Friedrich-Alexander-Universität
Technische Fakultät

Winter 2024

Computer Vision Project

Vincent Christlein, Thomas Gorges, Mathias Seuret, Thomas Köhler, Mathias Zinnen

Pattern Recognition Lab (CS5)
14th October 2024

Today‘s Outline

(1) Organization of this project

(2) Motivation

(3) What is Computer Vision?

(4) **Lecture 1**: Image Processing for Grayscale and Distance Images

Organization
Project Structure and Content

Organization – Grades and Certificates

Project for Computer Science and Medical Engineering

Option 1 (5 ECTS)	Option 2 (10 ECTS)	Option 3 (10 ECTS)
Academic Lab i. Lectures ii. Exercises (group)	Academic & Research Lab i. Lectures ii. Exercises (group) iii. Additional exercises (individual)	Academic & Research Lab i. Lectures ii. Exercises (group) iii. Research project (individual)

Grade for Hochschulpraktikum, if necessary, based on short talk (about a state-of-the-art research paper)

The research project is mostly related to one of the discussed topics

Forschungspraktikum only: Contact us or any other PhD student at the lab for a research project. Feel free to come up with your own idea or participate in a suitable challenge. (Overview: https://lme.tf.fau.de/teaching/thesis/)

Organization – Exercises

- Exercises are coded in **Python**
- You should work in **groups of two** (individual is also okay if really want that)

- Tasks are complementary to the lecture of the previous week. It is mandatory to **complete all of them**
  - You can work from **home** or in the **CIP pool**
  - It is strongly recommended to begin with the programming tasks before the exercise!

- Submissions are to be uploaded to StudOn and presented to the respective lecturer
- All tasks have to be finished by their individual deadlines (more later)

Organization – Exercises

- Prepare questions or problems **before** the exercise

- **Own virtual hardware**
  Virtual CIP-Pools: https://remote.cip.cs.fau.de/ (Xpra)

- You can ask additional questions via **StudOn Forum** (make use of this option!)

Organization – Lectures and Exercises

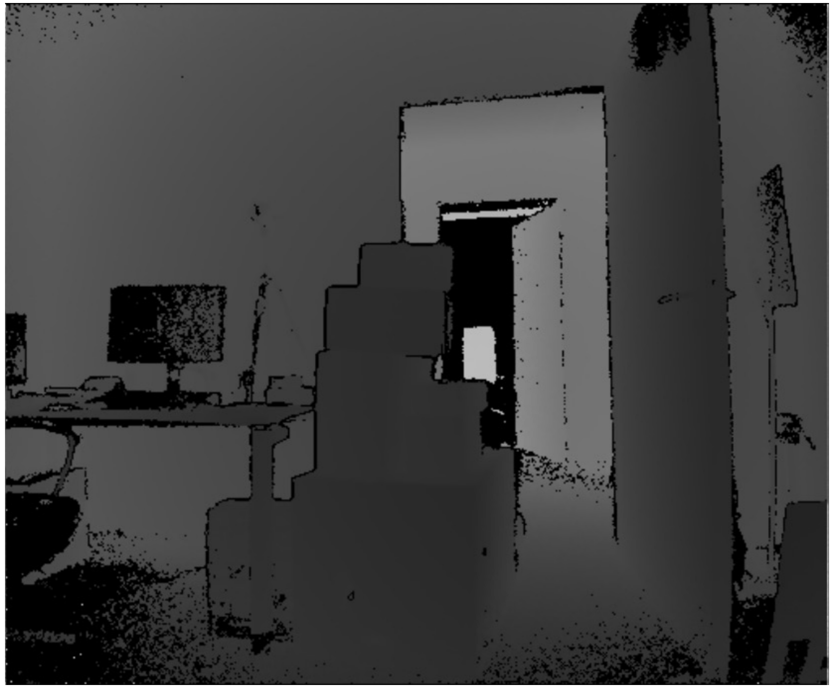
Preliminary schedule

Date	Lecture (In person)	Exercise (Online)
14.10.	Lecture 1 – Thomas Gorges	TODAY
21.10.	---	Ex. 1
28.10.	Lecture 2 – Mathias Seuret	**Deadline Ex. 1**
04.11.	---	Ex. 2
11.11.	---	Ex. 2
18.11.	Lecture 3 – Vincent Christlein	**Deadline Ex. 2**
25.11.	---	Ex. 3
02.12.	---	Ex. 3
09.12.	Lecture 4 – Thomas Köhler	**Deadline Ex. 3**
16.12.	---	Ex. 4
13.01.	Lecture 5 – Mathias Zinnen	**Deadline Ex. 4**
20.01.	---	Ex. 5
27.01.	---	**Deadline Ex. 5**
17.02.	---	**Deadline Individual Ex.**

- All exercise deadlines are at **2 pm**
- Each project team has to **present their exercise submission** to the respective lecturer (separate meeting in the week of the deadline)

Organization - Topics

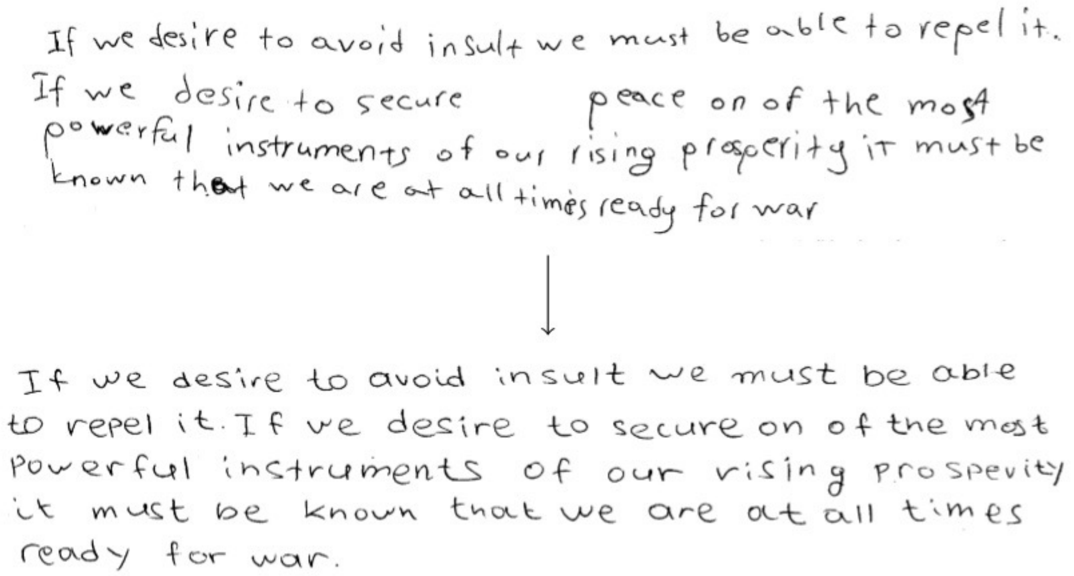
- Basic image processing of grayscale and distance images
- Writer Identification/Retrieval
- Color image reconstruction via demosaicing
- Face Recognition
- Image Recognition for Humanities



2D distance image

Organization - Topics

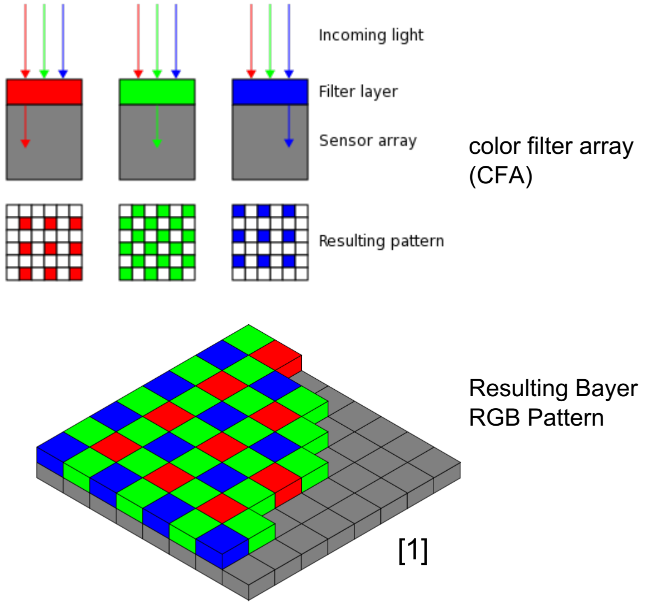
- Basic image processing of grayscale and distance images
- Writer Identification/Retrieval
- Color image reconstruction via demosaicing
- Face Recognition
- Image Recognition for Humanities

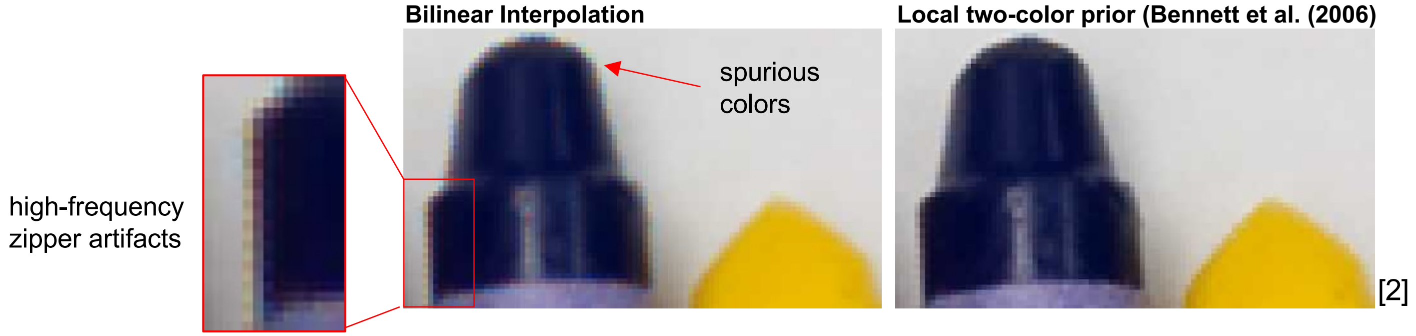


Writer identification/retrieval system

Organization - Topics

- Basic image processing of grayscale and distance images
- Writer Identification/Retrieval
- Color image reconstruction via demosaicing
- Face Recognition
- Image Recognition for Humanities





[1] https://rawpedia.rawtherapee.com/Demosaicing
[2] http://szeliski.org/Book

Organization - Topics

- Basic image processing of grayscale and distance images
- Writer Identification/Retrieval
- Color image reconstruction via demosaicing
- Face Recognition
- Image Recognition for Humanities

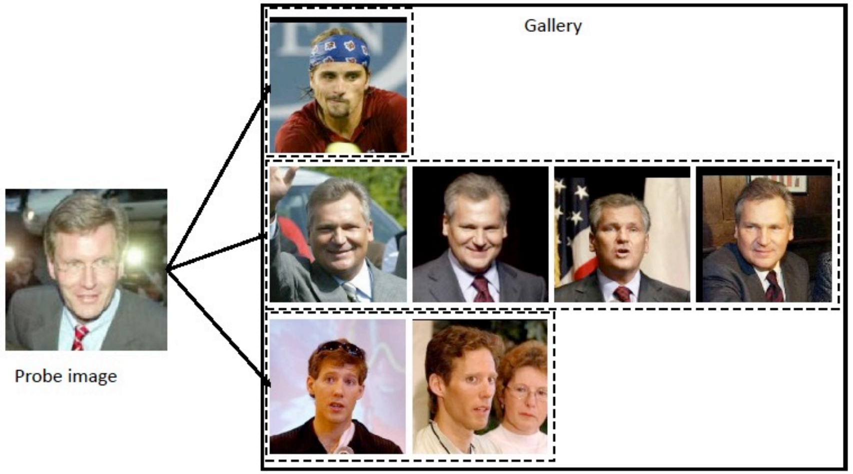


Figure: Supervised learning (face identification)

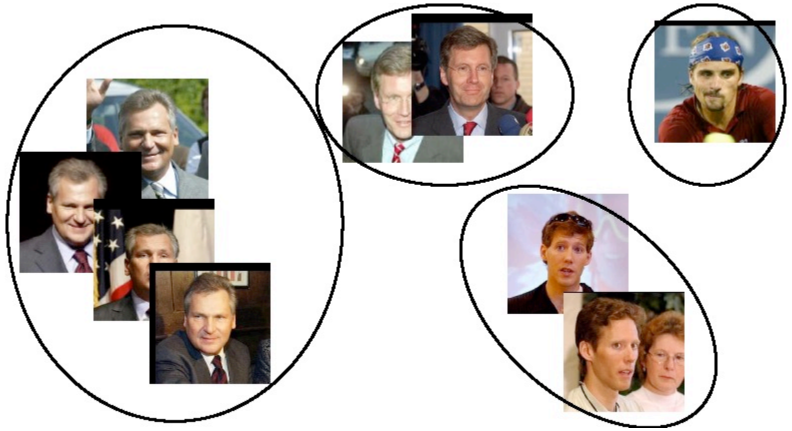


Figure: Unsupervised learning (face clustering)

Organization - Topics

- Basic image processing of grayscale and distance images
- Writer Identification/Retrieval
- Color image reconstruction via demosaicing
- Face Recognition
- Image Recognition for Humanities

Organization
Questions?

Motivation
Applications of Computer Vision

Example 1: Collision Avoidance

→ Use cameras to observe a human/machine environment.

→ Avoid collisions by fusing camera and robot position information.

Example 2: Autonomous Driving

→ Use **radar/lidar/camera sensors** to capture the surrounding.

→ Utilize computer vision techniques to **semantically analyze the scene** and potentially make autonomous decisions.

Motivation: Hand Tracking

→ Detect salient keypoints on the hand anatomy.

→ Track individual keypoints across video frames.

Computer Vision Primer

Computer Vision is about extracting information from images

- **RGB**
- Grayscale
- Multispectral
- 2.5D and 3D

Computer Vision Primer

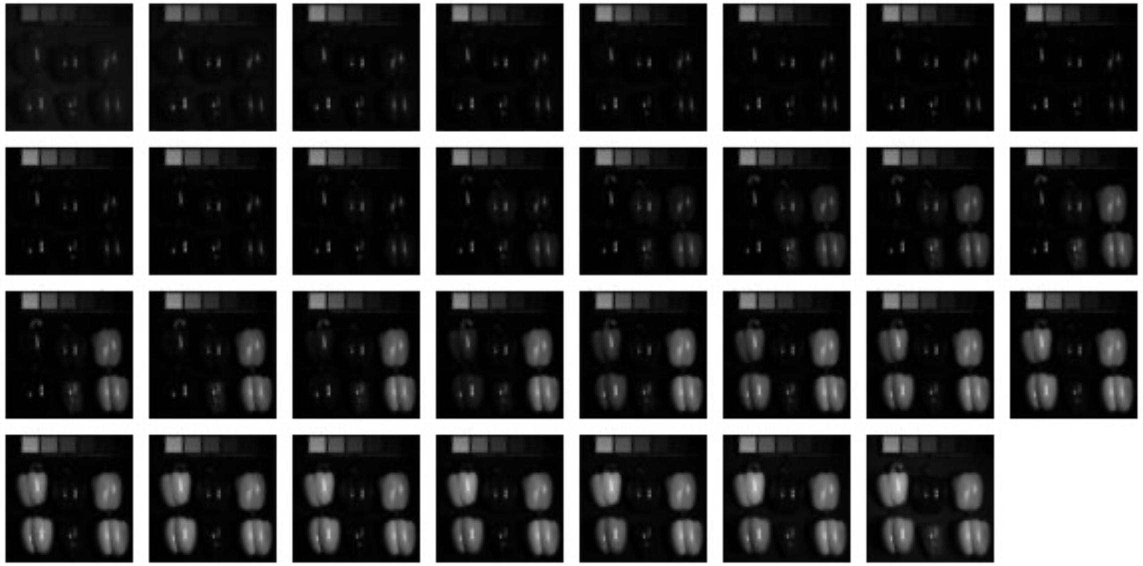
Computer Vision is about extracting information from images

- RGB
- **Grayscale**
- Multispectral
- 2.5D and 3D

Computer Vision Primer

Computer Vision is about extracting information from images

- RGB
- Grayscale
- **Multispectral**
- 2.5D and 3D



Computer Vision Primer

Computer Vision is about extracting information from images

- RGB
- Grayscale
- Multispectral
- **2.5D and 3D**

Dataset by Johannes Schauer, Andreas Nüchter from the University of Würzburg, Germany

Computer Vision Primer

Computer Vision is about extracting information from images
→ What kind of information?

Content	object detection, face identification
Text	sign detection, text extraction
Movement	tracking, motion capturing
Geometry	multi-camera setups, shape extraction, SLAM
Medical Imaging	segmentation, tumor detection
Quality Control	find production errors

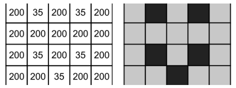
...

Lecture 1

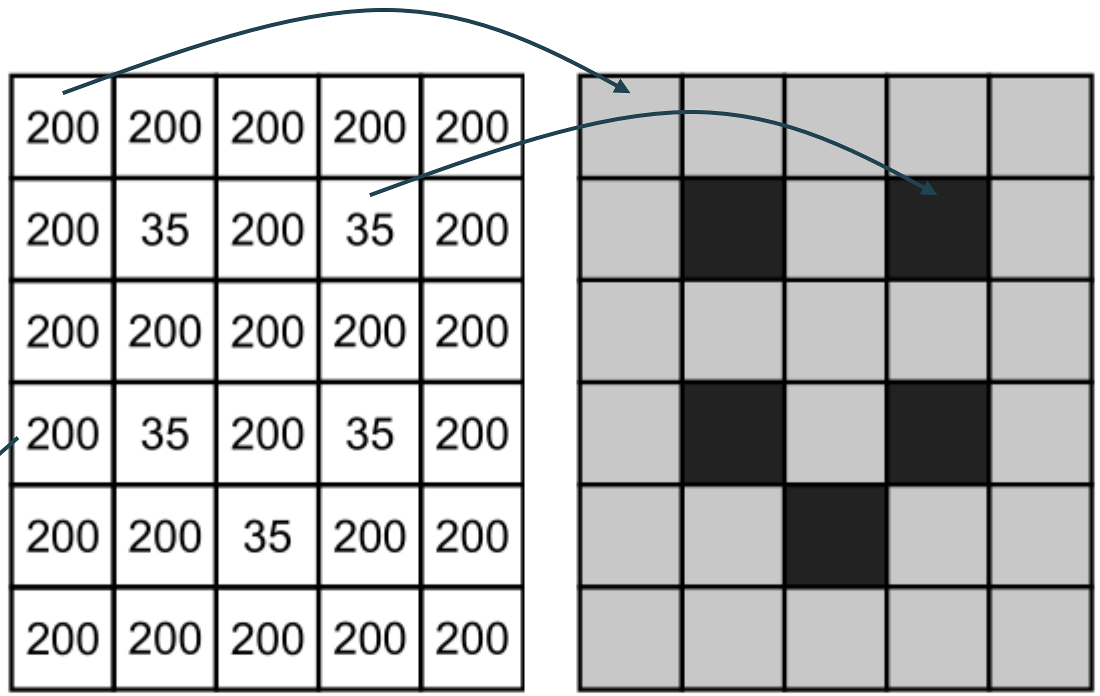
Processing on Grayscale and Distance Images

Image Formation

Images



Images can be represented as 2D arrays of a certain type, e.g., byte for grayscale images.



Gray scale image (ubyte/int8)

Num. of bits	Min. value	Max. value
8 bit	0	255
16 bit	0	65535
32 bit	0	4294967295
64 bit	0	2^64 - 1

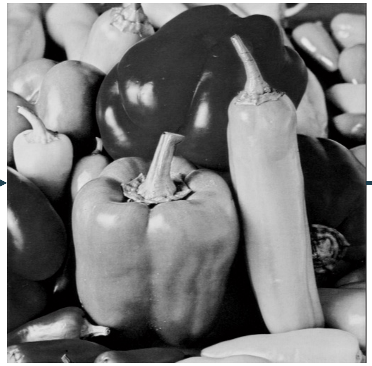
* Excluding floating point numbers

(A) Thresholding

Thresholding Simple operation that is used to separate regions of different „colors“

Returns a mask array of the same size as the original image

Input



Isolated channel: **Green**

Threshold
Value at $(x, y) > t$

Example t = 150

Resulting mask

Apply mask: Green peppers

(A) Thresholding on Distance Images

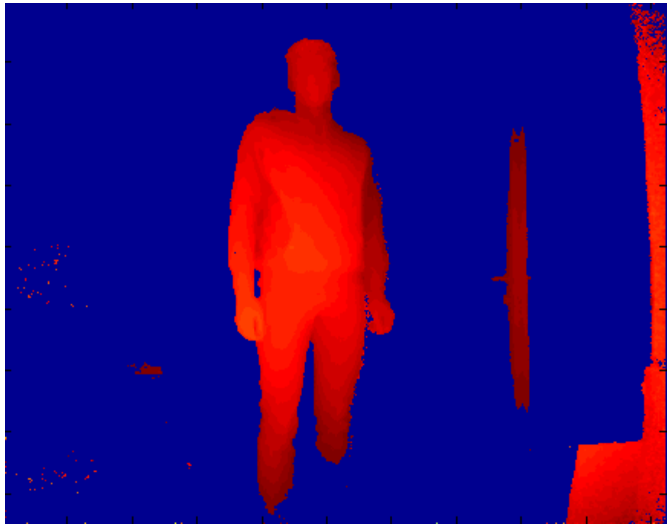
Pixels contain distance information. „Color" encodes the distance.

Example: On distance images thresholding can be used to separate fore- from background

Distance image with person

Threshold
Value at $(x, y) < t$

Example t = 1.85m



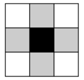
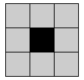
Scene foreground

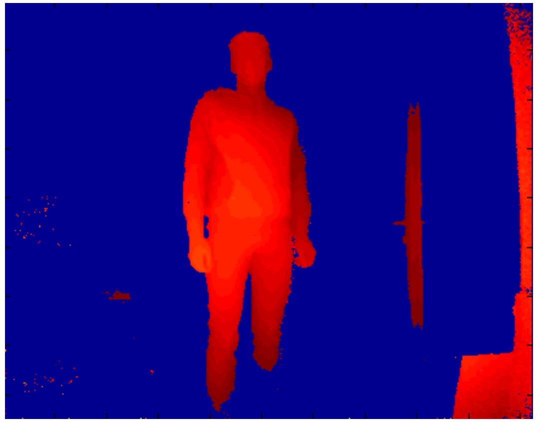
(B) Connected Components

Connected components are regions in an image that are „**connected**“

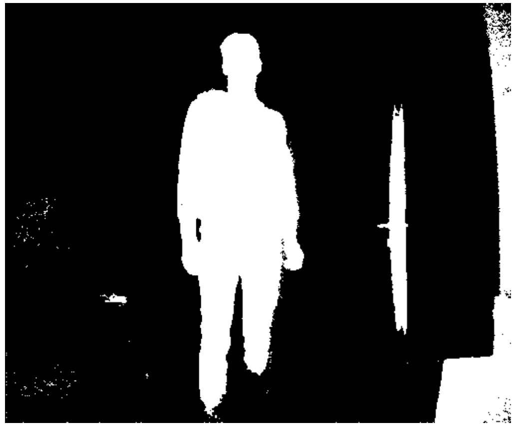
A pixel is **connected** to another pixel if they are in the same **neighborhood**

The **neighborhood** can be defined arbitrarily. Common neighborhoods are:

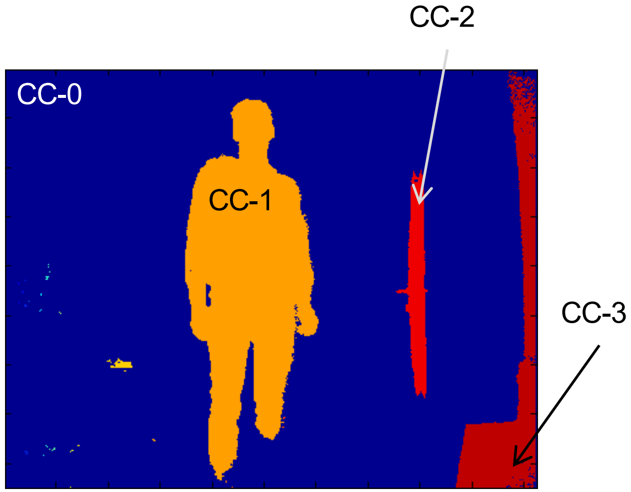
- 4-connected neighborhood 
- 8-connected neighborhood 



Input image



Mask image



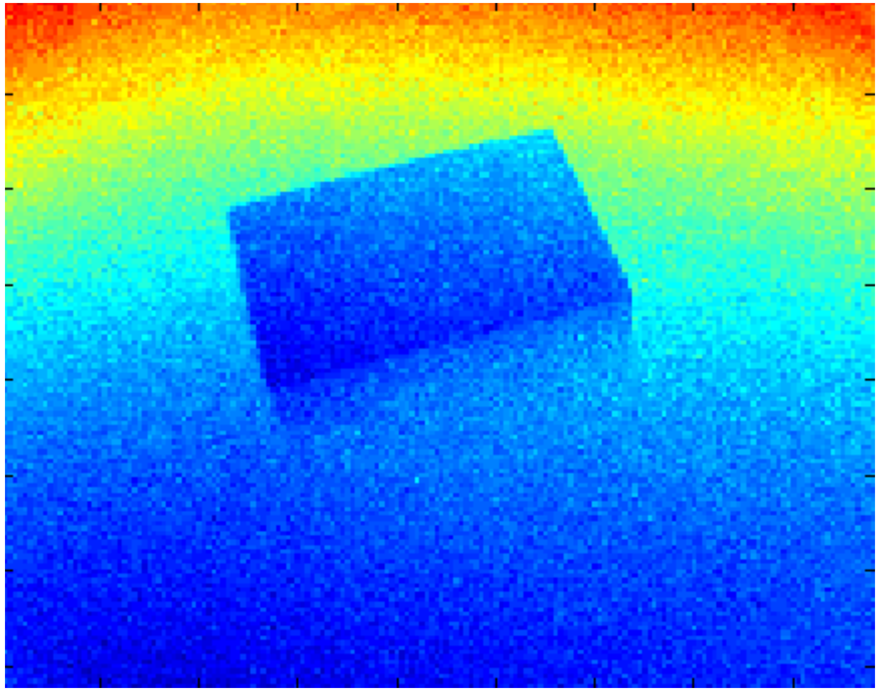
Connected Components

(C) Smoothing: Mean Filter

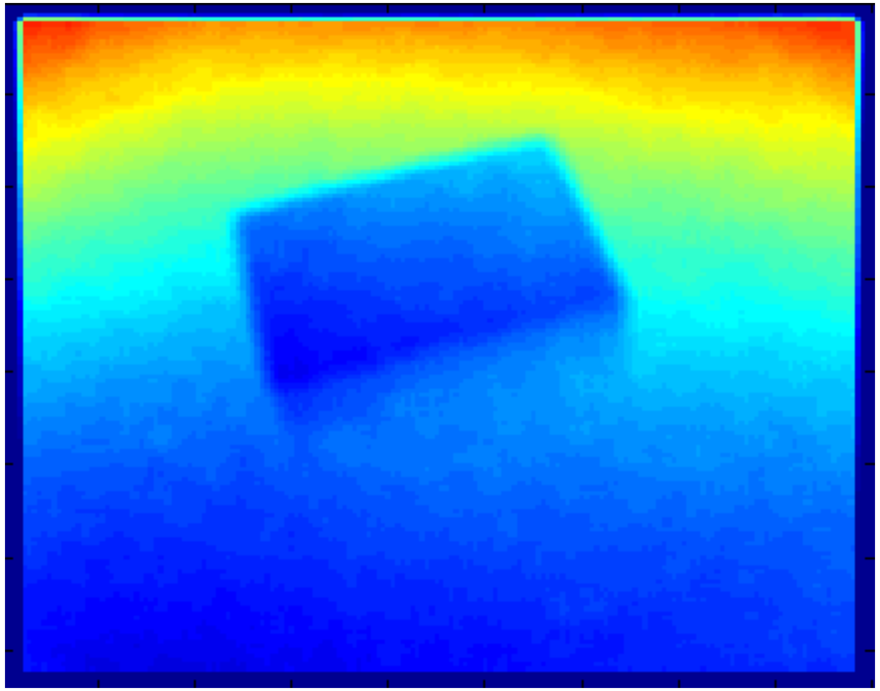
Mean Filter

Smoothes the image by **averaging pixel values** within a **certain window**

Commonly represented as a **convolution**

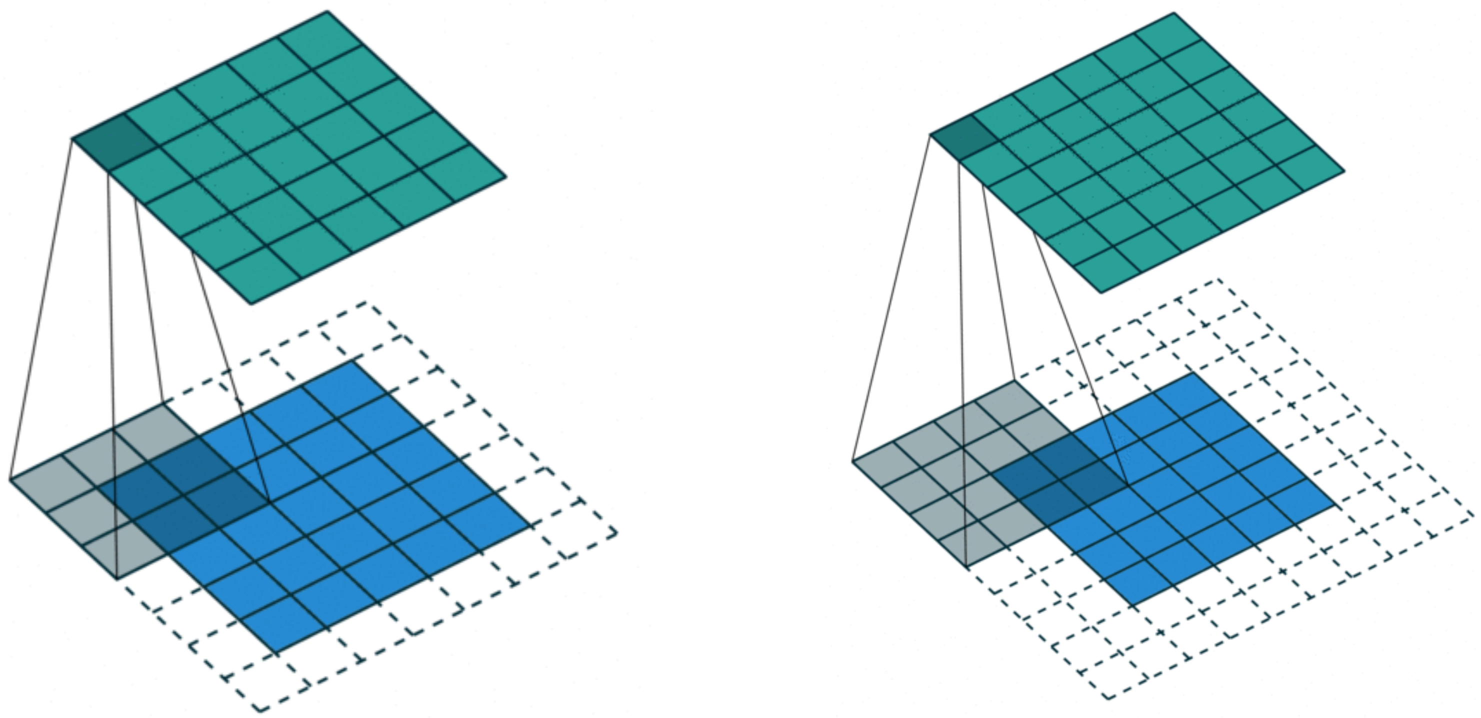


Original image from noisy measurement



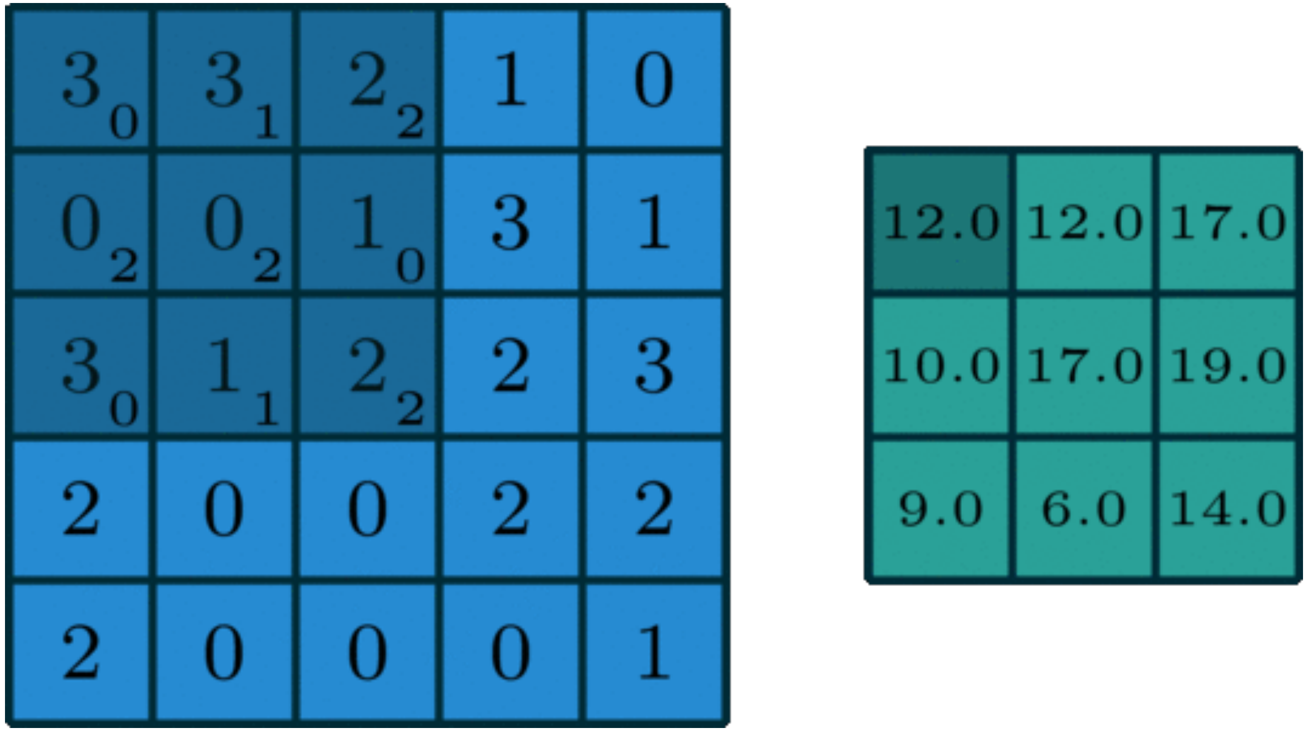
Mean filter (3x3 kernel)

(C) Recap Convolution / Cross-correlation



https://github.com/vdumoulin/conv_arithmetic

(C) Recap Convolution / Cross-correlation

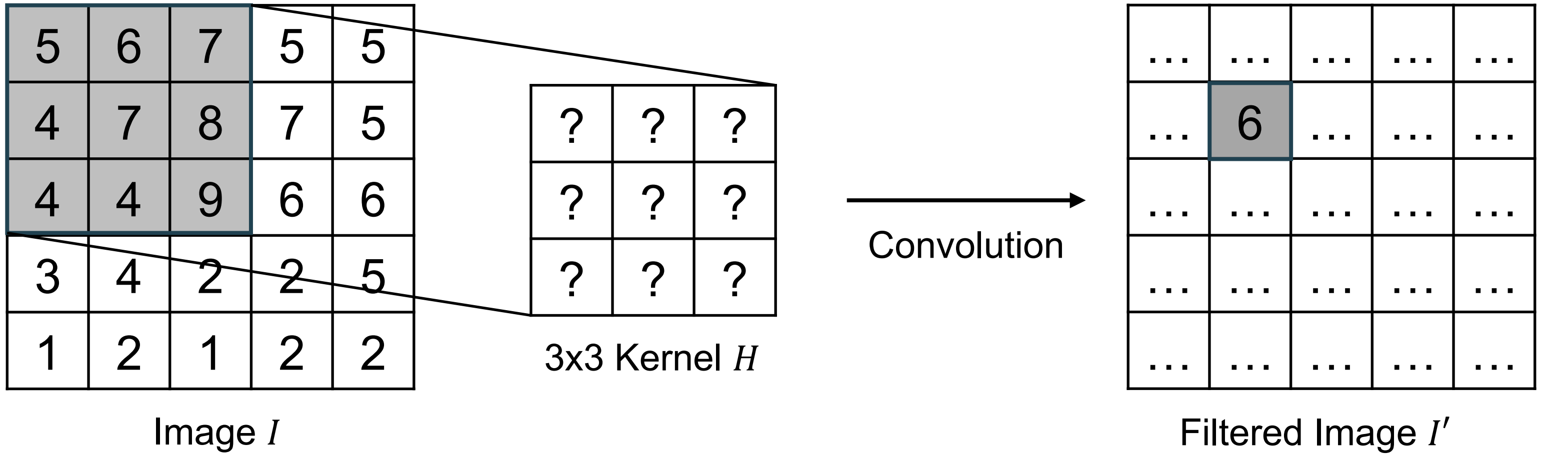


https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1

(C) Smoothing: Mean Filter

Smoothes the image by averaging pixel values within a certain window

Commonly represented as a convolution



(C) Smoothing: Mean Filter

Smoothes the image by averaging pixel values within a certain window

Commonly represented as a convolution

5	6	7	5	5
4	7	8	7	5
4	4	9	6	6
3	4	2	2	5
1	2	1	2	2

Image I

$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$

3x3 Kernel H

Convolution

…	…	…	…	…
…	6	…	…	…
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…

Filtered Image I'

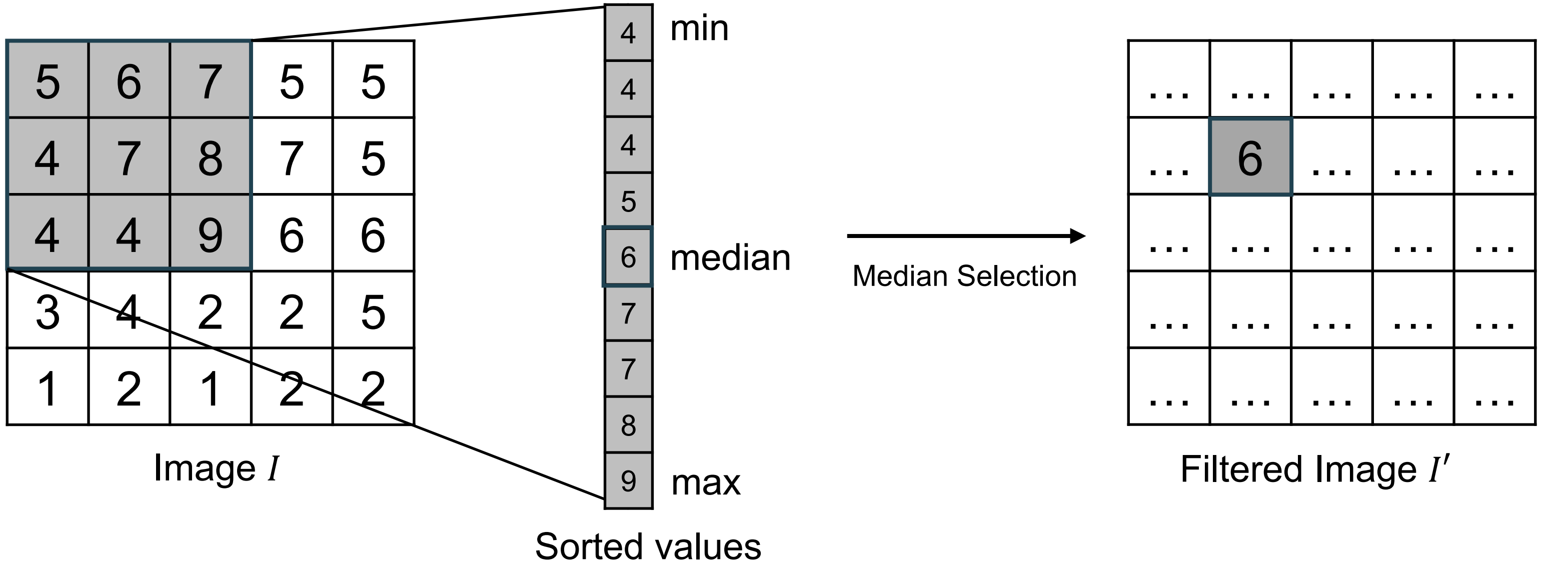
$$I' = I \boxed{*} H = I * \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Asterix denotes the convolution operator

(C) Smoothing: Median Filter

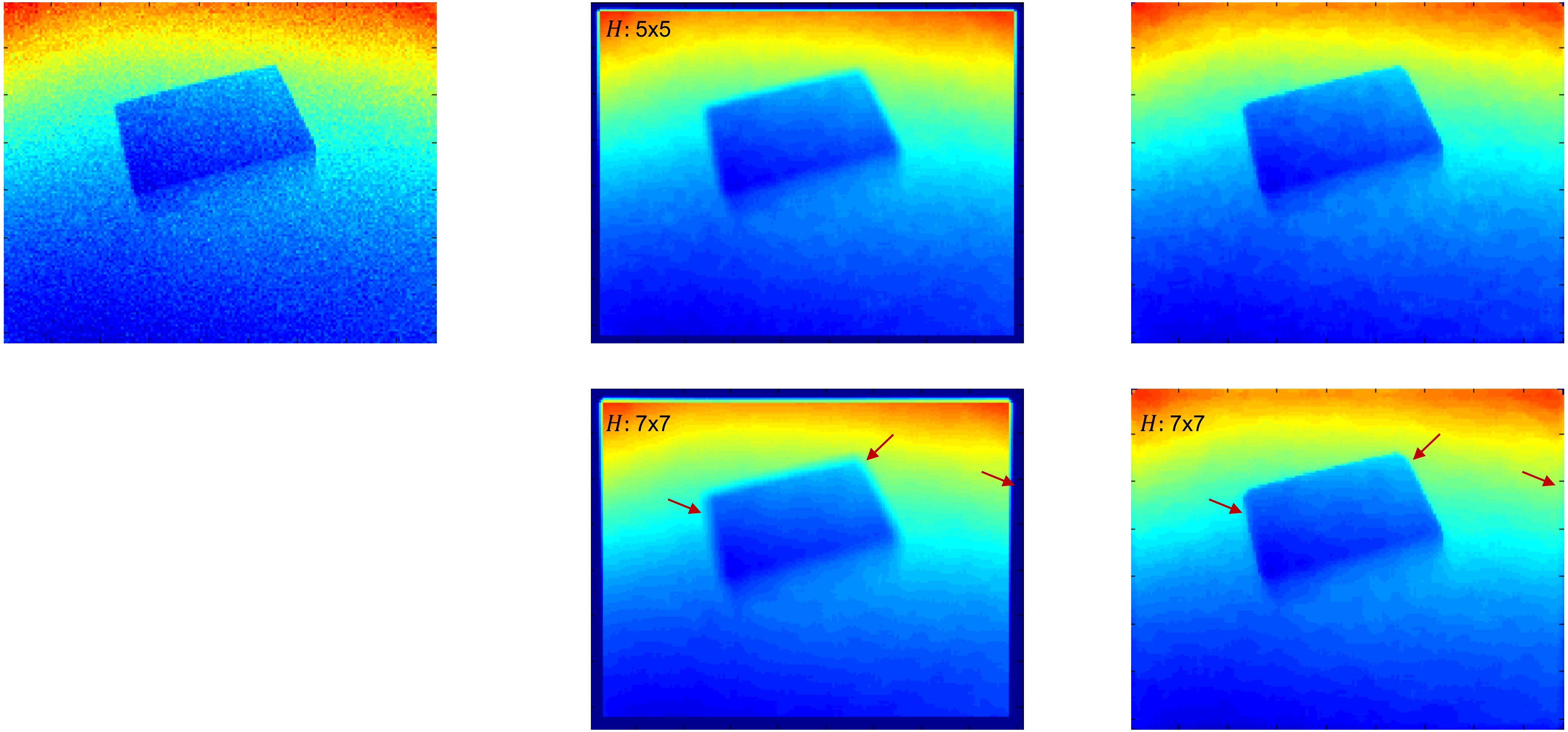
Sorts all pixel values within a certain window and uses the median of the sorted list as new pixel value

- Nonlinear filter → cannot be represented using a convolution
- Popular to remove salt-and-pepper noise



(C) Smoothing: Mean vs. Median Filtering

Distance image (box on surface) | Mean filter (linear) | Median filter (non-linear)



(C) Smoothing

There exist more smoothing filters and algorithms out of which some are based on convolutions while others are not.

- Gaussian smoothing
- Bilateral filtering
- Guided filtering
- ...

(D) Morphological Operators

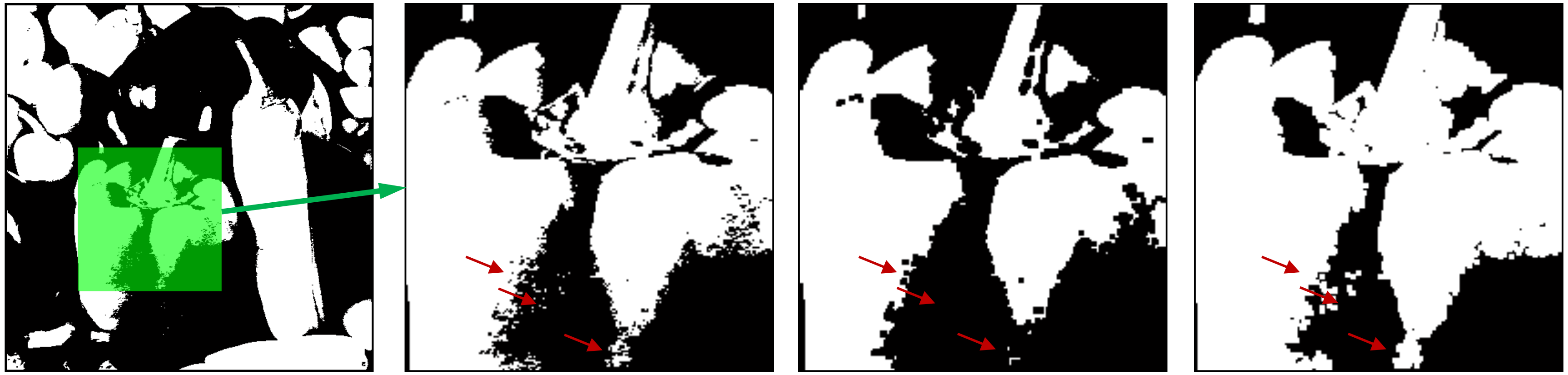
Morphological operators typically work on **binary images** (e.g. masks with only values of 0 a. 255)

→ Analyze pixels within a certain window/structuring element B

Basic operators are:

Erosion → Smooth out fuzzy edges

Dilation → Smooth out nooks and crannies



Binary image | Input region | *Erosion* (3x3) | **Dilation** (3x3)

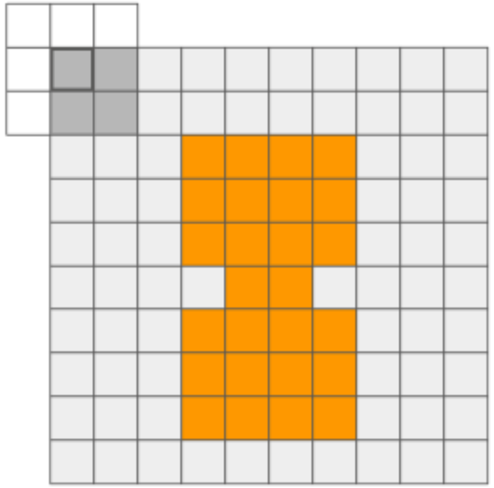
(D) Erosion

Erosion

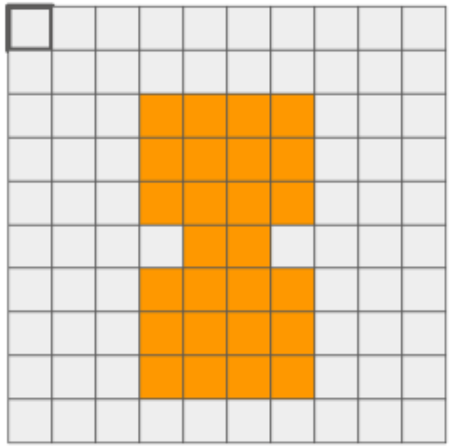
Morphological operation denoted by operator $\ominus$

→ **Rule**: Given an input image $\boldsymbol{A}$**,** a pixel in the output $\boldsymbol{A'}$ is **1** (is retained) if a structuring element $\boldsymbol{B}$ is completely contained in $\boldsymbol{A}$

$\boldsymbol{A}$



$\boldsymbol{A'}$



Struct. Elem. $\boldsymbol{B}$

1	1	1
1	1	1
1	1	1

https://penny-xu.github.io/blog/mathematical-morphology

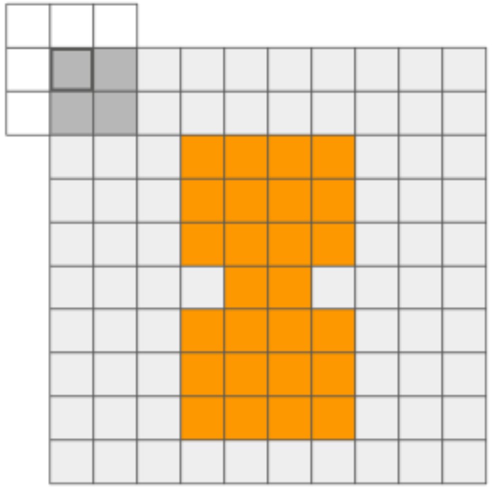
(D) Dilation

Dilation

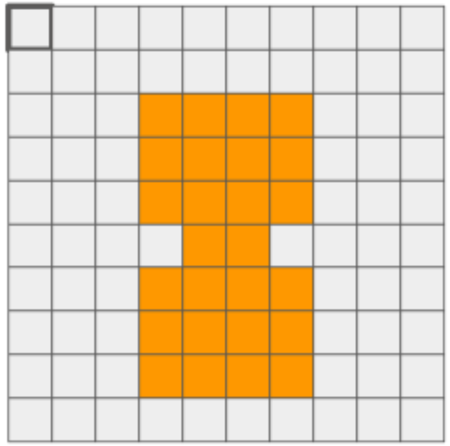
Morpholocial operation denoted by operator $\oplus$

→ **Condition**: Superimpose $\boldsymbol{B}$ on each pixel in $\boldsymbol{A}$ that has the value of 1**.** The output image $\boldsymbol{A'}$ includes every pixel of $\boldsymbol{B}$ that is **1**

$\boldsymbol{A}$



$\boldsymbol{A'}$



Struct. Elem. $\boldsymbol{B}$

1	1	1
1	1	1
1	1	1

https://penny-xu.github.io/blog/mathematical-morphology

(D) Morphological Operators

Opening

First **erode**, then **dilate:** $A' = (A \ominus B) \oplus B$ → Remove noise smaller than the structuring element

Closing

First **dilate**, then **erode**: $A' = (A \oplus B) \ominus B$ → Close holes („black dots“) in the foreground

Binary image | Input region | Opening (3x3) | Closing (3x3)

(D) Further Example

(a) (b) (c) (d) (e)

http://szeliski.org/Book/

a) Input image
b) Dilation (5x5)
c) Erosion (5x5)
d) Opening (5x5)
e) Closing (5x5)

(E) Further Reading

There exist different filters for a multitude of tasks

Edge detection (Sobel, Scharr, Canny)

Second order derivatives (Laplacian of Gaussian)

Corner detectors [Harris88]

Circle detectors [Loy03]

...

Edges: Sobel

Edges: Canny

Corners: Harris

(F) Optimization with RANSAC

RANdom Sampling And Consensus is a simple optimization strategy for noisy data

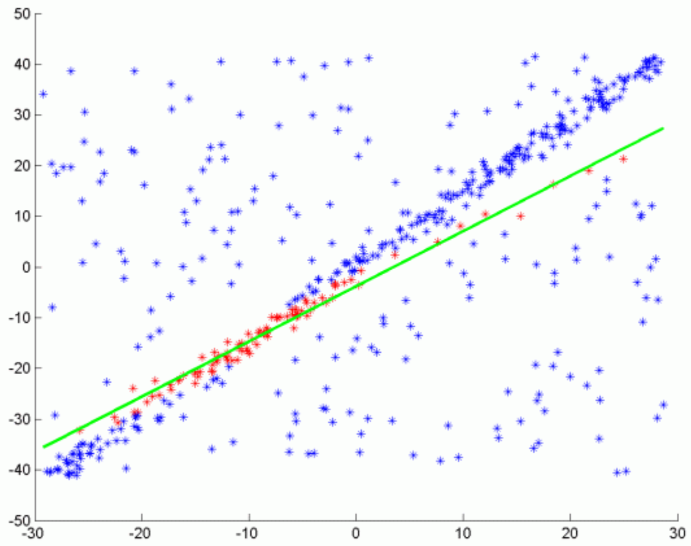
Assumes that a data set contains

(1) **Inliers**

(2) **Outliers** (noisy samples)

Can be used for all kinds of models: lines, planes, rotations, projection matrices, ...

→ We will learn more about this in the exercise!



https://de.wikipedia.org/wiki/RANSAC-Algorithmus

```
max_inliers = 0
best_model = null
while (i < N) {
        samples = minimal_sample(k, all_points)
        model = estimate_model(samples)
        n_inliers = computeInliers(all_points, model)
        if (n_inliers > max_inliers)  {
                best_model = model
                max_inliers = n_inliers
        }
}
```

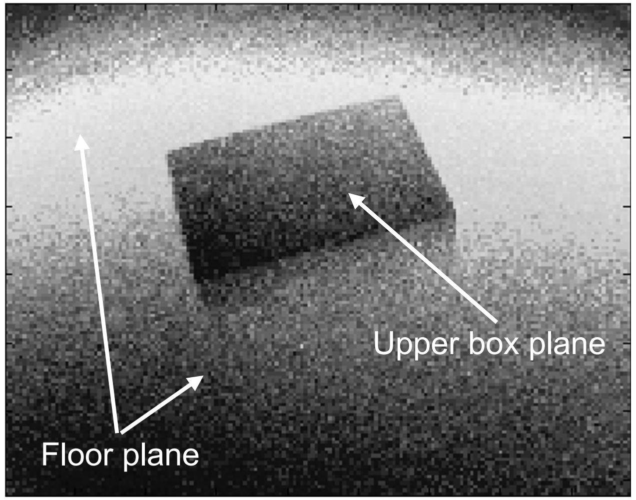
Lecture 1
Questions?

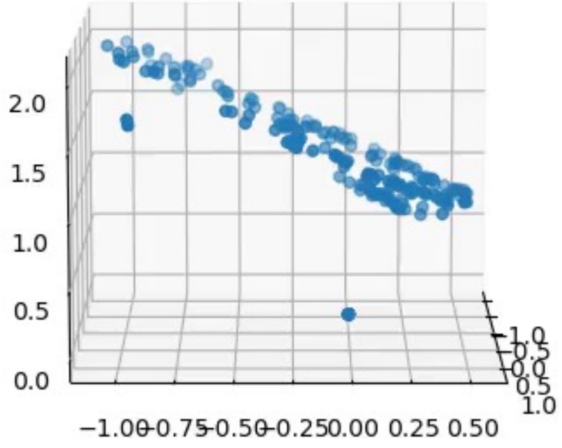
Exercise 1
Teaser

Exercise 1: Box Detection

What you get

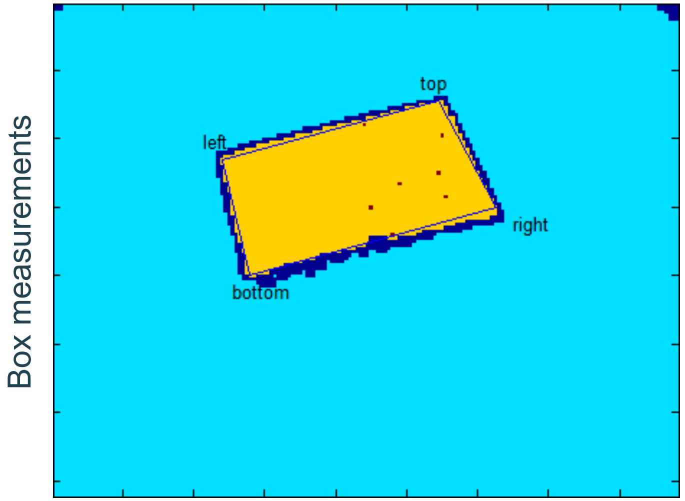
- 2D distance image of a simple packaging box on a table
- Its corresponding 3D point cloud measurement





Your goal is to

- Find the floor plane
- Find the box
- Estimate the size of the box



...with the tools you have learned today
  (**RANSAC, Filtering, Smoothing,** etc.)

References

References

[Harris88] C. Harris and M. Stephens (1988). "*A combined corner and edge detector*“, 1988

[Loy03] G. Loy, „*Fast radial symmetry for detecting points of interest*,” 2003