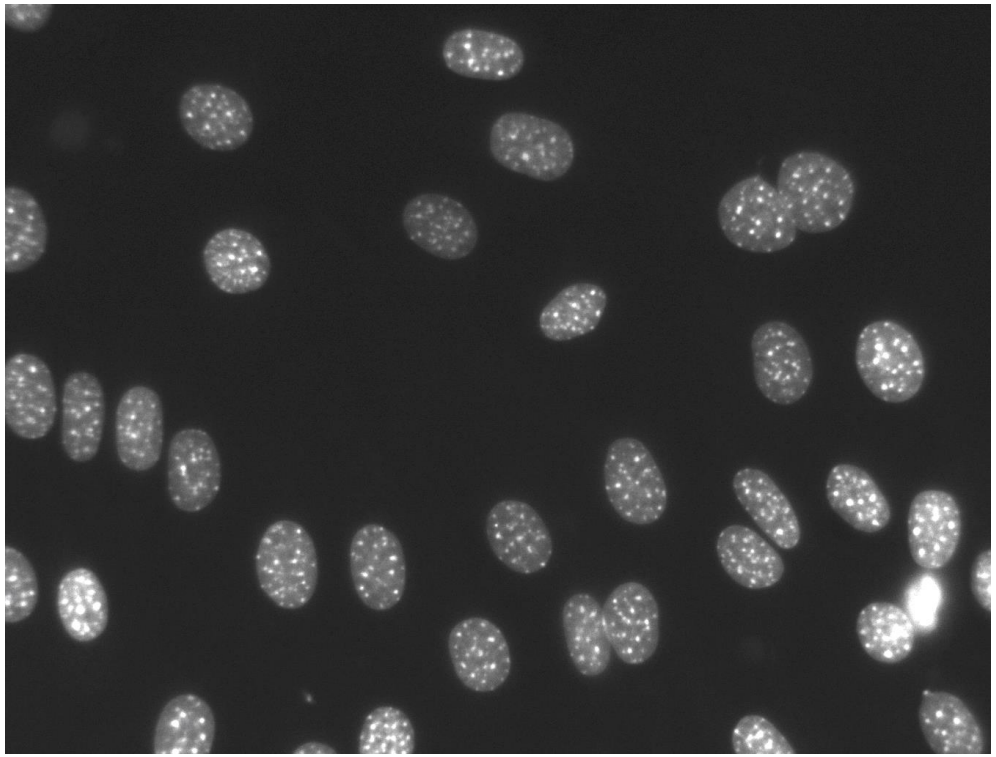
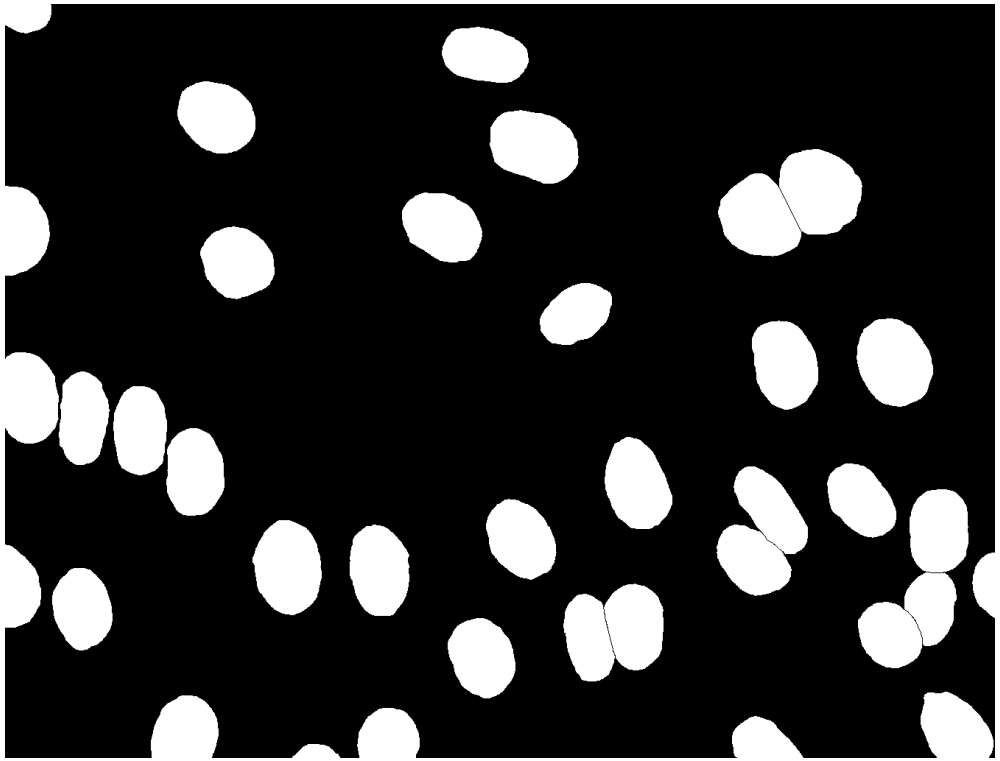
Image segmentation

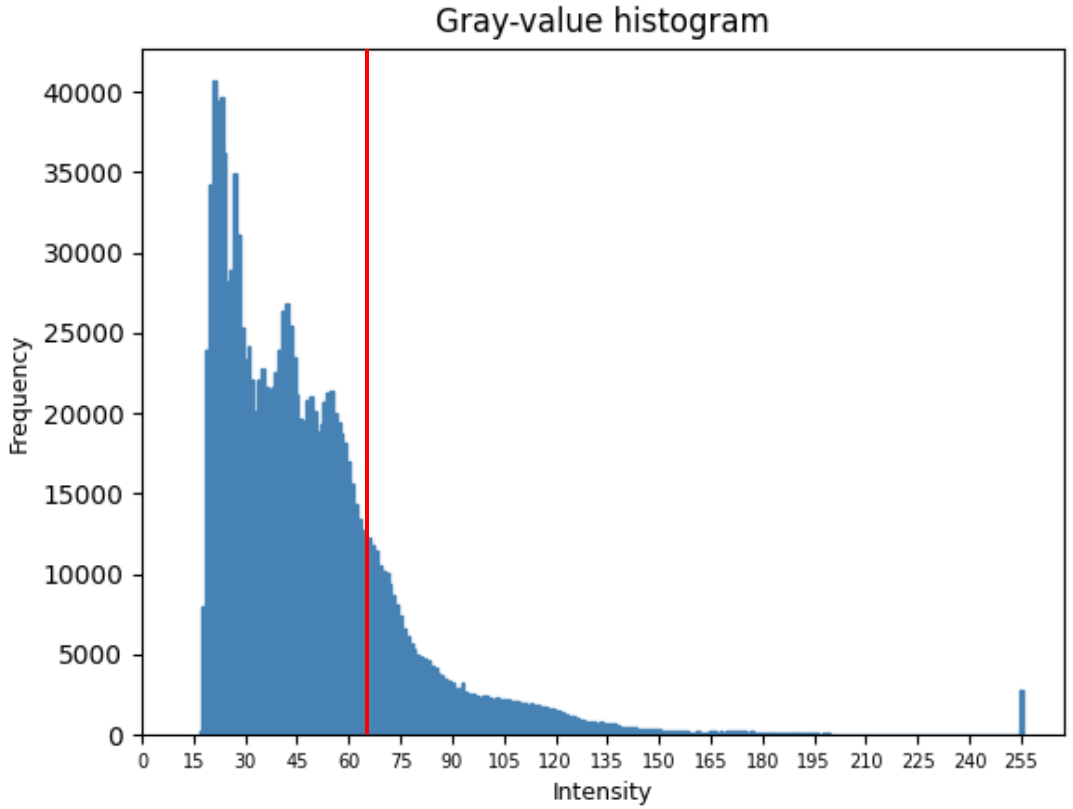




Methods

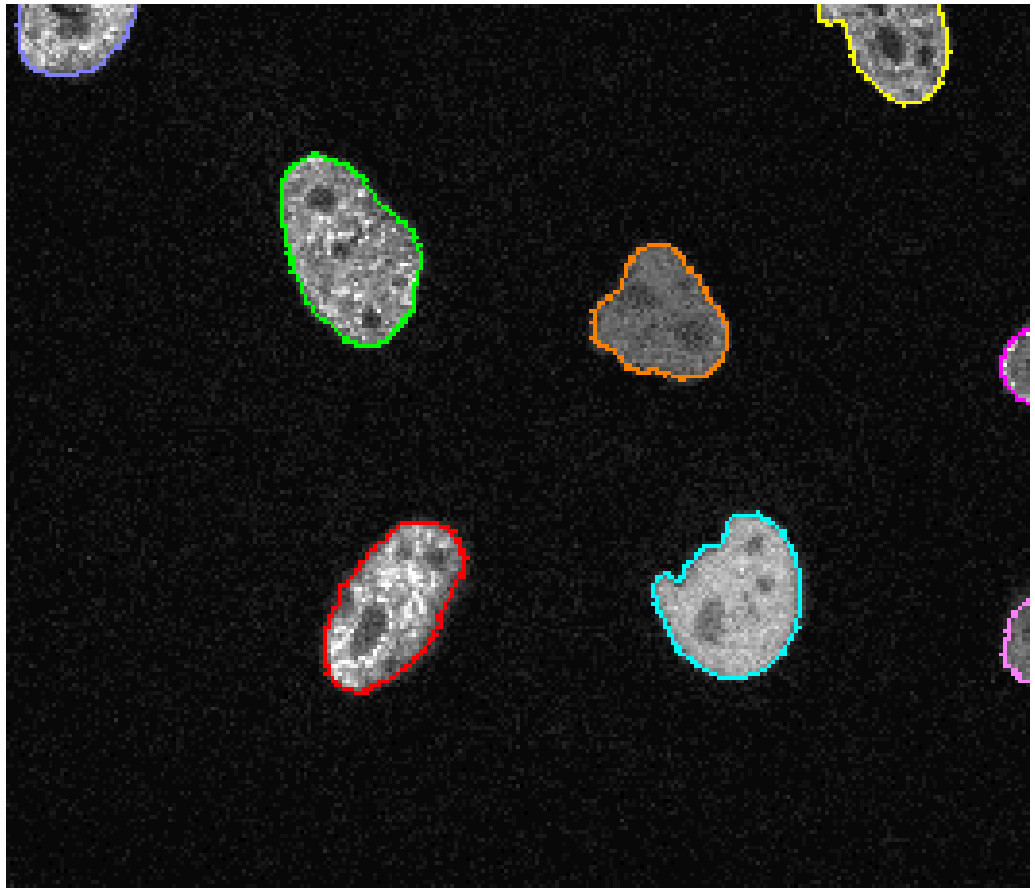
- Thresholding → Intensity clipping

Otsu's thresholding



Applications

- High-throughput cytometry
  - Cell size
  - Cell counting
- Cell tracking



Workflow

01
Input

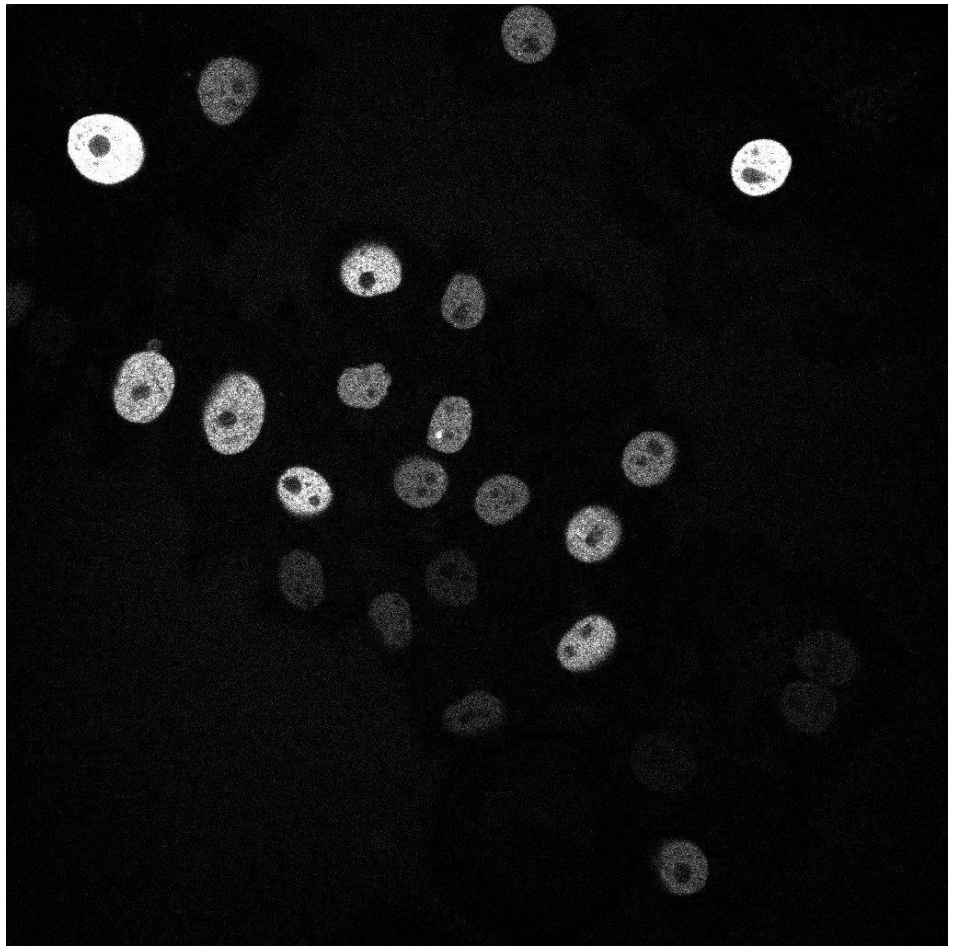
02
Preprocessing

03
Otsu‘s Thresholding

04
Dice Score

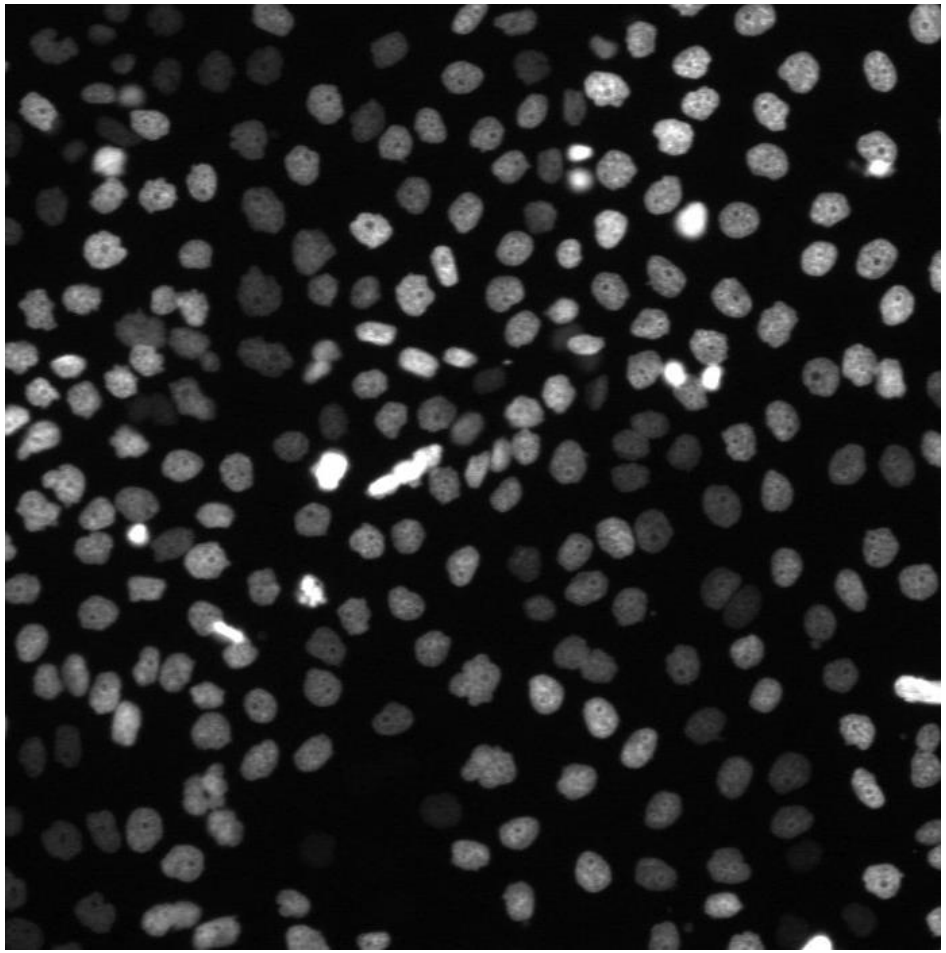
05
Comparison

Input 1: N2DH-GOWT1 cells



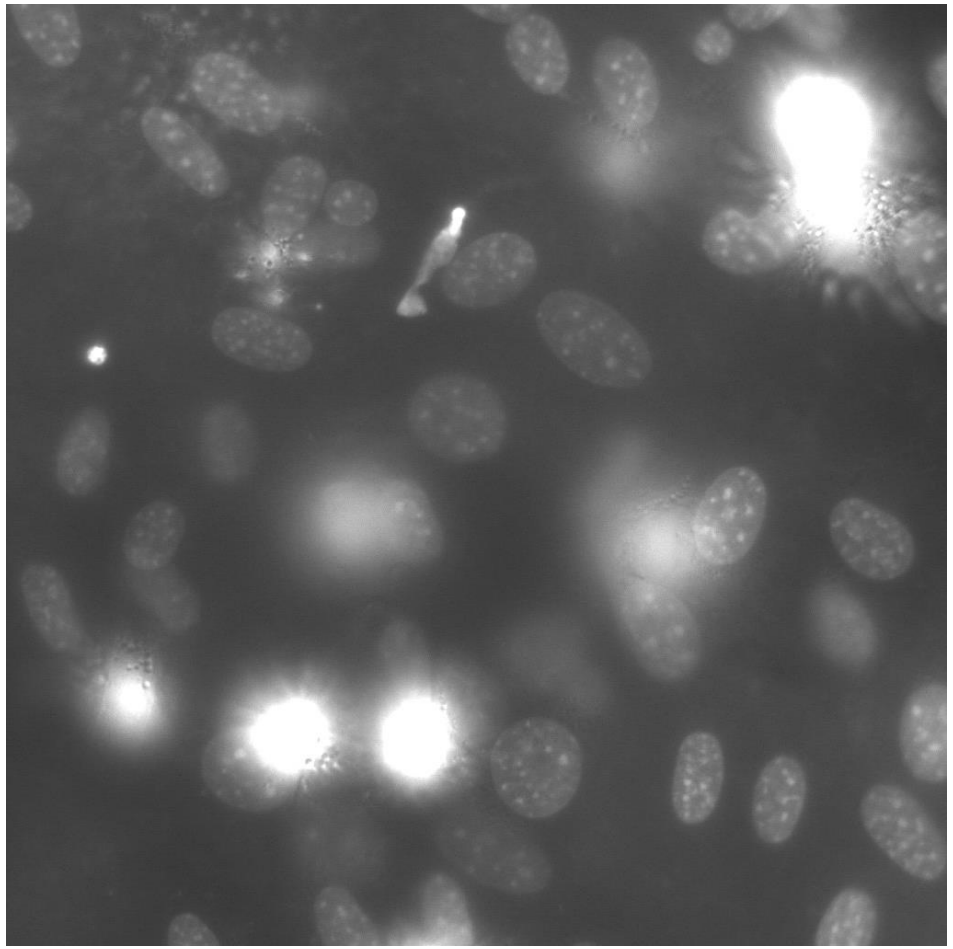
- GFP-Gowt1 mouse embryonic stem cells
- Time-lapse confocal microscopy and GFP-staining
- Leica TCS SP5 microscope
- Investigate genomic integrity of the cells

Input 2: N2DH-HeLa cells



- Human epithelial cells of cervical cancer
- Live imaging of fluorescently labelled chromosomes
- Olympus IX81 microscope
- Phenotypic profiling of the human genome

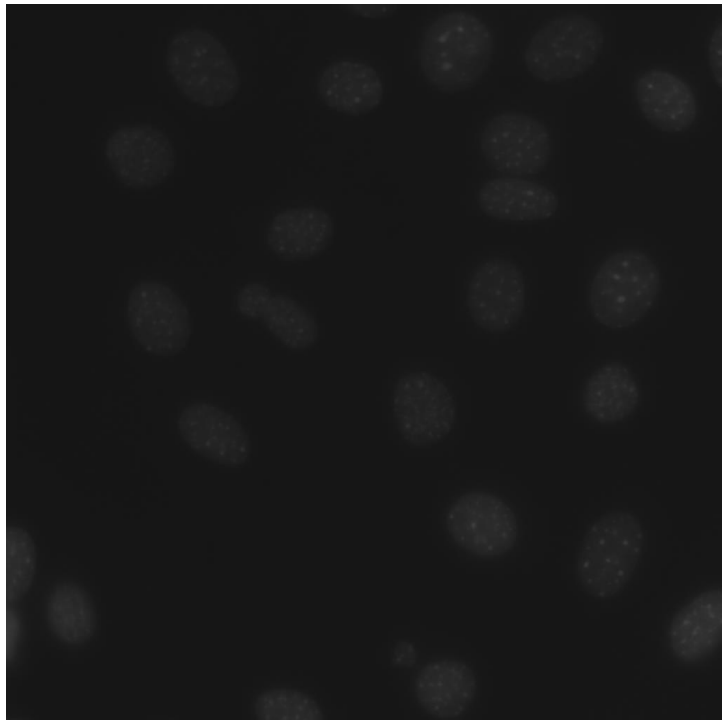
Input 3: NIH3T3 cells



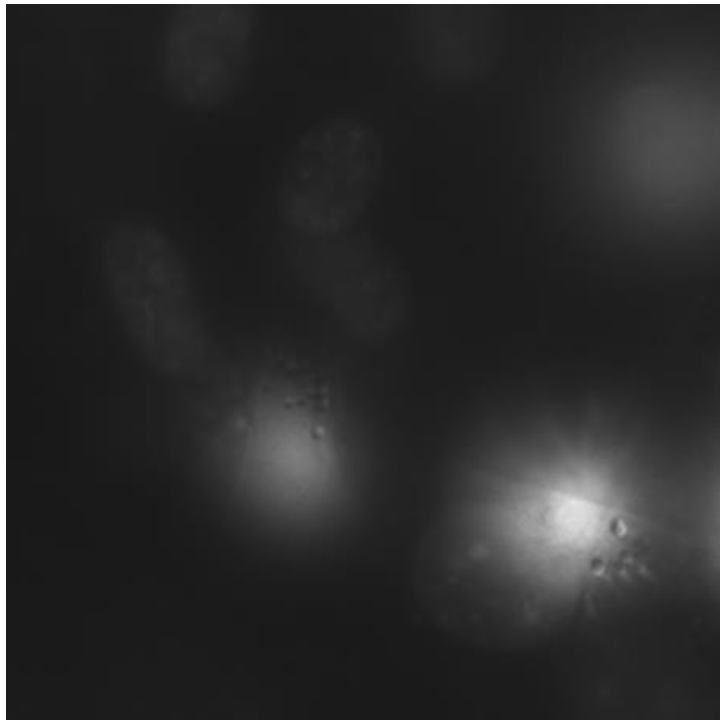
- Several mouse embryonic fibroblast cells
- Fluorescence microscopy images
- Evaluation of image analysis pipelines

Problems

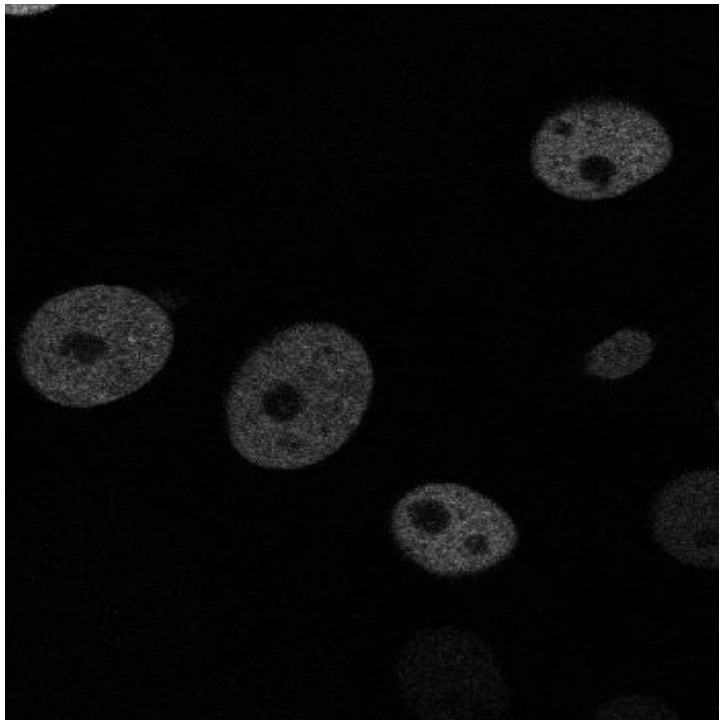
Low contrast



Reflections



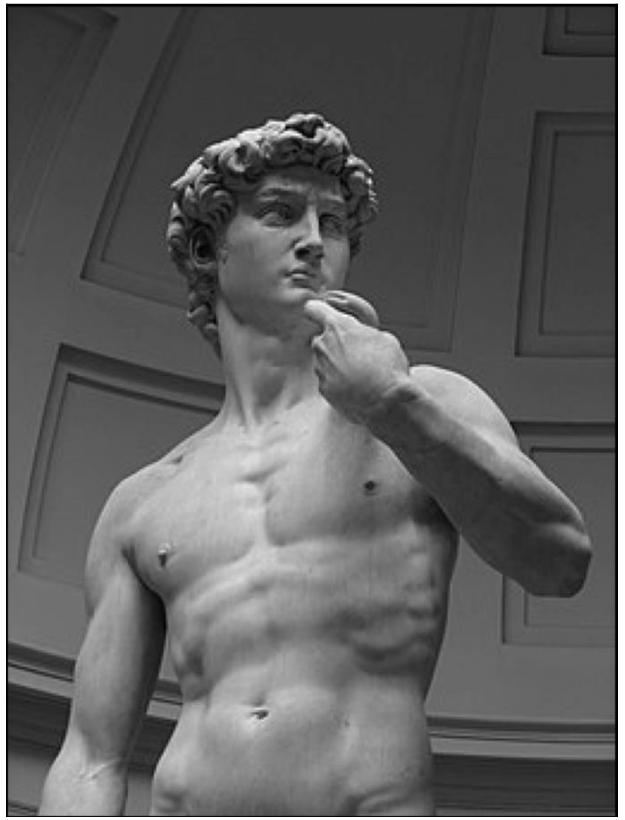
Random noise



Preprocessing

Solutions:

- Random noise
- → **Gaussian filter**
- Reflections
- Low contrast



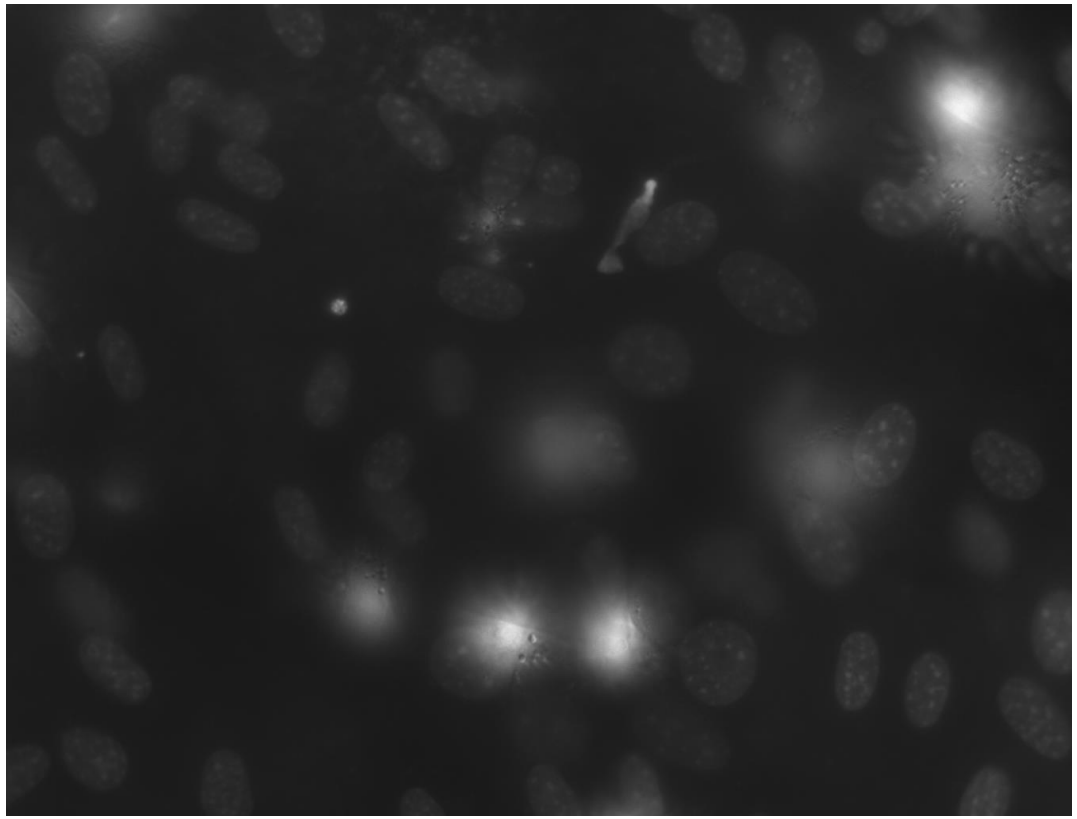
Original

Gaussian filter (σ = 3)

Preprocessing

Solutions:

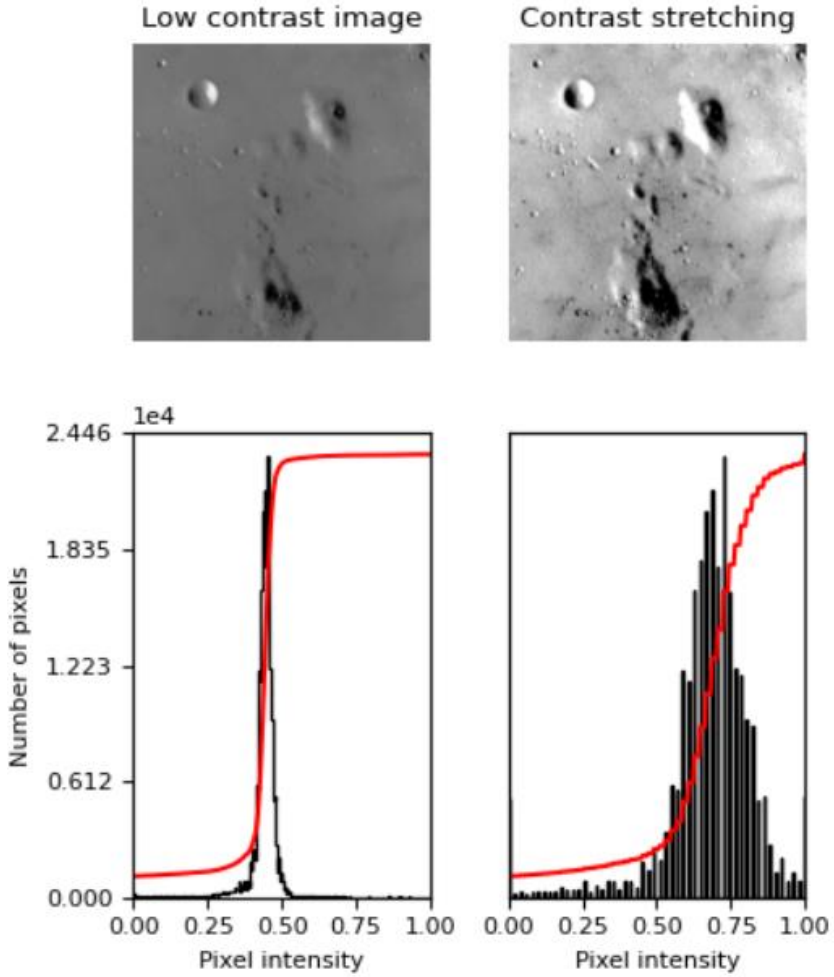
- Random noise
  → **Gauss filter, median filter**
- Reflections
  → **Thresholding**
- Low contrast



Preprocessing

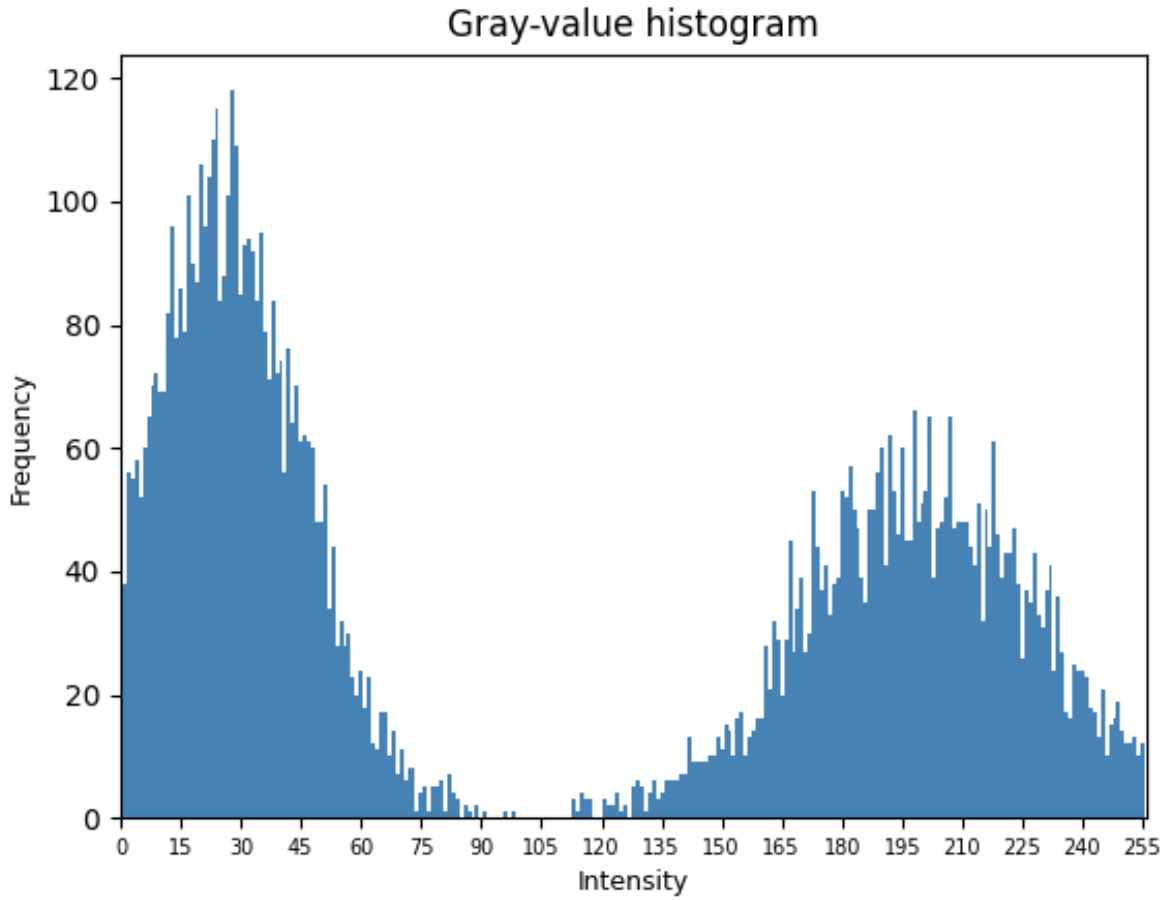
Solutions:

- Random noise
  → **Gauss filter, median filter**
- Reflections
  → **Thresholding**
- Low contrast
  → **Histogram stretching**

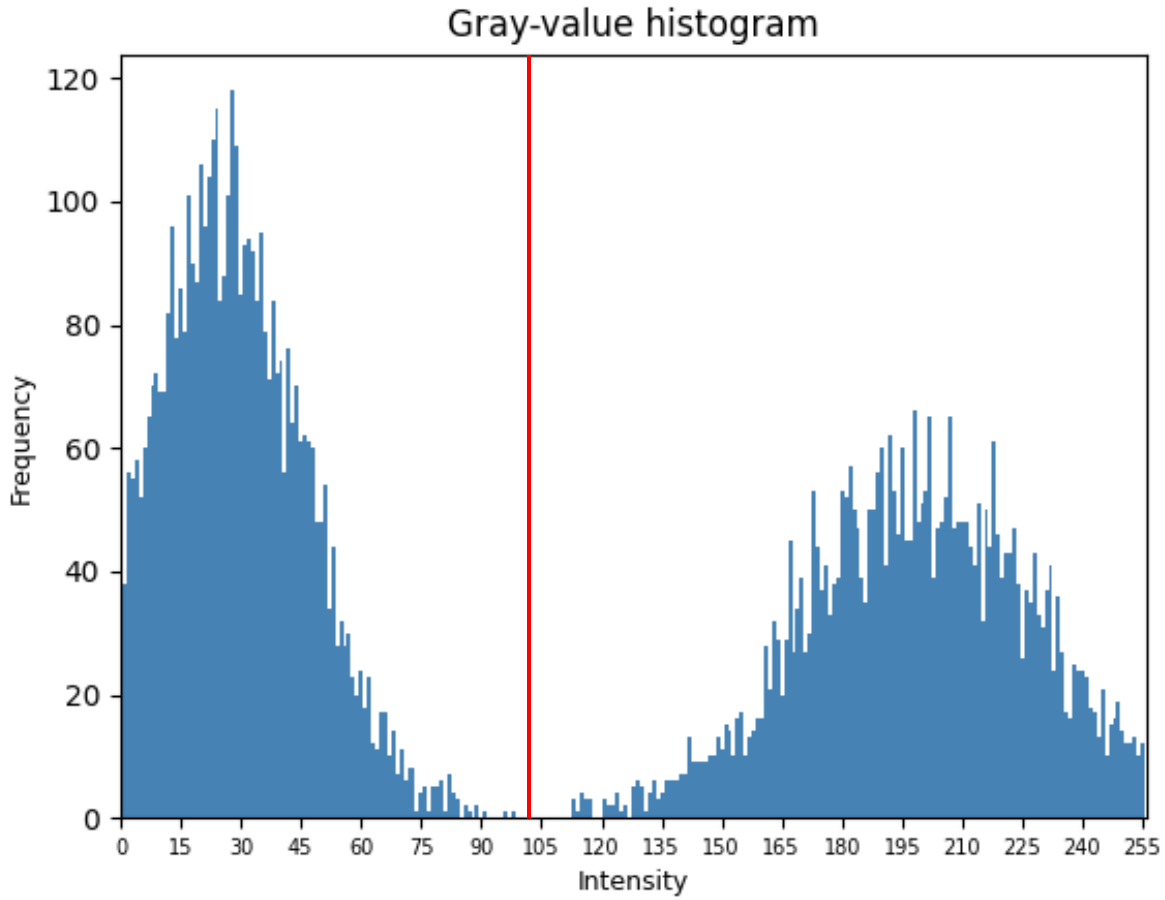


Adapted from: https://scikit-image.org/docs/dev/auto_examples/color_exposure/plot_equalize.html

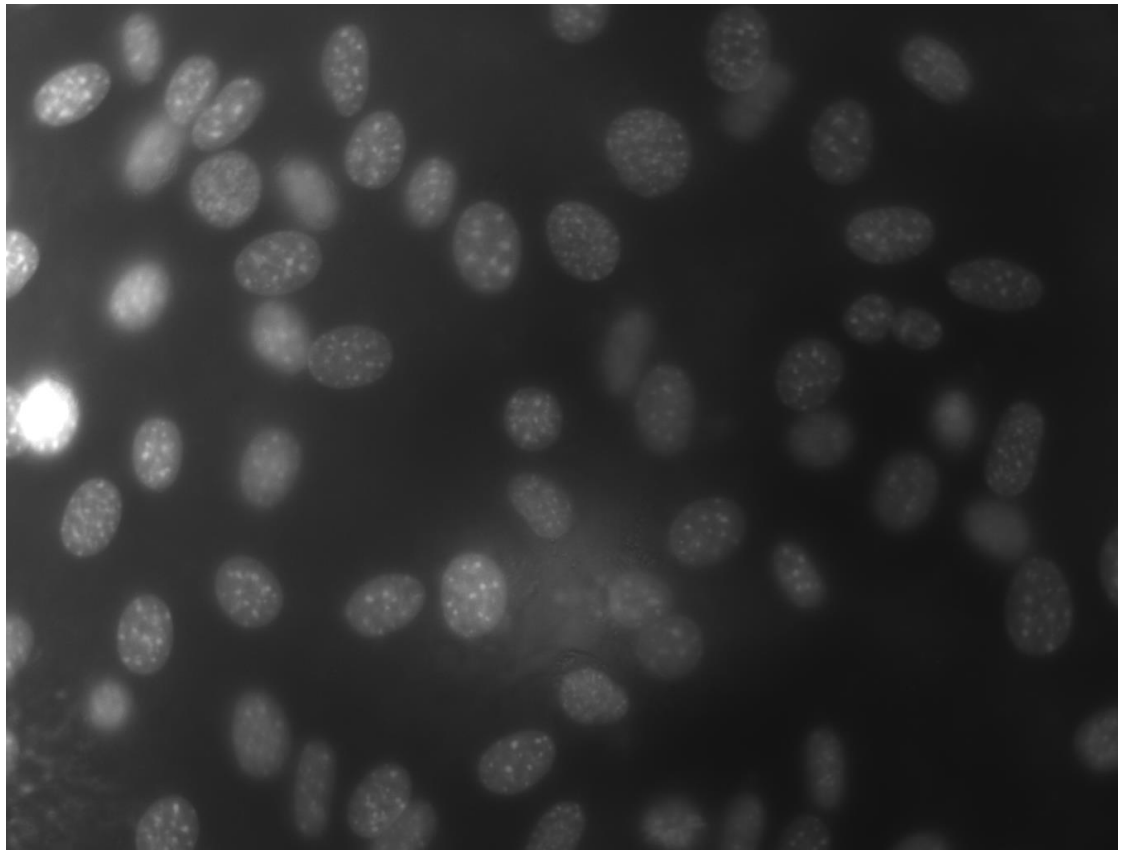
Otsu's Thresholding

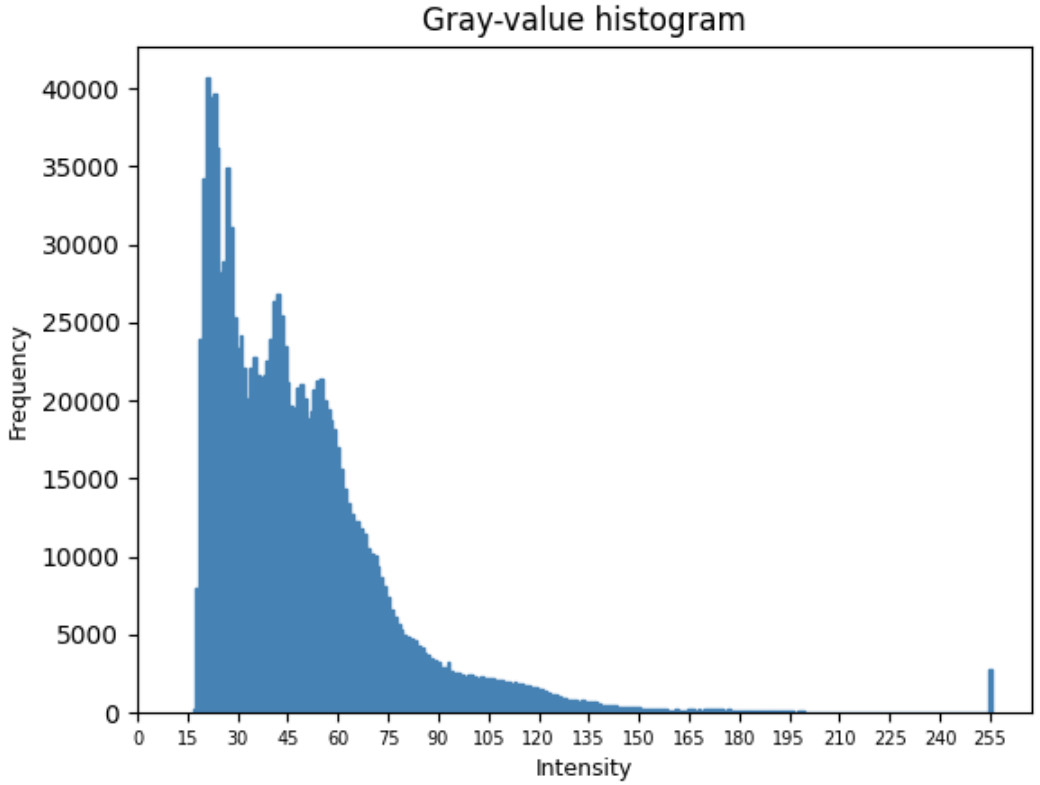


Otsu's Thresholding



Otsu's Thresholding





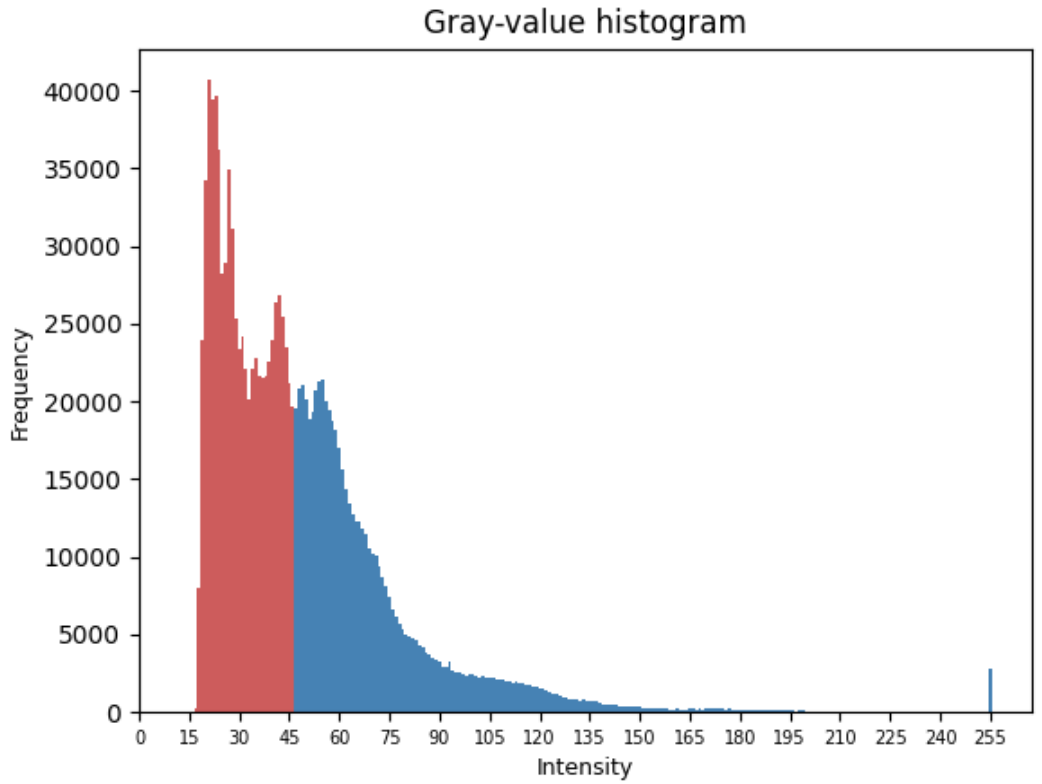
Threshold value k $\in [0,255]$

Otsu's Thresholding

Between-class variance

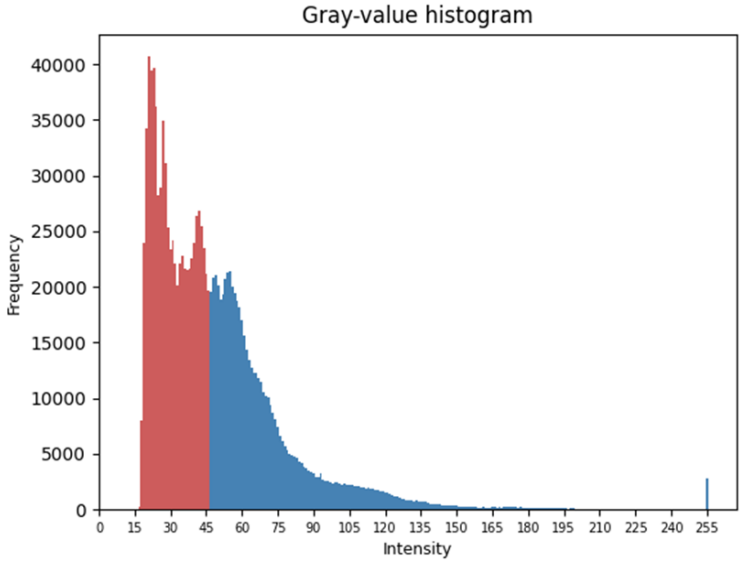
$$\sigma_B = \omega_0 \omega_1 (\mu_1 - \mu_0)^2$$

$\omega_{0,1}$ = probability of class occurrence
$\mu_{0,1}$ = mean intensity values

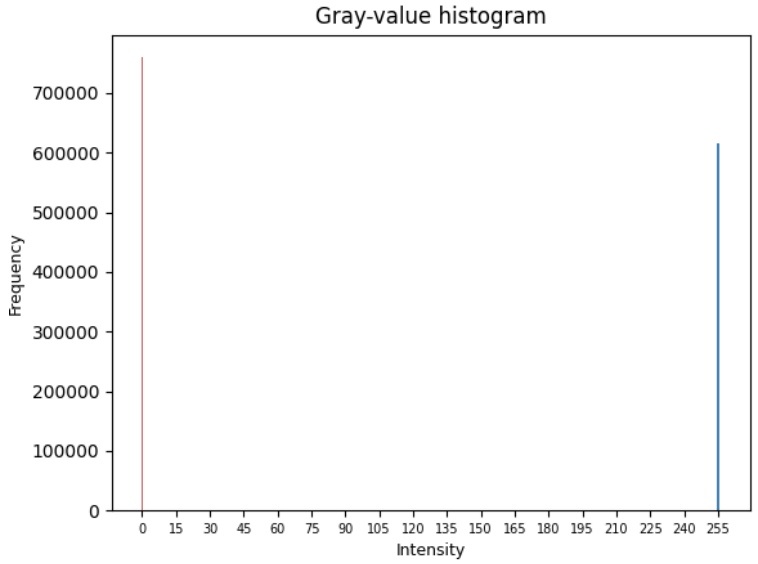


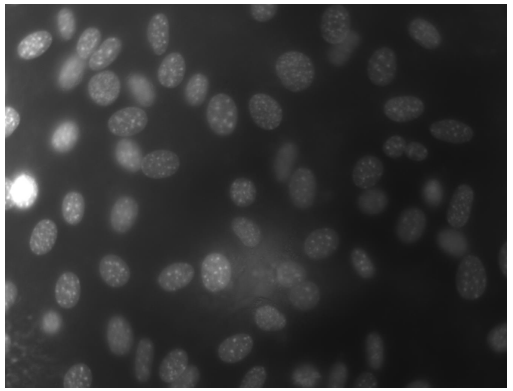
Threshold value k $\in [0,255]$

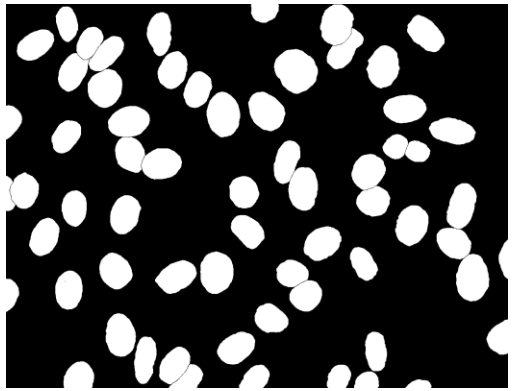
Image clipping



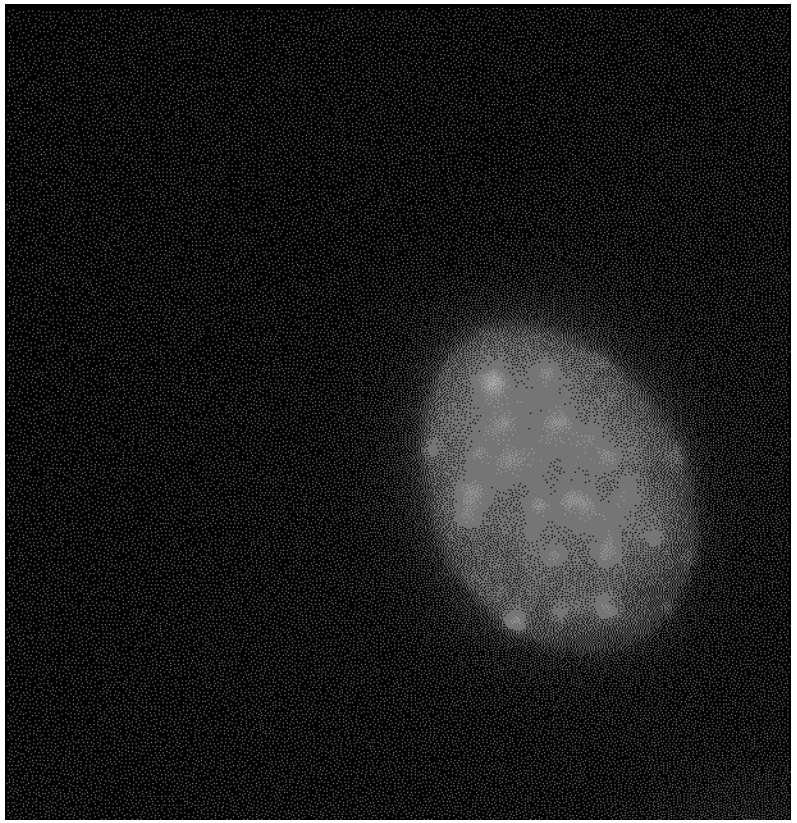
$$g_{clip}(x,y) = \begin{cases} 0 & \text{if } g(x,y) \leq k \\ 255 & \text{if } g(x,y) > k \end{cases}$$



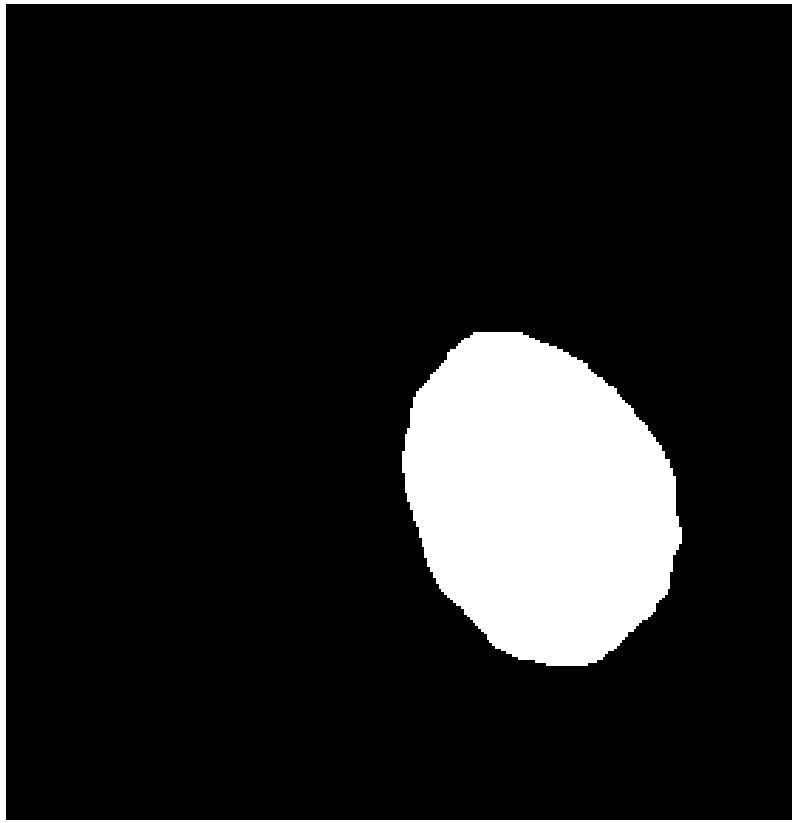




Implementation of the Dice Score



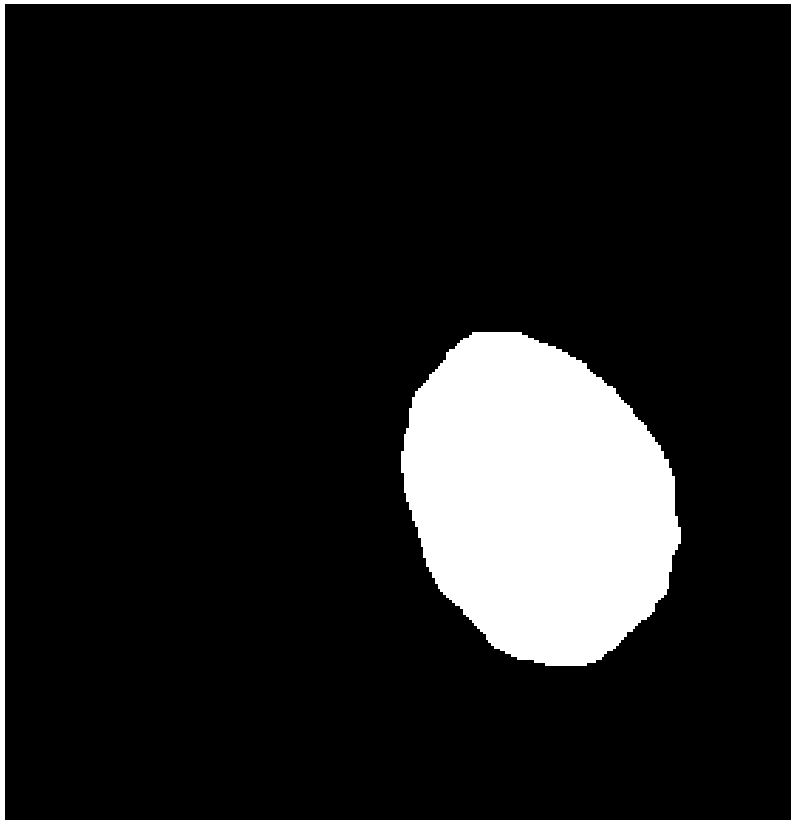
Original image

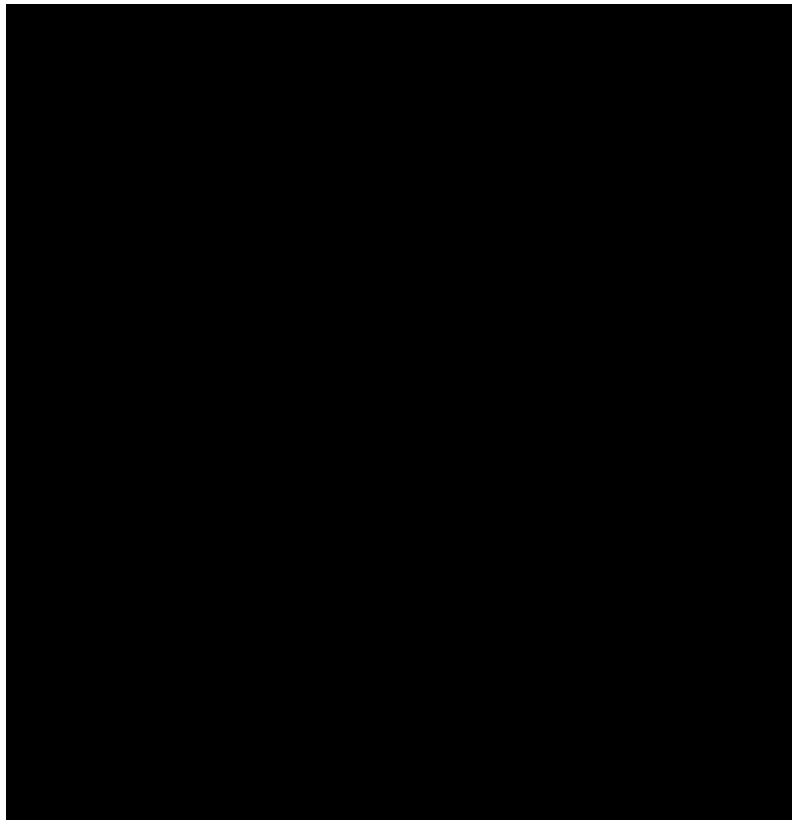


Ground truth

Implementation of the Dice Score

≈80 % accuracy



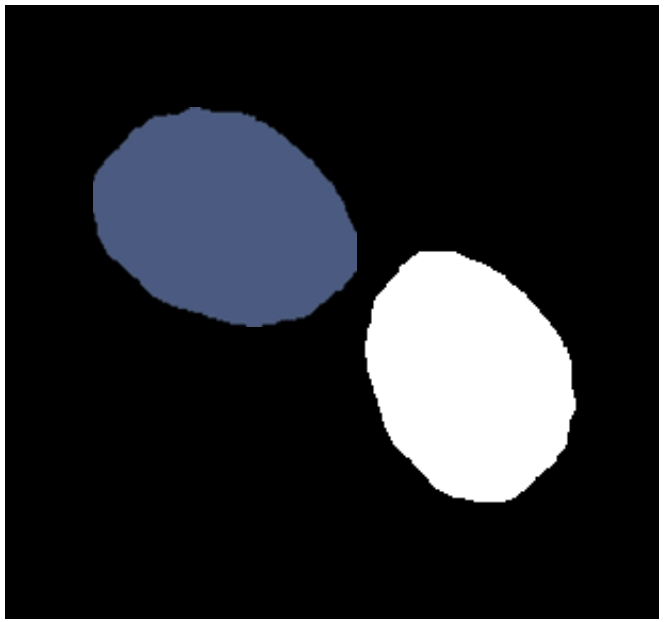


Ground truth

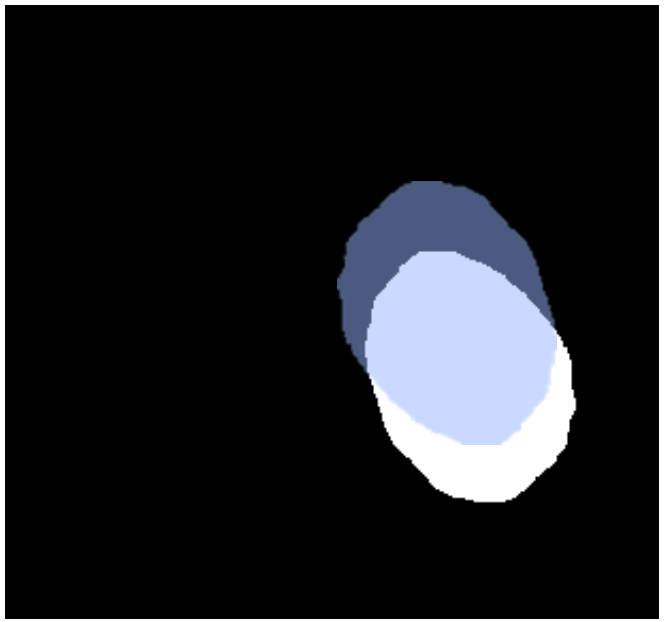
Prediction

Implementation of the Dice Score

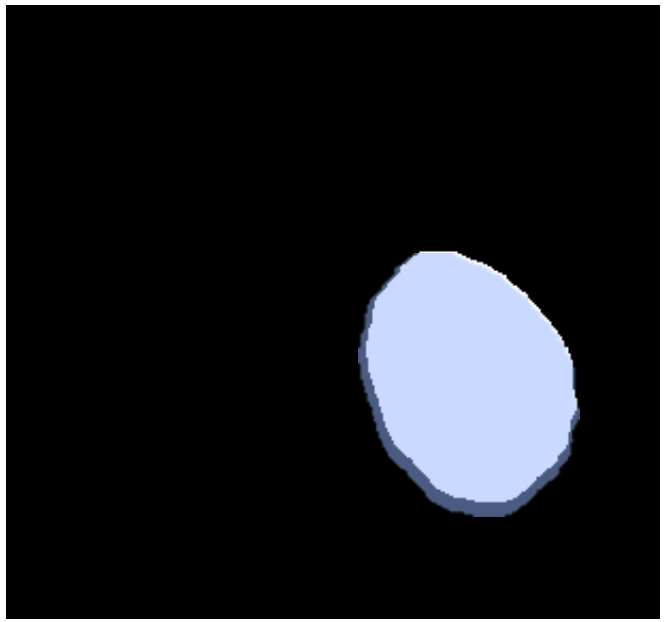
Bad prediction



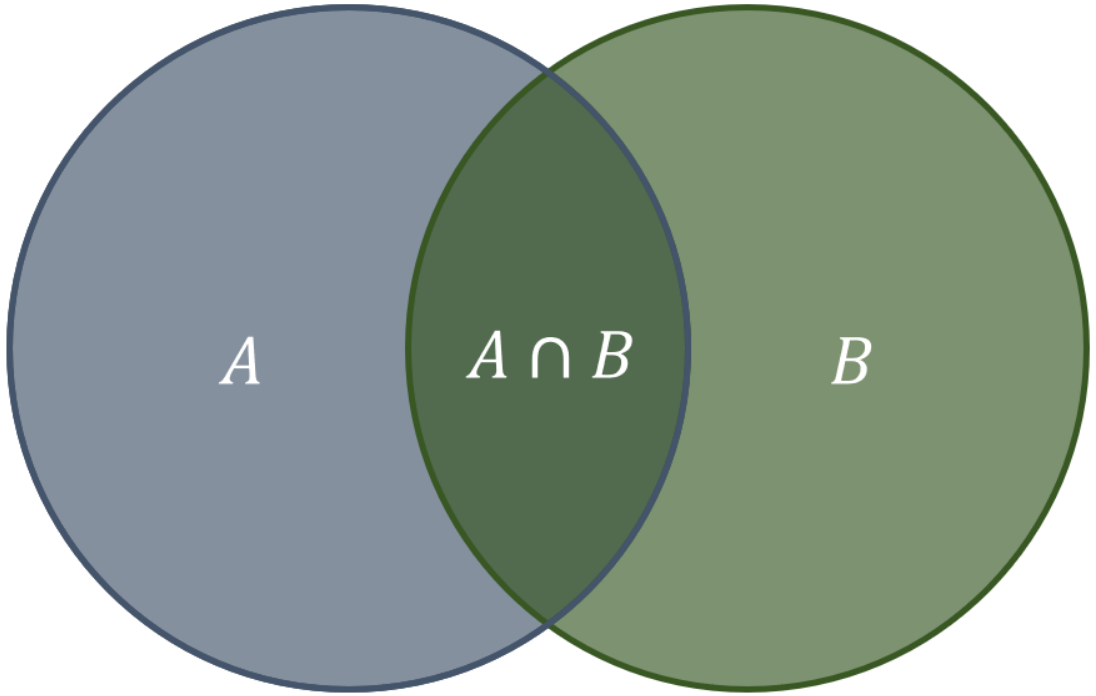
Better prediction



Good prediction



Implementation of the Dice Score

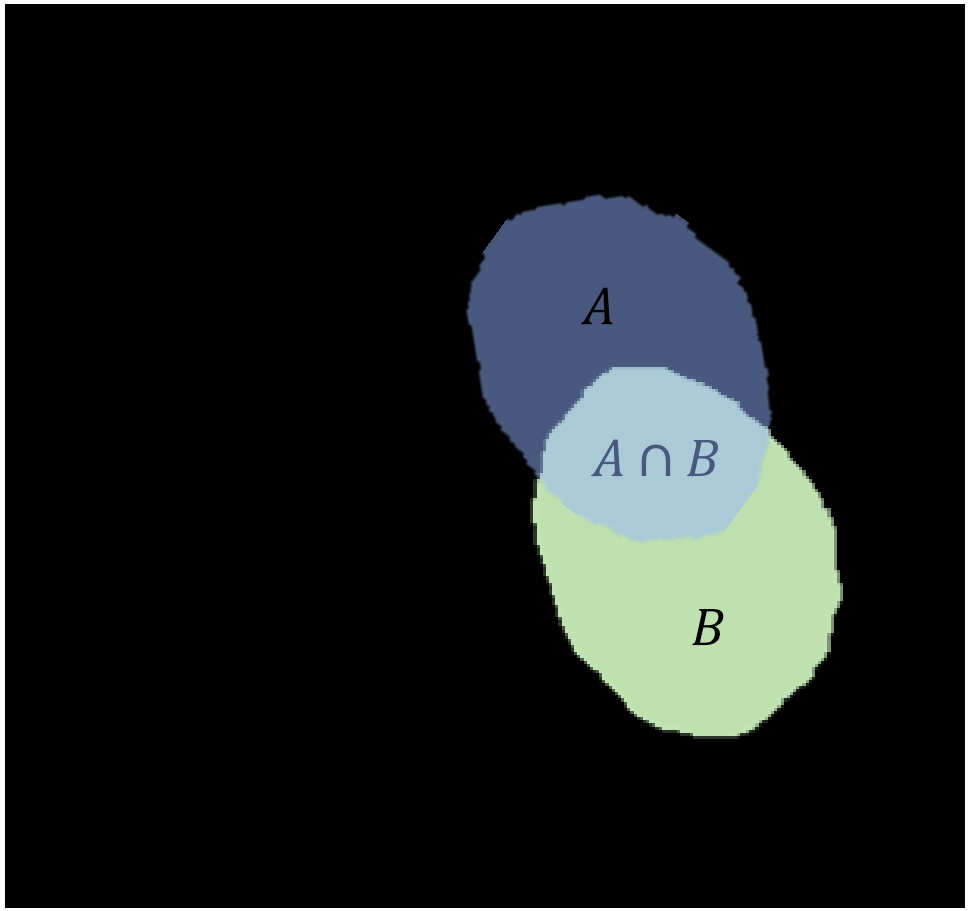


$$DSC = \frac{2 \times |A \cap B|}{|A| + |B|}$$

A: Predicted shape

B: Ground truth

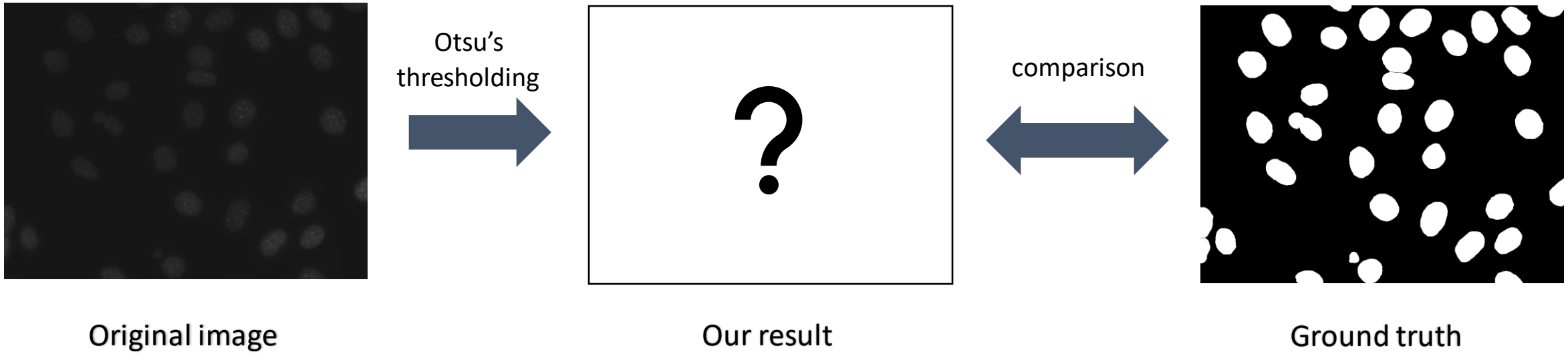
Implementation of the Dice Score



$$DSC = \frac{2 \times |A \cap B|}{|A| + |B|}$$

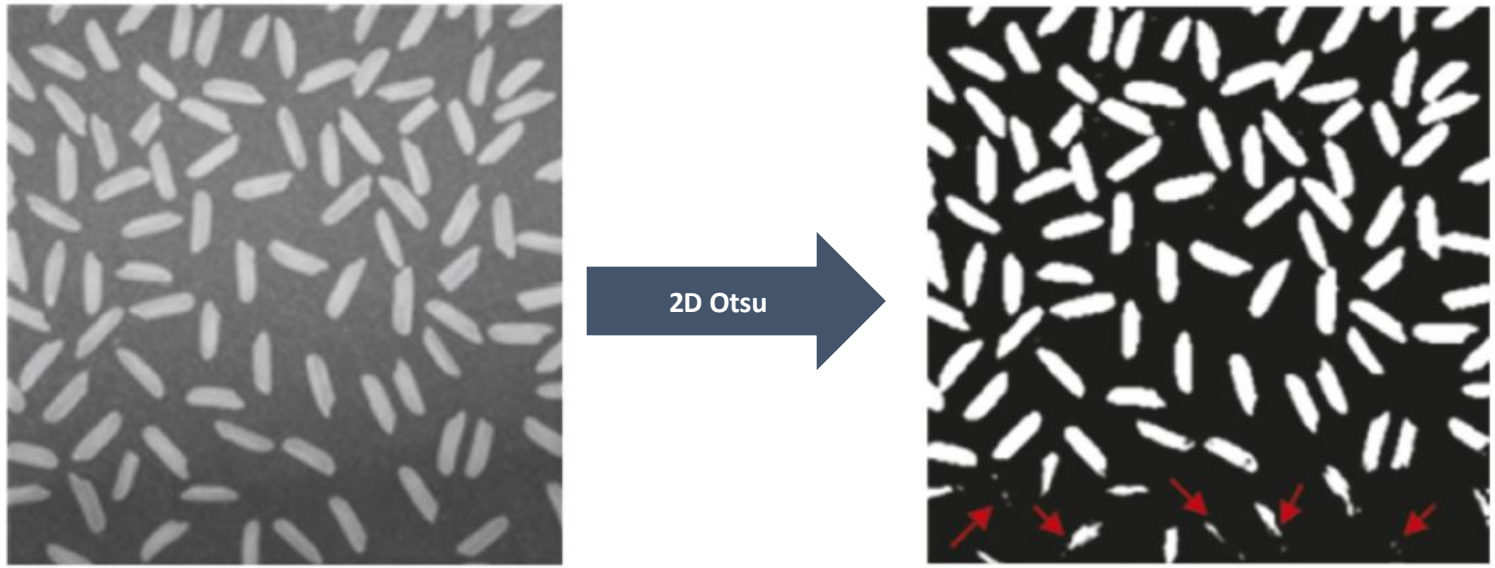
Implementation of the Dice Score

Our goal: compare ground truth images with our results

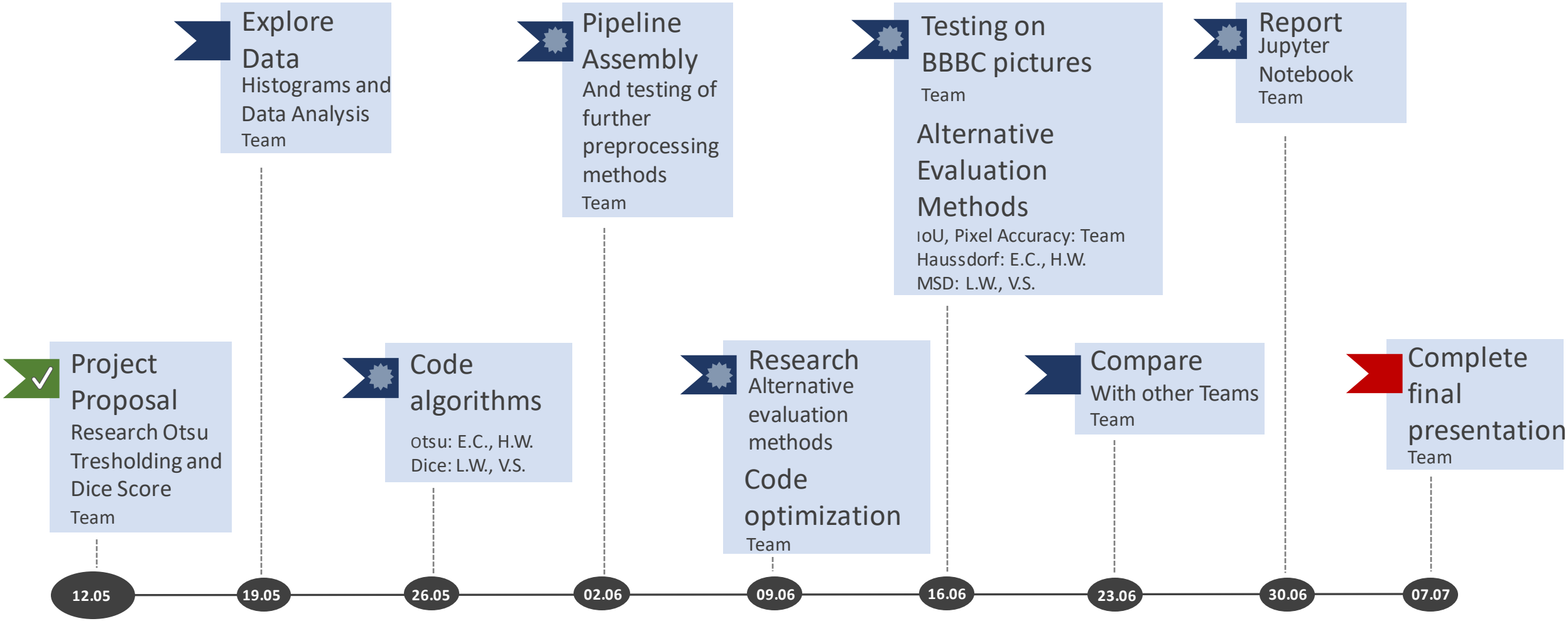


Further ideas

- 2D Otsu
- Algorithm for counting cells



Timeline


Project Proposal
Research Otsu Tresholding and Dice Score
Team
12.05
Explore Data
Histograms and Data Analysis
Team
19.05
Code algorithms
Otsu: E.C., H.W.
Dice: L.W., V.S.
26.05
Pipeline Assembly
And testing of further preprocessing methods
Team
02.06
Research
Alternative evaluation methods
Code optimization
Team
09.06
Testing on BBBC pictures
Team
Alternative Evaluation Methods
IoU, Pixel Accuracy: Team
Haussdorf: E.C., H.W.
MSD: L.W., V.S.
16.06
Compare
With other Teams
Team
23.06
Report
Jupyter Notebook
Team
30.06
Complete final presentation
Team
07.07

Thank you for your attention!

Laura Wächter, Hannah L. Winter, Elizaveta Chernova, Veronika Schuler

Additional slide – Histogram stretching

a = 0, b =255

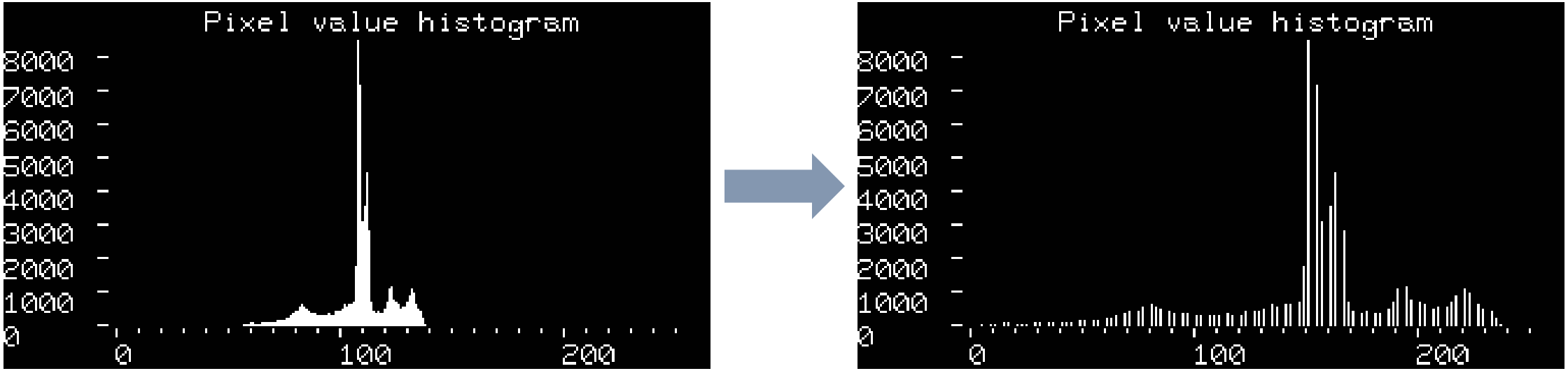
c – lowest pixel intensity in the image

d – highest pixel intensity in the image

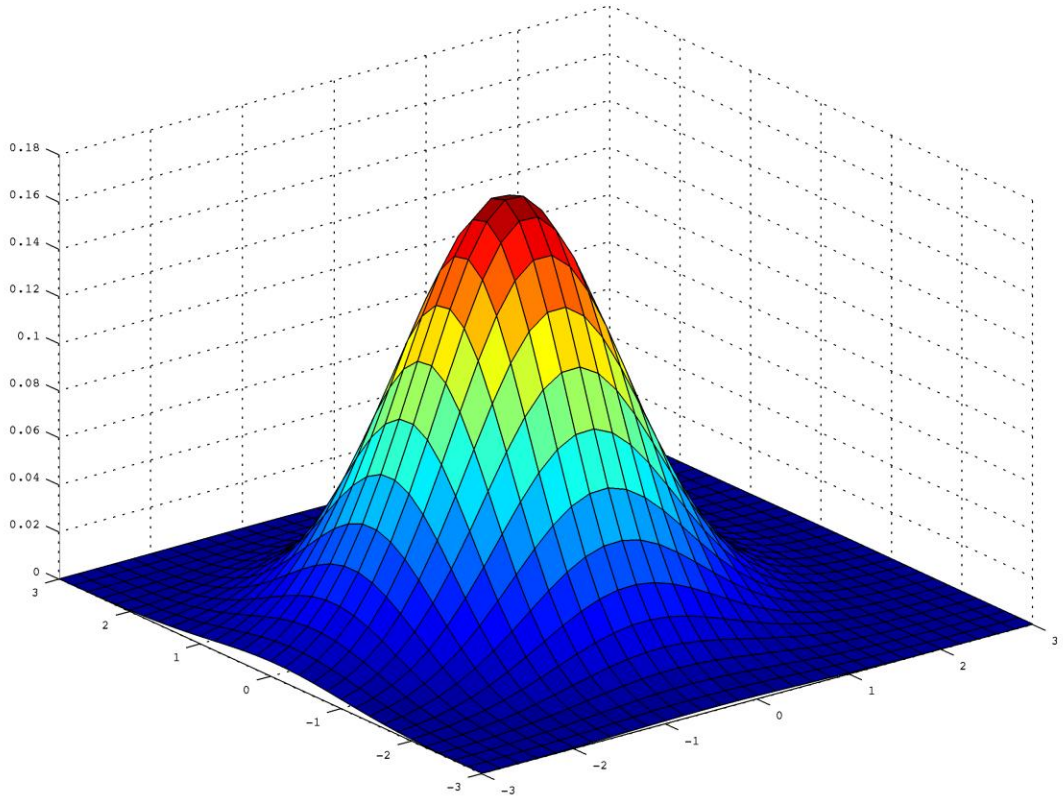
$$P_{out} = (P_{in} - c)\left(\frac{b-a}{d-c}\right) + a$$

https://homepages.inf.ed.ac.uk/rbf/HIPR2/stretch.htm

Additional slide – Histogram stretching



Additional slide – Gaussian filter



$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

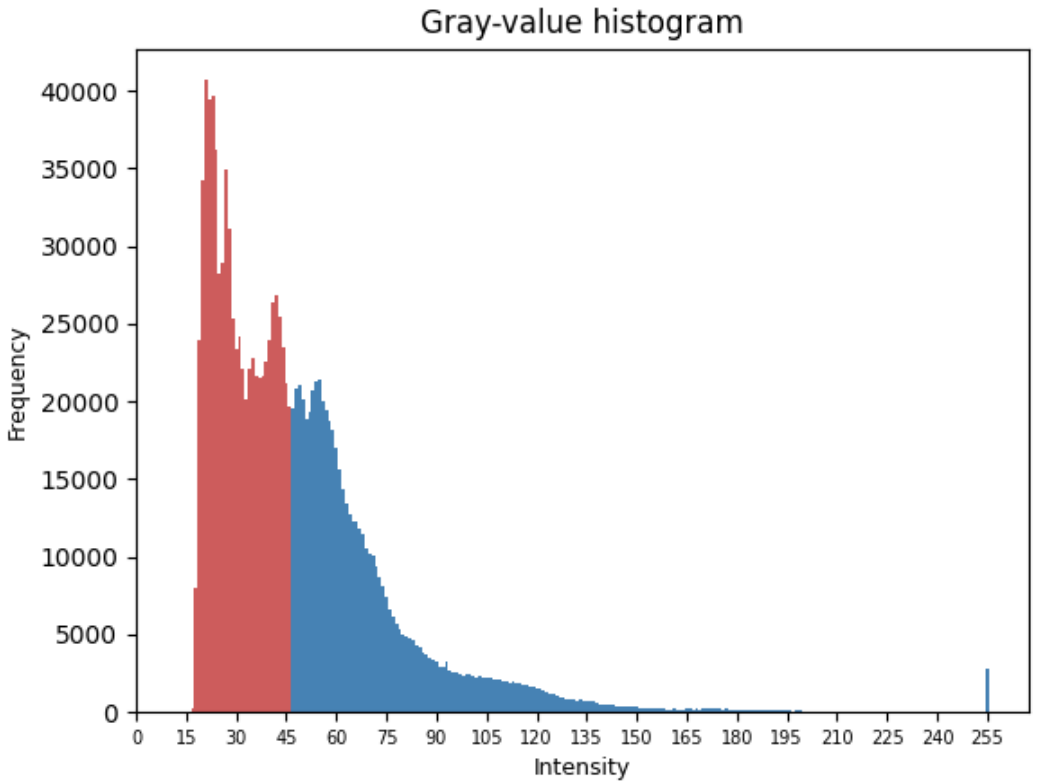
https://stats.stackexchange.com/questions/102632/plot-two-dimensional-gaussian-density-function-in-matlab

Additional slide – Criterion measure

Criterion measure

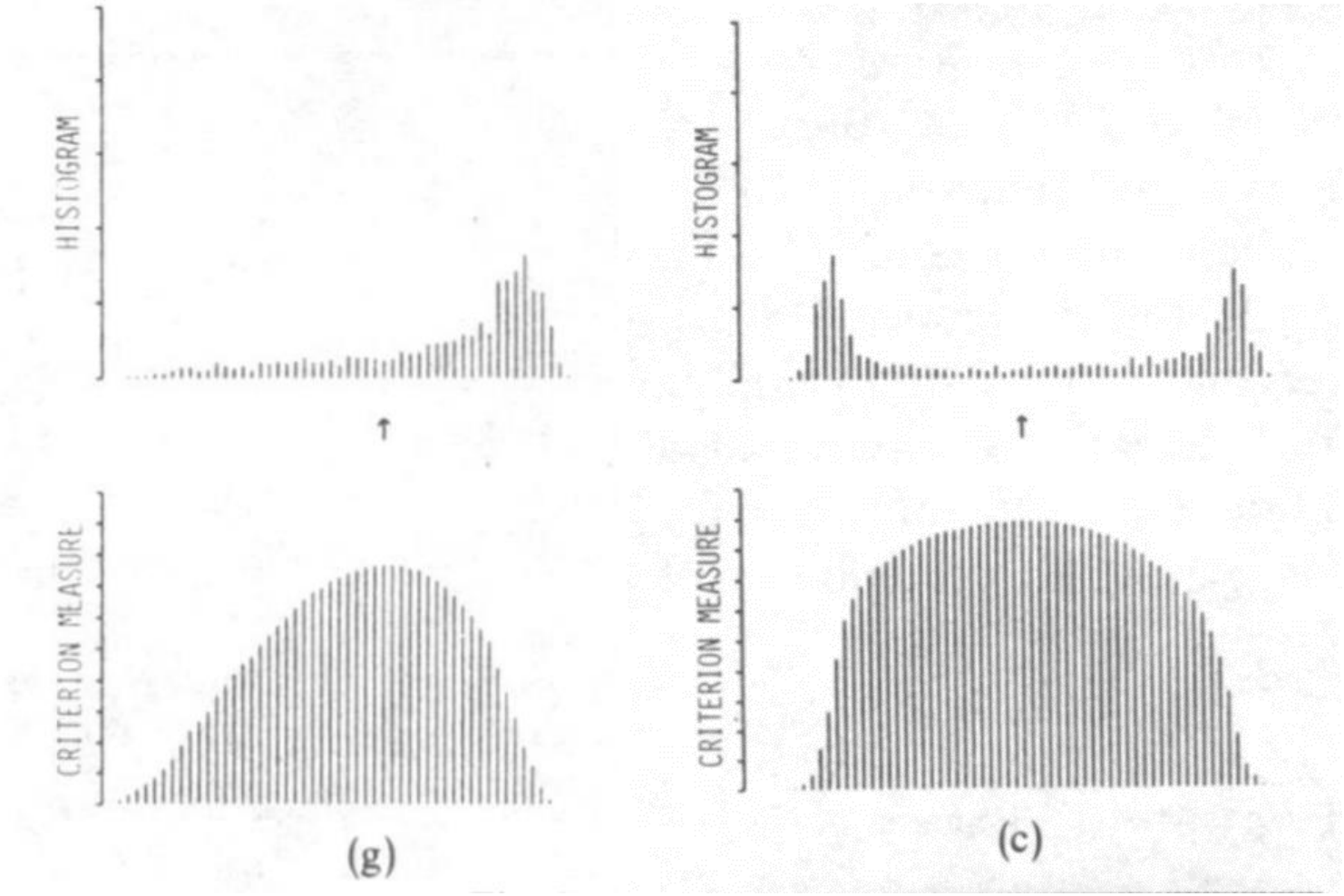
$$\eta(k) = \frac{{\sigma_B}^2(k)}{{\sigma_T}^2}$$

σ_B = between-class variance
σ_T = total variance
$\eta(k) \in [0,1]$



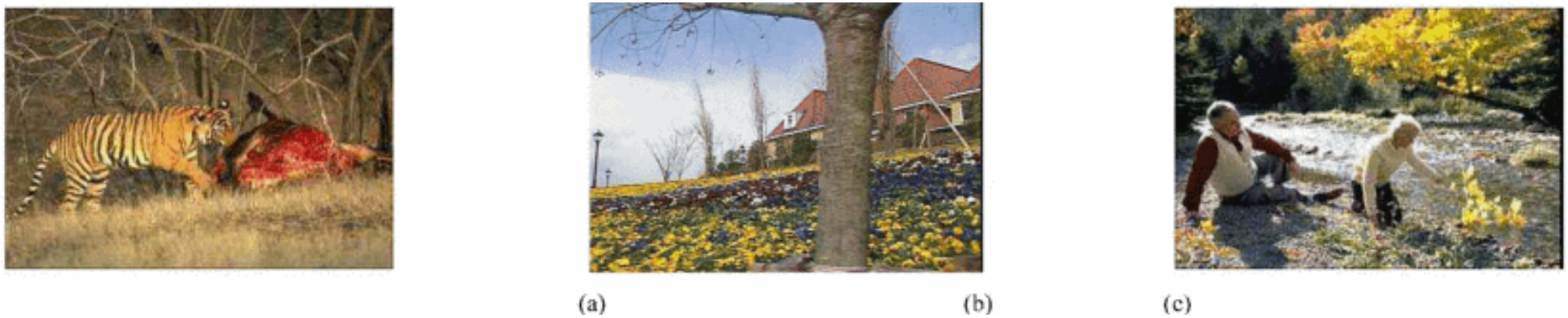
Threshold value $k \in [0,255]$

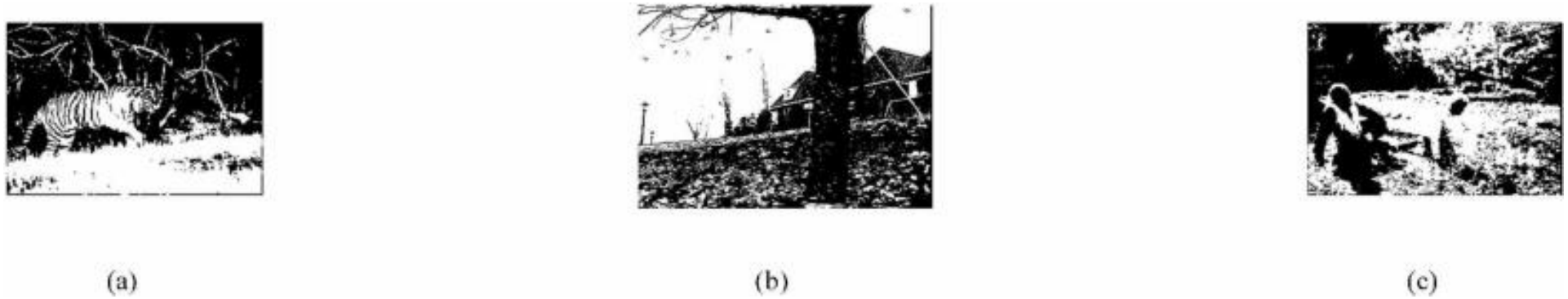
Additional slide – Criterion measure



Otsu, 1979

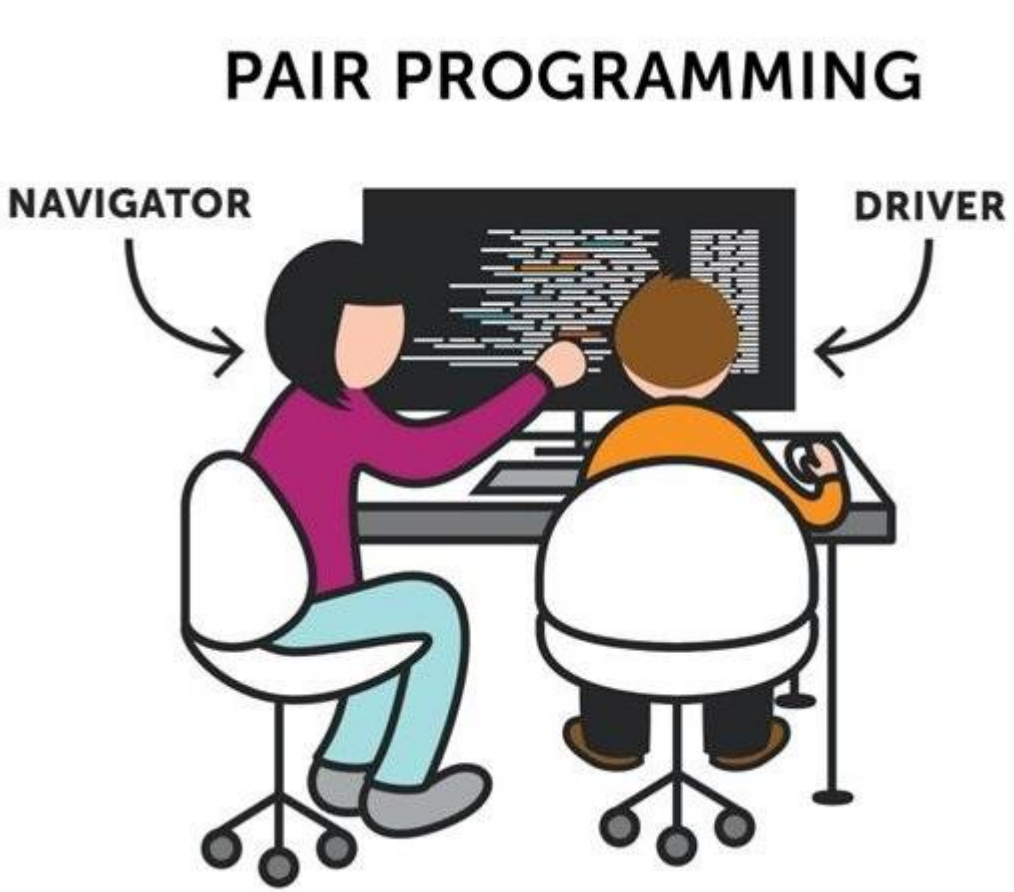
Additional slide – Otsu disadvantages





Not suitable for images with high complexity

Additional slide – Pair Programming



Additional slide – 2D Otsu

Intensity level of pixel is compared with immediate neighborhood pixels

Algorithm:

- For each pixel calculate average gray-level of neighborhood
- Gray level of pixel and average gray levels are divided in L discrete values
- Form pairs: pixel gray level i and neighborhood average j
- There are $L \times L$ possible pairs
- Frequency $f_{i,j}$ of a pair (i, j) divided by the total pixel number N defines probability mass function in a 2D histogram:

$$P_{i,j} = \frac{f_{i,j}}{N} \qquad \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} P_{i,j} = 1$$

Additional slide – IoU

IoU = Intersection-Over-Union

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

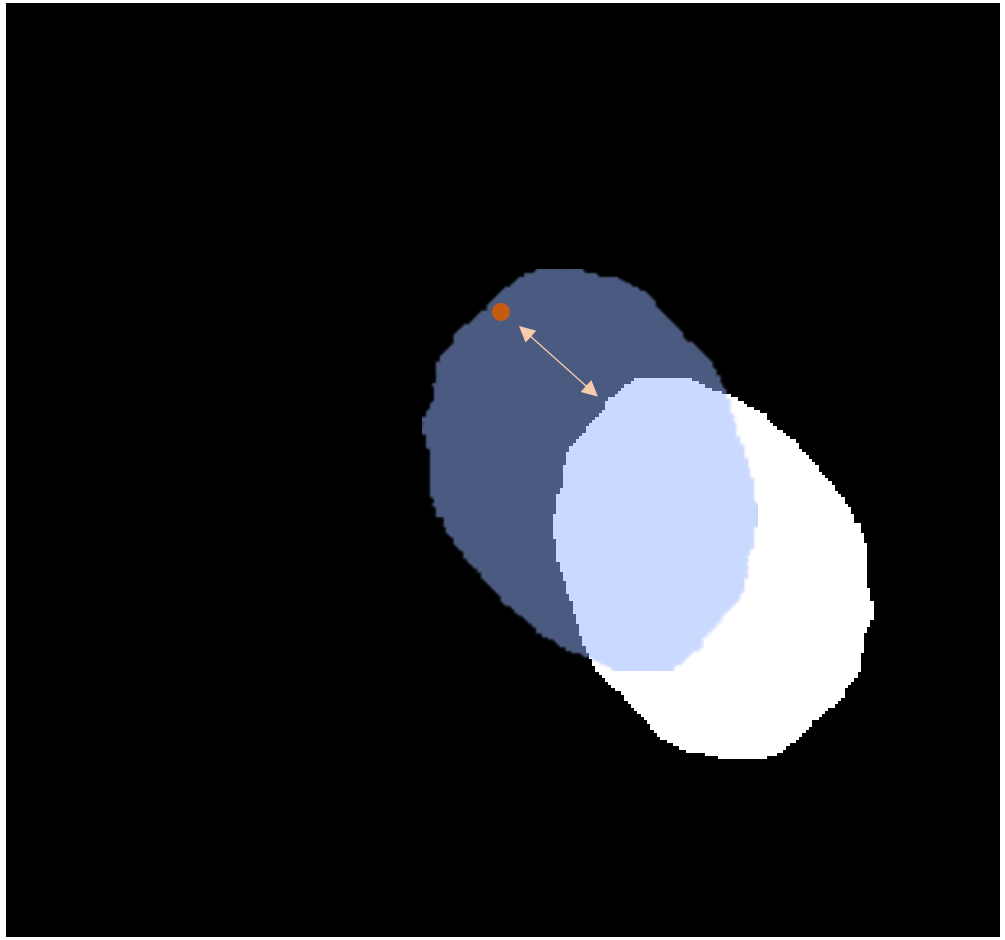
https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/

Additional slide - MSD

MSD = mean surface distance

$$d(p, S') = \min_{p' \in S'} ||p - p'||_2$$

$$\text{MSD} = \frac{1}{n_S + n_{S'}} \left(\sum_{p=1}^{n_S} d(p, S') + \sum_{p'=1}^{n_{S'}} d(p', S) \right)$$



Additional slide – Hausdorff

$$HD=\max[d(S,S'),d(S',S)]$$