Exercise to Visual Computing

- Two sessions, same content:
  - Tuesday, 13:00 – 16:00
  - Thursday, 09:00 – 12:00

- Two parts:
  - Computer Vision: Until mid November (Head TA: Philipp Lindenberger)
  - Computer Graphics: Until Christmas (Head TA: Rafael Wampfler)

- Tuesday morning: Release of execercise

- Thursday evening: Release of solution

- Prerequisites: Python (Computer Vision), C++ (Graphics)

Exercise 1: Today's schedule

- First part(~ 45min):
  - Recap on useful concepts from the lecture
  - Present coding assignment
  - Present sample exam question

- Second part (~ 1h 45min):
  - Individual work on coding and exam question
  - We go around to help you when you need help!

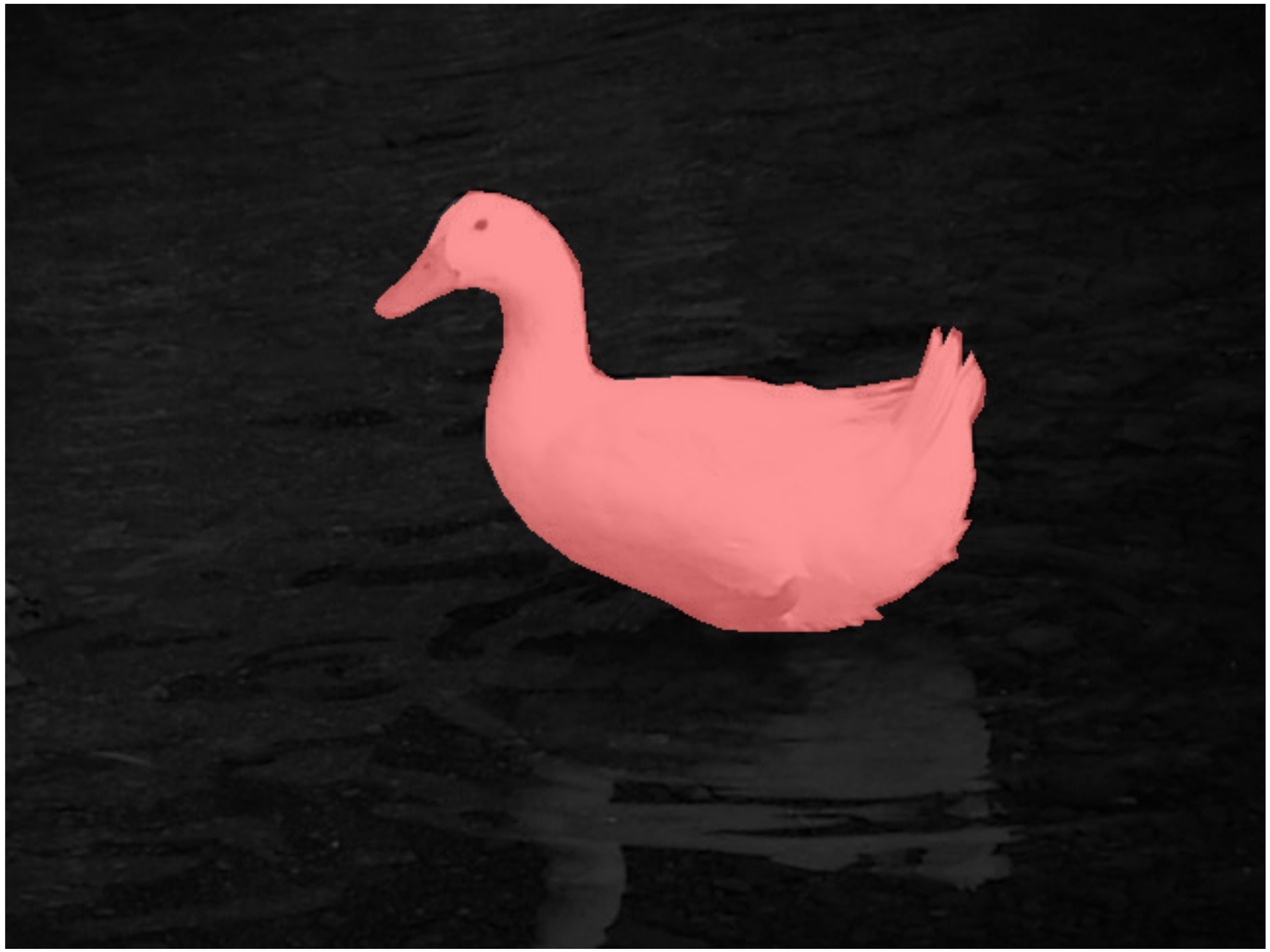
- Third part (~ 15 min):
  - We go over the solutions!

What is a digital image?

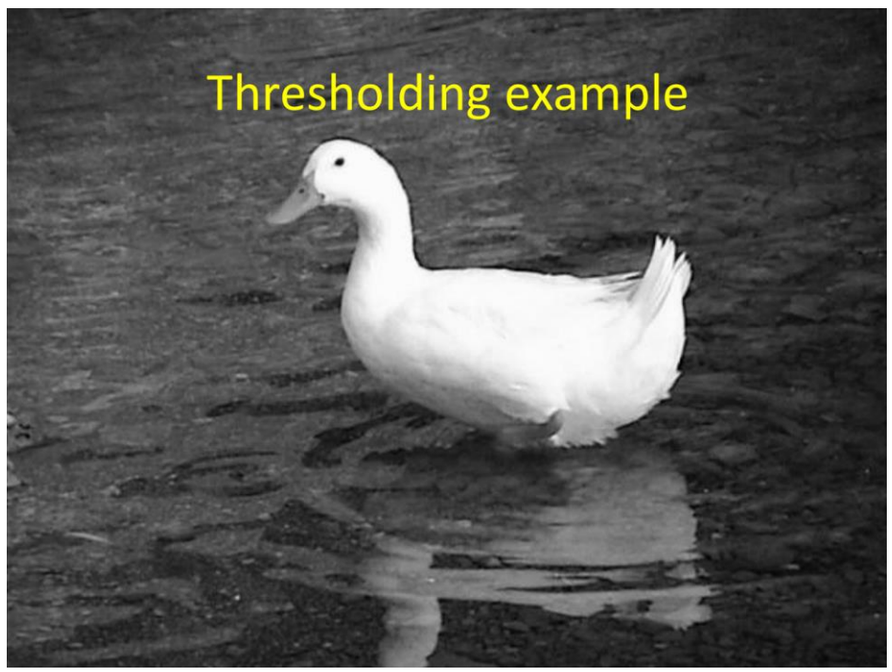
- Image is a Tensor: [H x W x 3]
- Each pixel is a rgb vector [0 ... 255] * 3 (“color channels”)
- Coordinate system (x/y flipped):

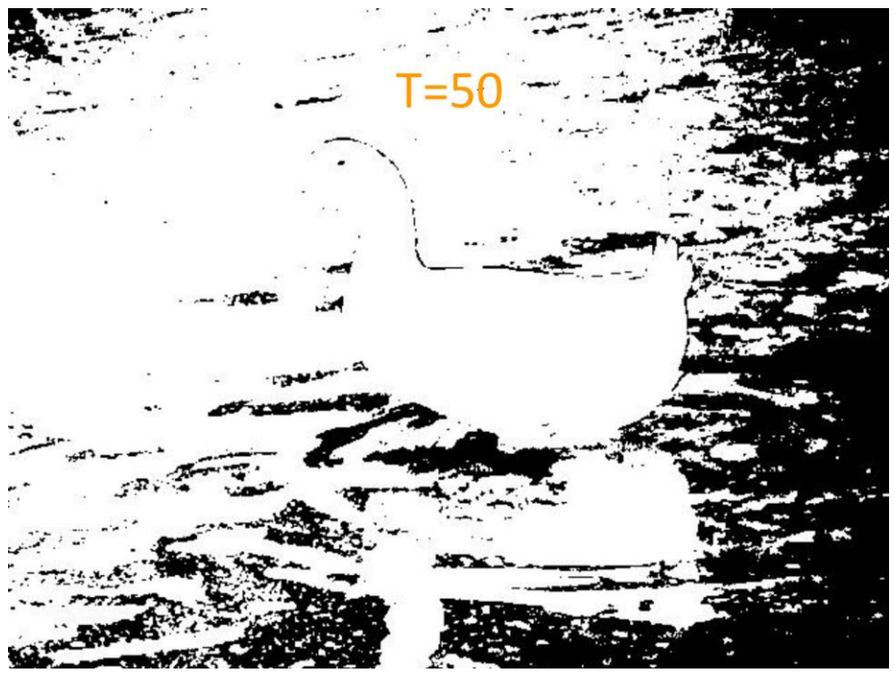
Tutorial 1 – Segmentation

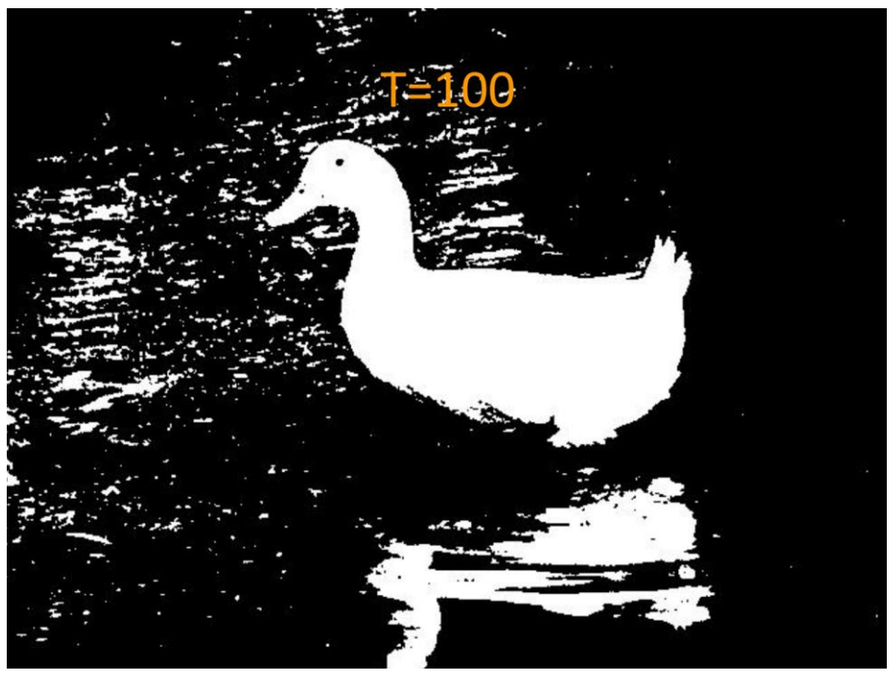
Background Removal

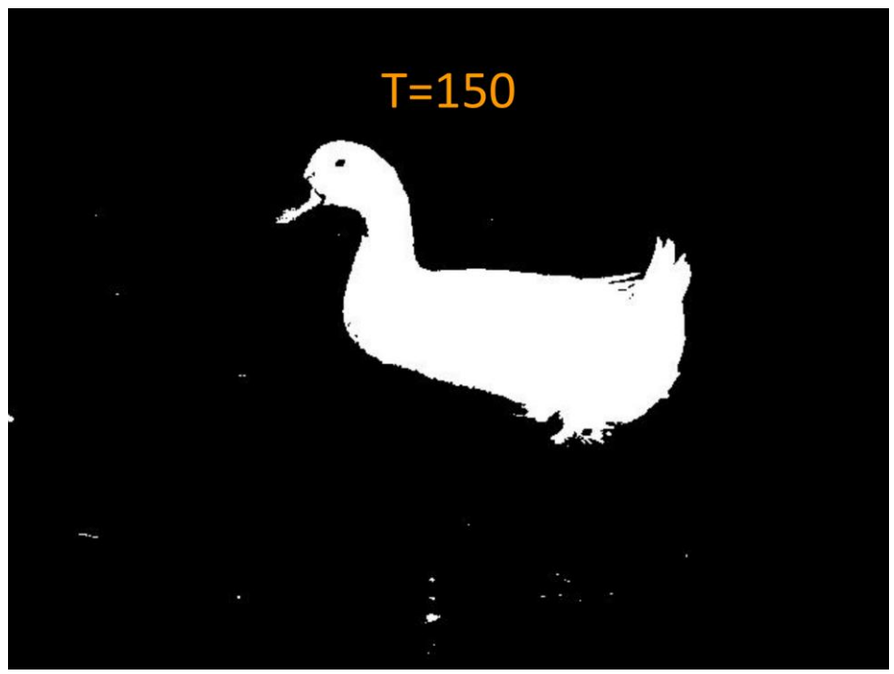


Thresholding

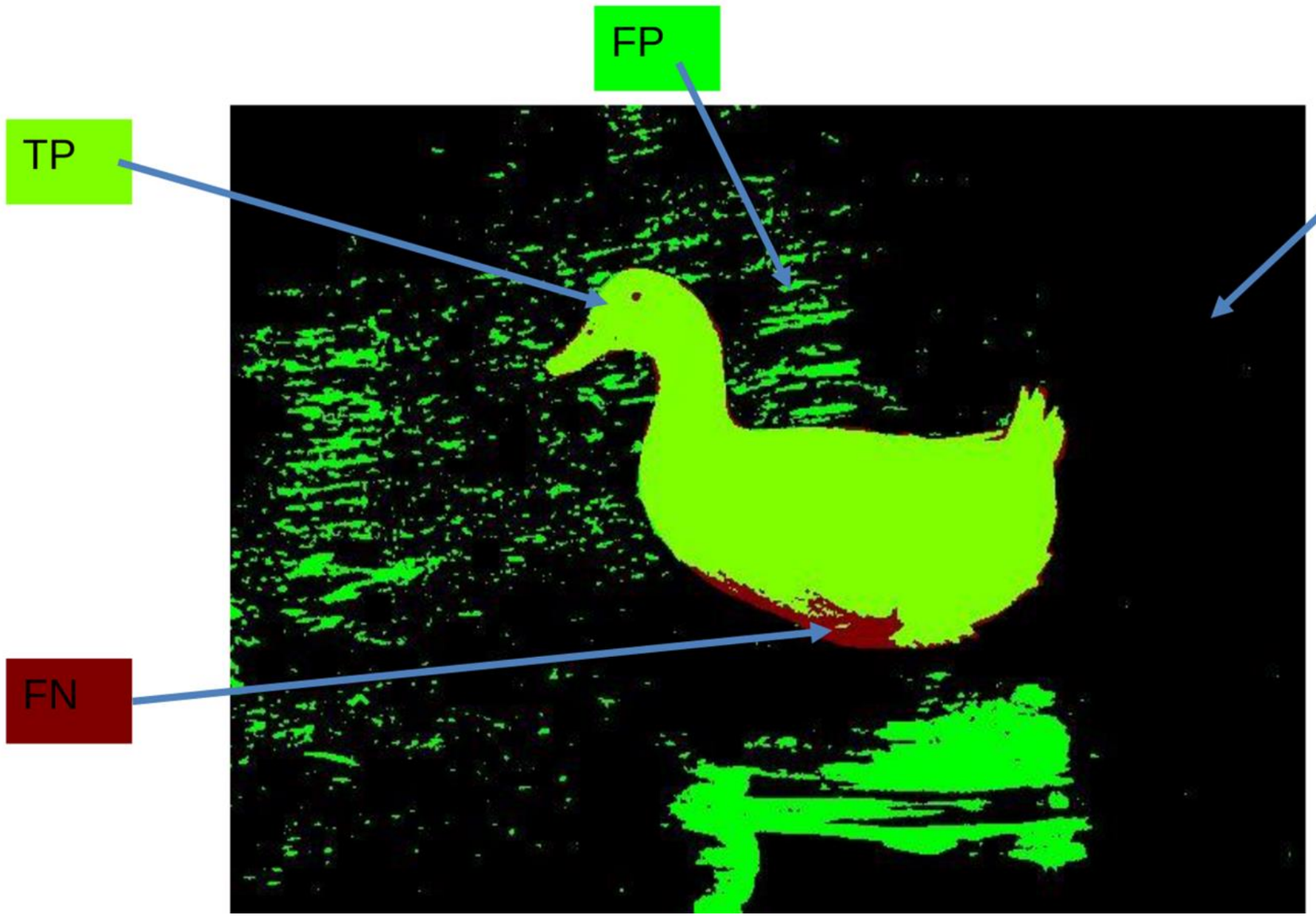


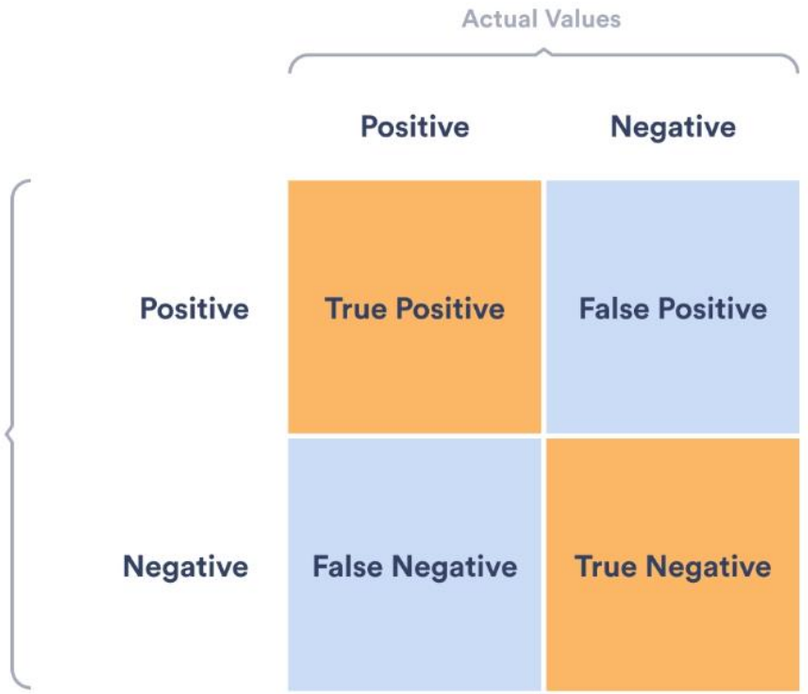






The Confusion Matrix



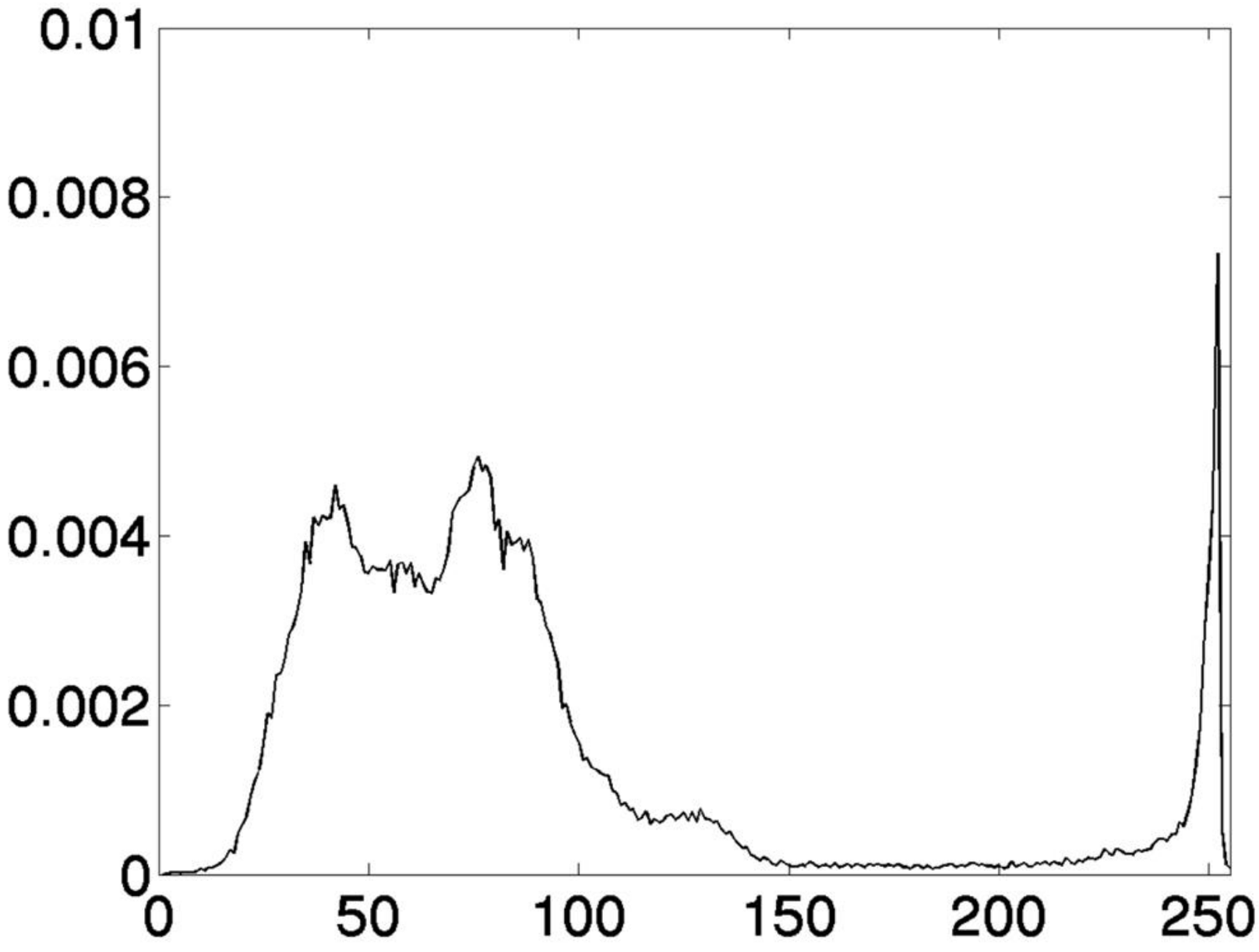


The ROC curve

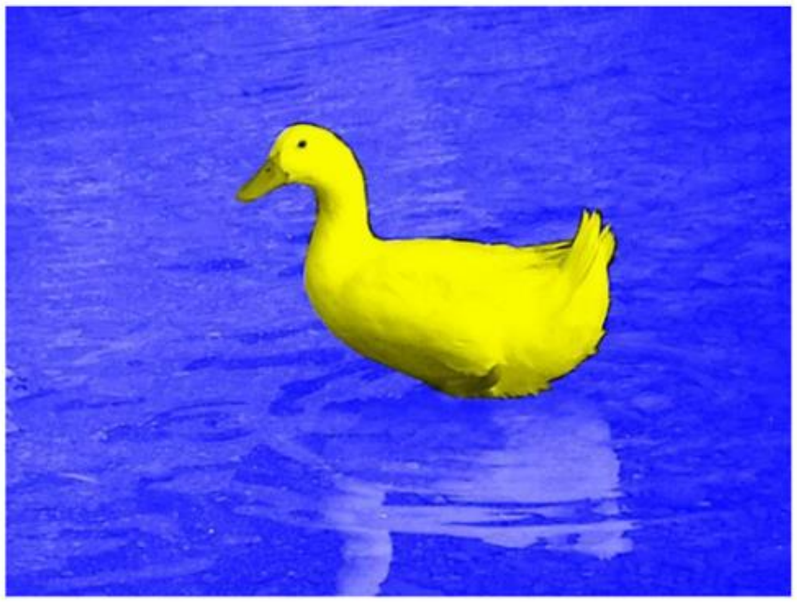
$$\frac{\text{True positive count}}{P = TP + FN}$$

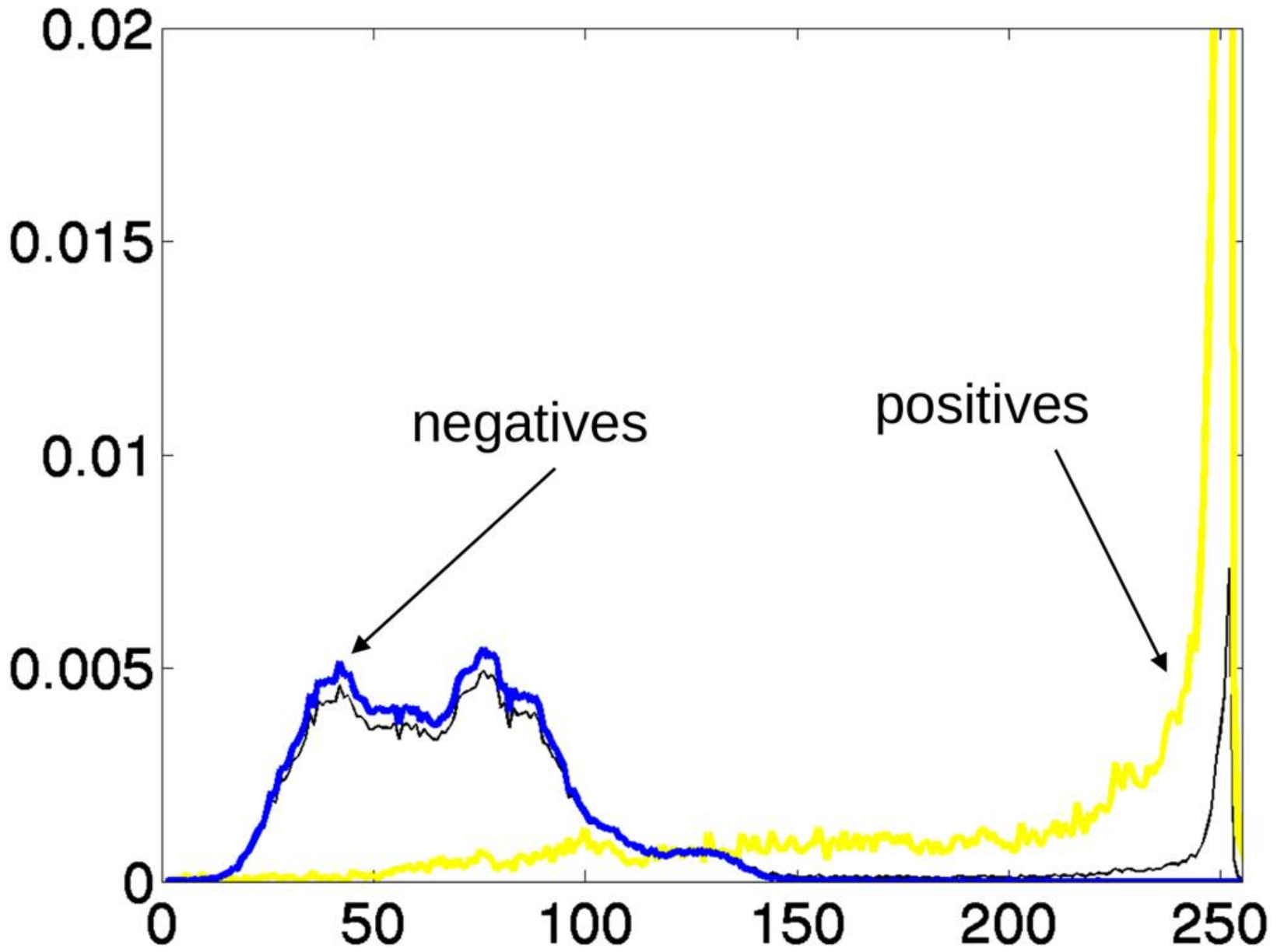
$$\frac{\text{False positive count}}{N = FP + TN}$$

Graylevel histograms



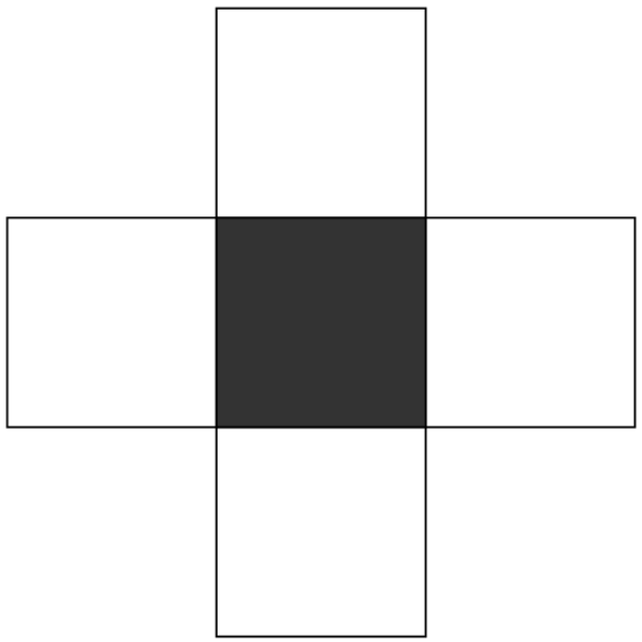
Graylevel histograms



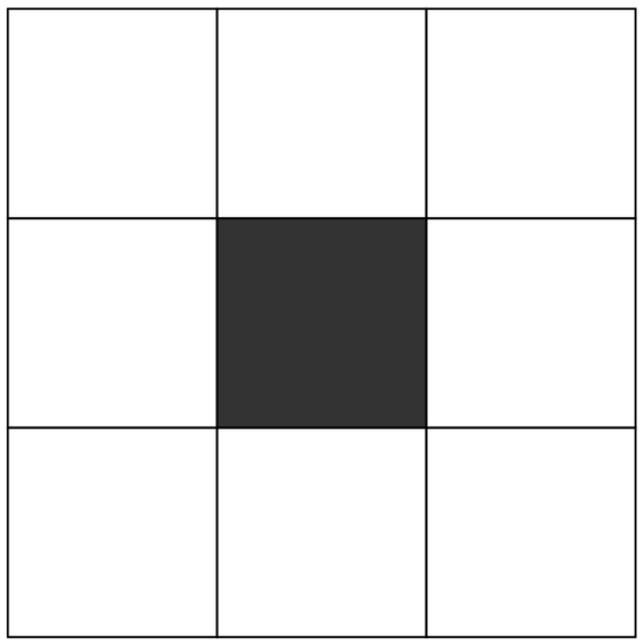


Region growing

- Start from a seed point
- Add neighboring pixels that share some properties.
- Iterate with the newly added pixels.



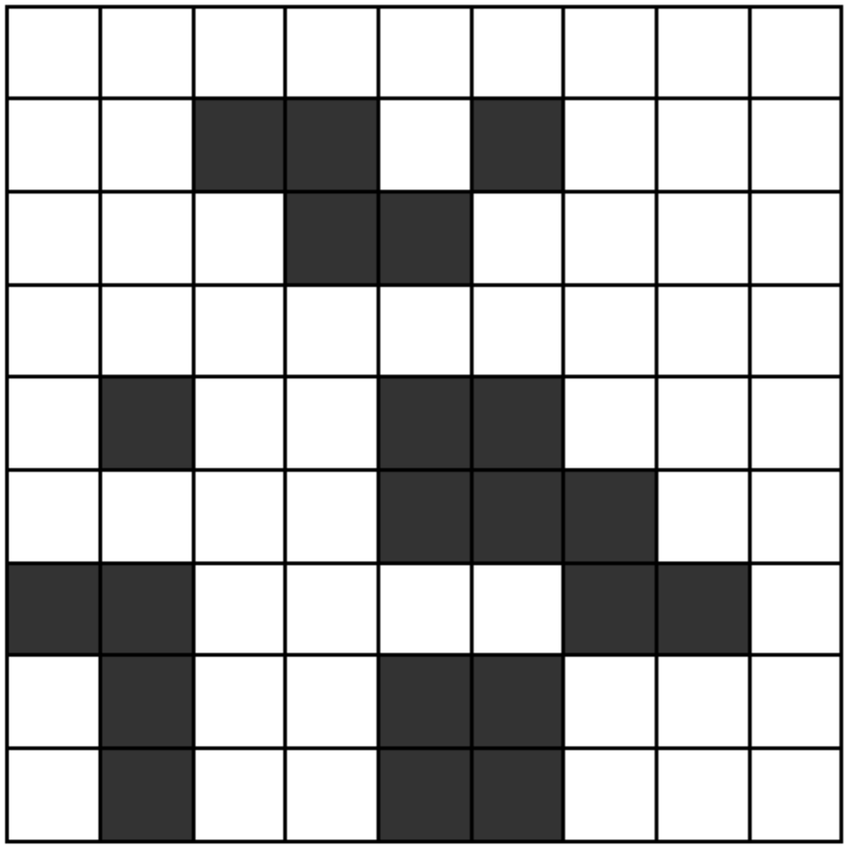
4-neighborhood



8-neighborhood

Connected regions

- Labels each connected component of a binary image with a separate number.



1	1	1	1	1	1	1	1	1
1	1	2	2	1	3	1	1	1
1	1	1	2	2	1	1	1	1
1	1	1	1	1	1	1	1	1
1	4	1	1	5	5	1	1	1
1	1	1	1	5	5	5	1	1
6	6	1	1	1	1	5	5	1
7	6	1	1	8	8	1	1	1
7	6	1	1	8	8	1	1	1

Exercise 1 – Background Subtraction

Bluescreen / Greenscreen

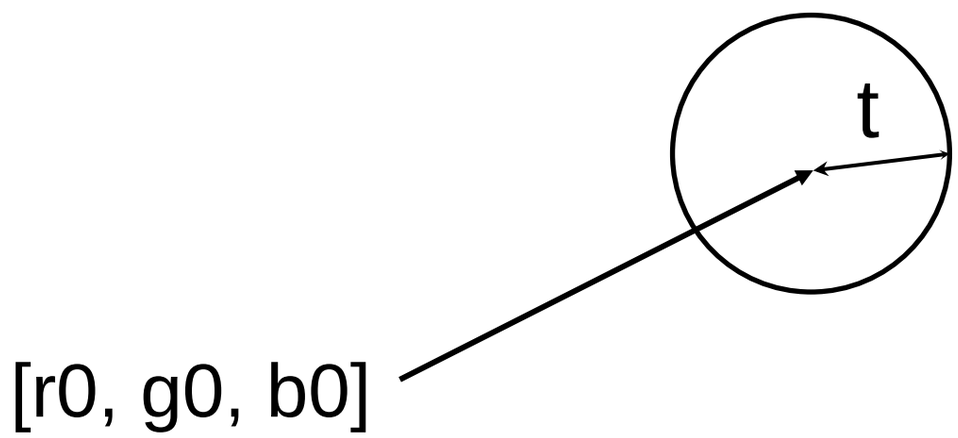
http://www.iwatchstuff.com/images/2006/01/superman-greenscreen.jpg

http://www.theavclub.tv/behind_the_scenes/greenscreen/

Bluescreen

- Represent background with a single color value
  - Classification based on absolute distances

$$|[r, g, b] - [r0, g0, b0]| < t.$$



Bluescreen

- Represent background with a set of color values
  - Classify new RGB values based on Mahalanobis distance

$$(\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) > t$$

- Covariance Matrix

$$\Sigma_{ij} = E\left[(X_i - \mu_i)(X_j - \mu_j)\right]$$

- Estimation from n data points

$$\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \overline{x})(x_i - \overline{x})^T$$

Bluescreen

Pixel-wise Color Model

- Mean and covariance for each pixel
- One threshold for all pixels (Mahalanobis distance)

Pixel-wise Color Model

Setting up the environment

Two options:

- GitHub + jupyter notebooks run locally

https://github.com/tavisualcomputing/viscomp2023

- Google Colab: Python notebook in the cloud

https://colab.research.google.com/github/tavisualcomputing/viscomp2023/blob/main/Exercises/W2/W2_exercise.ipynb