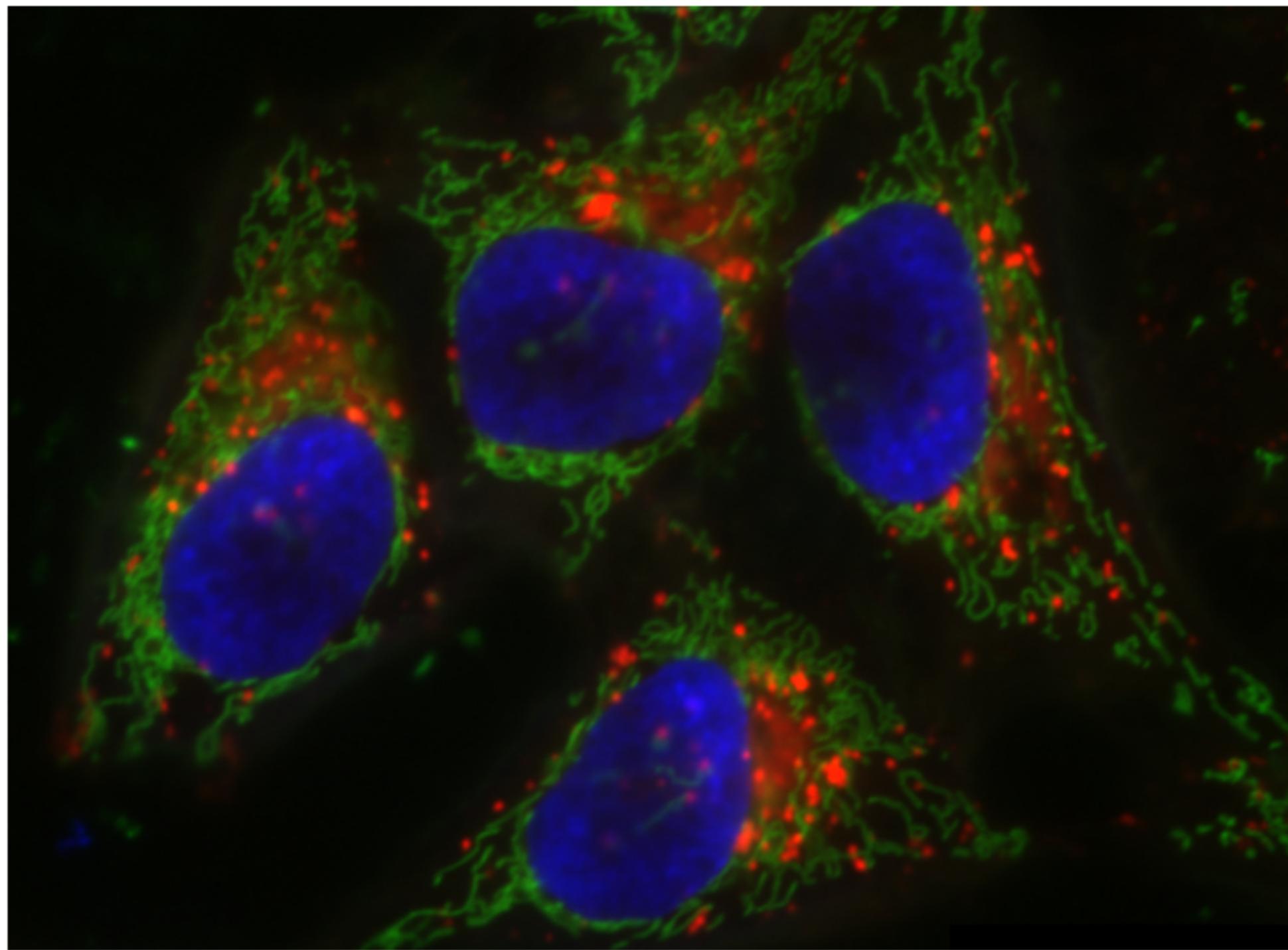
Segmented Nuclei (Blue Channel)



Using ChatGPT-4o for Image Analysis

GPT-4o, Oct. 2024 - prompted to close larger holes

make sure to close larger holes, and increase smoothing sigma to 3

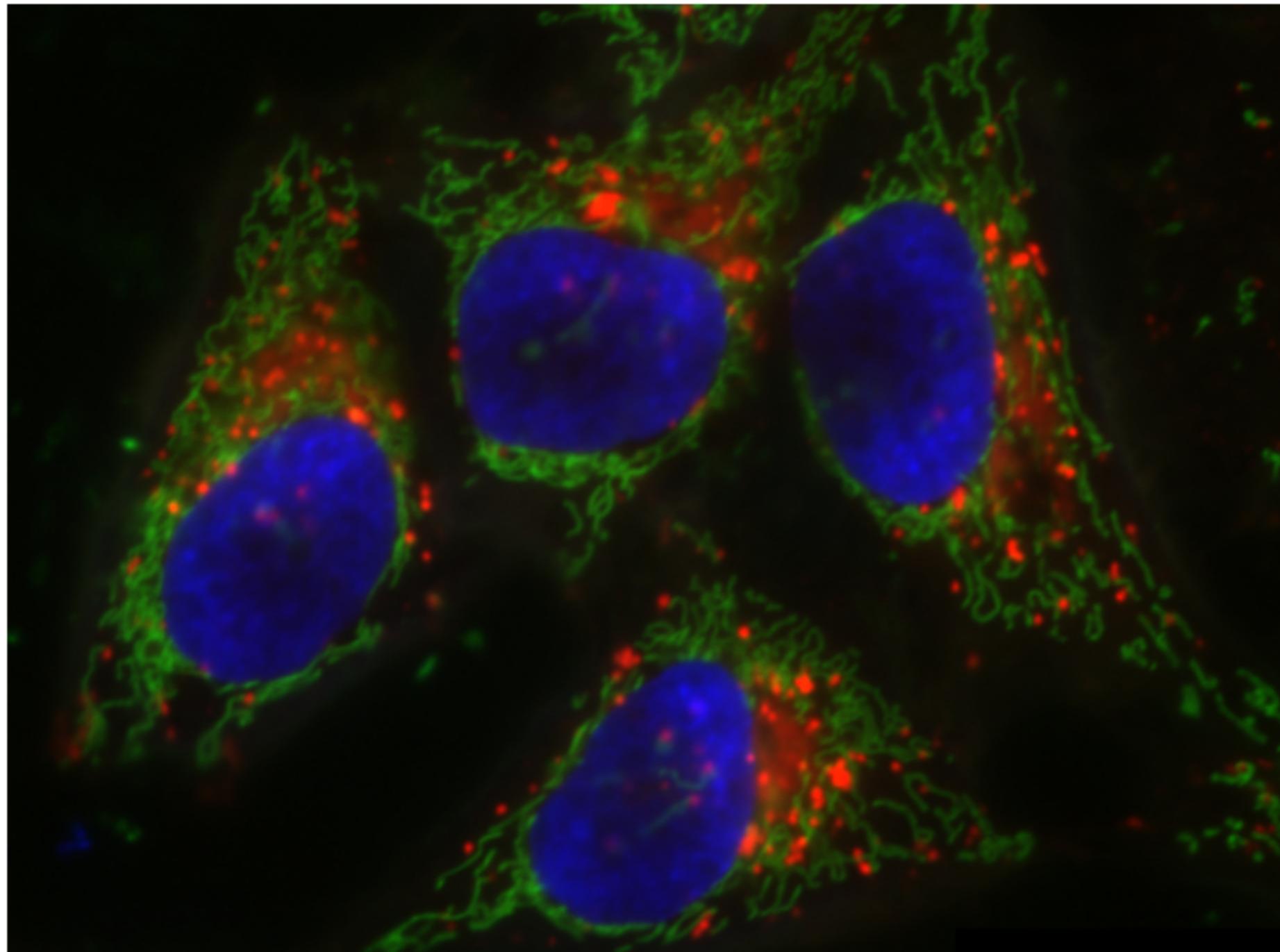


Using ChatGPT-4o for Image Analysis

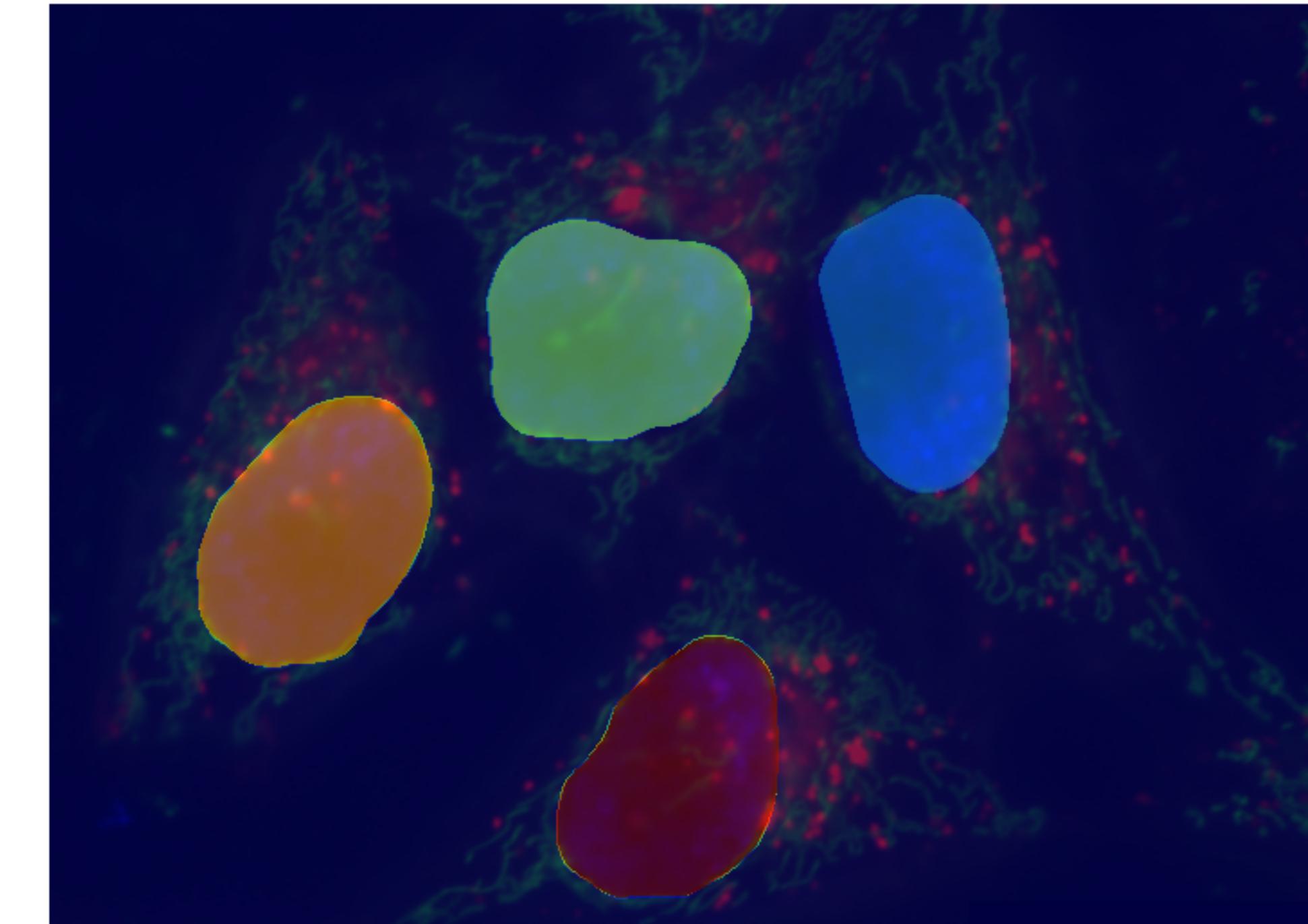
GPT-4o, Oct. 2024 - it helps to know what to ask

compute the convex hull for shapes with a low circularity, to
approximate the characteristic shape of a nuclei

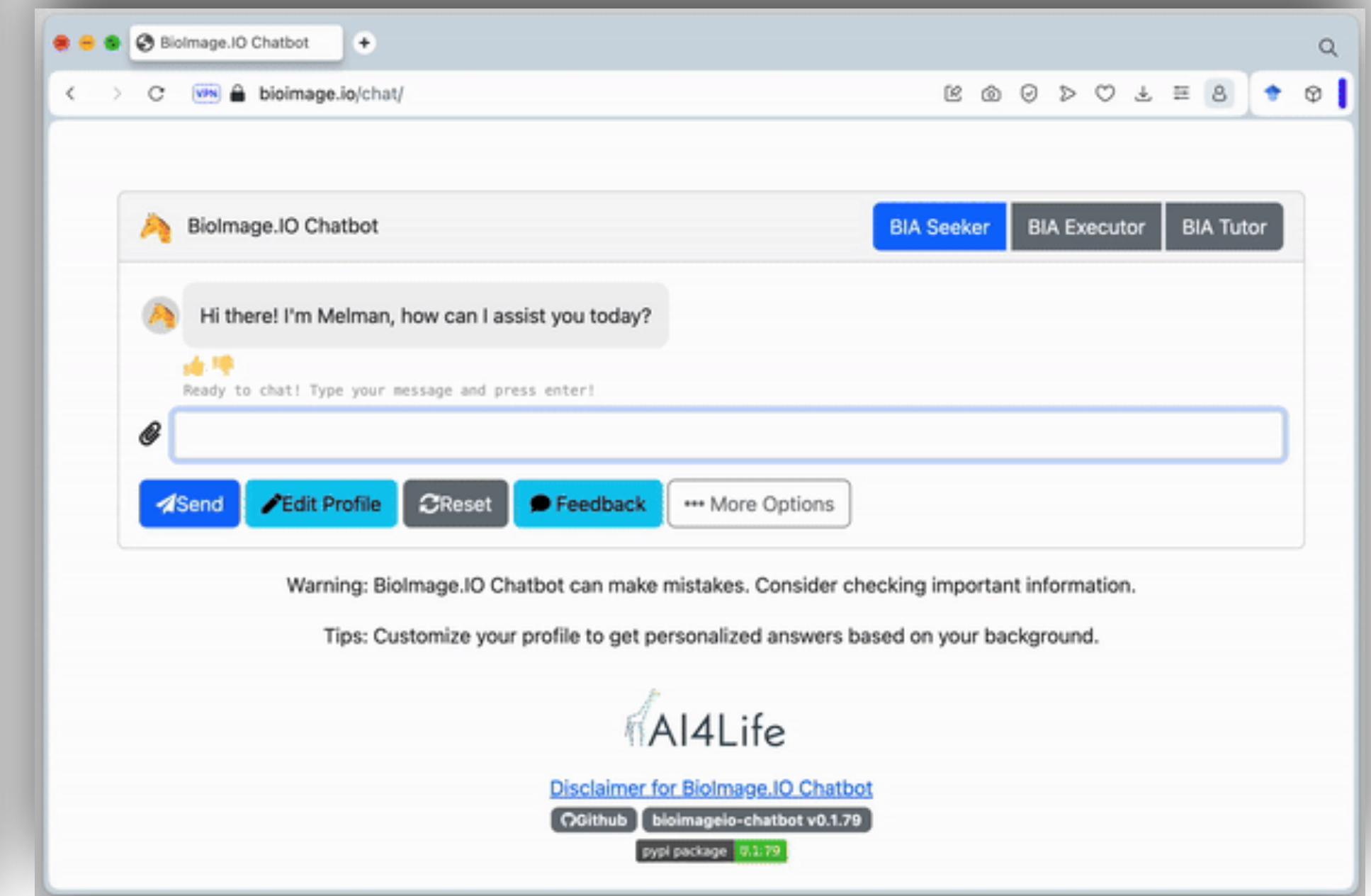
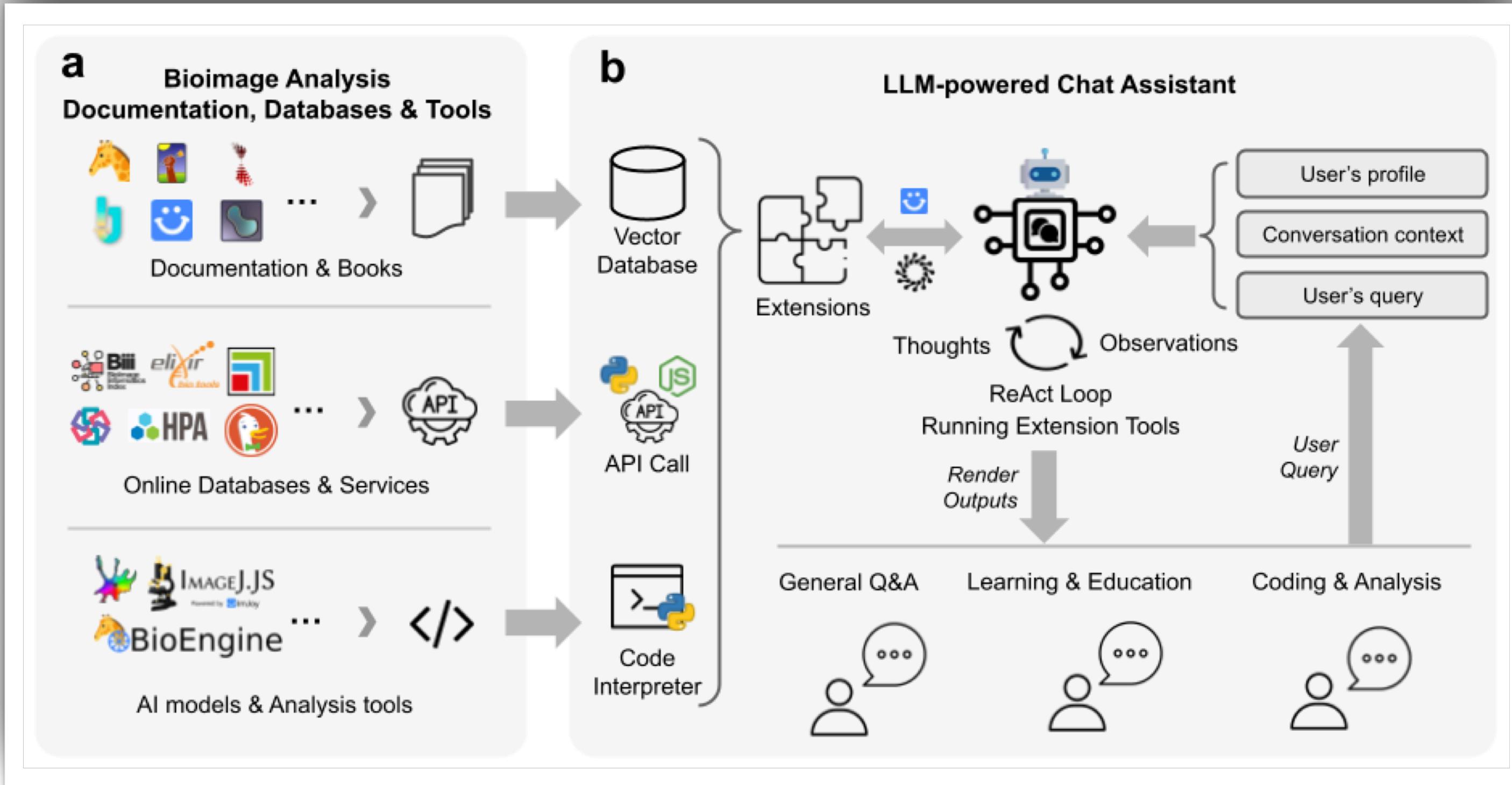
Original Image



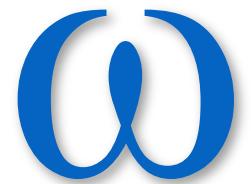
Convex Hull for Low Circular Nuclei



Biolmage Chatbot



napari-omega



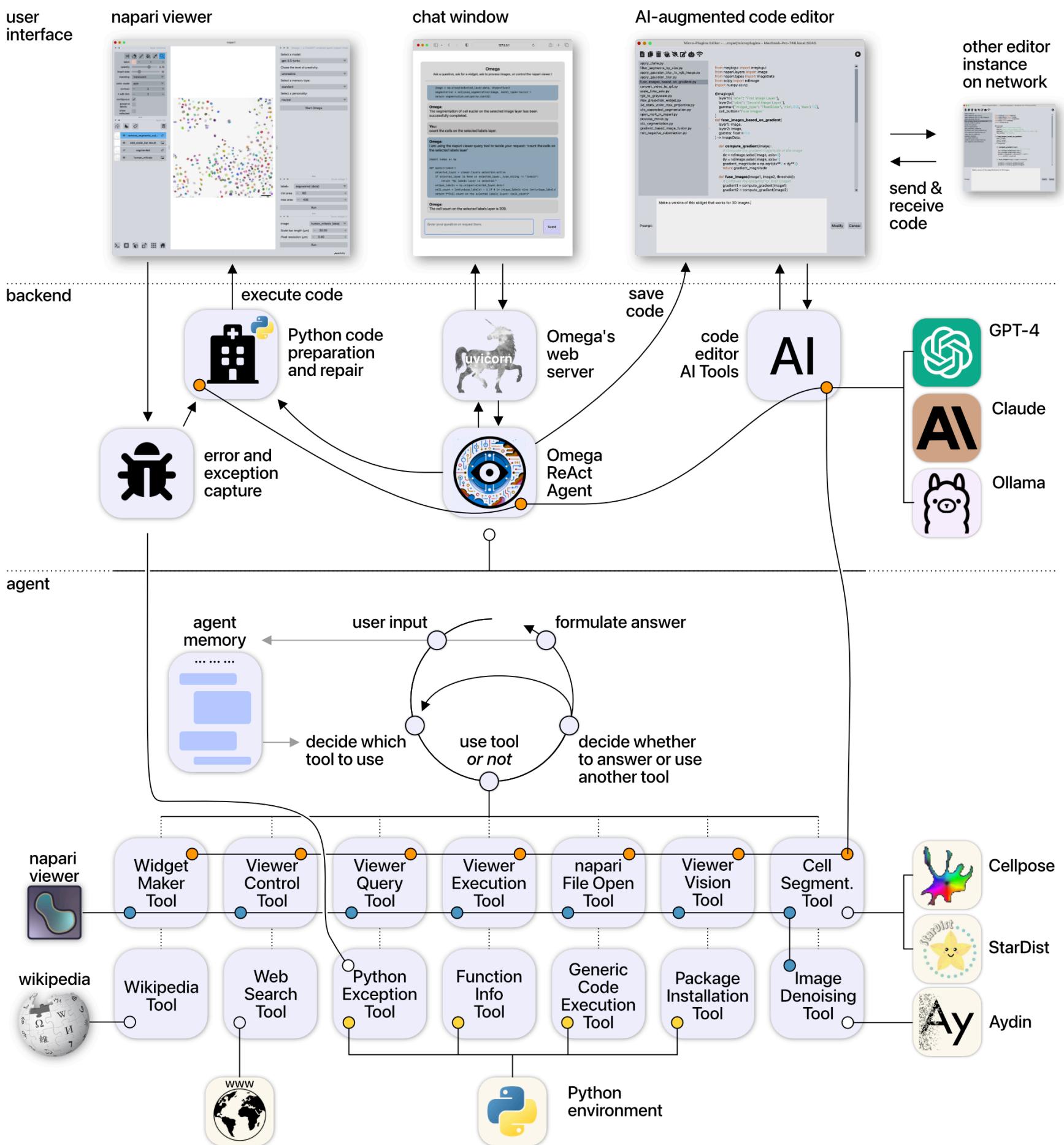
Omega – Harnessing for Bioimage Analysis

Loïc A. Royer^{1,*}

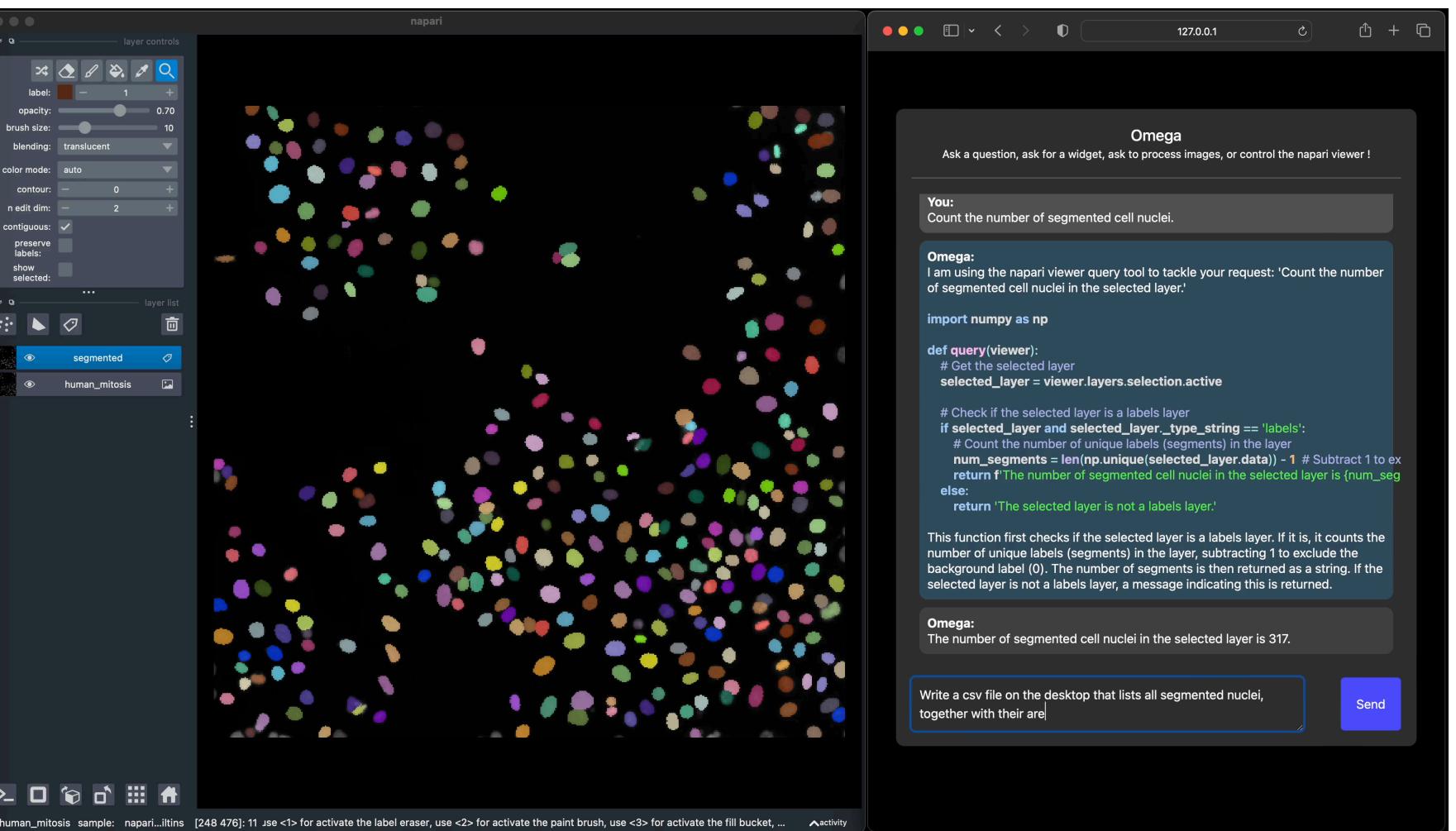
¹Chan Zuckerberg Biohub, San Francisco, USA.

*Correspondence: loic.royer@czbiohub.org

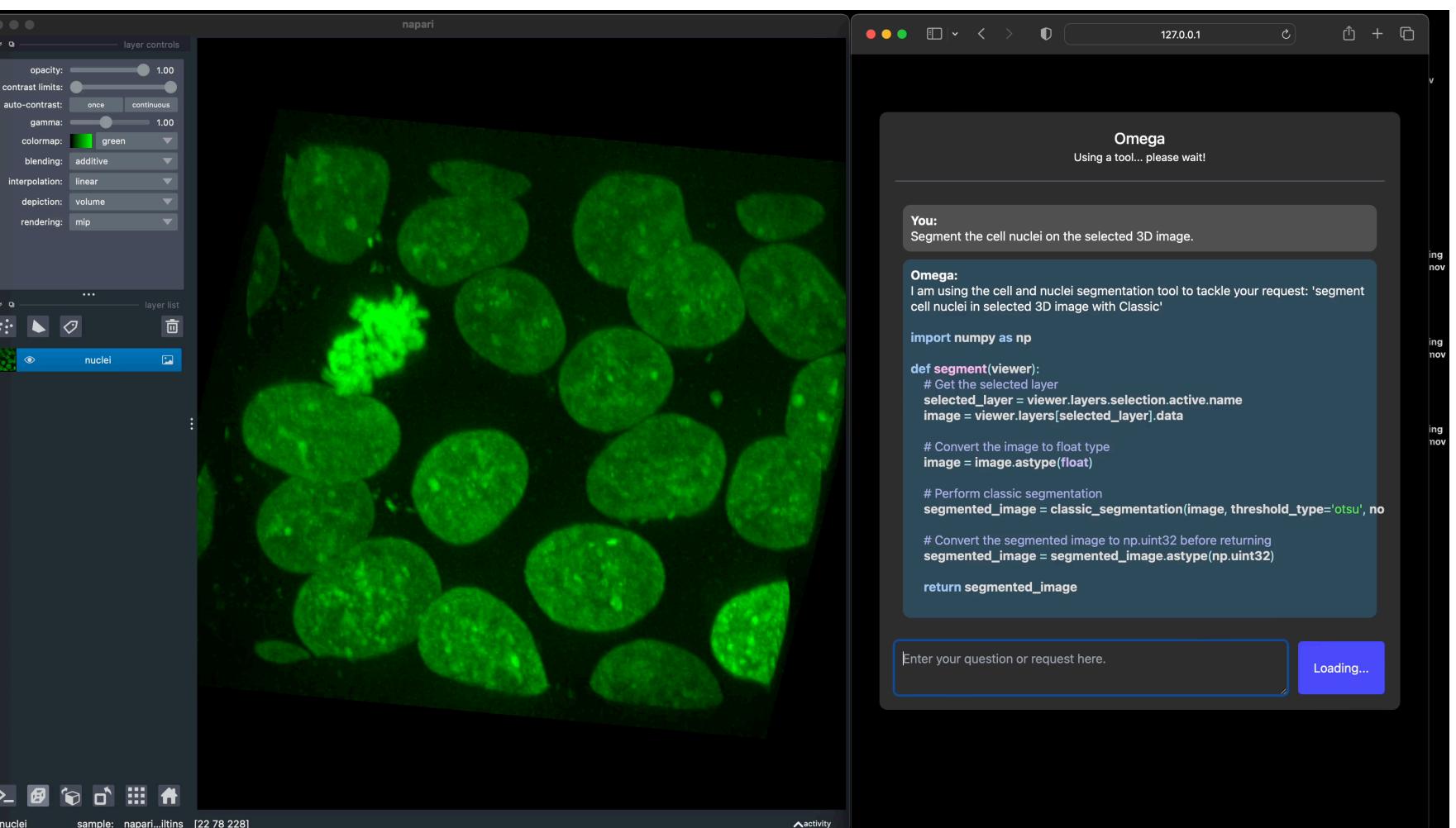
[10.5281/zenodo.10828225](https://doi.org/10.5281/zenodo.10828225)



Supplementary Figure 3. Omega's System Architecture. Diagram illustrating the Omega system architecture, encompassing both the user interface and backend components. The user interface is displayed at the top, consisting of the napari viewer, chat window, and AI-enhanced code editor. Key backend elements include the Omega ReAct agent, web server, AI tools within the editor, and the Python code repair module. The architecture employs a cascaded design, with a central dialog loop orchestrated by a primary language model. This model coordinates a suite of specialized tools, each powered by a secondary tool-specific language model that can access online resources, interact with the napari viewer, and integrate functionalities from third-party libraries such as Cellpose and StarDist.



Supplementary Video 1. Omega can segment nuclei with StarDist and perform follow-up analysis. The video showcases Omega's ability to segment cell nuclei in a 2D image using [StarDist](#). Omega successfully segments the nuclei and adds a label layer to the napari viewer. With further instructions, Omega can count the segmented nuclei and create a CSV file on the desktop folder of the machine. This file contains coordinates and areas of all segments, sorted by decreasing area, with one segment per row. Omega also opens the file using the system's default CSV viewer. The video has been sped up by a factor of 2.



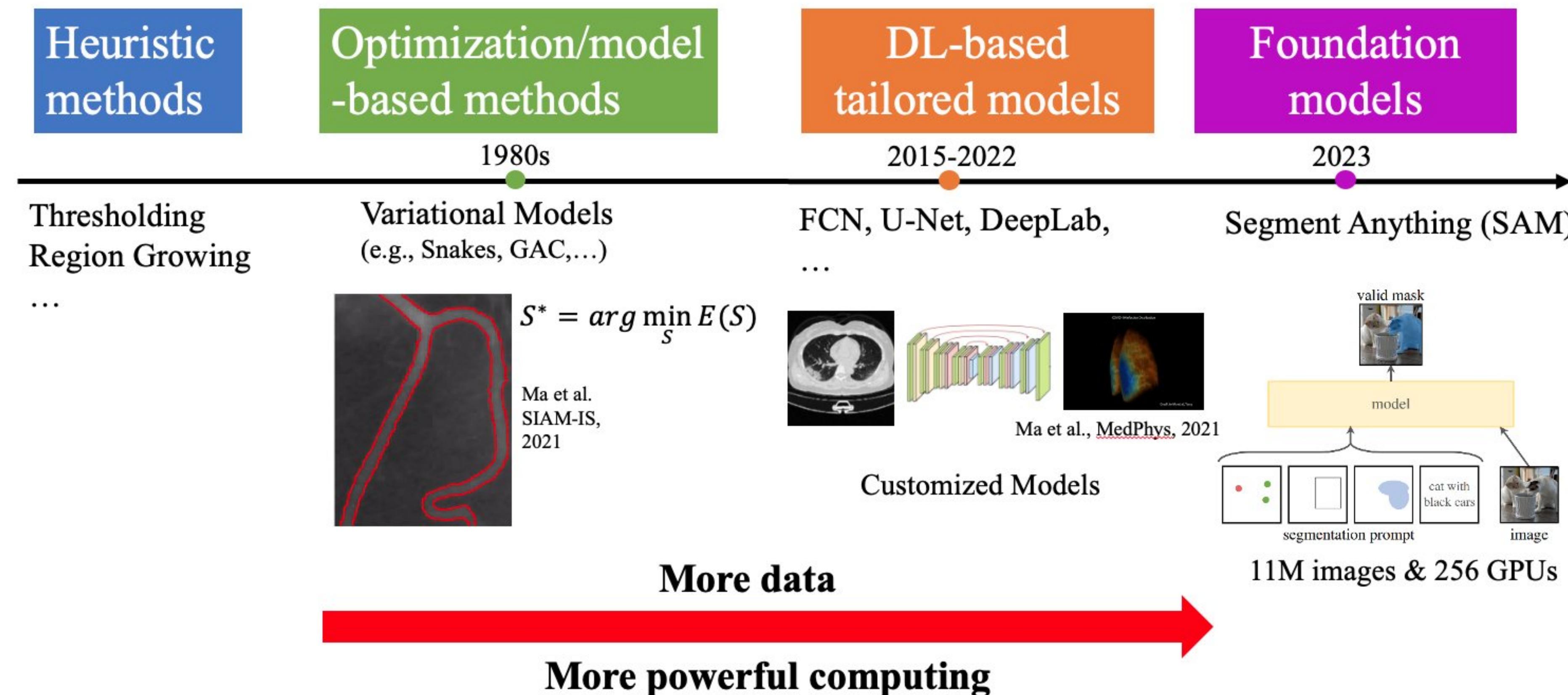
Supplementary Video 2. Omega can segment nuclei in a 3D image. This video shows how Omega segments the nuclei in a 3D image displayed in the napari viewer. Omega uses a specialized tool for cell and nuclei segmentation and employs a 'classic' approach that combines single thresholding, specifically [Otsu](#), with watershed splitting to prevent under-segmentation. After segmentation, Omega adds a labels layer to the viewer, and we inquire about the number of segments detected. The response is 27. The video has been sped up by a factor of 2.

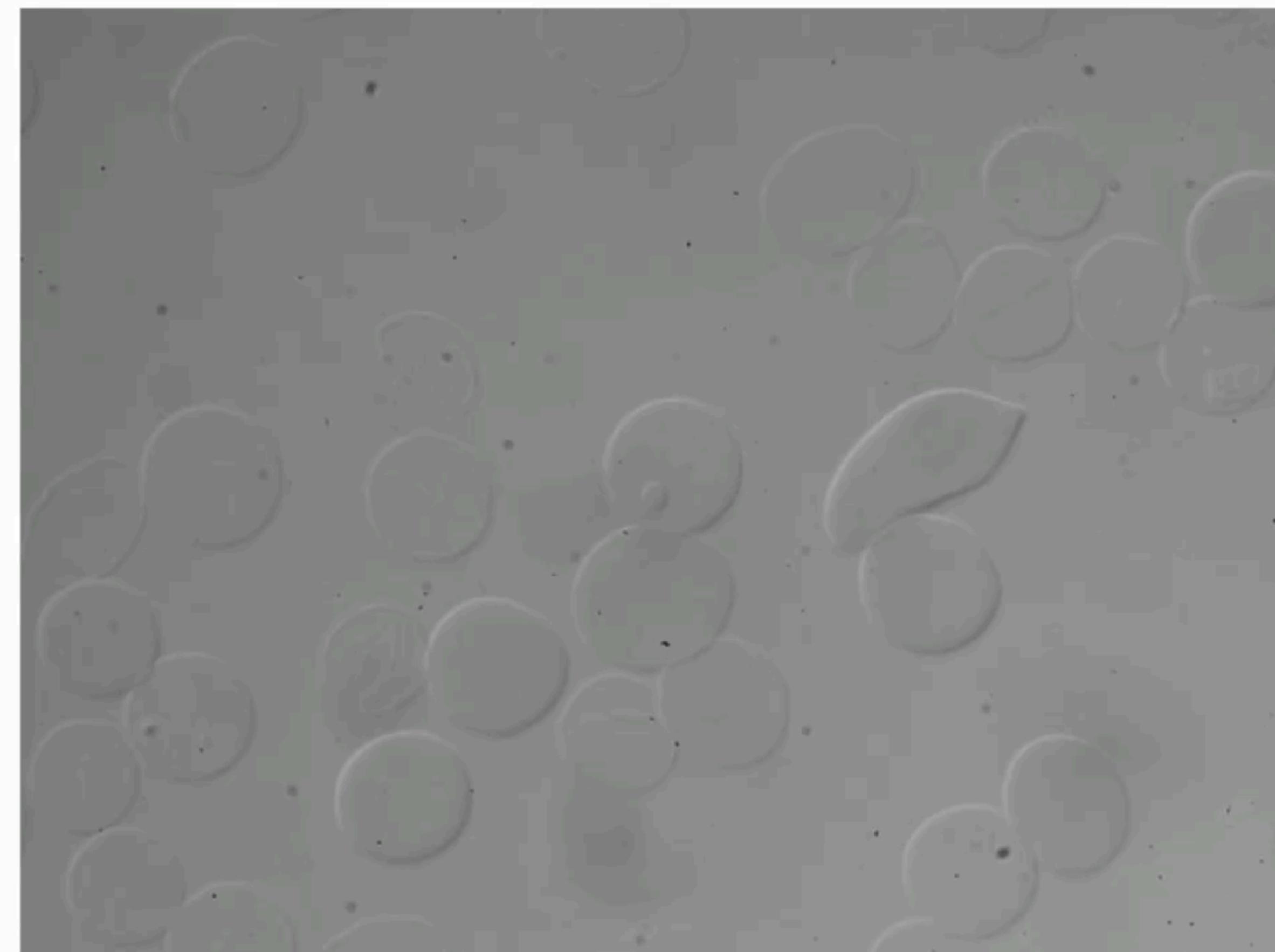
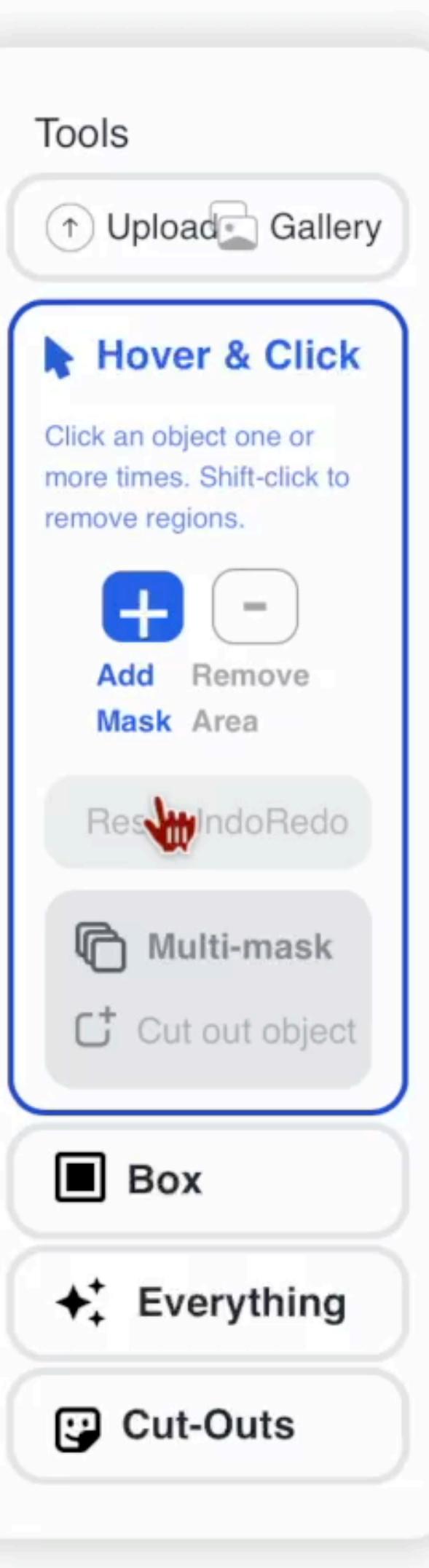


HARVARD
MEDICAL SCHOOL

History of Segmentation

Segmentation Paradigm Over the Past Half Century





"[BBC009v1](#) from the Broad Bioimage Benchmark Collection [[Ljosa et al., Nature Methods, 2012](#)]."

Segment Anything

Research by Meta AI

[Home](#)[Demo](#)[Dataset](#)[Blog](#)[Paper](#)

Tools

Upload Gallery

Hover & Click

Click an object one or more times. Shift-click to remove regions.



Add Remove
Mask Area

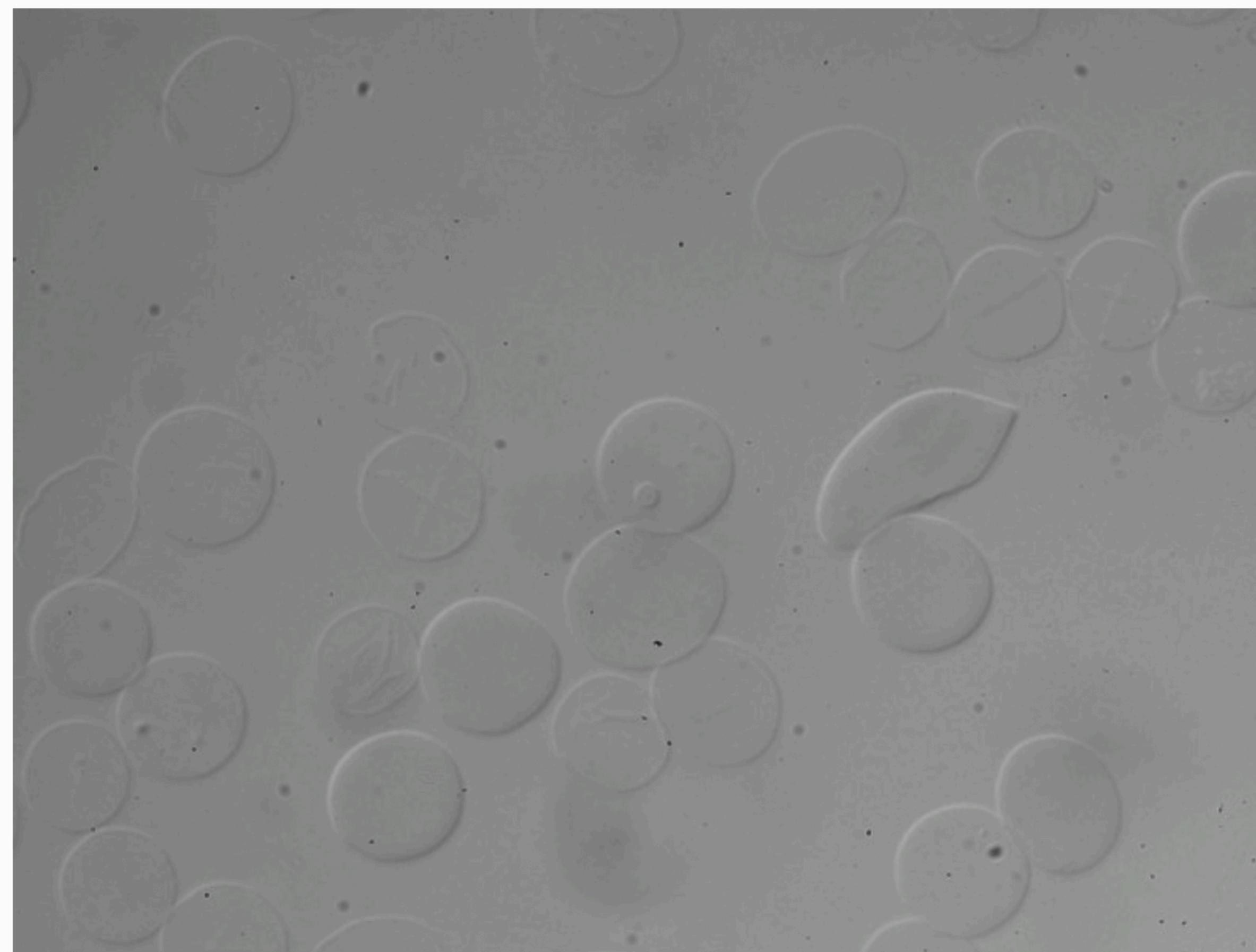
Reset Undo Redo



Multi-mask
Cut out object

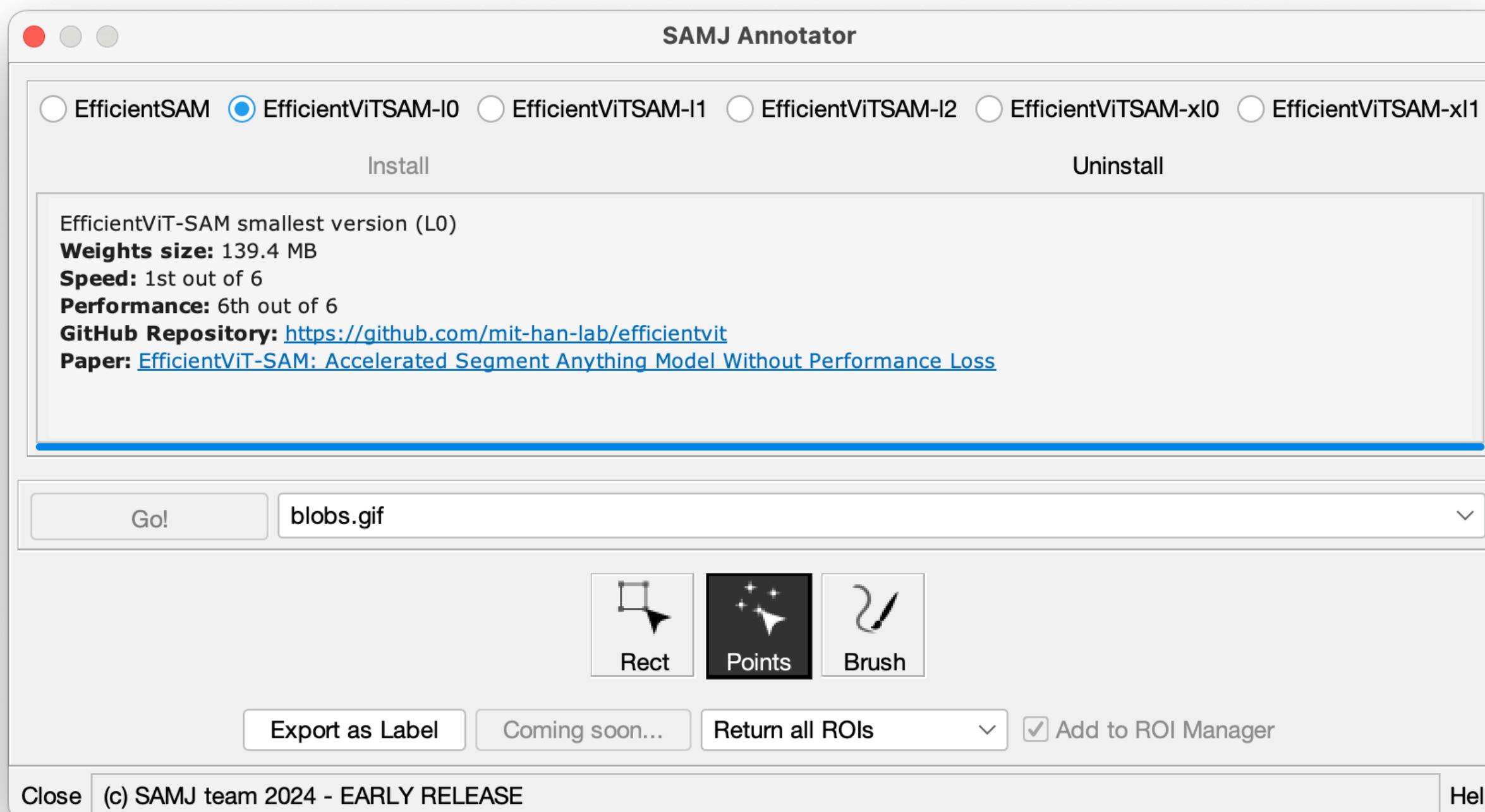


Box
Everything



Segment Anything in ImageJ/Fiji

Released around March 14th, 2024
 Graphical User Interface
 No coding required



Example of point annotations

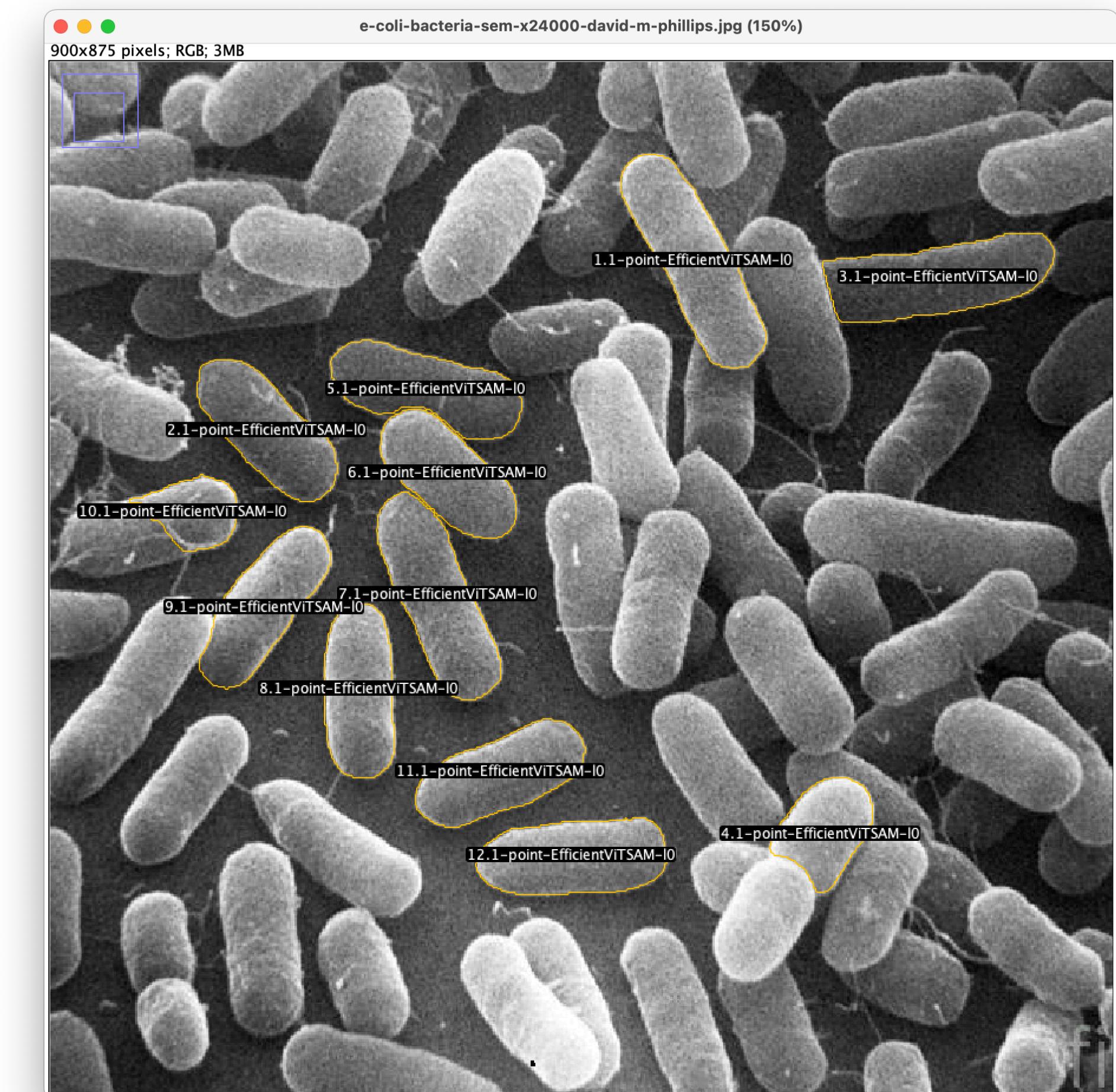


Image Analysis Workflow

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



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

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



- **Data-wrangling in general**
 - Image format conversion, incl. lossless compression
 - Renaming
- **Restoration and Reconstruction**
 - Tomographic reconstruction
 - Registration (3D stacks)
 - Stitching (tiled images)
 - Illumination correction
 - Intensity normalisation
 - Deconvolution
 - Filtering (smoothing, sharpening, etc)
 - Grayscale morphological operations
- **Color deconvolution**
- **Channel alignment (chromatic aberrations)**
- ...



- **Image Segmentation**
 - Thresholding
 - Pixel Clustering
 - Edge detection
 - Region growing
 - Level set methods
 - Watershed
 - Model based
 - Multi-scale
 - Trainable pixel-classification
- **Spot detection**
- **Tracking**
- ...



- **Filtering** detected objects based on
 - Size, shape, intensity
 - Texture, location, neighbours
- **Binary morphological operations**
 - Open/close
 - Skeletonise
 - Hit-or-miss transform
- ...



- **Object measures**
 - Size, shape, intensity
 - Texture, positions
 - Local neighbourhood
- **Image measures**
 - Object count
 - Area covered
 - Total intensity
- **Co-localisation**
- ...



- Exploratory data analysis
- Filtering of numerical data based on measured values
- Summary statistics
- Plotting numerical values
- Making movies and montages of images
- Record entire work-flow for later reference
 - Write down what you did and why (each step)
 - Use a version control system
 - Document your code
 - ...

Processing vs. Analysis

Image Formation

object in → image out

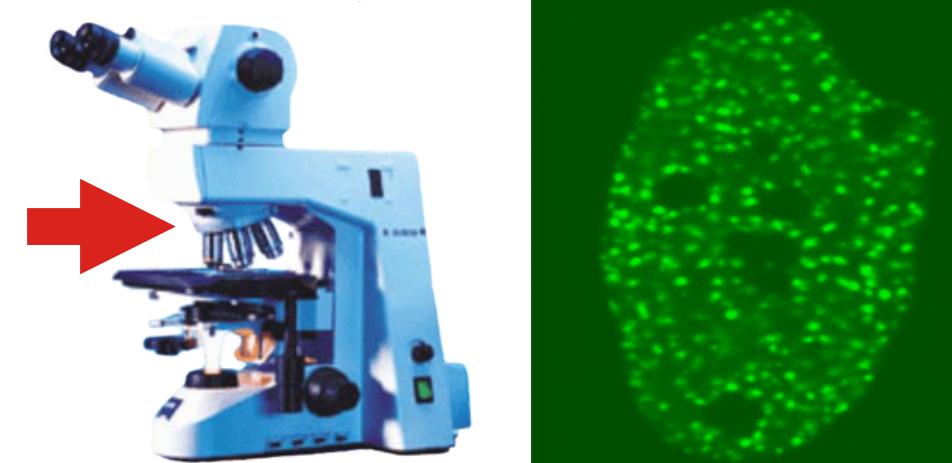


Image Processing

image in → image out

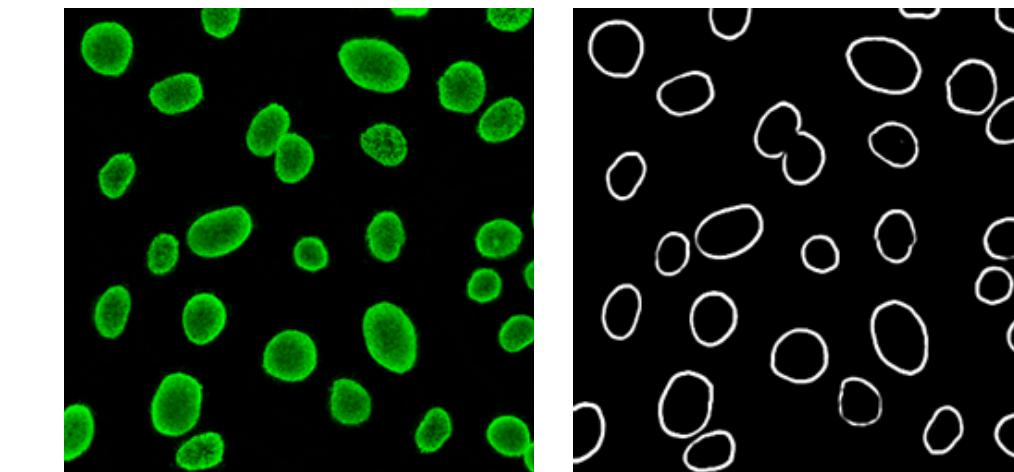
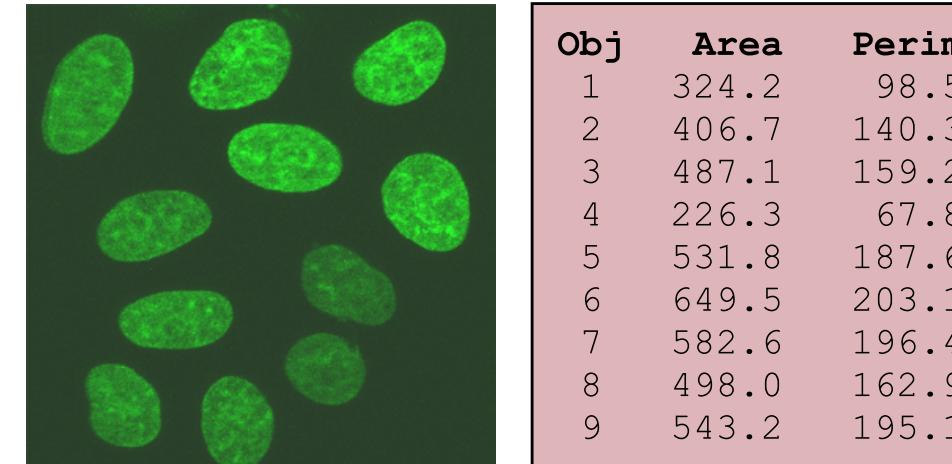


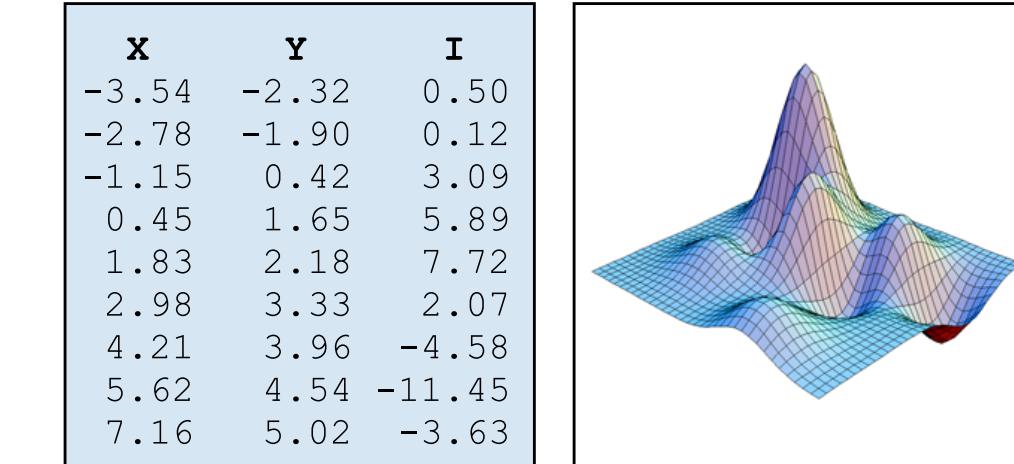
Image Analysis

image in → features out



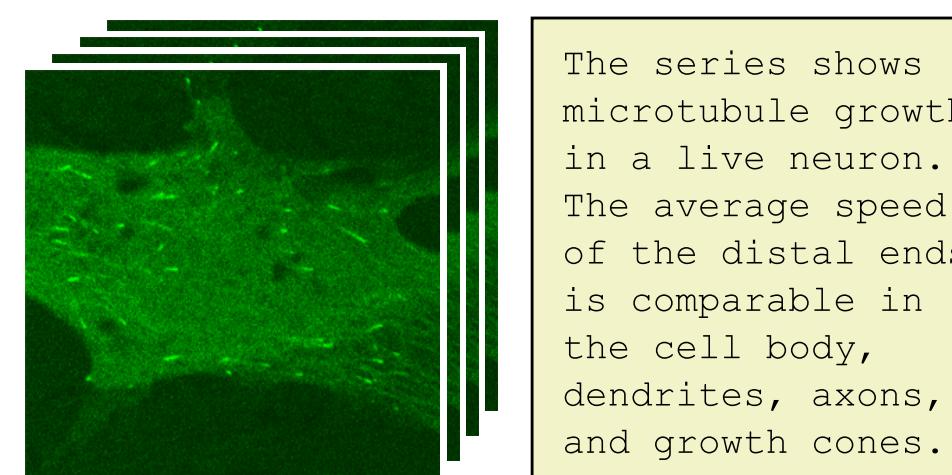
Computer Graphics

numbers in → image out



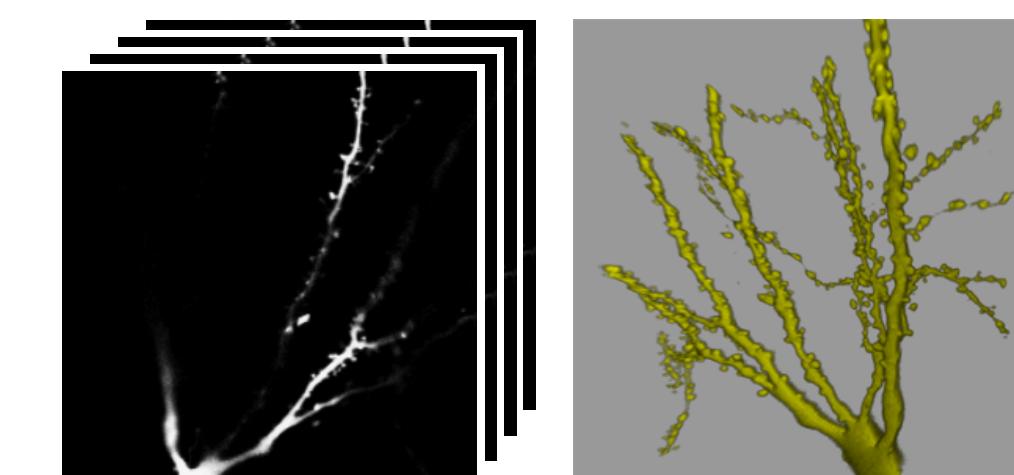
Computer Vision

image in → interpretation out



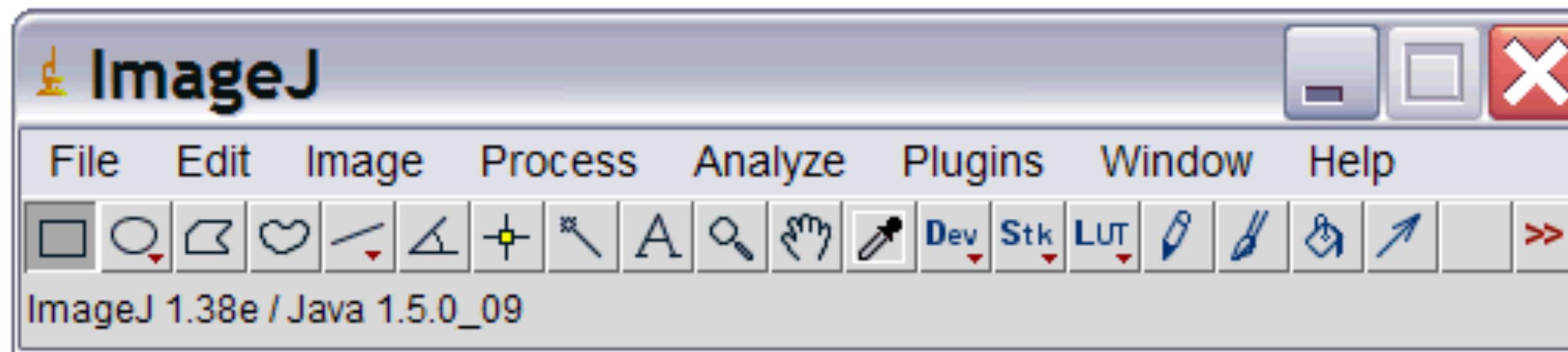
Visualization

image in → representation out

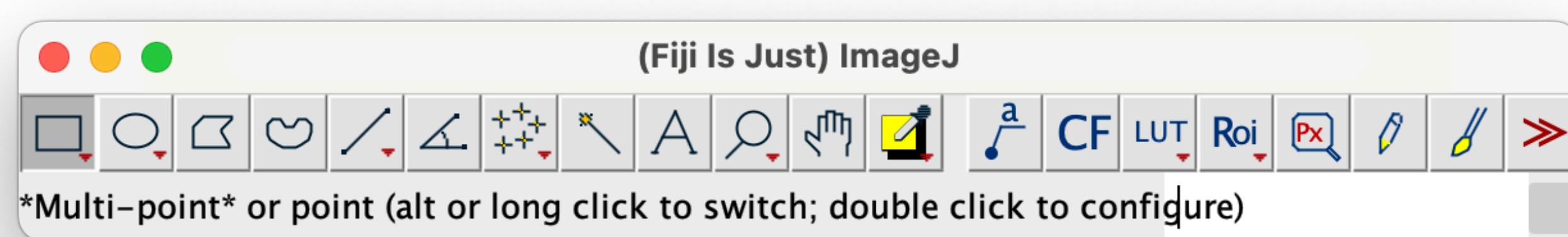


Why ImageJ/Fiji?

ImageJ—the Swiss Army knife of bioimage analysis



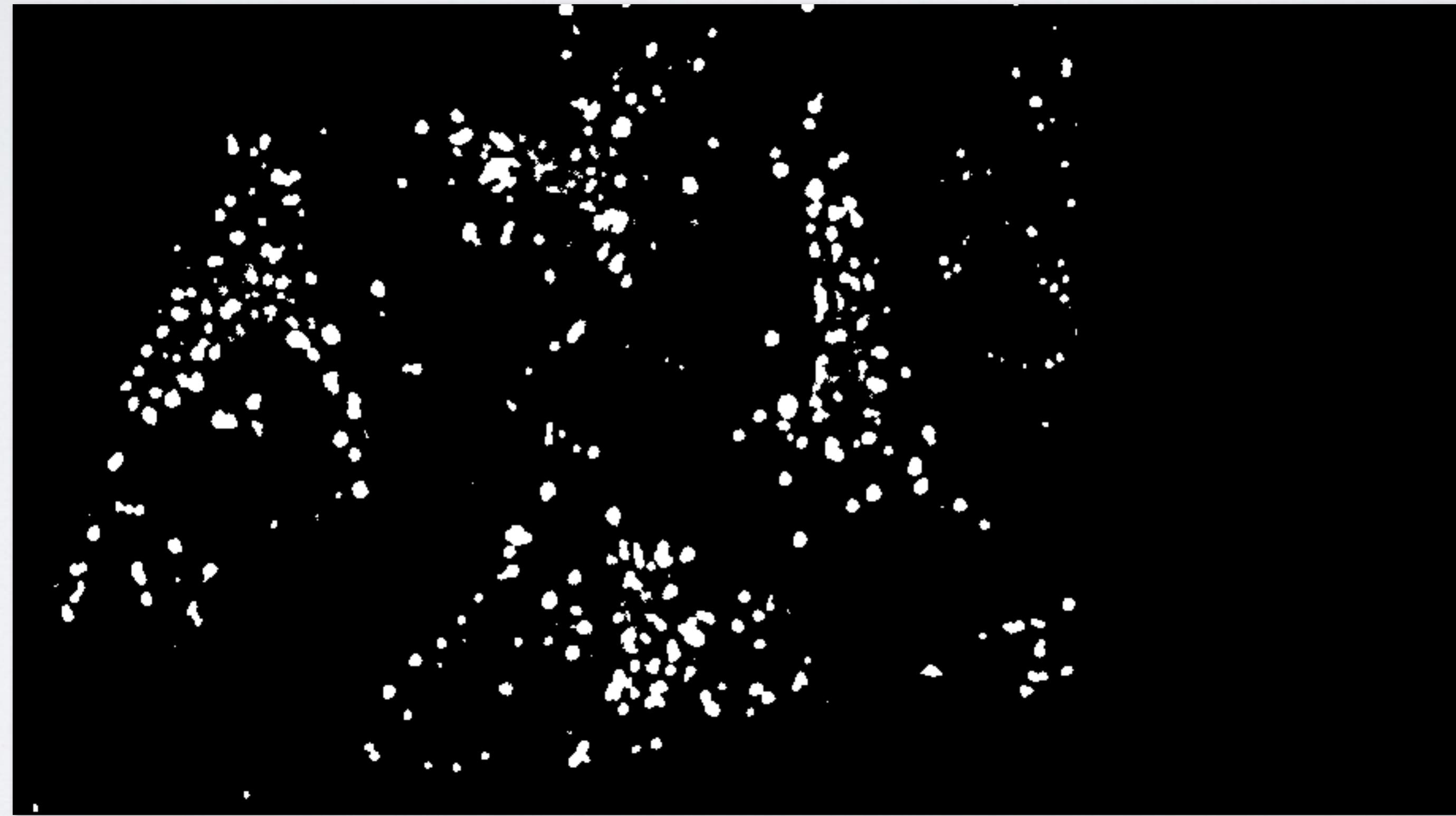
Fiji is just ImageJ—with batteries included



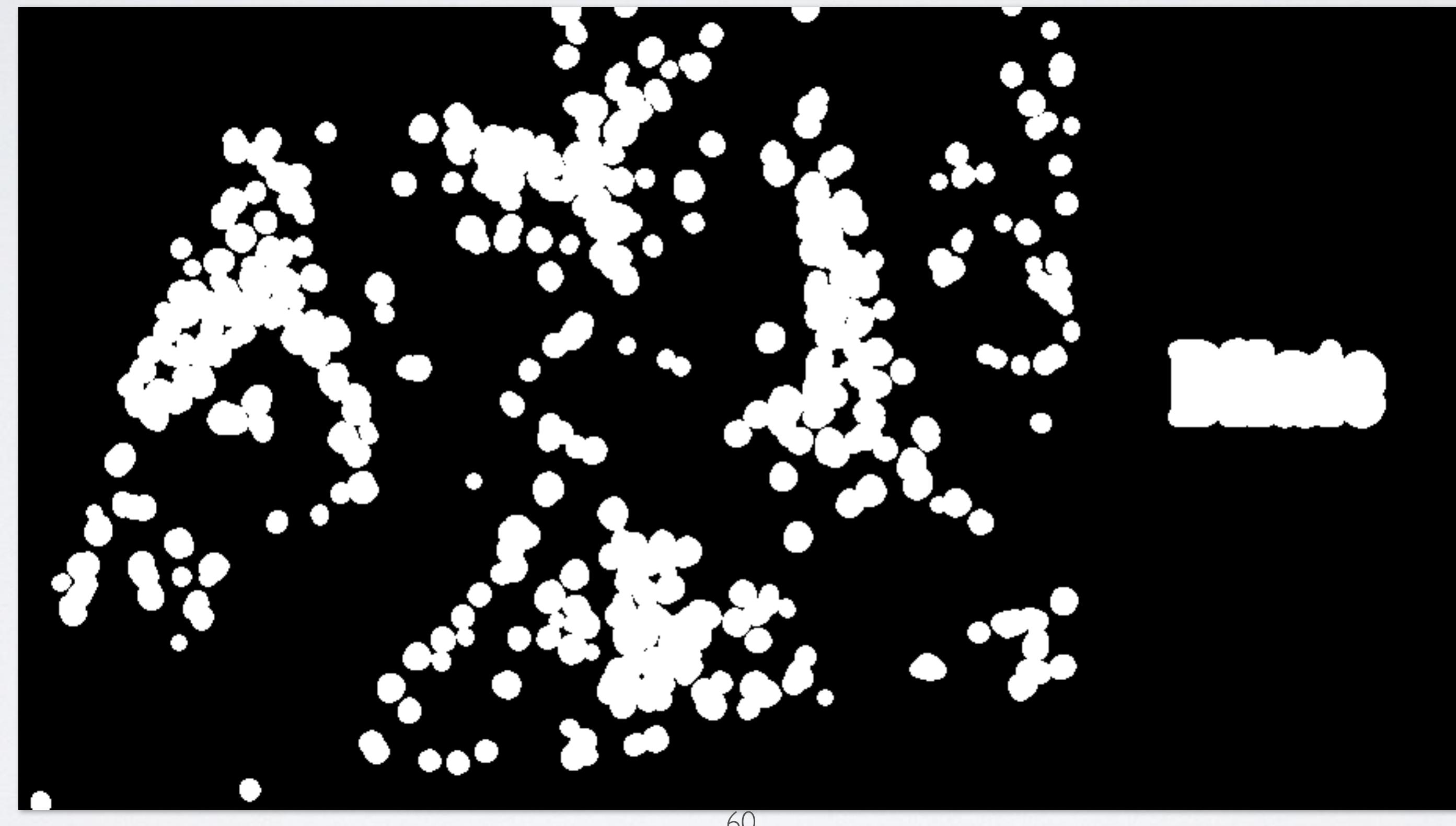
Debugging Fiji and getting help — a workflow

1. **Restart Fiji**
2. **Ask you right neighbor**
3. **Ask you left neighbor**
4. **Ask an instructor**
5. **Search <https://forum.image.sc/>**

Binary images can be cleaned
up using morphological operations



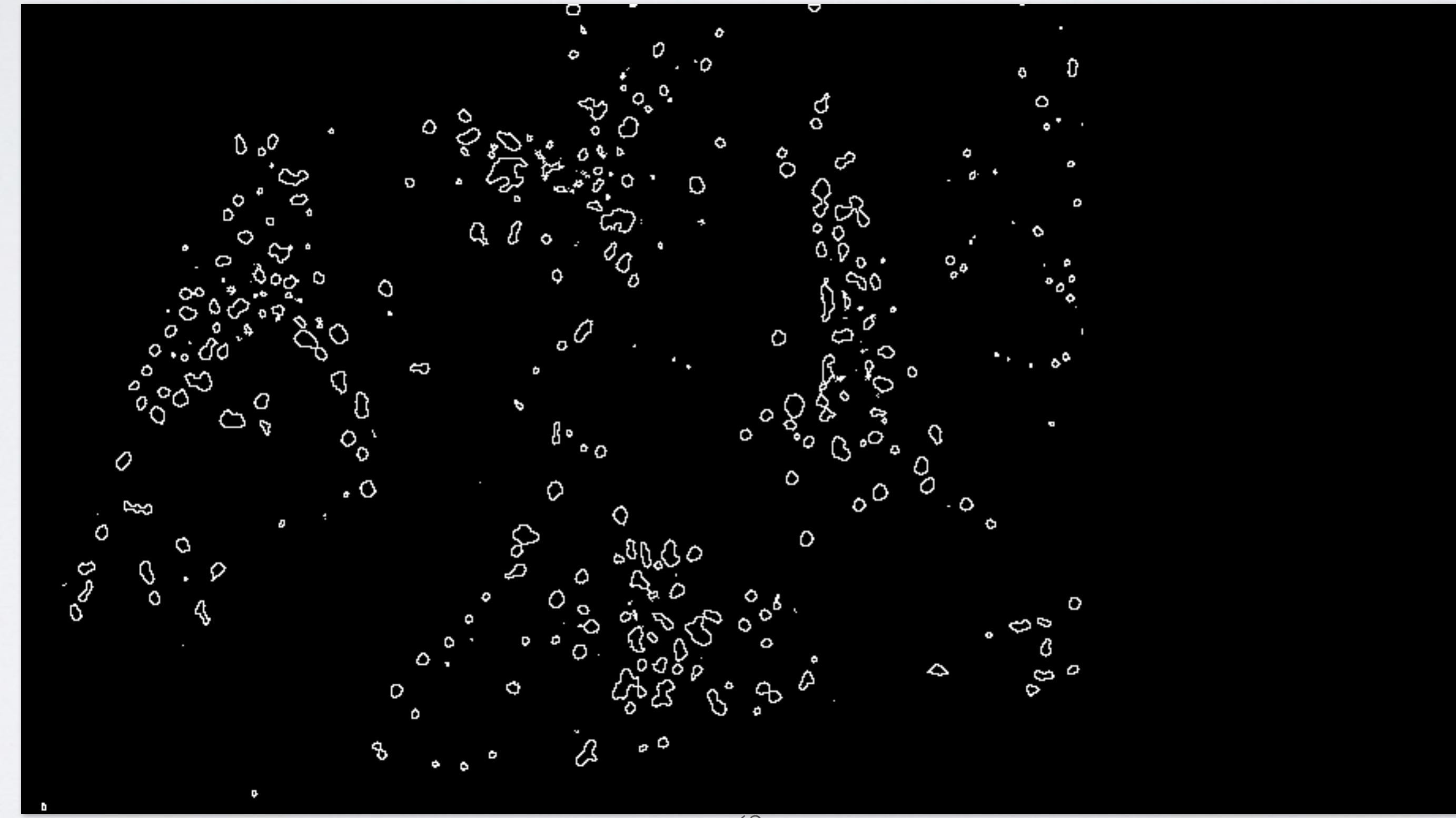
Dilation is equivalent to a maximum filter & increases object sizes



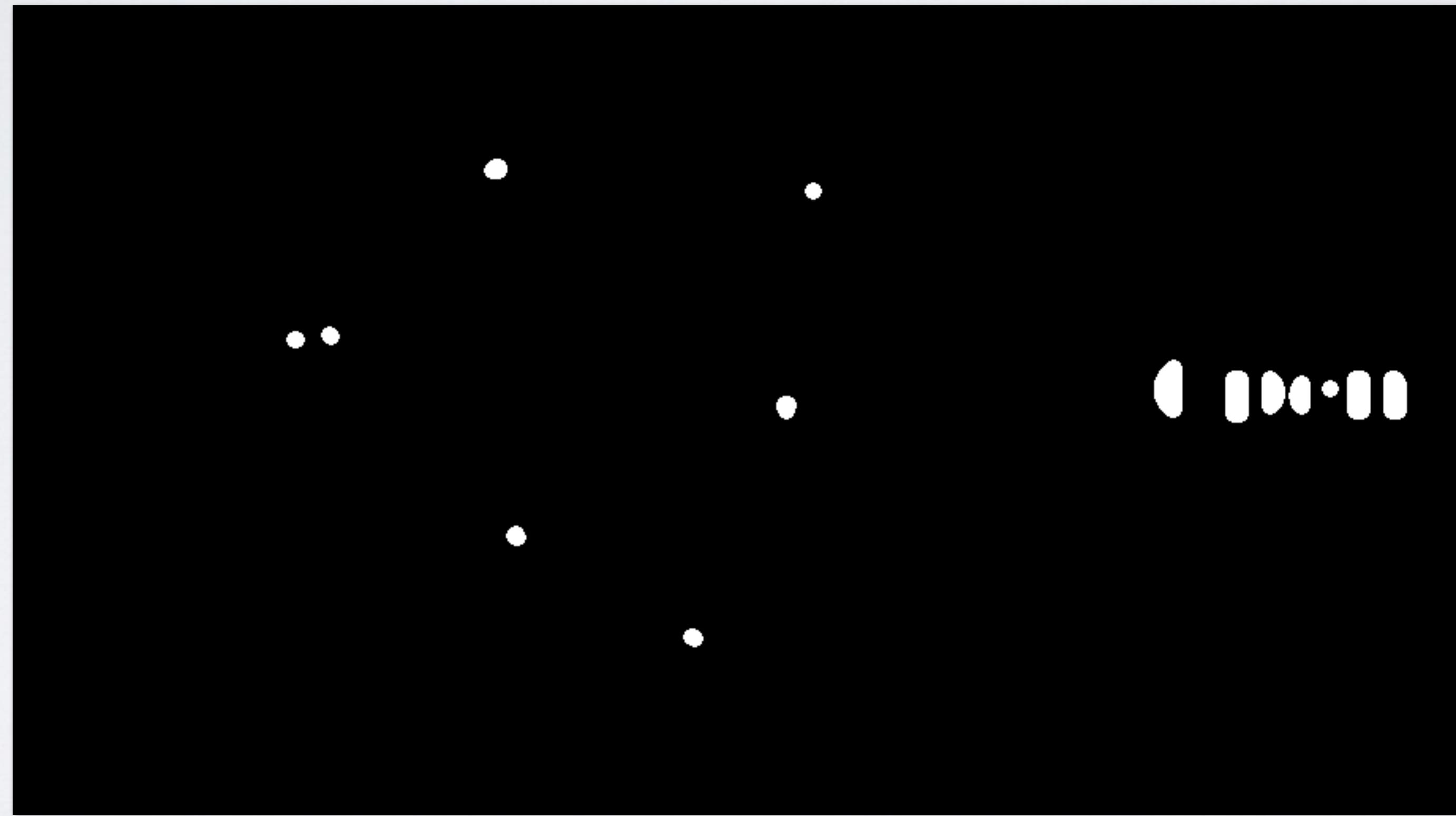
Erosion is equivalent to a minimum filter & reduces object sizes



Subtracting an eroded image
from the original gives outlines



Opening consists of
an erosion followed by a *dilation*



*Closing consists of
a dilation followed by an erosion*

