# SSY097 - Image Analysis

## Lecture 1 - Introduction and Linear Classifiers

*Torsten Sattler*
*(slides adapted from Olof Enqvist)*

# Team

Jennifer Alvén

alven@chalmers.se

Carl Toft

carl.toft@chalmers.se

José Pedro Lopes Iglesias

jose.iglesias@chalmers.se

Torsten Sattler torsat@chalmers.se

# Course Information

- Exercises (optional, non-graded)

# Course Information

- Exercises (optional, non-graded)
- Four mandatory lab assignments
  - Hand in code and short report
  - Pass or fail

  - Deadlines (23:59): February 2nd, February 13th, February 22nd, March 10th

# Course Information

- Exercises (optional, non-graded)
- Four mandatory lab assignments
  - Hand in code and short report
  - Pass or fail

  - Deadlines (23:59): February 2nd, February 13th, February 22nd, March 10th
- Mandatory mini project (two choices):
  - Hand in code, report, poster

  - Up to 3 bonus points for exams (do not help passing exam)

# Course Information

- Exercises (optional, non-graded)
- Four mandatory lab assignments
  - Hand in code and short report
  - Pass or fail

  - Deadlines (23:59): February 2nd, February 13th, February 22nd, March 10th
- Mandatory mini project (two choices):
  - Hand in code, report, poster

  - Up to 3 bonus points for exams (do not help passing exam)

- Recommend doing the labs and project in pairs

# Course Information

- Exercises (optional, non-graded)
- Four mandatory lab assignments
  - Hand in code and short report
  - Pass or fail

  - Deadlines (23:59): February 2nd, February 13th, February 22nd, March 10th
- Mandatory mini project (two choices):
  - Hand in code, report, poster

  - Up to 3 bonus points for exams (do not help passing exam)

- Recommend doing the labs and project in pairs
- Graded exam on March 23rd (pen and paper)

# Check PingPong Course Page Regularly

**SSY097 Bildanalys V19**


Event

What's new

Overview

Content

- Course information
- Schedule
- **Projects**
- **Course Representatives**
- **Lab 1**
- **Lab 2**
- **Lab 3**
- **Lab 4**
- **Mini project**

Documents

FAQ

Communication

Members

Message Board

Discussion

Project groups

Toolbox

Preferences

Answer questions

Project groups

Send message

Log book

g.chalmers.se/courseId/10509/contentStart.do

Previous Next Search Maximize the content Save as PDF Edit Print


## Course information

### Essential information

- There are four mandatory lab assignments.
  After completing the lab you submit a mini report together with your code. Deadlines below.
- There is a mandatory mini project. You choose to do either Project A or Project B.
- The exam takes place on March 23 (time: TBA). Allowed materials: Pencil eraser.
  You can get 3 bonus credits on the exam by completing extra tasks connected to the mini project.
  These bonus credits are only valid for grades 4 and 5. They do not help in passing the exam.
  The bonus credits are valid at all three exams in 2019 (March 23, June 10, Aug 19).
- We recommend that you do the labs and projects in pairs, but in special cases groups of 1-3 persons are ok.

Further information, such as aims and learning outcomes can be found on the course page at student.portal.chalmers.se

### Staff

- **Torsten Sattler**, lecturer, torsat@chalmers.se
- **Jennifer Alvén**, teaching assistant, alven@chalmers.se
- **Carl Toft**, teaching assistant, carl.toft@chalmers.se
- **José Pedro Lopes Iglesias**, teaching assistant, jose.iglesias@chalmers.se
- **Fredrik Kahl**, examiner, fredrik.kahl@chalmers.se

### Course material

The necessary course material will be provided during the course. This includeslab pm's and exercises.
If you like to read more about image analysis, you can use Szeliskis book.

- Richard Szeliski, Computer Vision: Algorithms and Applications, available at Cremona or as a free pdf.

### Deadlines

You need to submit your code and reports before the following dates. If there are minor errors, you will be given a chance to correct them later.
If you need more than one iteration to correct your mistakes, we do not promise to mark them before the re-exam.

1. Lab 1: Sunday, February 2, 23:59
2. Lab 2: Wednesday, February 13, 23:59
3. Lab 3: Sunday, February 22, 23:59

# Ask Questions

- Feel free to ask questions any time
- Please give us feedback (positive and negative) about the lecture, the labs, …
  - Talk to me before or after the lecture
  - Write me an email
  - Come to my office
  - Talk to the course representatives (see PingPong page)
- Happy to adapt based on your suggestions

# Today

- What is Image Analysis
- Modeling Images
- Filtering and Linear Classifiers

# What is Image Analysis?

# Image Classification

Is there a robin in this image?

# Object Detection

Where is the robin?

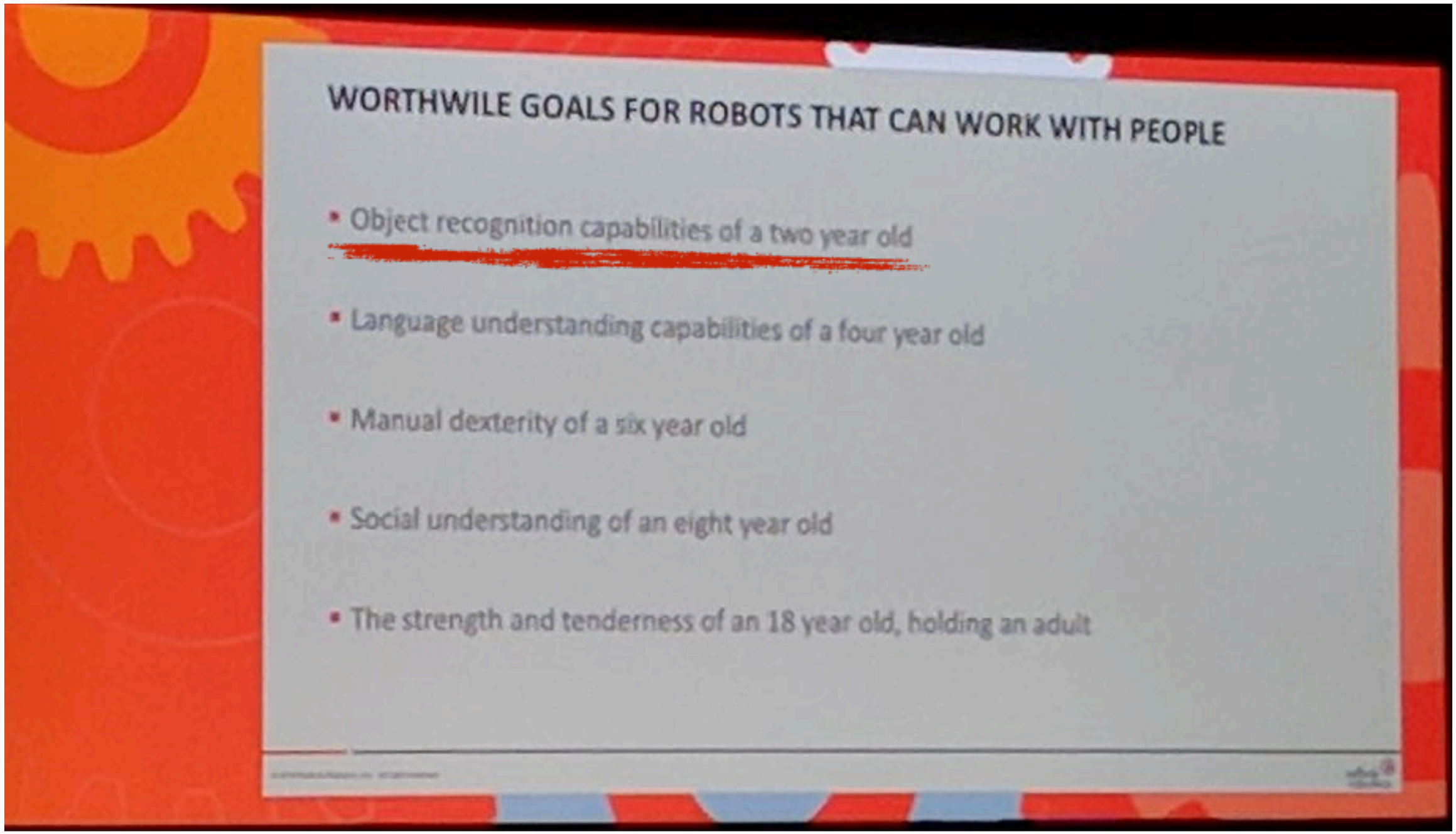
# Object Detection


YOLOv3

# Object Detection


YOLOv3

# Image Classification




ICRA 2018 plenary talk by Rodney Brooks

# Image Segmentation

**Full-Resolution Residual Networks for Semantic Segmentation in Street Scenes**

Tobias Pohlen, Alexander Hermans,
Markus Mathias, Bastian Leibe

Visual Computing Institute, Computer Vision Group
RWTH Aachen University

Visual Computing Institute
Computer Vision
Prof. Dr. Bastian Leibe

RWTH AACHEN UNIVERSITY

[Full-Resolution Residual Networks (FRRNs) for Semantic Image Segmentation in Street Scenes]

# Image Segmentation

**Full-Resolution Residual Networks for Semantic Segmentation in Street Scenes**

Tobias Pohlen, Alexander Hermans,
Markus Mathias, Bastian Leibe

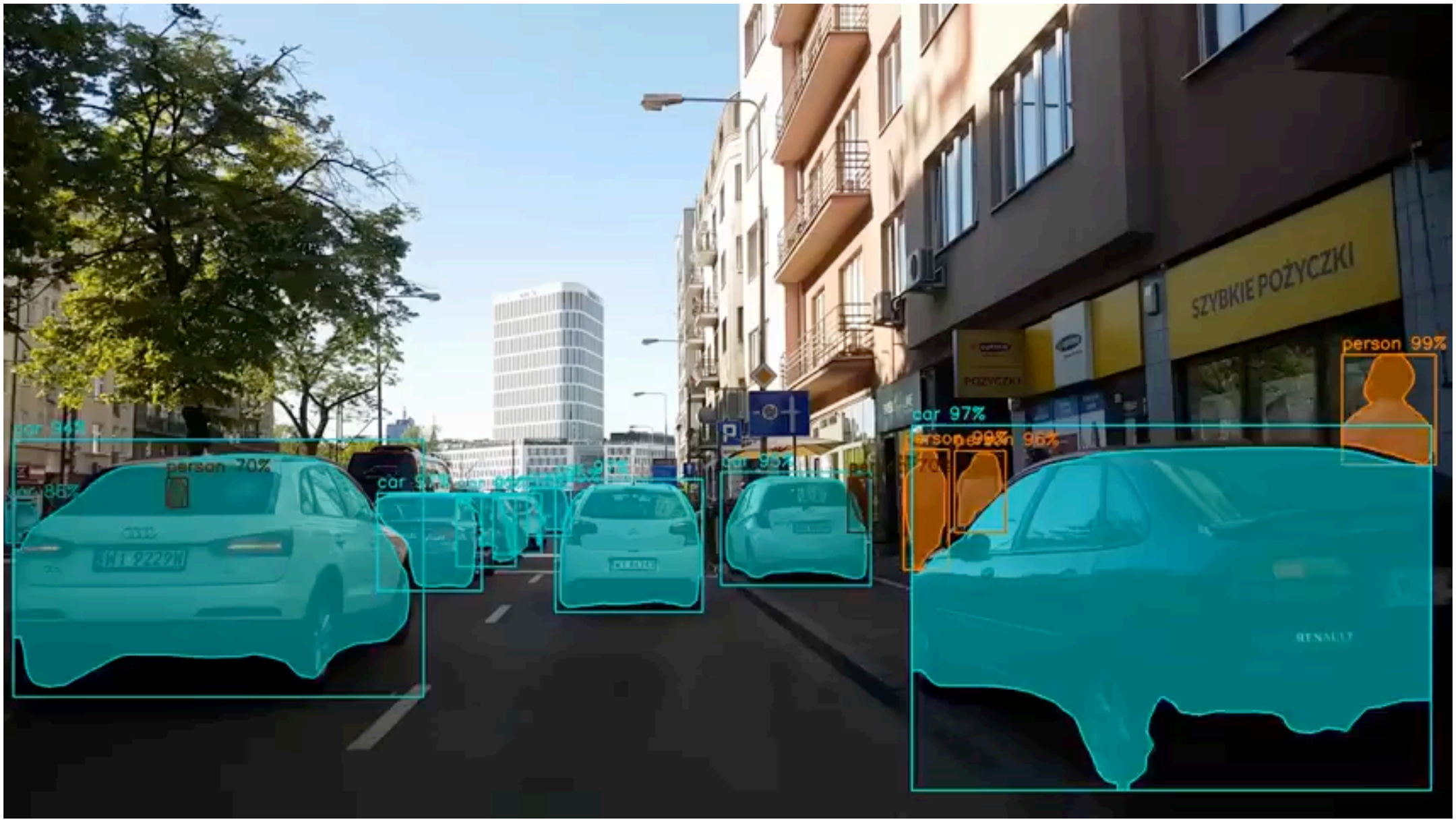
Visual Computing Institute, Computer Vision Group
RWTH Aachen University

Visual Computing Institute
Computer Vision
Prof. Dr. Bastian Leibe

RWTH AACHEN UNIVERSITY

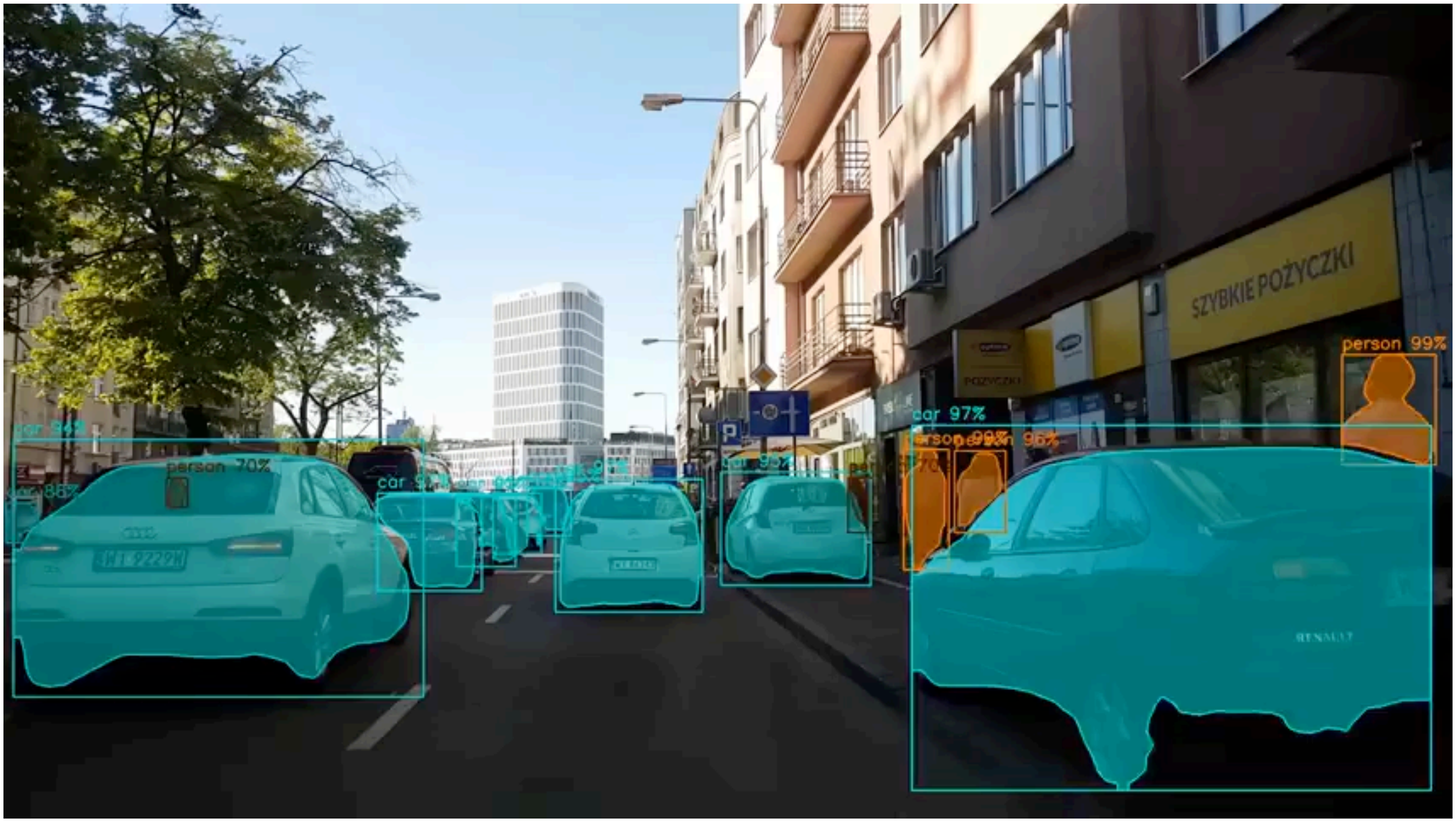
[Full-Resolution Residual Networks (FRRNs) for Semantic Image Segmentation in Street Scenes]

# Instance-Level Segmentation



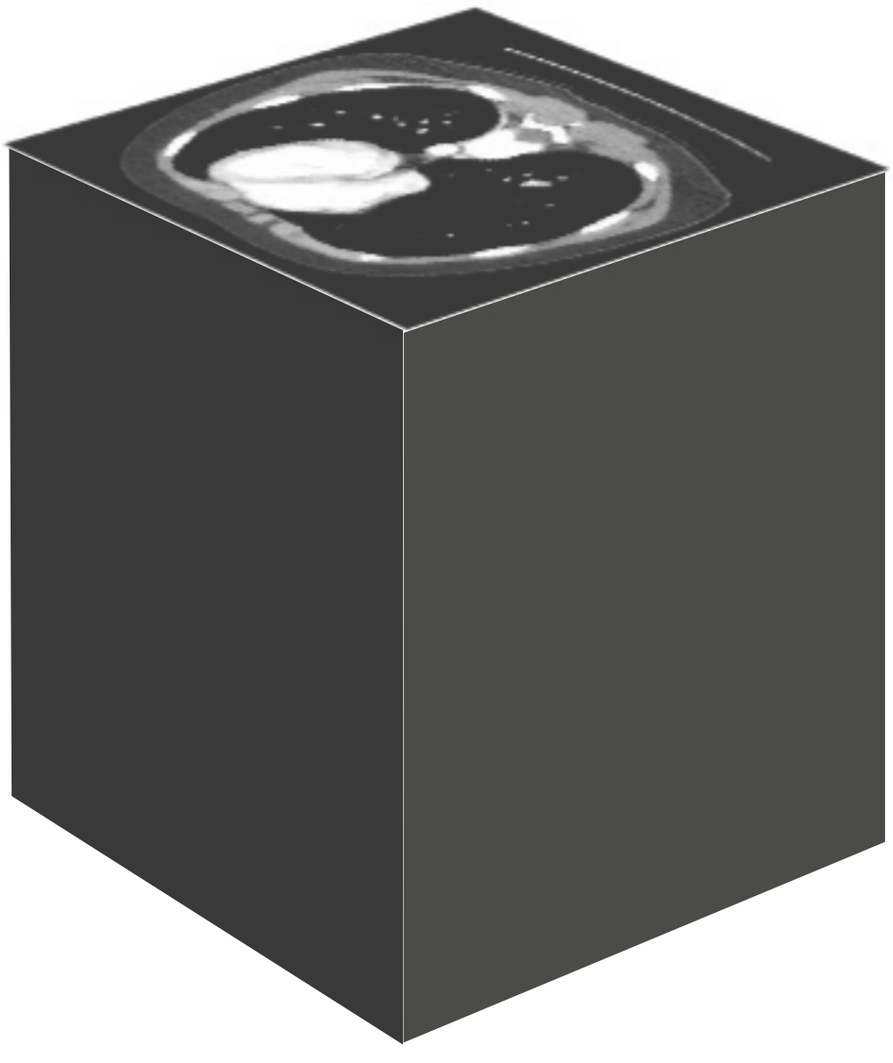

[Masked R-CNN]

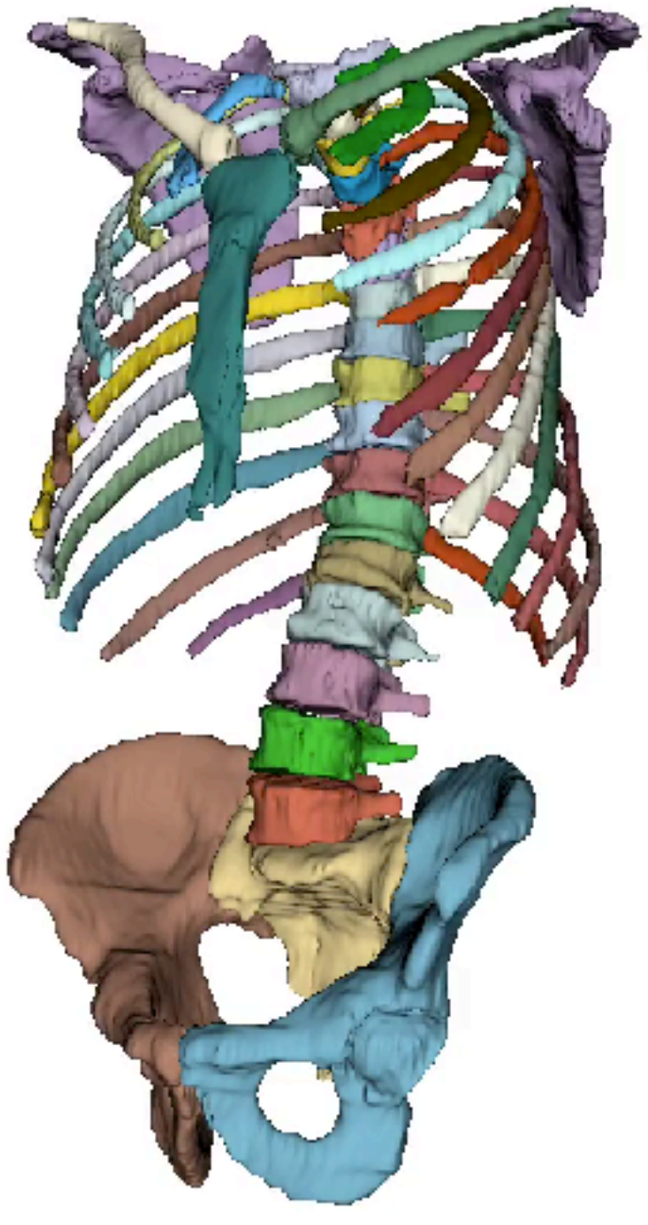
# Instance-Level Segmentation



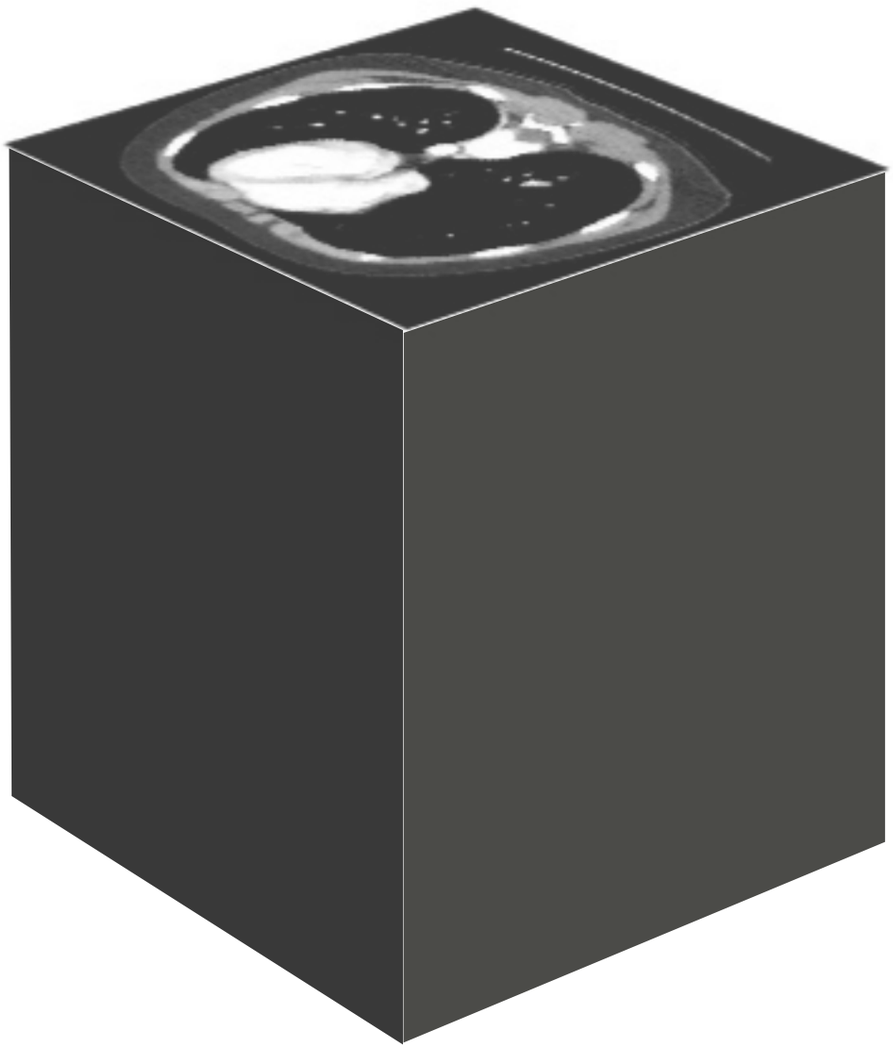

[Masked R-CNN]

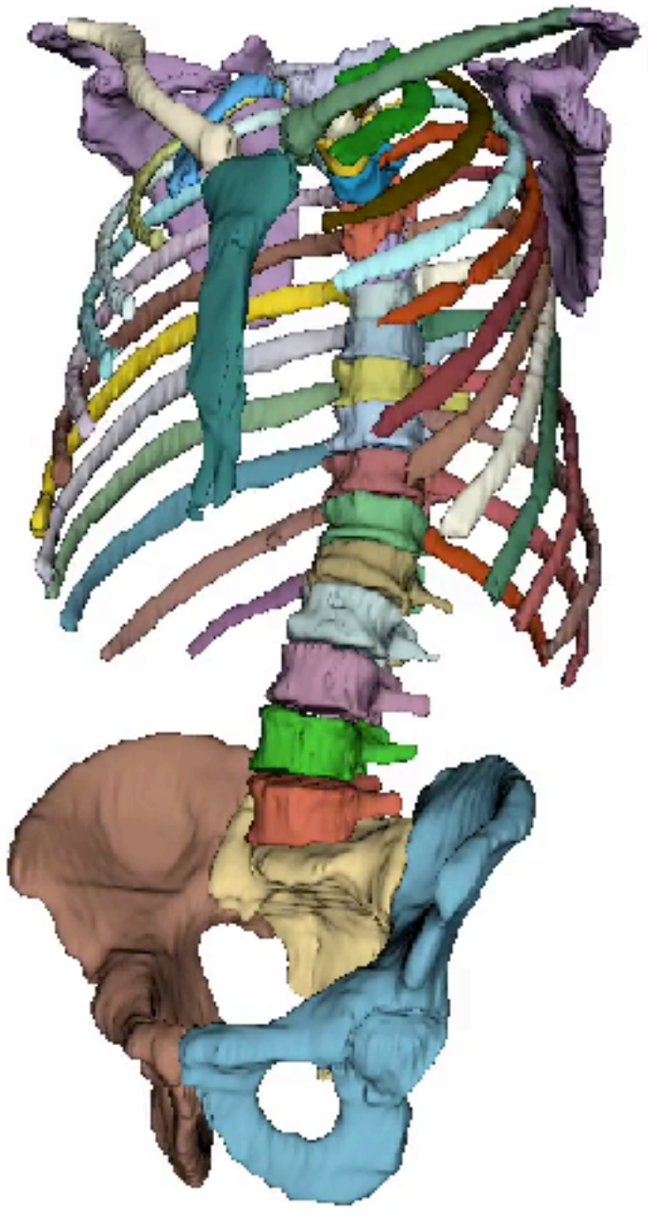
# 3D Segmentation





# 3D Segmentation





# 3D Reconstruction

## Structure-from-Motion Revisited

Johannes L. Schönberger, Jan-Michael Frahm

CVPR 2016

Code available at:
https://github.com/colmap/colmap

[Structure-from-Motion Revisited]

# 3D Reconstruction

## Structure-from-Motion Revisited

Johannes L. Schönberger, Jan-Michael Frahm

CVPR 2016

Code available at:
https://github.com/colmap/colmap

[Structure-from-Motion Revisited]

# 3D Reconstruction

## Pixelwise View Selection for Unstructured Multi-View Stereo

Johannes L. Schönberger, Enliang Zheng,
Marc Pollefeys, Jan-Michael Frahm

***ECCV 2016***

[Pixelwise View Selection for Unstructured Multi-View Stereo]

# 3D Reconstruction

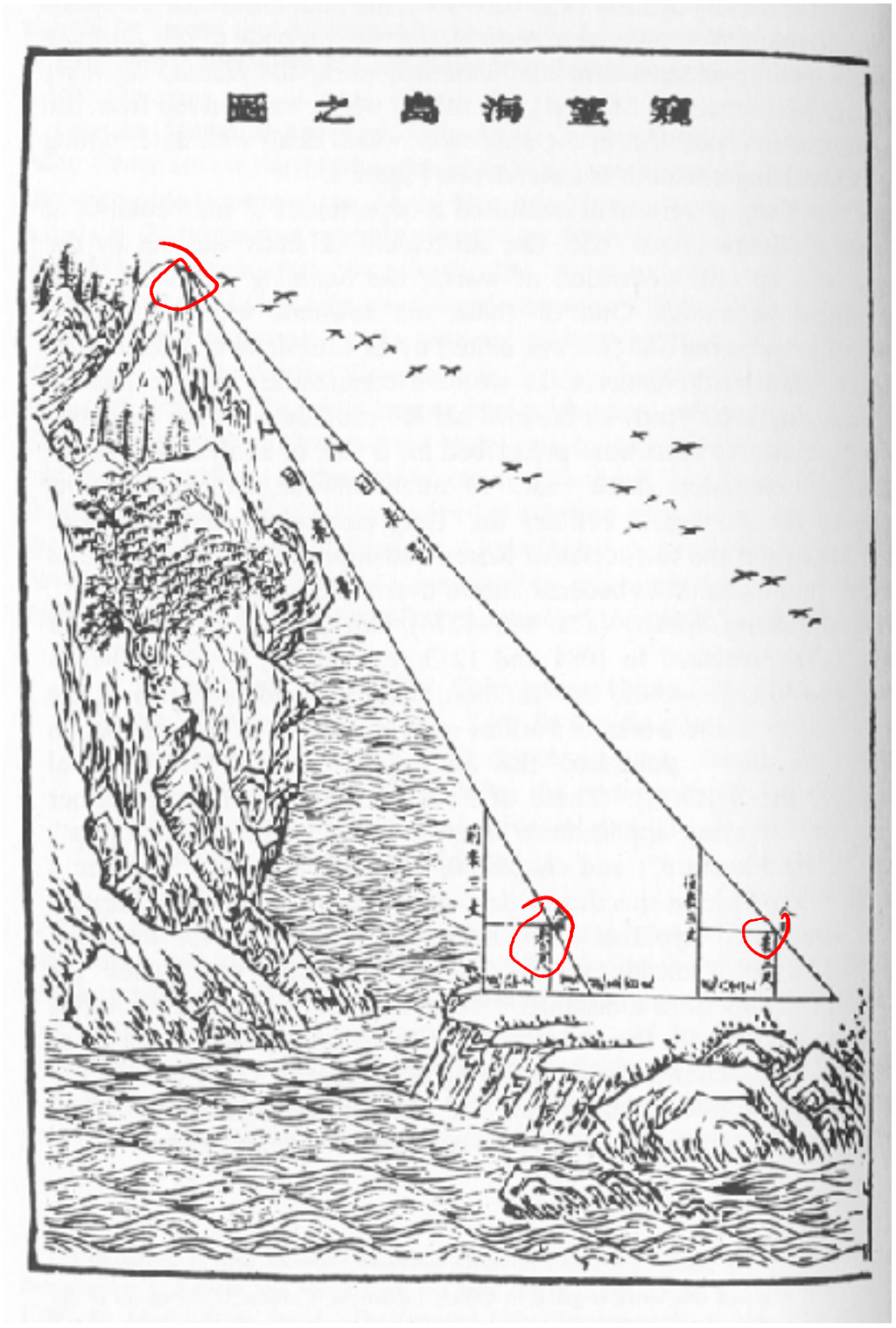
## Pixelwise View Selection for Unstructured Multi-View Stereo

Johannes L. Schönberger, Enliang Zheng,
Marc Pollefeys, Jan-Michael Frahm

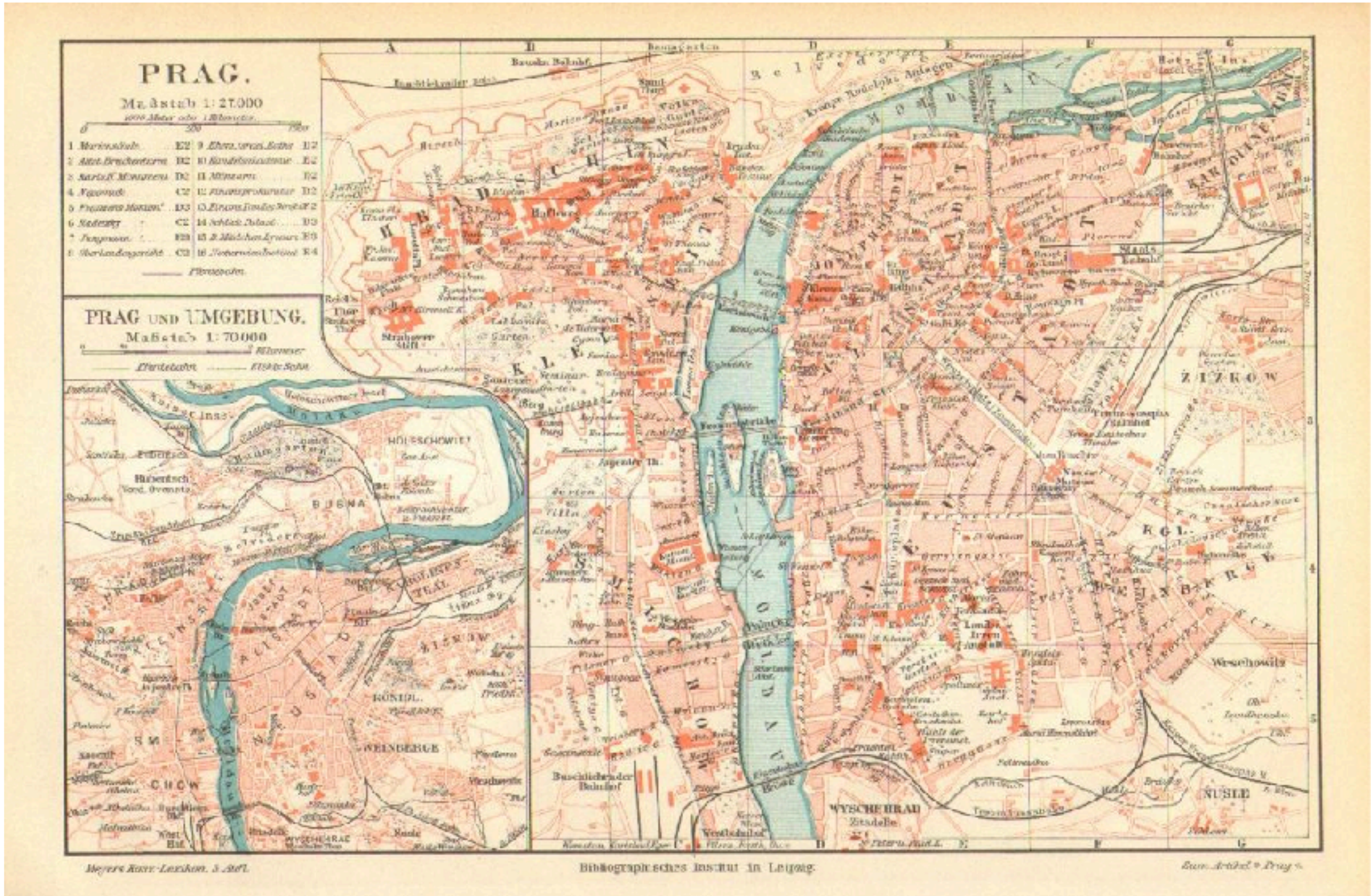
***ECCV 2016***

[Pixelwise View Selection for Unstructured Multi-View Stereo]

# 3D Reconstruction



# 3D Reconstruction




Prague in the 19th century

# Mixed Reality

[Microsoft HoloLens]

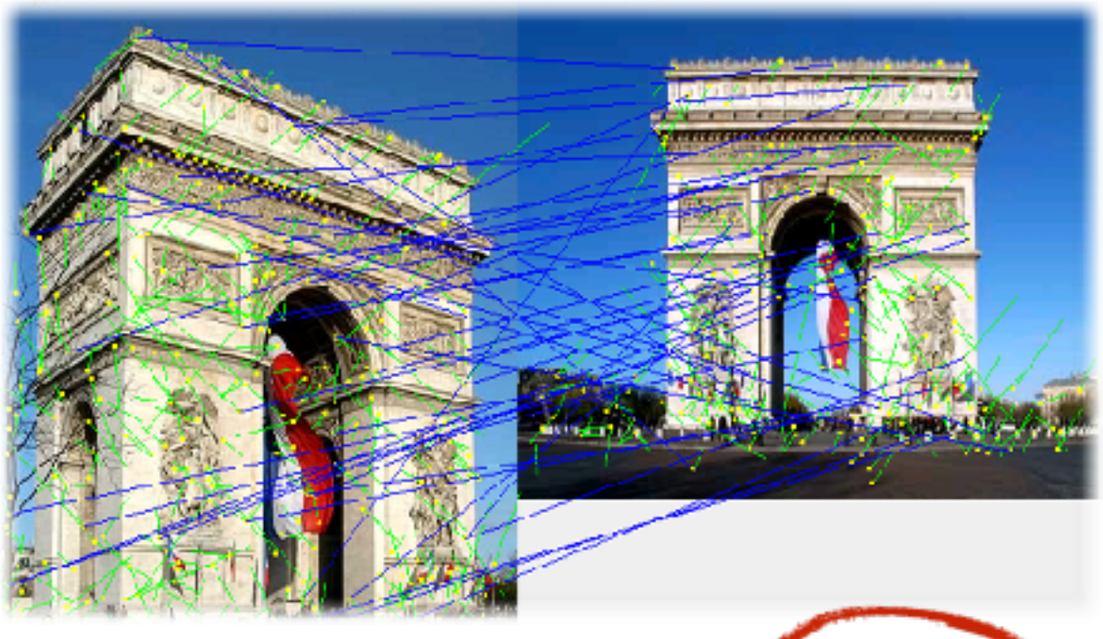
# Mixed Reality

[Microsoft HoloLens]

# Part 1. Basic image handling and sparse features

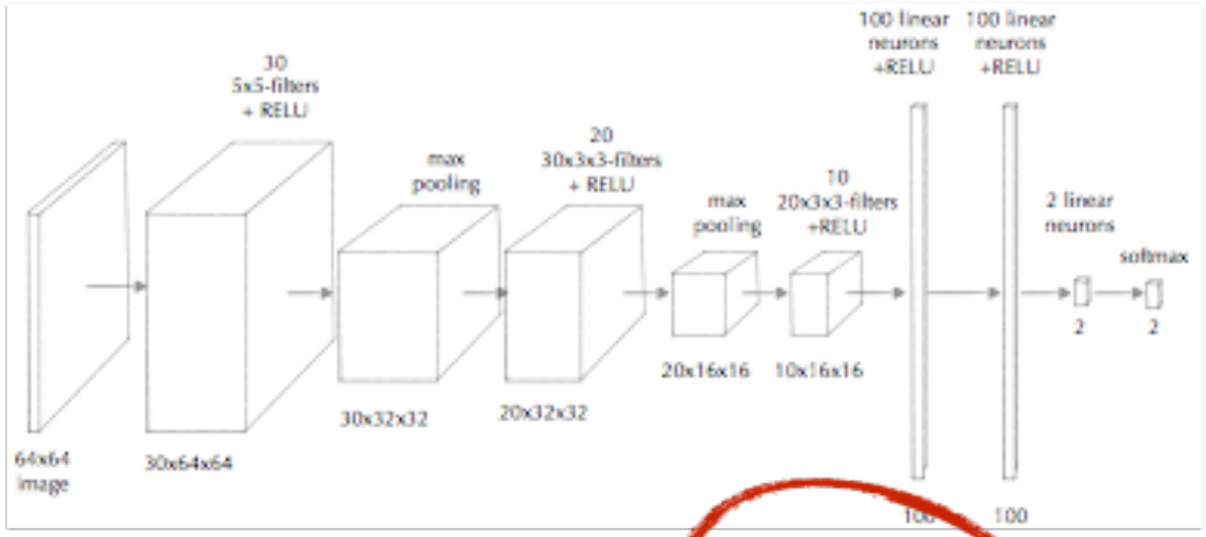
Lab 1

Part 1. Basic image handling and sparse features

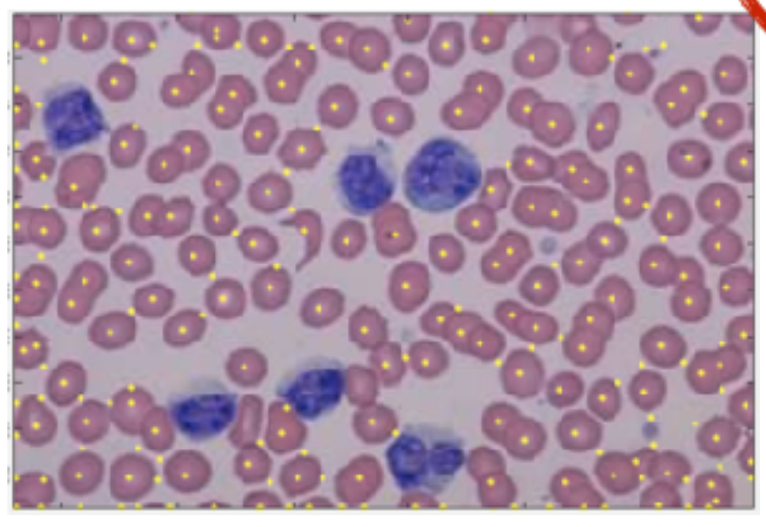


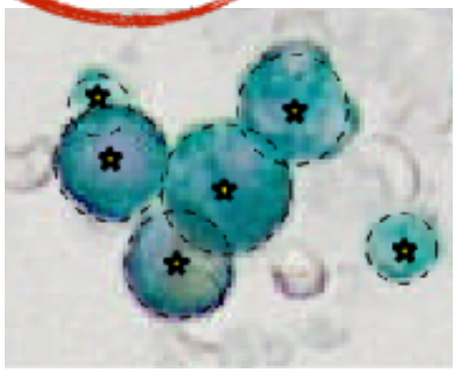
Lab 1

Part 2. Machine learning for classification, detection, and segmentation

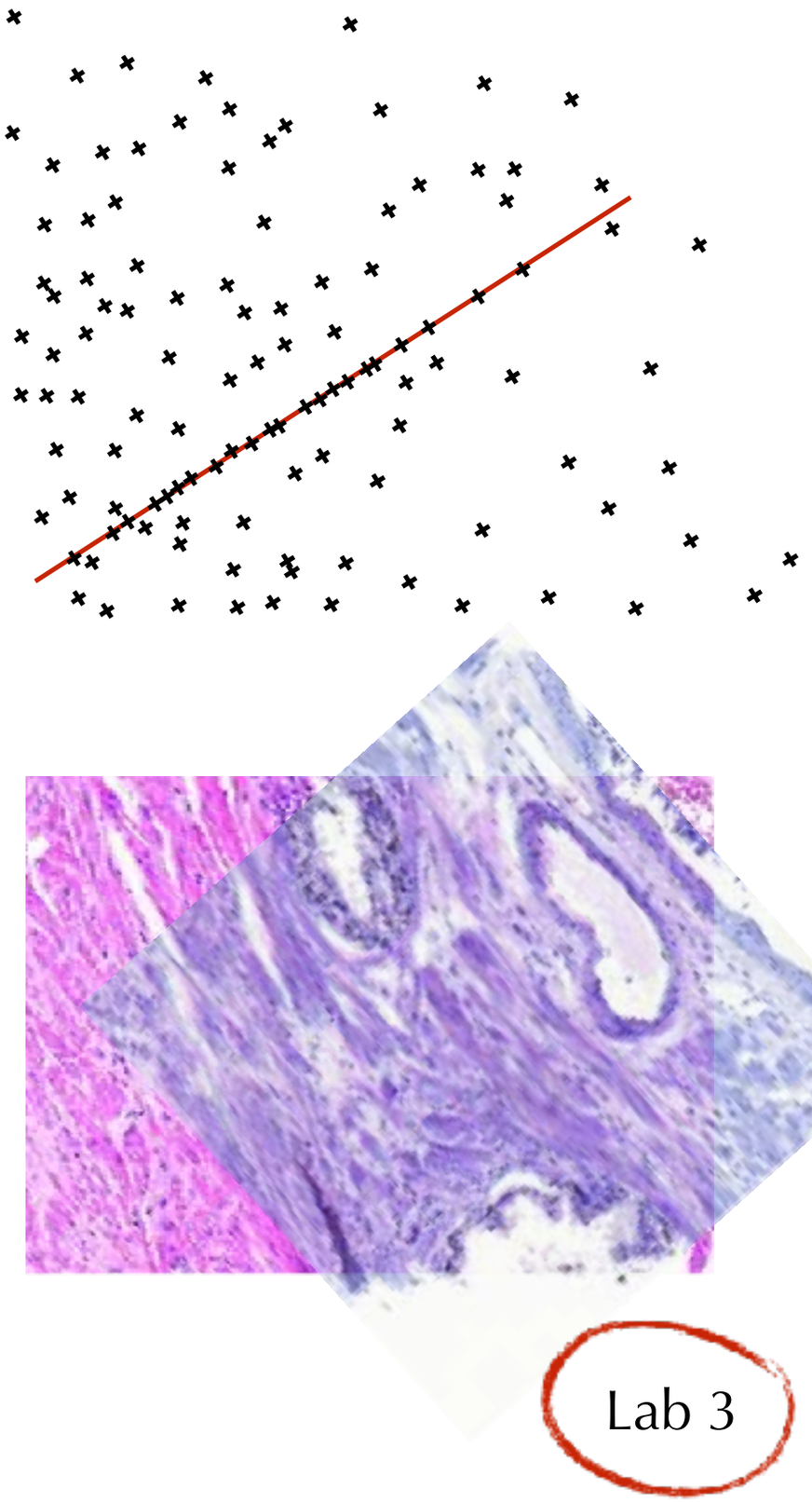



Lab 2
Project A



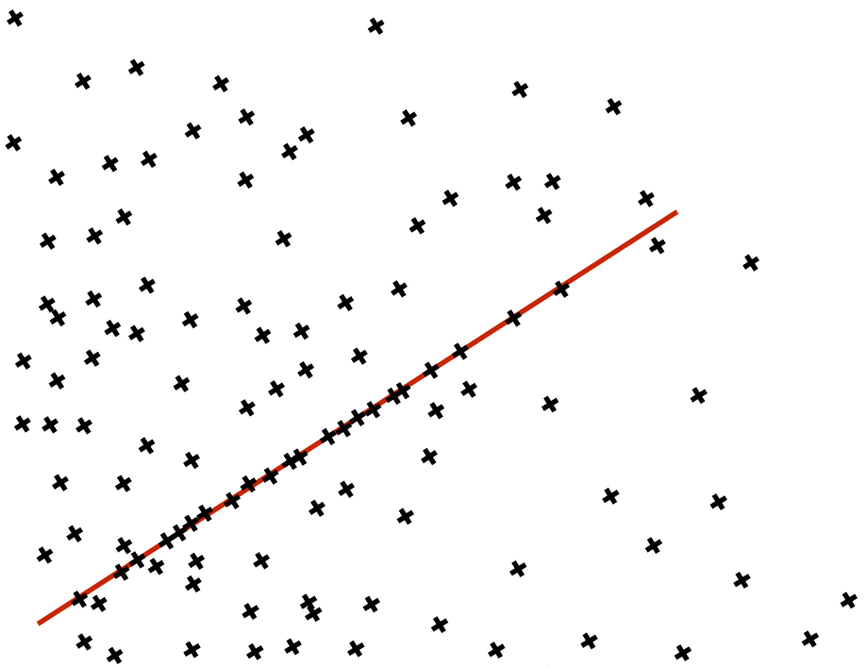


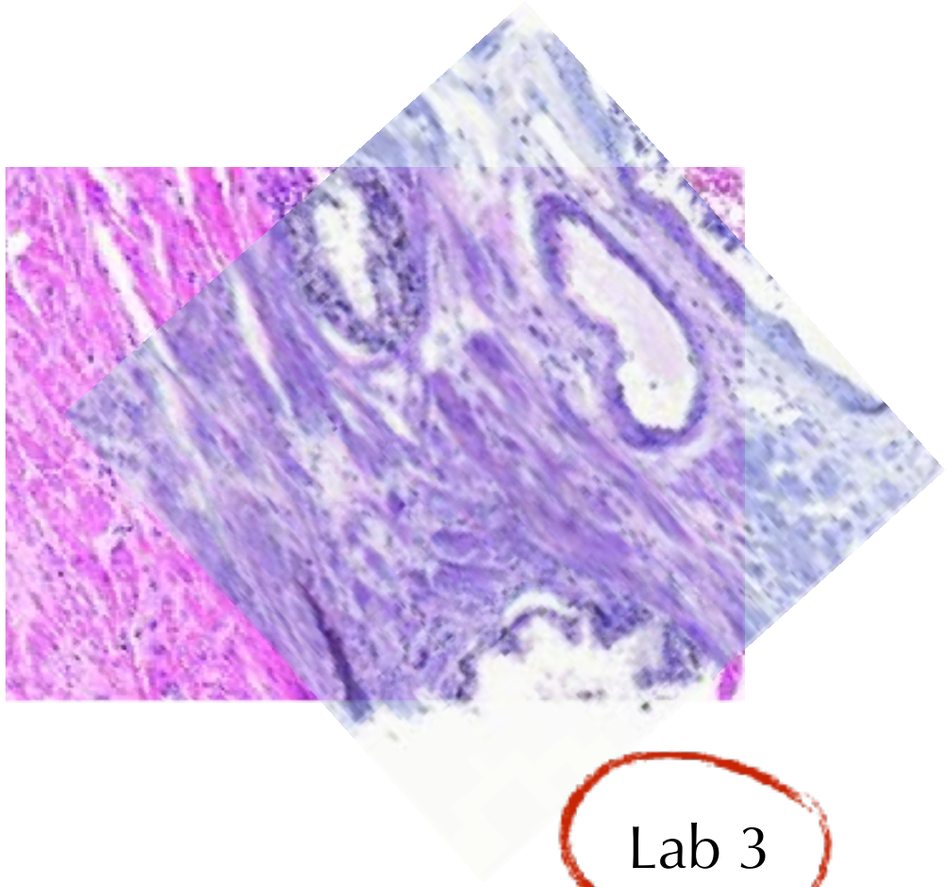
# Part 3. Robust model estimation and image registration



Lab 3

## Part 3. Robust model estimation and image registration



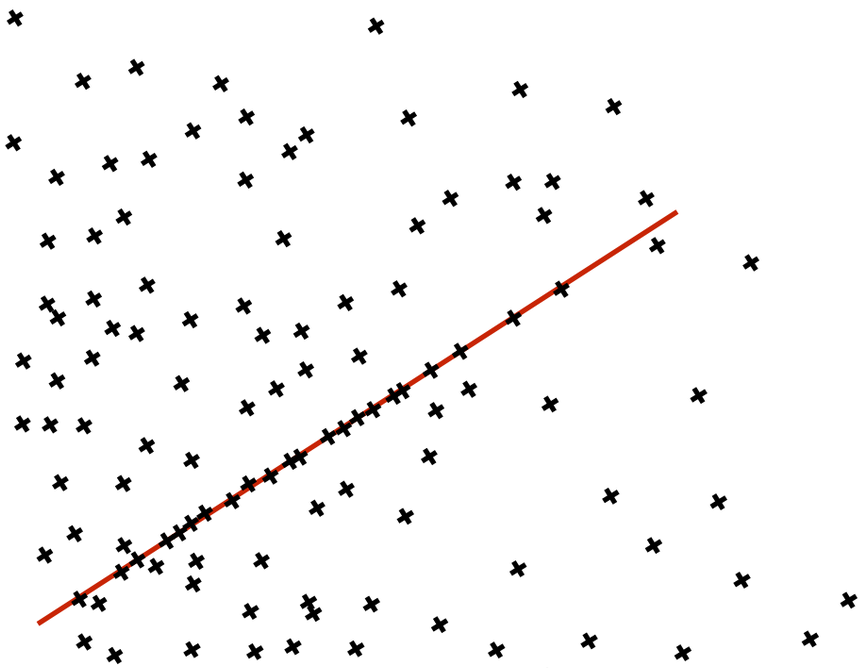


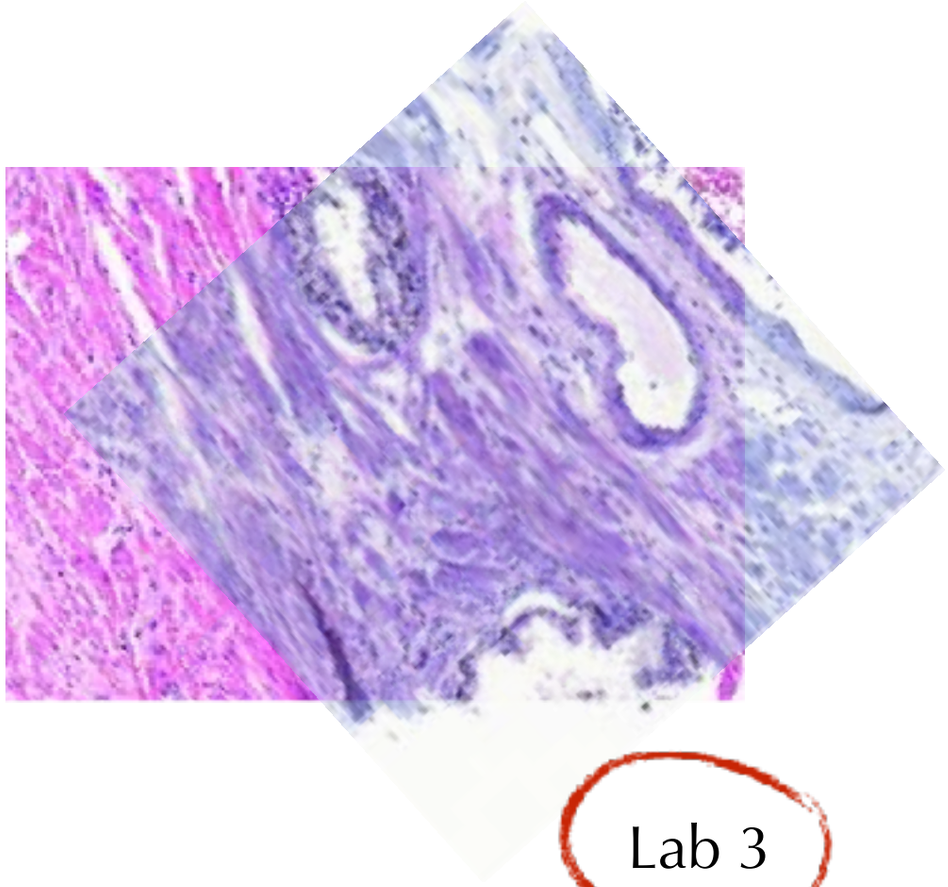
Lab 3

## Part 4. Camera geometry and basic 3D reconstruction

Lab 4
Project B

Part 3. Robust model estimation and image registration





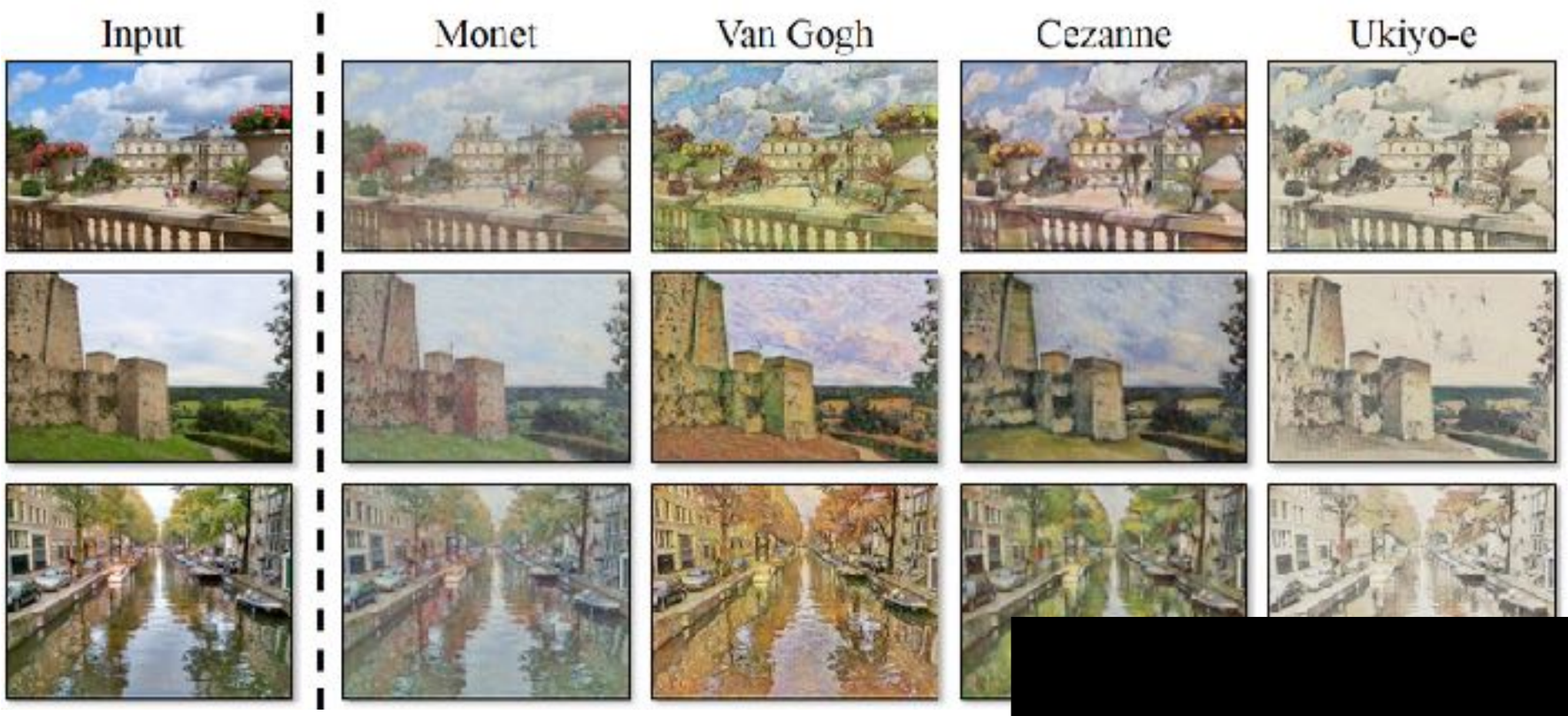
Lab 3

Part 4. Camera geometry and basic 3D reconstruction

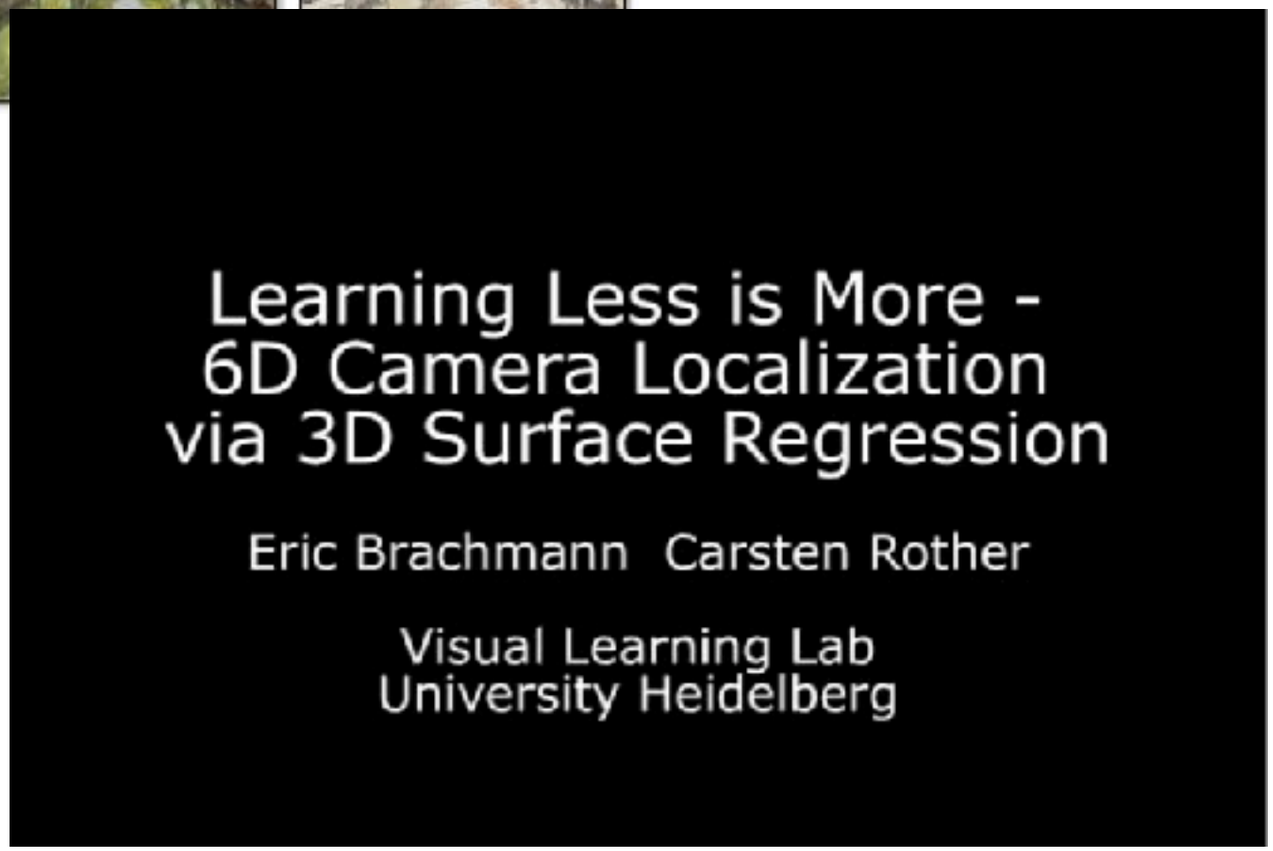
Lab 4
Project B

EEN020 - Computer Vision

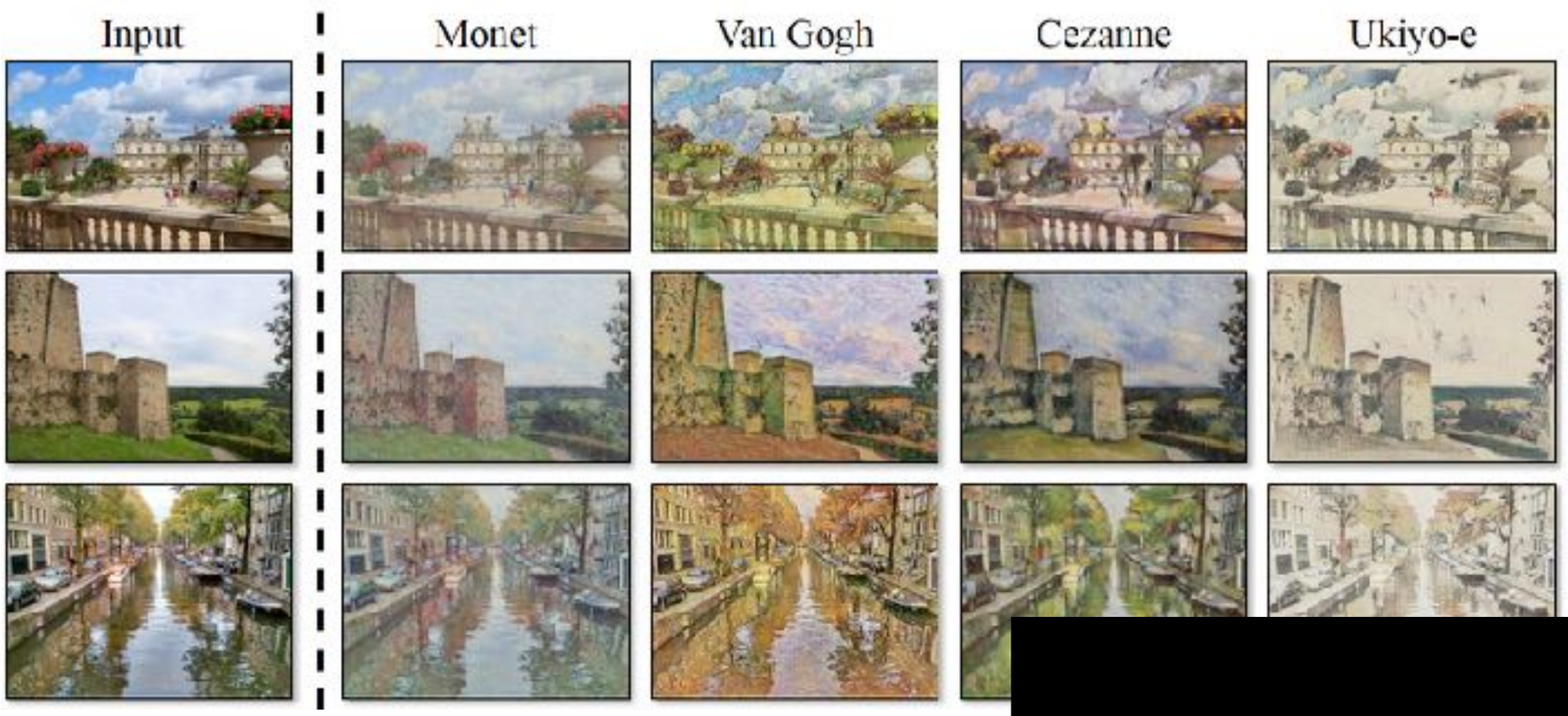
# Part 5. Generative models for machine learning and machine learning in 3D vision



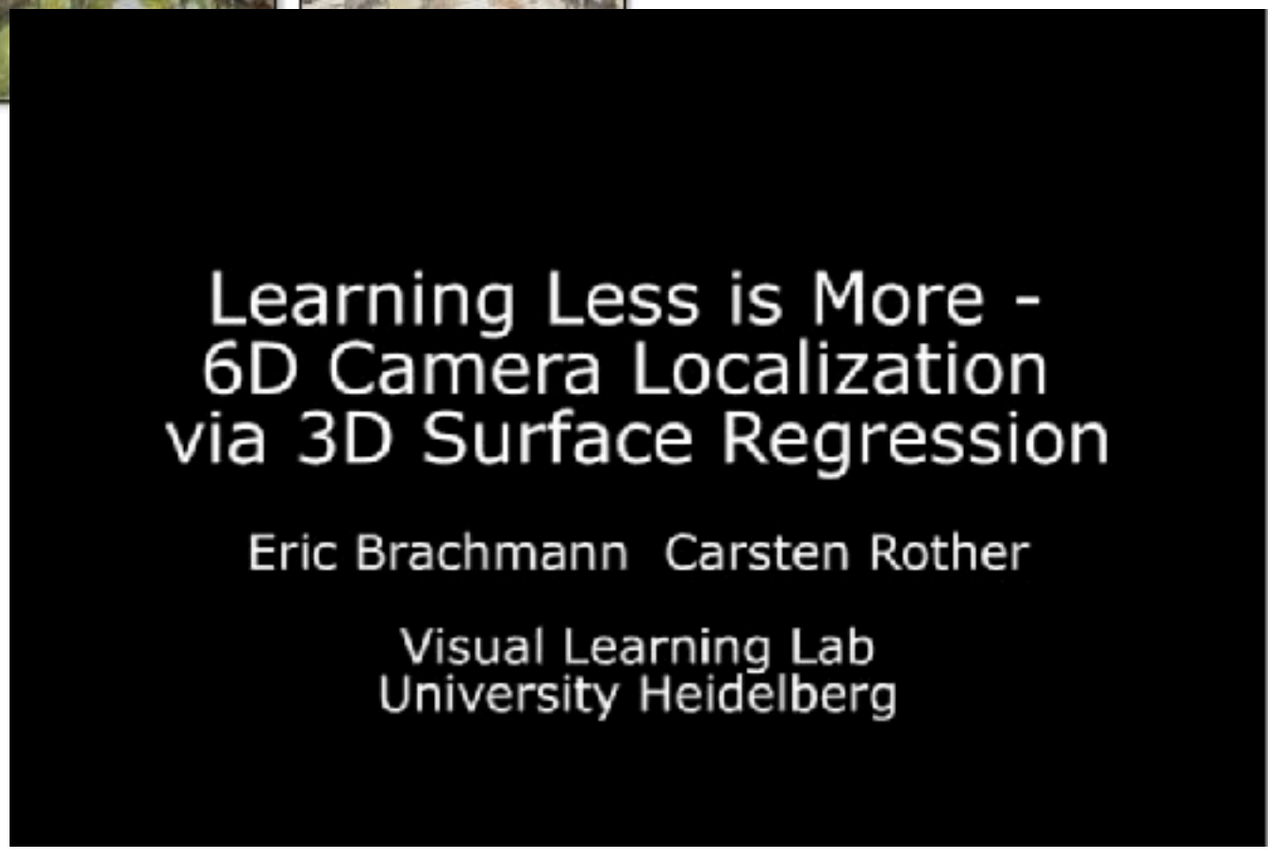

[Zhu et al., ICCV 2017]



# Part 5. Generative models for machine learning and machine learning in 3D vision

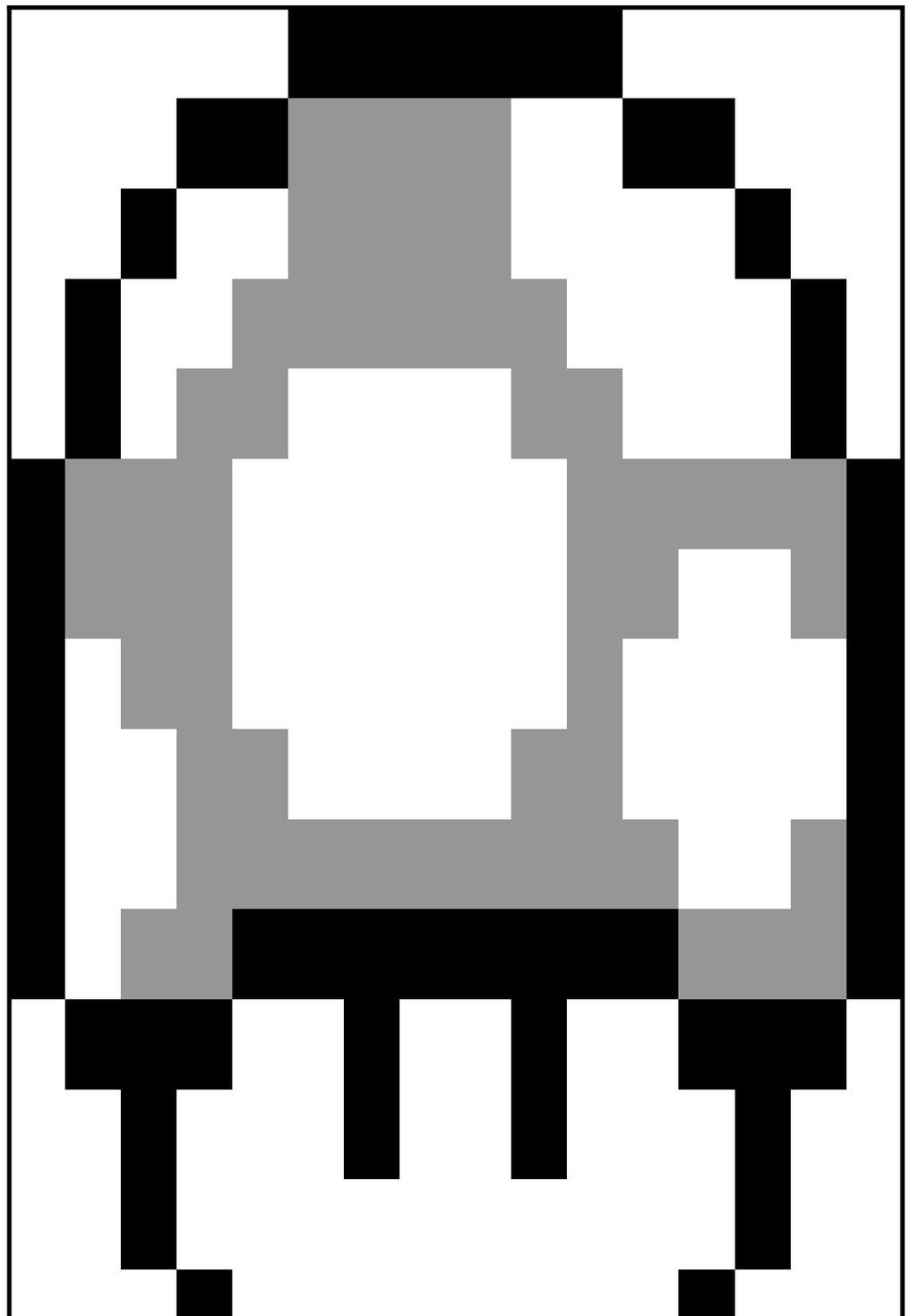



[Zhu et al., ICCV 2017]

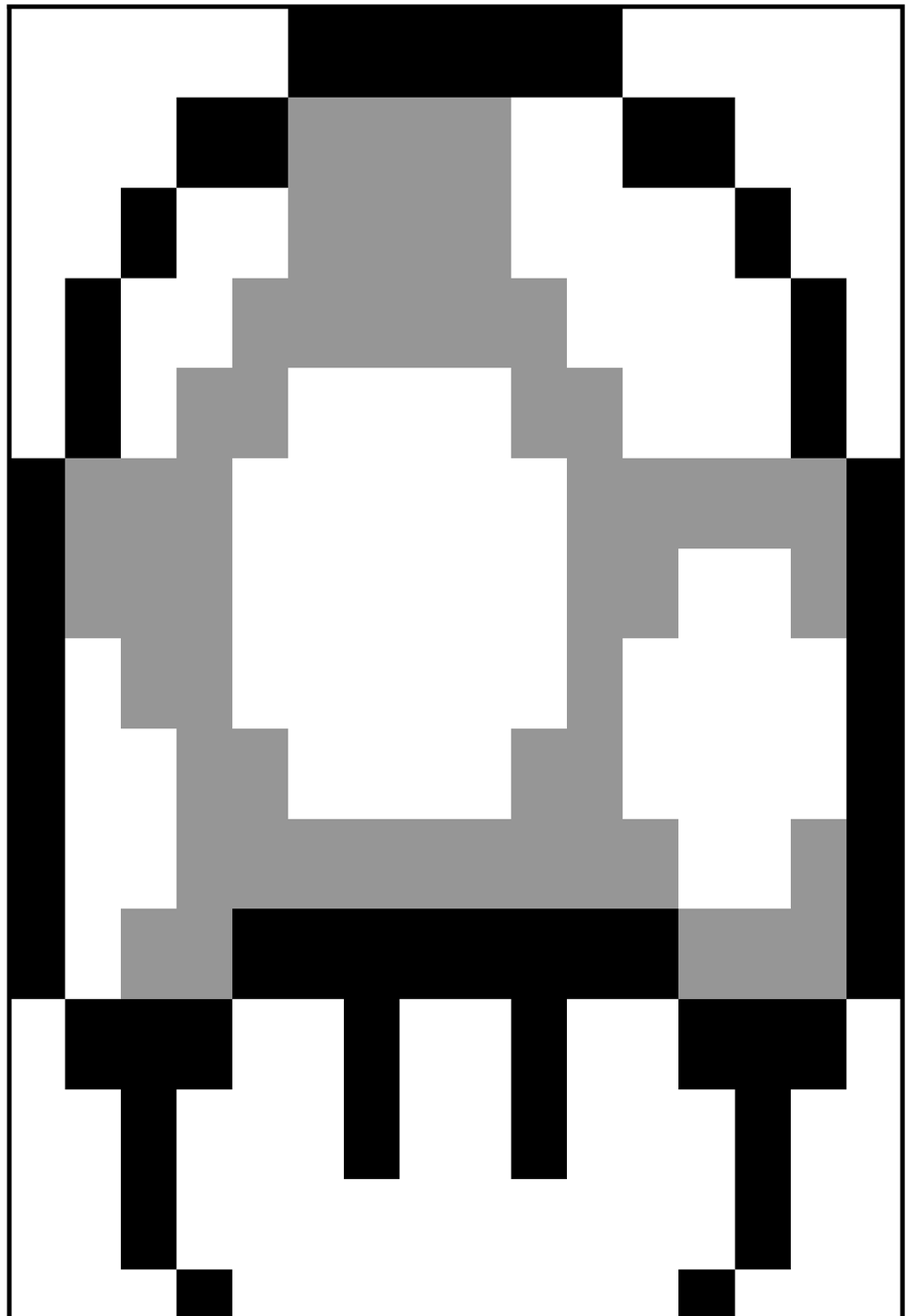


# Modeling Images

# What is an Image?



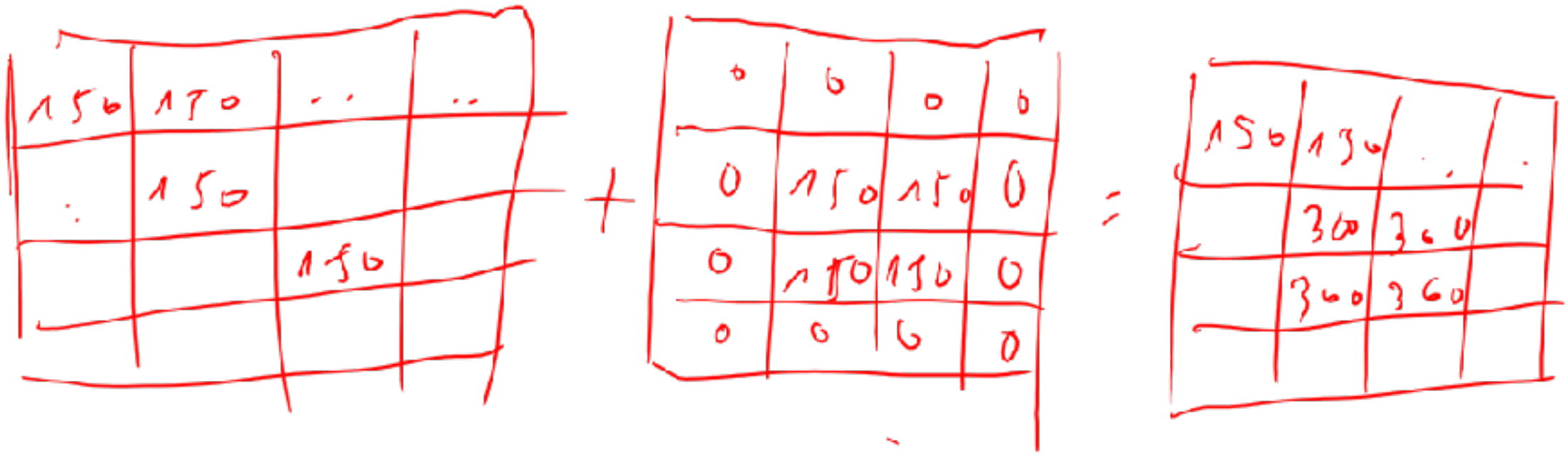
# What is an Image?



| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 255 | 255 | 255 | 255 | 255 | 0 | 0 | 0 | 0 | 0 | 0 | 255 | 255 | 255 | 255 | 255 |
| 255 | 255 | 255 | 0 | 0 | 150 | 150 | 150 | 150 | 255 | 255 | 0 | 0 | 255 | 255 | 255 |
| 255 | 255 | 0 | 255 | 255 | 150 | 150 | 150 | 150 | 255 | 255 | 255 | 255 | 0 | 255 | 255 |
| 255 | 0 | 255 | 255 | 150 | 150 | 150 | 150 | 150 | 150 | 255 | 255 | 255 | 255 | 0 | 255 |
| 255 | 0 | 255 | 150 | 150 | 255 | 255 | 255 | 255 | 150 | 150 | 255 | 255 | 255 | 0 | 255 |
| 0 | 150 | 150 | 150 | 255 | 255 | 255 | 255 | 255 | 255 | 150 | 150 | 150 | 150 | 150 | 0 |
| 0 | 150 | 150 | 150 | 255 | 255 | 255 | 255 | 255 | 255 | 150 | 150 | 255 | 255 | 150 | 0 |
| 0 | 255 | 150 | 150 | 255 | 255 | 255 | 255 | 255 | 255 | 150 | 255 | 255 | 255 | 255 | 0 |
| 0 | 255 | 255 | 150 | 150 | 255 | 255 | 255 | 255 | 150 | 150 | 255 | 255 | 255 | 255 | 0 |
| 0 | 255 | 150 | 150 | 150 | 150 | 150 | 150 | 150 | 150 | 150 | 150 | 255 | 255 | 150 | 0 |
| 0 | 255 | 150 | 150 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 150 | 150 | 150 | 0 |
| 255 | 0 | 0 | 0 | 255 | 255 | 0 | 255 | 255 | 0 | 255 | 255 | 0 | 0 | 0 | 255 |
| 255 | 255 | 0 | 255 | 255 | 255 | 0 | 255 | 255 | 0 | 255 | 255 | 255 | 0 | 255 | 255 |
| 255 | 255 | 0 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 0 | 255 | 255 |
| 255 | 255 | 255 | 0 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 0 | 255 | 255 | 255 |
| 255 | 255 | 255 | 255 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 255 | 255 | 255 | 255 |

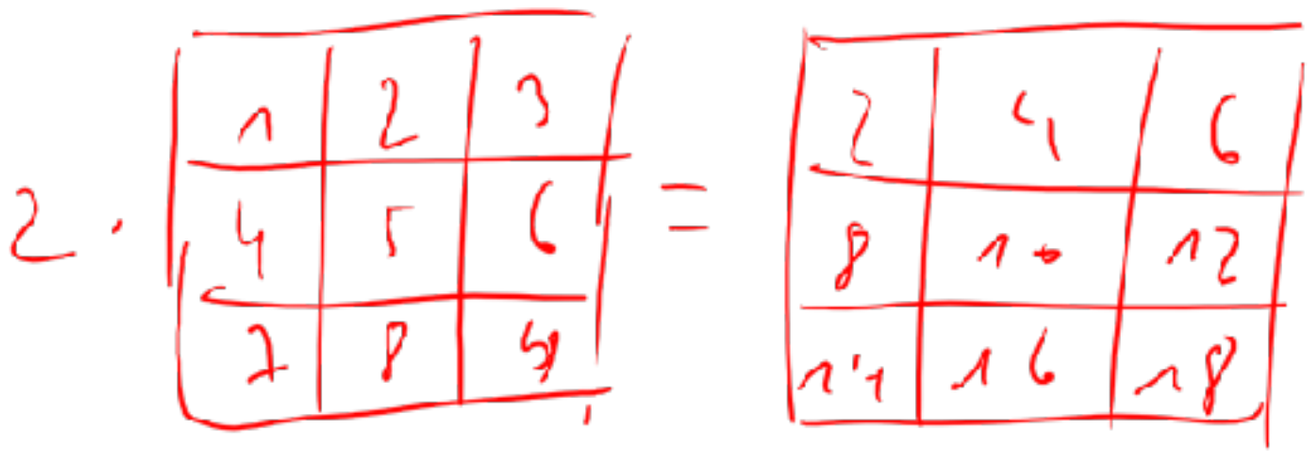
# Images are Vectors

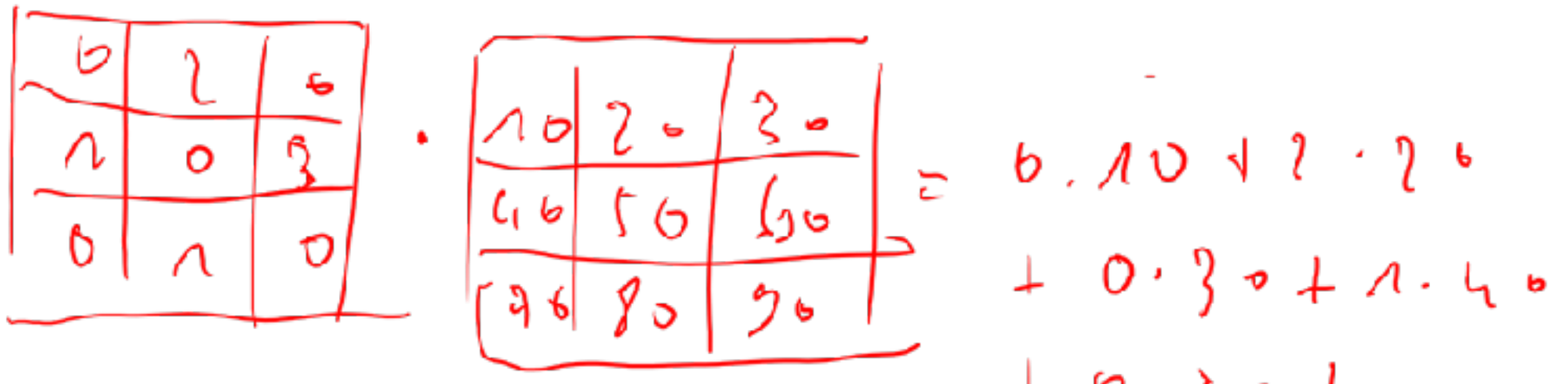
Addition, Subtraction



# Images are Vectors

Scalar multiplication, dot product





# Images are Vectors


Average
speed limit sign

# Images are Vectors

Average
speed limit sign

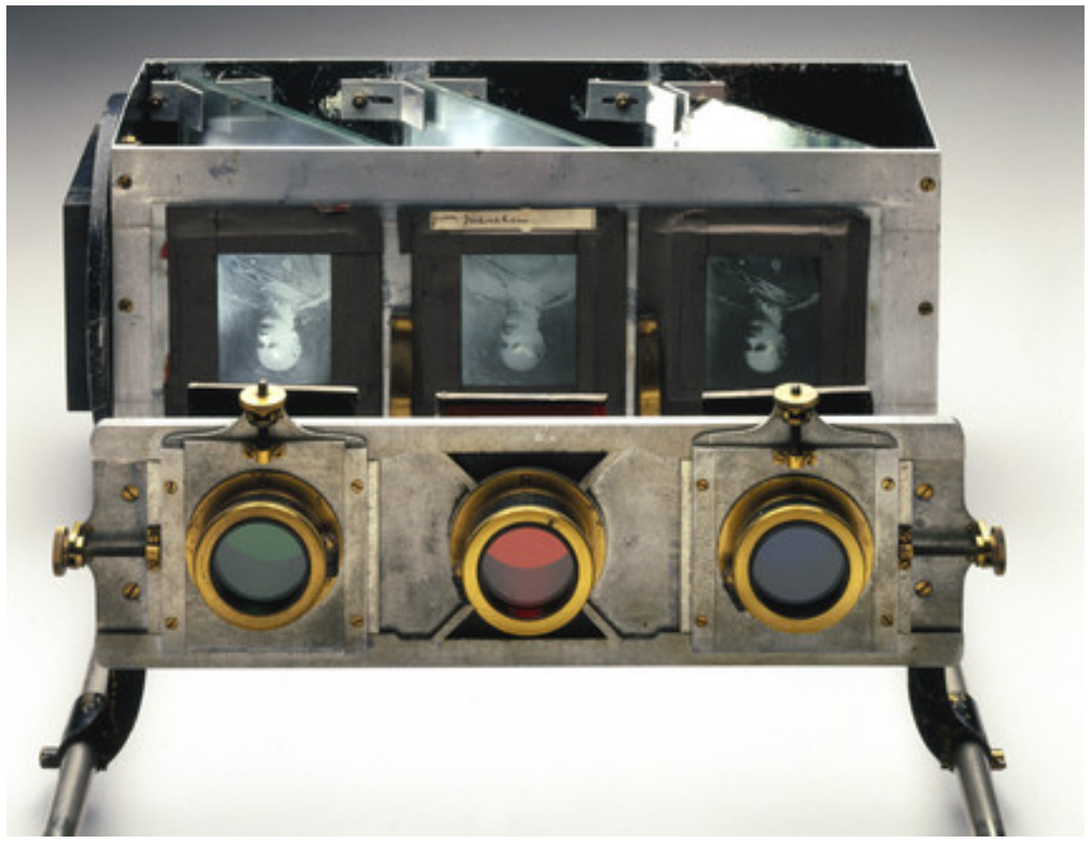
image source: pinterest

Average
president

# Color

# Color

# Color



# RGB

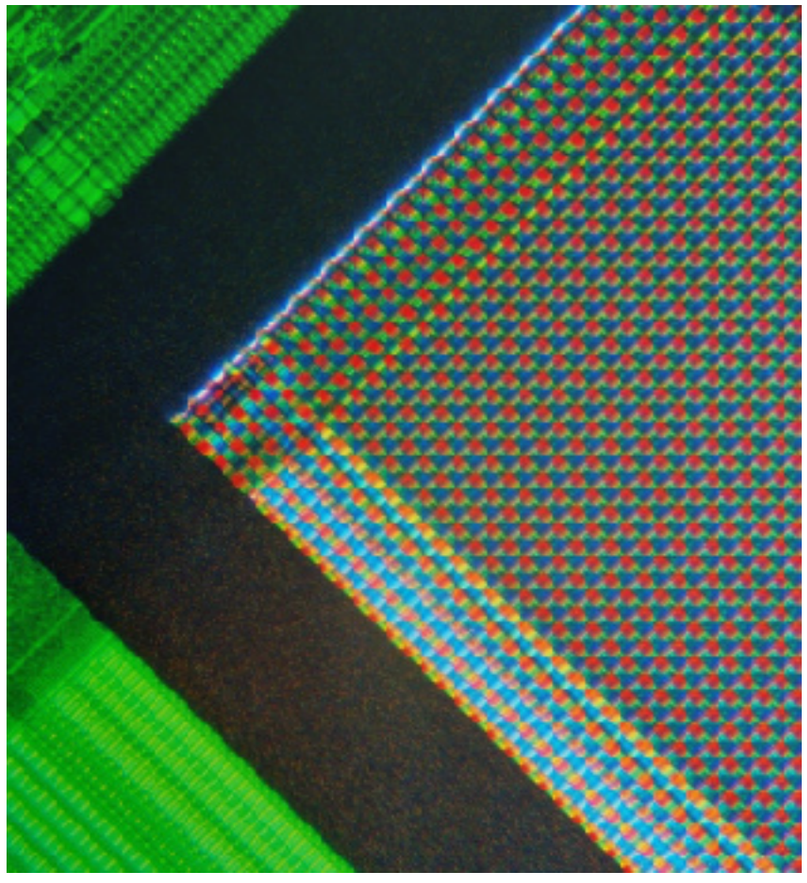
| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | 122 | 124 | 145 | 153 | 100 | 123 | 109 |
| | 233 | 232 | 211 | 198 | 188 | 177 | 155 |
| 54 | 121 | 121 | 121 | 78 | 13 | 87 | 76 |
| 54 | 56 | 78 | 89 | 23 | 23 | 12 | 12 |
| 255 78 | 18 | 24 | 25 | 53 | 23 | 54 | 23 |
| 255 255 | 23 | 34 | 24 | 35 | 54 | 23 | 34 |
| 255 244 | 34 | 53 | 35 | 35 | 52 | 43 | 63 |
| 255 | 34 | 56 | 34 | 87 | 76 | 56 | 87 |
| 244 | 98 | 9 | 76 | 58 | 48 | 59 | 28 |
| | 67 | 28 | 38 | 28 | 28 | 38 | 38 |
| | 28 | 8 | 0 | 29 | 9 | 82 | 0 |

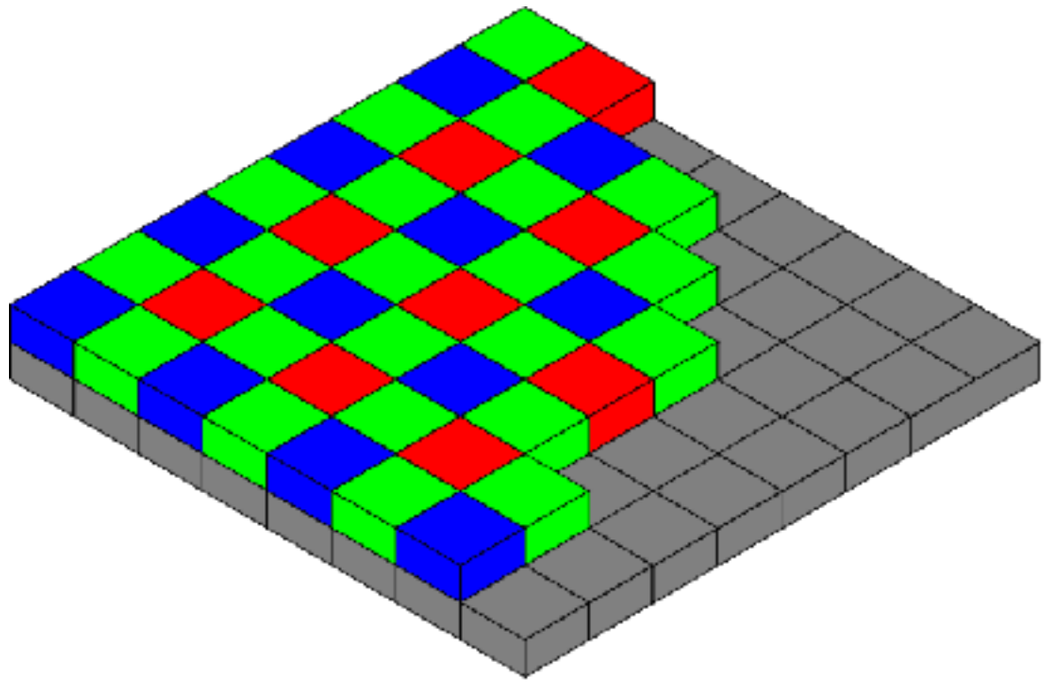
# RGB

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | 122 | 124 | 145 | 153 | 100 | 123 | 109 |
| | 233 | 232 | 211 | 198 | 188 | 177 | 155 |
| 54 | 121 | 121 | 121 | 78 | 13 | 87 | 76 |
| 54 | 56 | 78 | 89 | 23 | 23 | 12 | 12 |
| 255 78 | 18 | 24 | 25 | 53 | 23 | 54 | 23 |
| 255 255 | 23 | 34 | 24 | 35 | 54 | 23 | 34 |
| 255 244 | 34 | 53 | 35 | 35 | 52 | 43 | 63 |
| 255 | 34 | 56 | 34 | 87 | 76 | 56 | 87 |
| 244 | 98 | 9 | 76 | 58 | 48 | 59 | 28 |
| | 67 | 28 | 38 | 28 | 28 | 38 | 38 |
| | 28 | 8 | 0 | 29 | 9 | 82 | 0 |

Channel-wise operations

# Digital Cameras

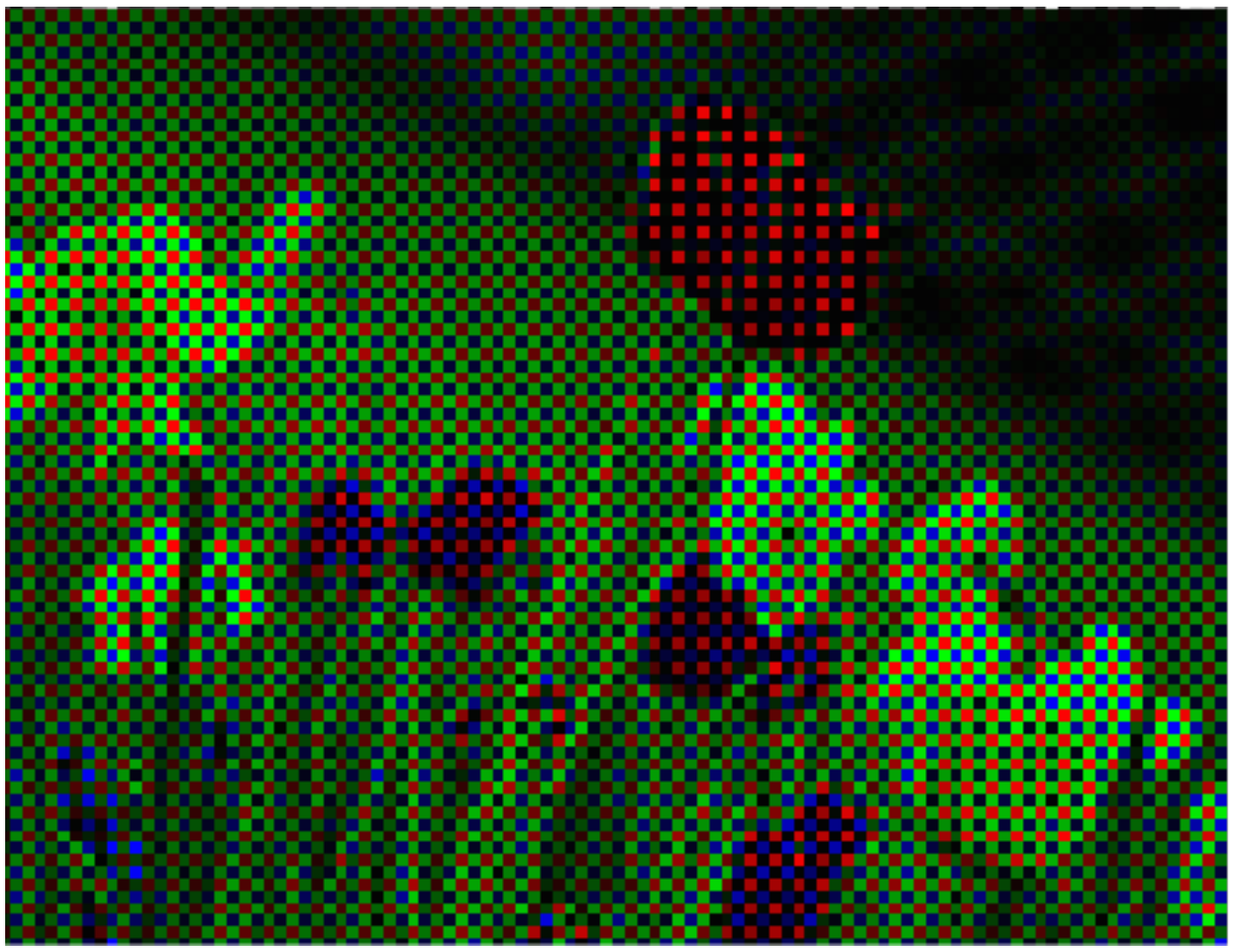




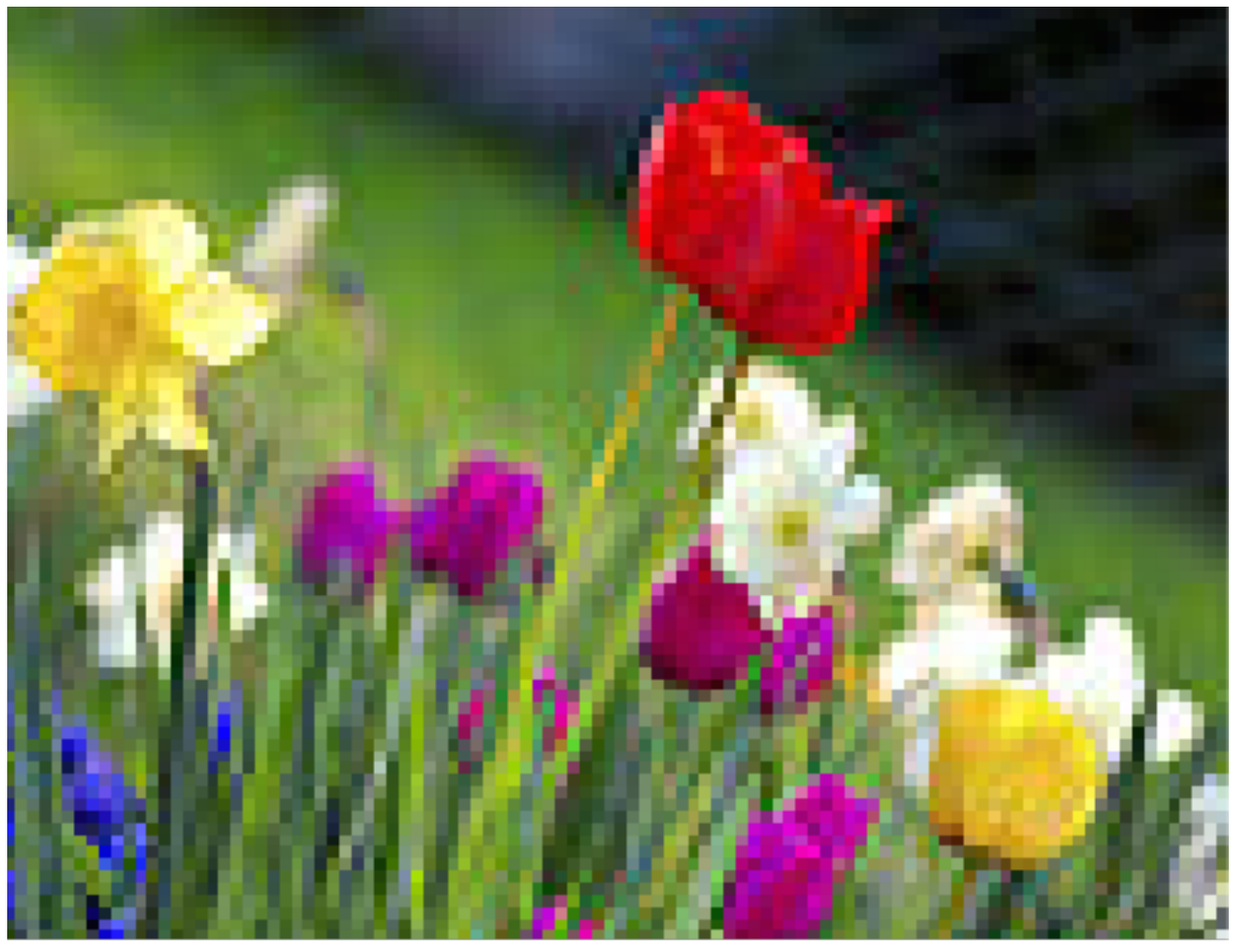
Bayer filter

# Digital Cameras

# Digital Cameras



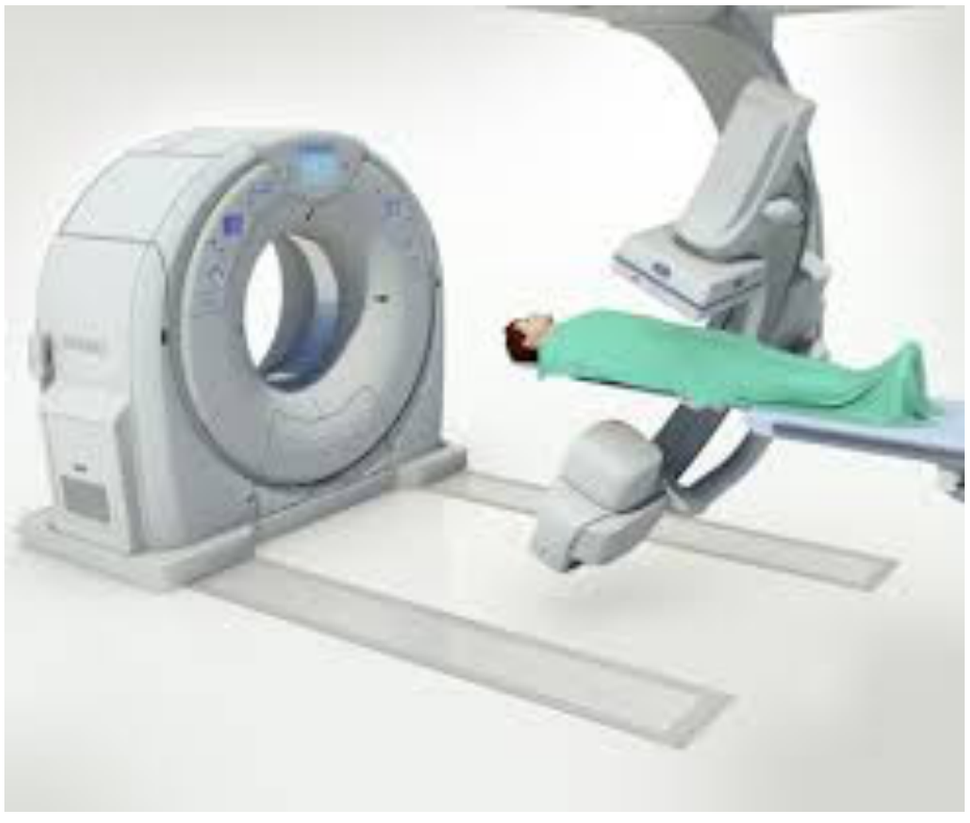
# Digital Cameras



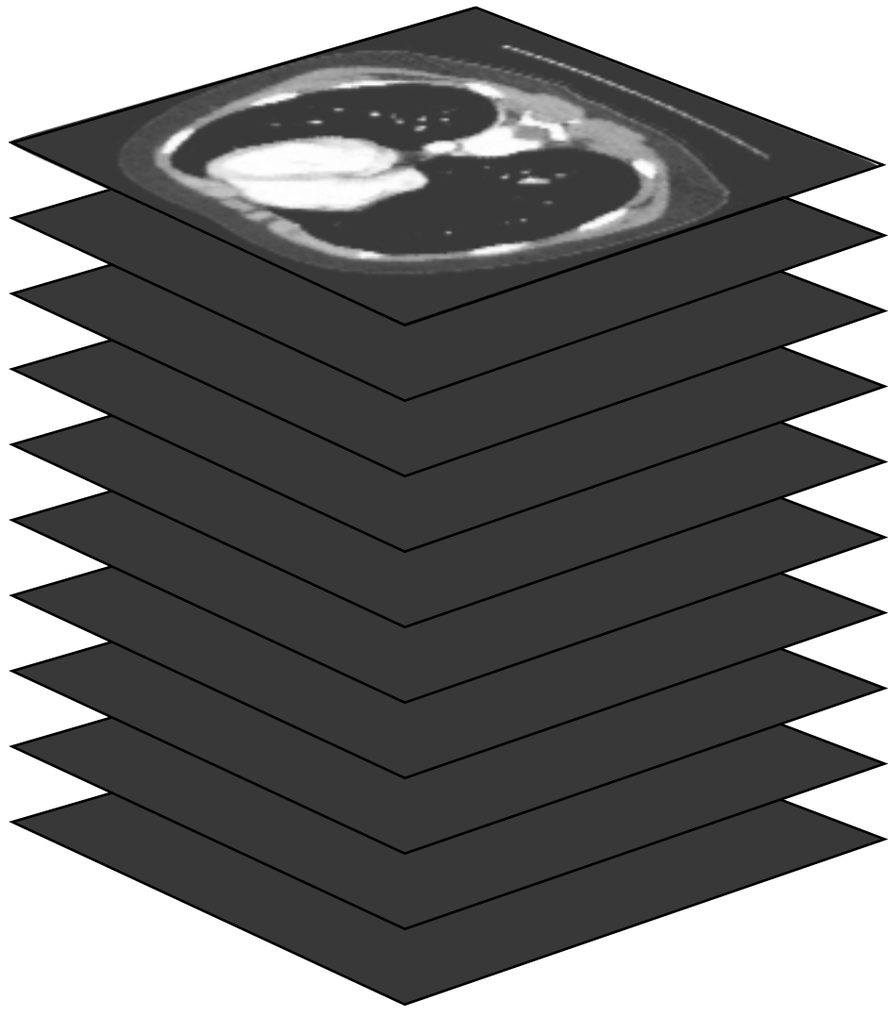
Demosaicing

# Lighting and White Balance

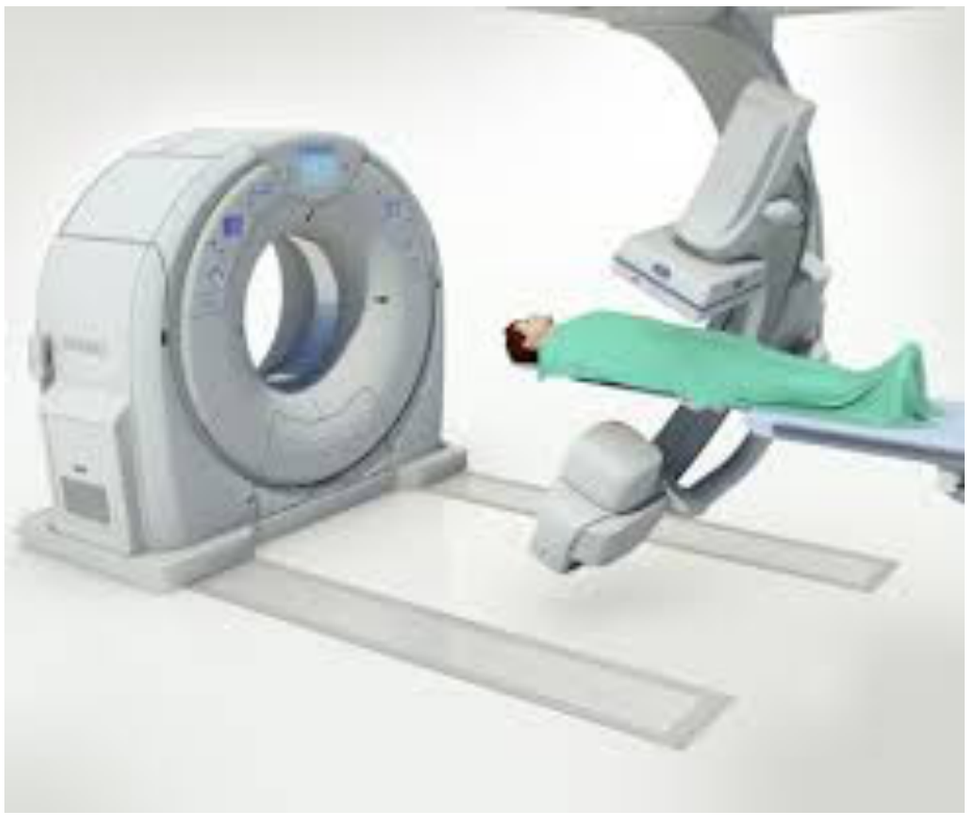
# 3D Images



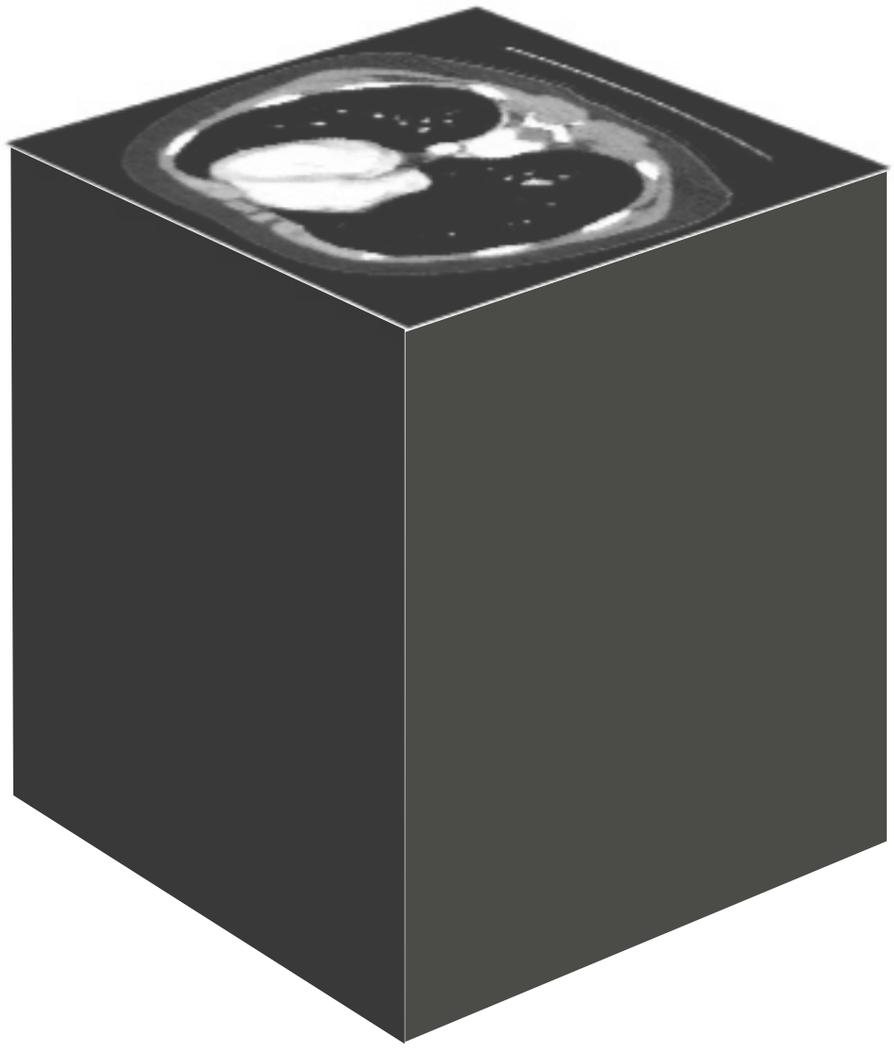
Computed tomography



# 3D Images

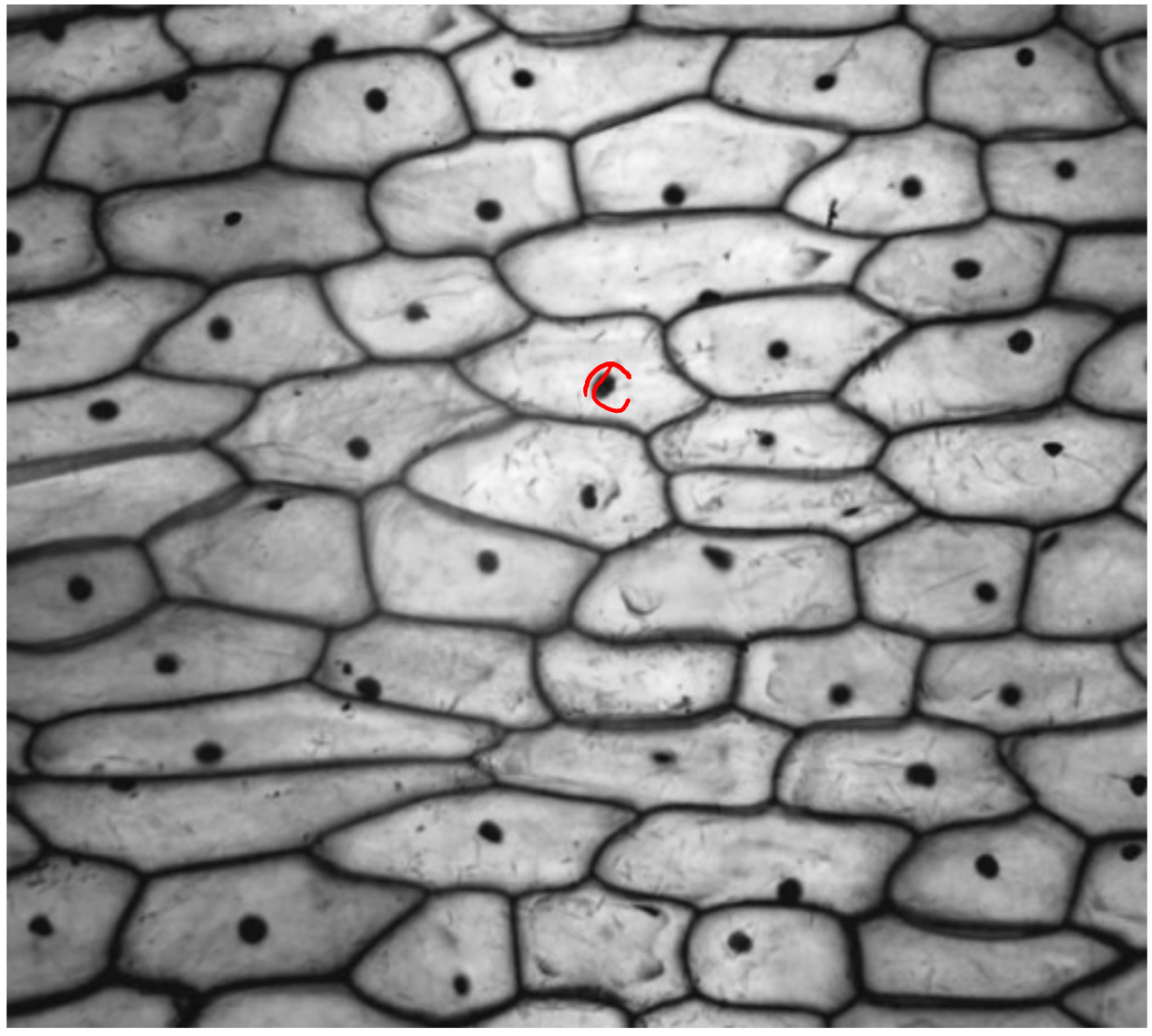


Computed tomography



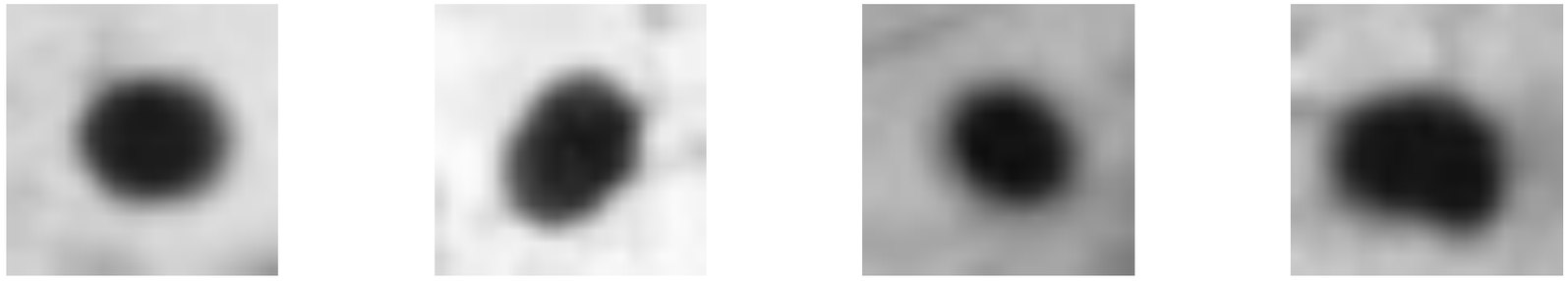
# Filtering and Linear Classifiers

# Count the Nuclei

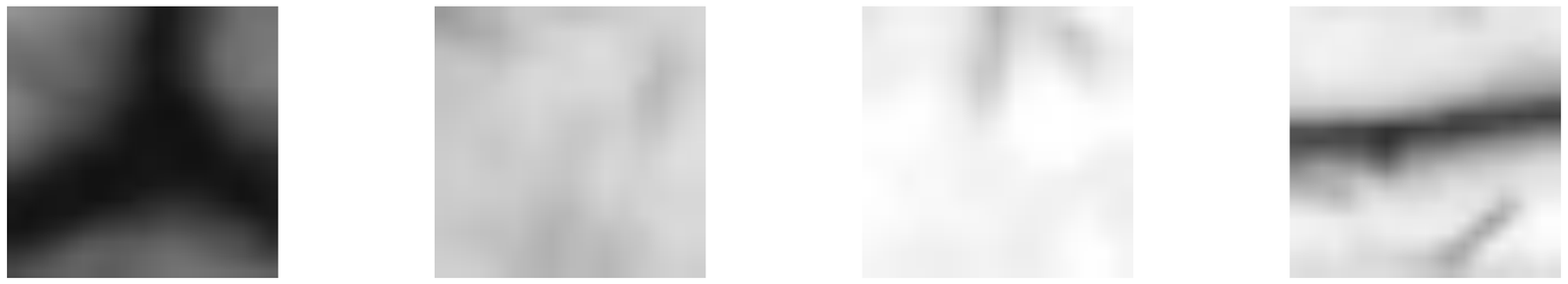


# Classification

nuclei



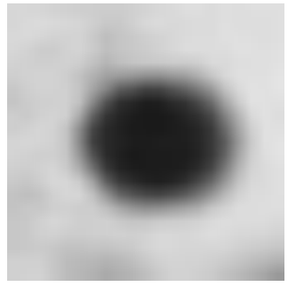
other

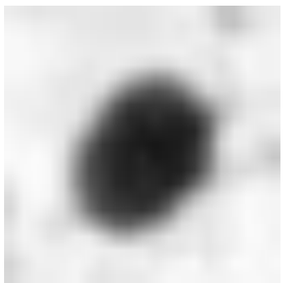


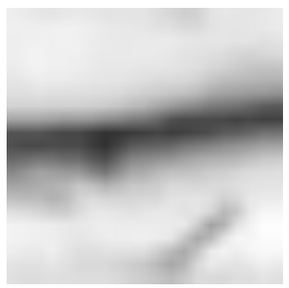
21 x 21 patches

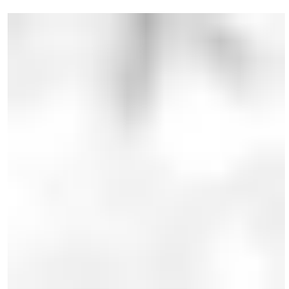
# A Linear Classifier

TARGET ->

$$w \cdot$$  $$> \tau$$

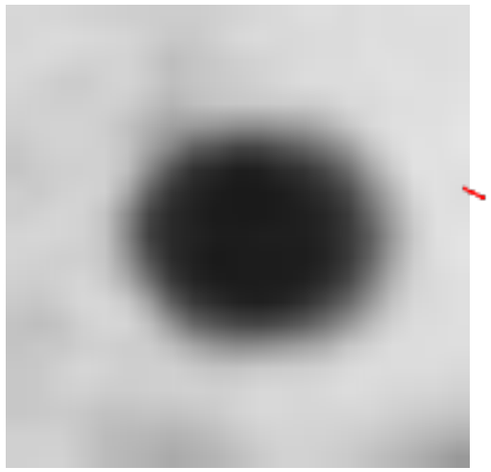
$$w \cdot$$  $$> \tau$$

$$w \cdot$$  $$< \tau$$

$$w \cdot$$  $$< \tau$$

# Designing a Classifier

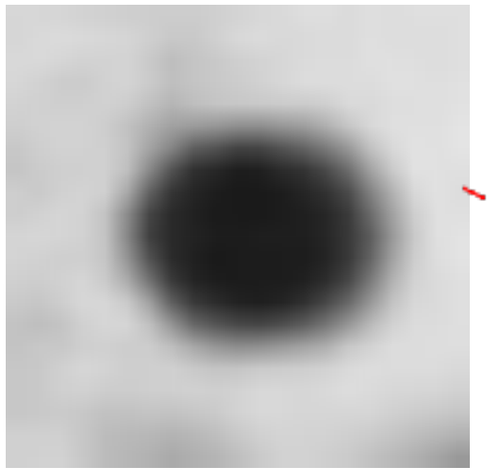

$\cdot$



$> \tau$

# Designing a Classifier


$\cdot$

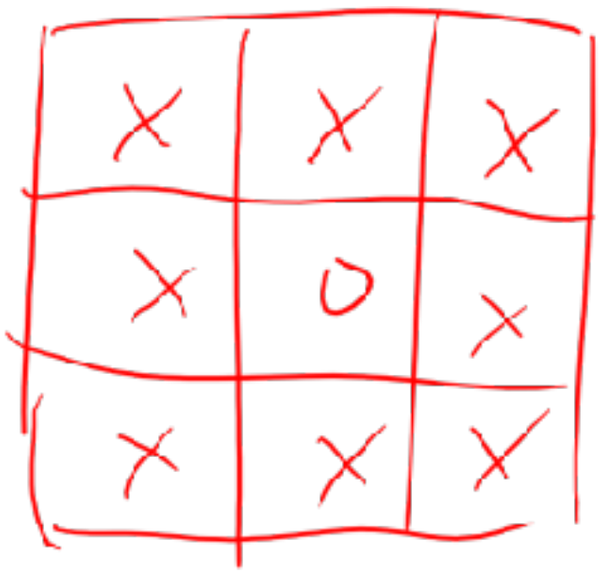


$> \tau$

# Designing a Better Classifier

$$w = \frac{1}{N}(w' - \mu_{w'} 1)$$

$w' =$



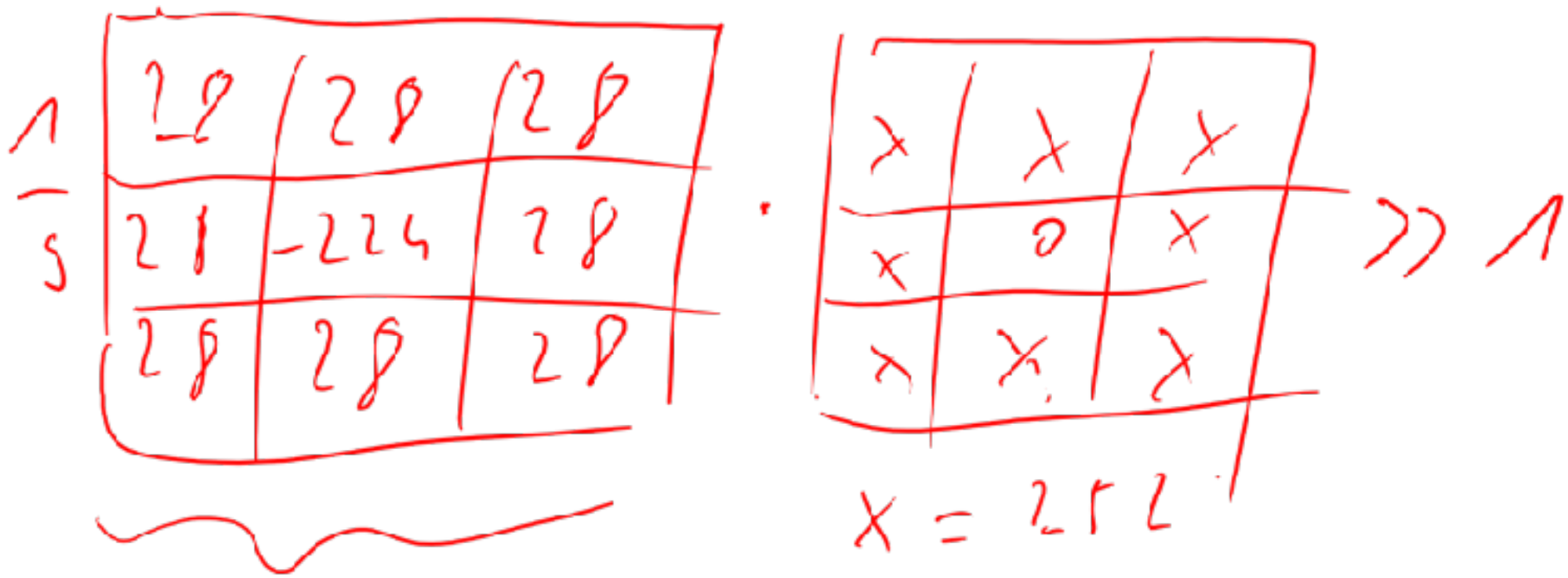
$x = 252$

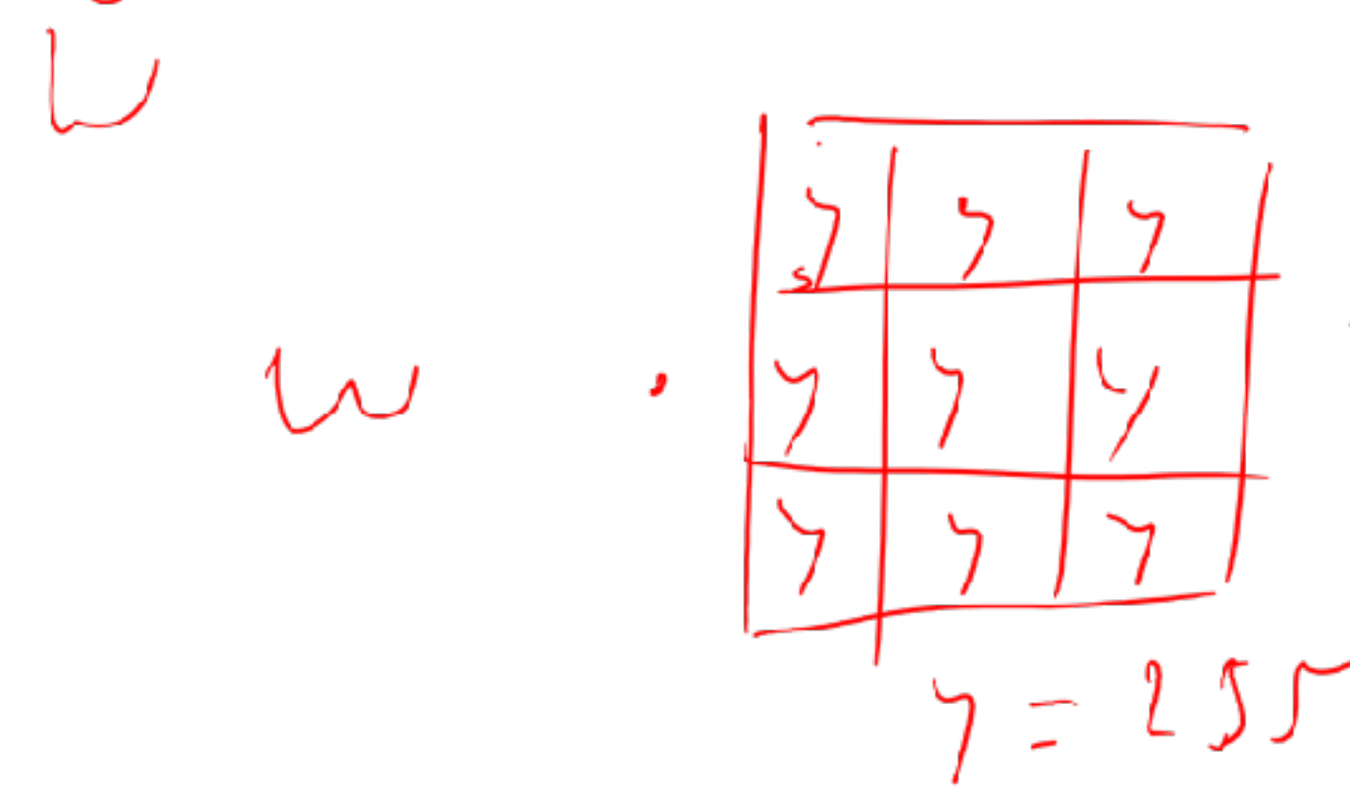
$w = \frac{1}{9}\left(\begin{bmatrix} x & x & x \\ x & o & x \\ x & x & x \end{bmatrix} - \begin{bmatrix} \mu_{w'} & \cdots \\ \vdots & \\ \vdots & \end{bmatrix}\right) =$

mean: $\mu_{w'} = 224$

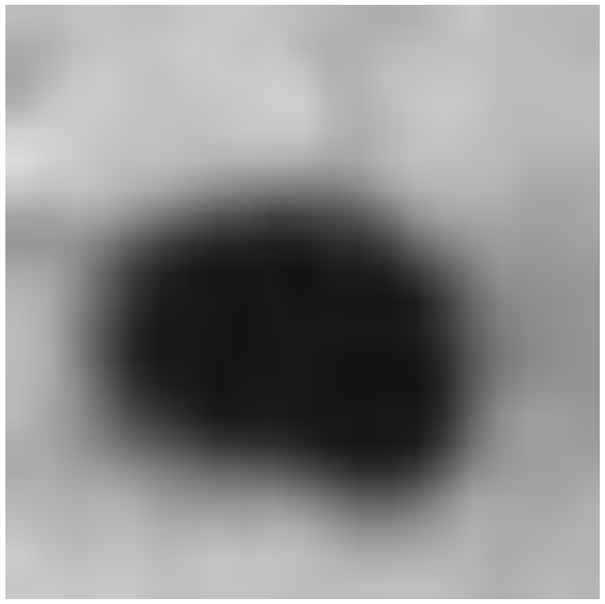
$= \frac{1}{3}\begin{bmatrix} 28 & \cdots \\ \vdots & -224 \\ \vdots & 28 \end{bmatrix}$

# Designing a Better Classifier



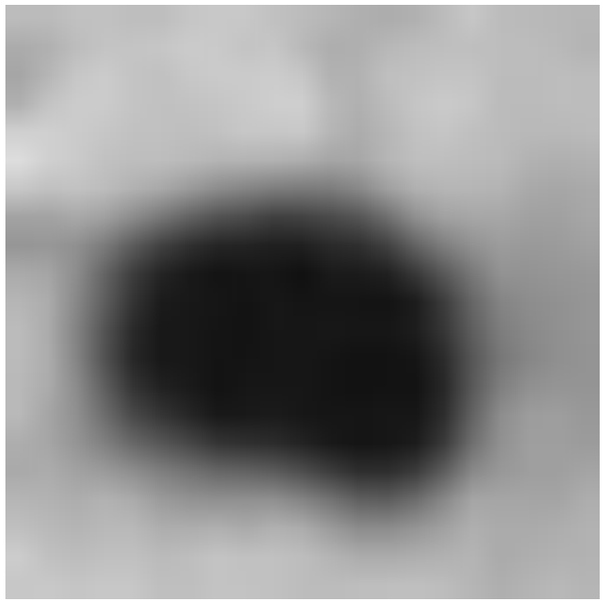


# Choosing A Template

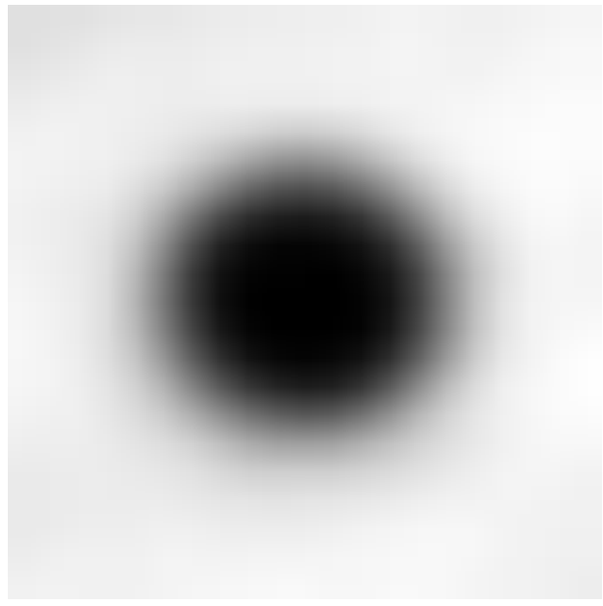
$w =$ 

some nucleus

# Choosing A Template

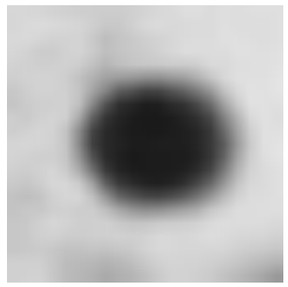
$w =$ 

some nucleus

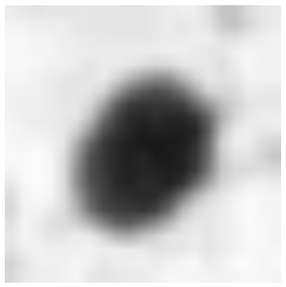
$w =$ 

mean nucleus

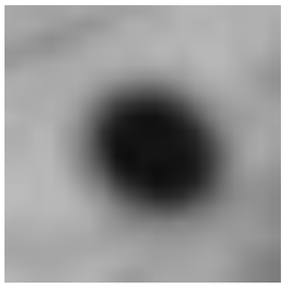
# Choosing a Threshold



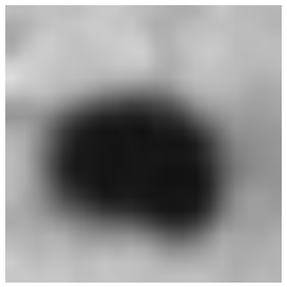
1.72



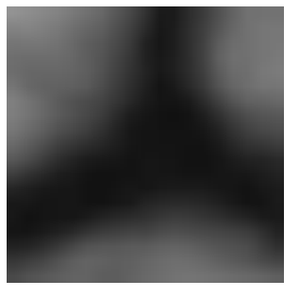
1.88



1.26



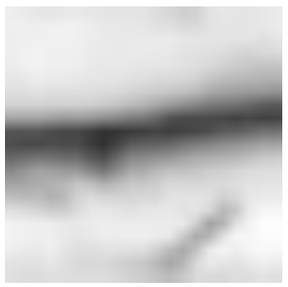
1.45



0.56

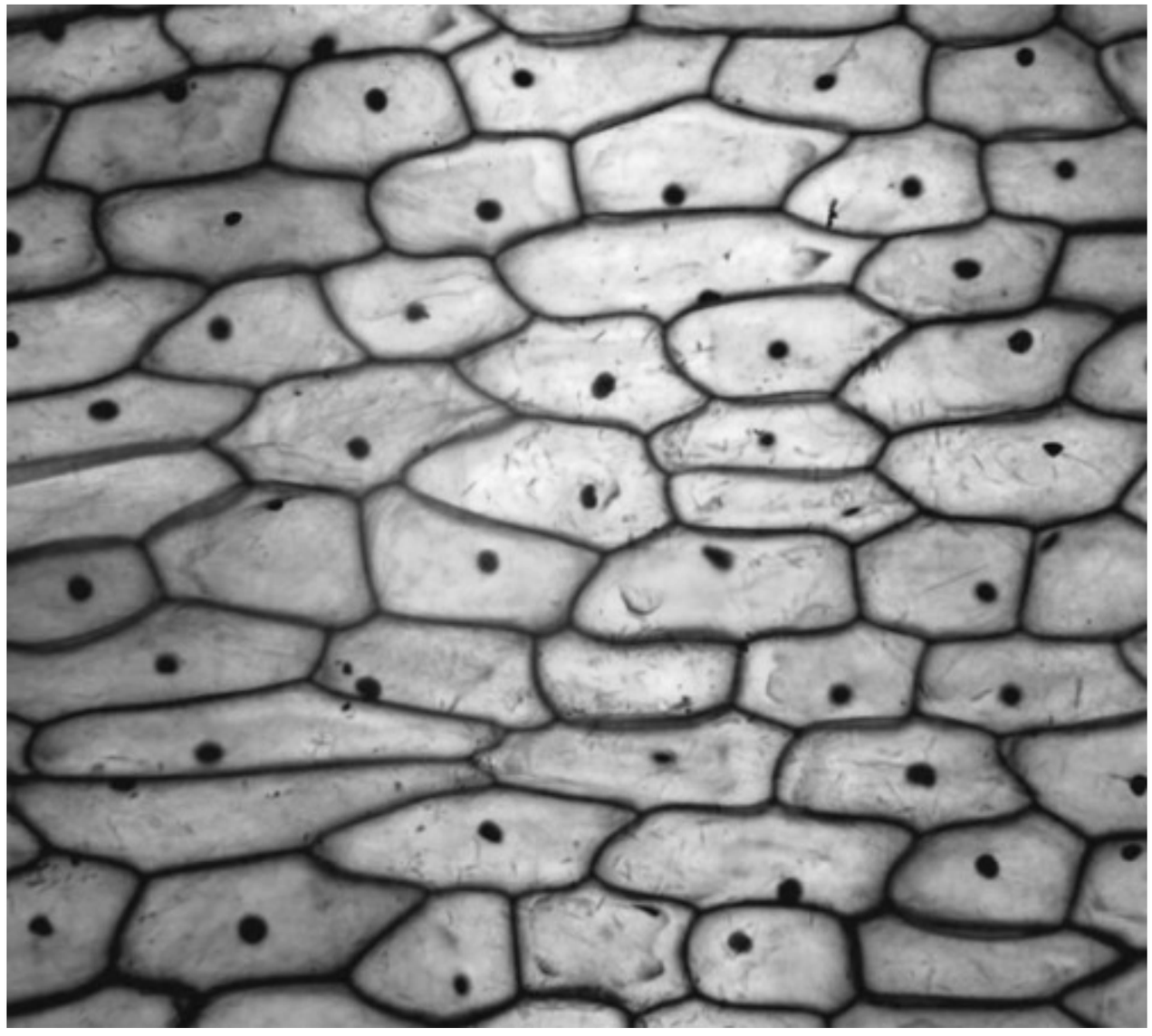
-0.07

-0.06

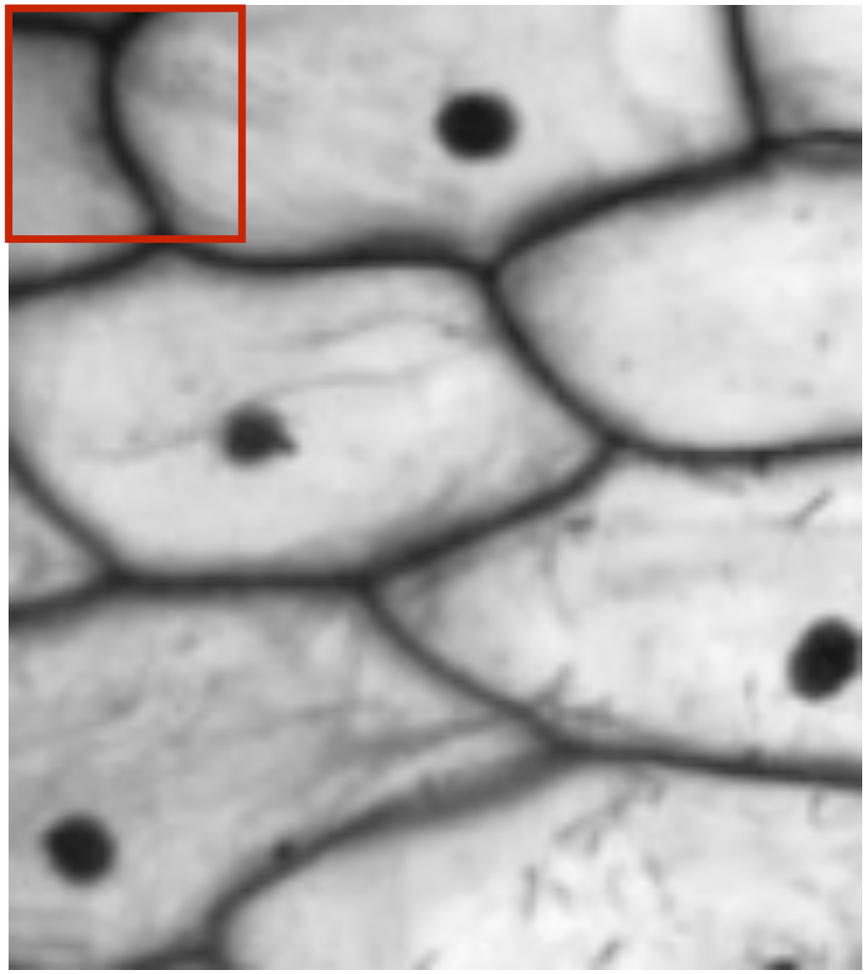


0.75

# Count the Nuclei

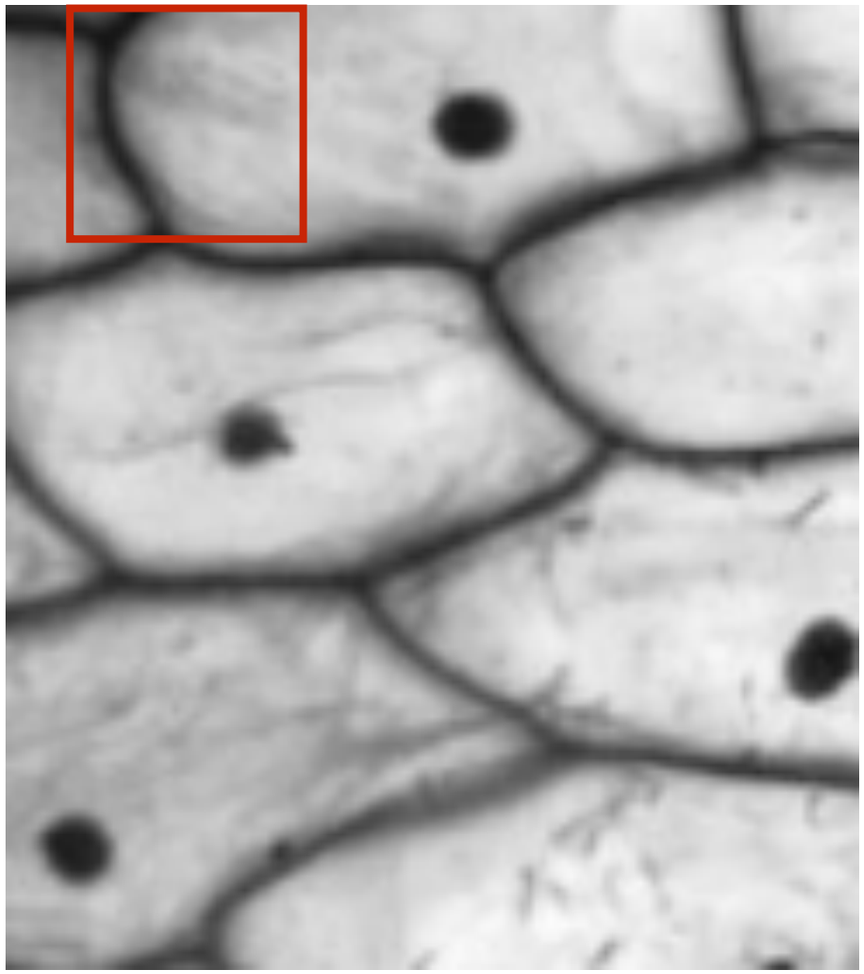


# Sliding Window Classification



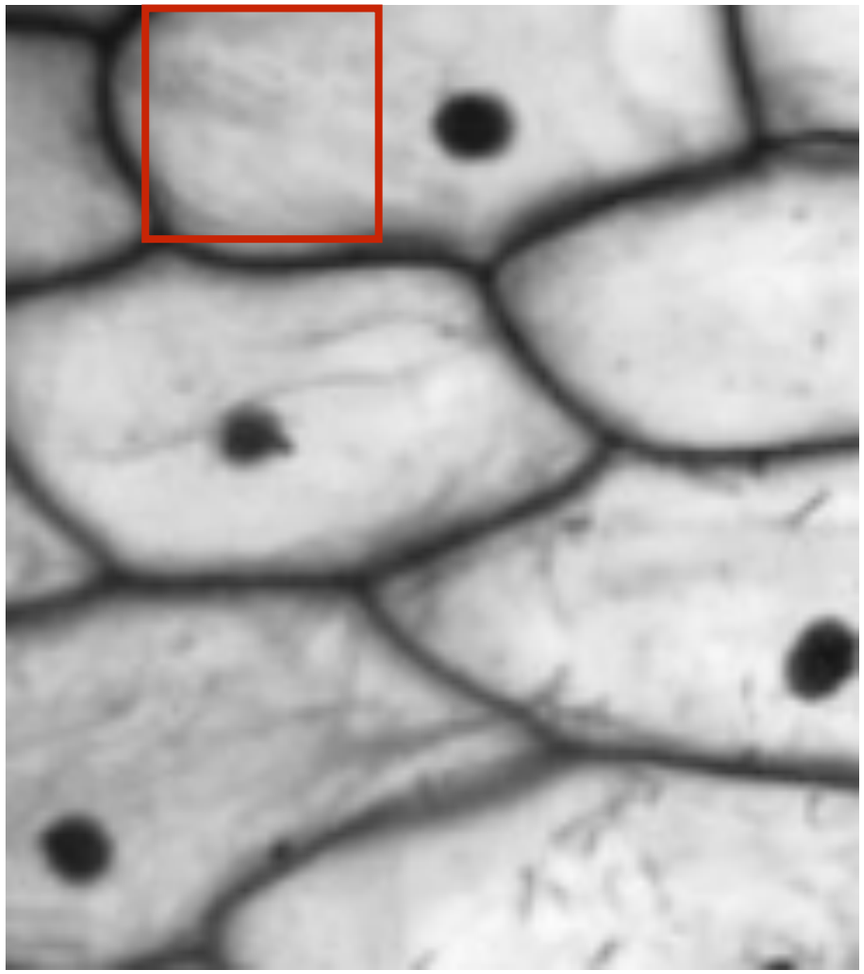
$$\cdot\, w < \tau$$

# Sliding Window Classification



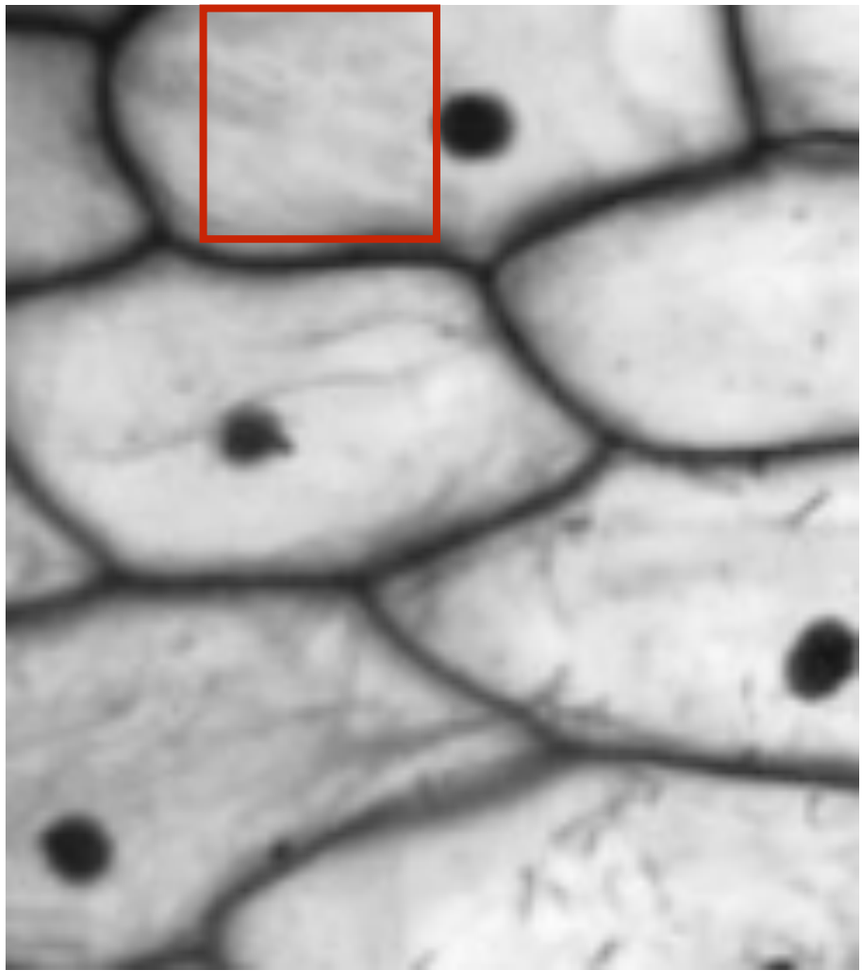
$$\cdot\, w < \tau$$

# Sliding Window Classification



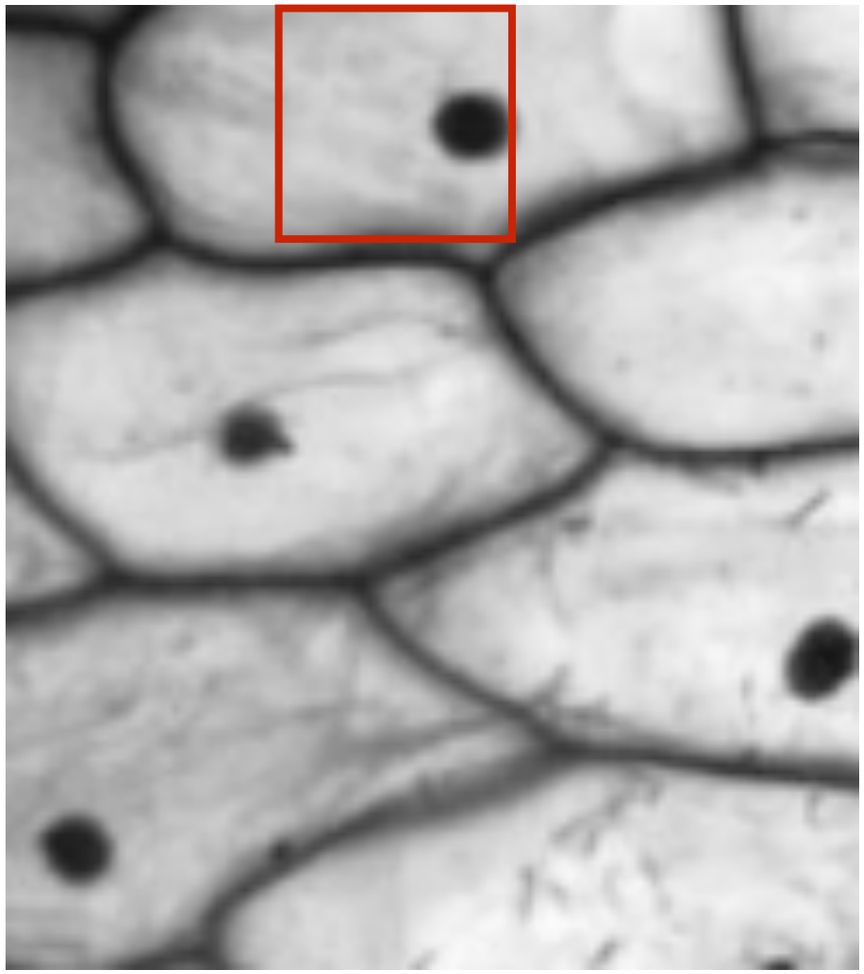
$$\cdot\, w < \tau$$

# Sliding Window Classification



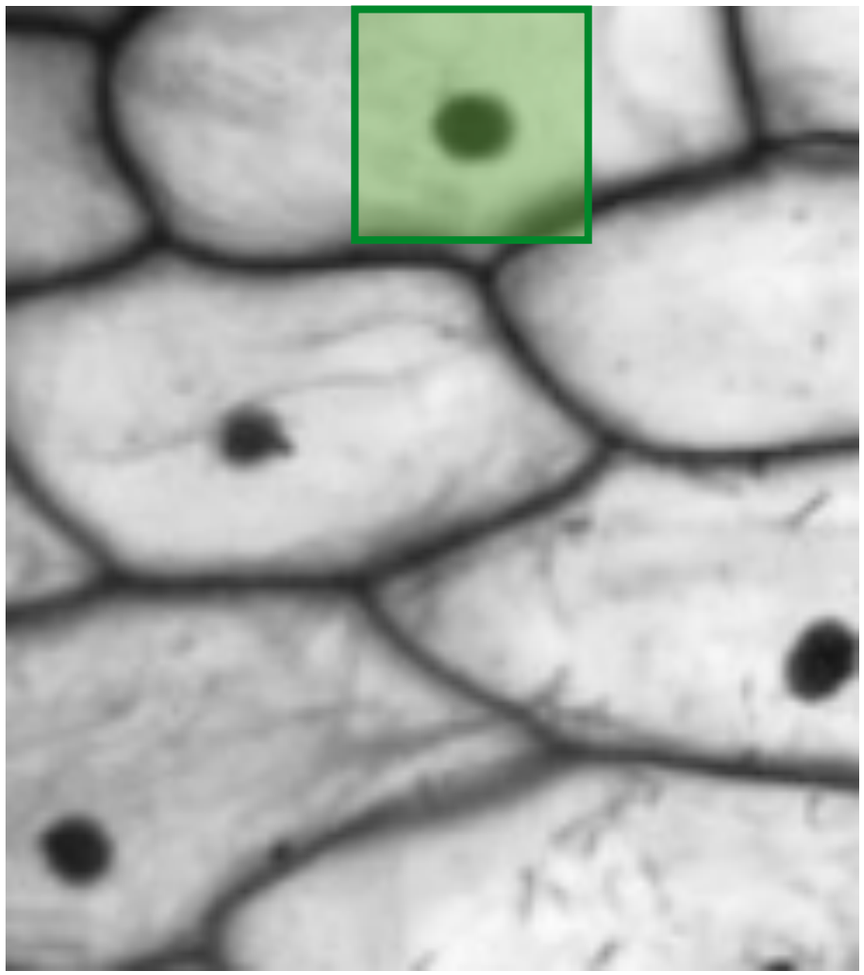
$$\cdot\, w < \tau$$

# Sliding Window Classification



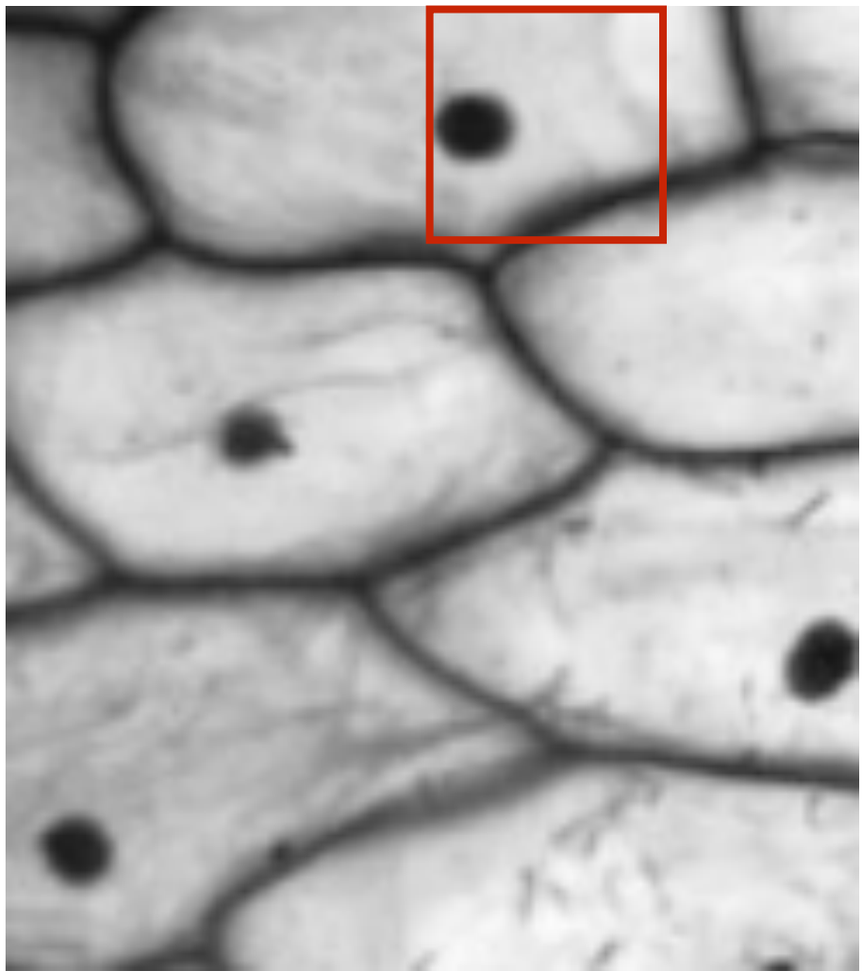
$$\cdot\, w < \tau$$

# Sliding Window Classification



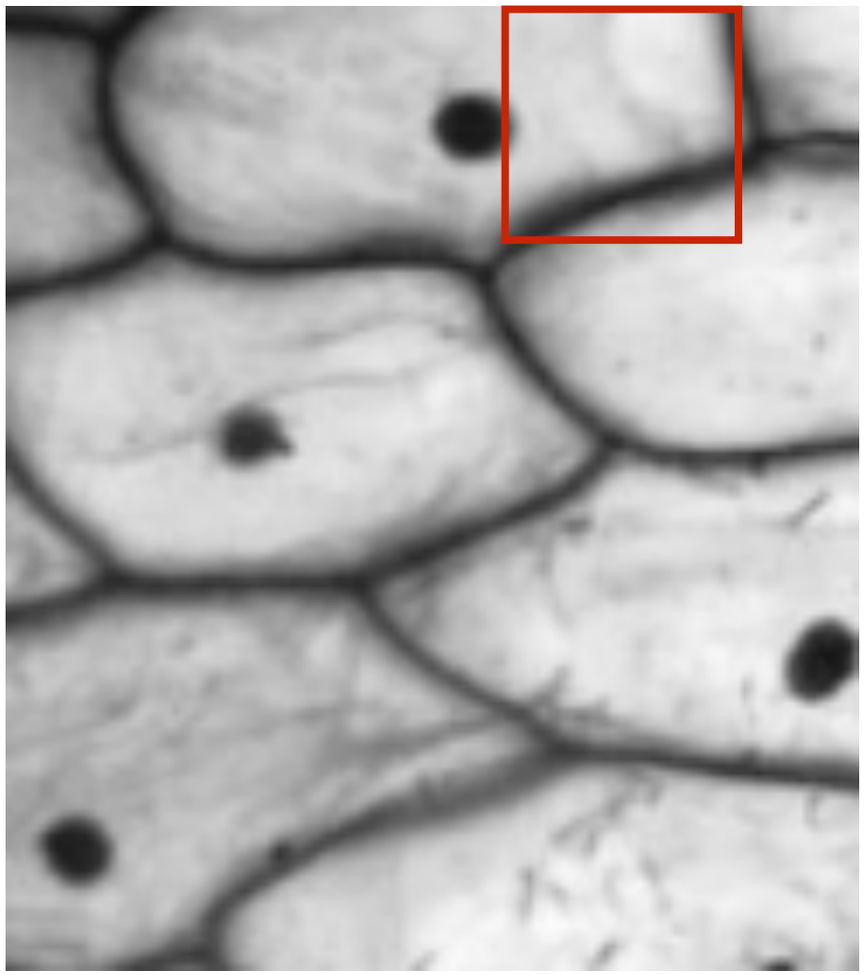
$$\cdot w > \tau$$

# Sliding Window Classification



$$\cdot\, w < \tau$$

# Sliding Window Classification



$$\cdot\, w < \tau$$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

$I \otimes w$

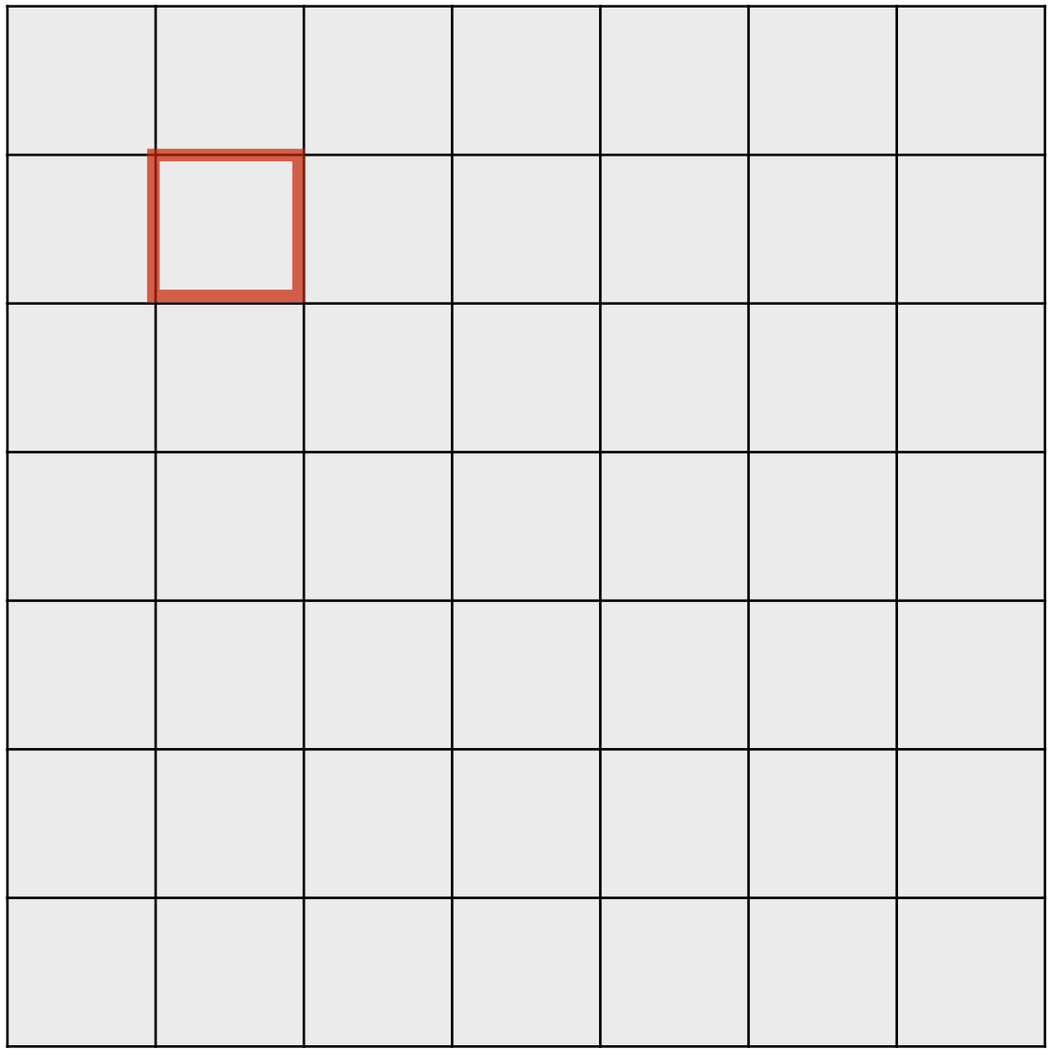
# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$



$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 1 | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 1 | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 1 | 2 | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 1 | 2 | 1 | 0 | |
| | 1 | 5 | 6 | 4 | 1 | |
| | 2 | 6 | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 1 | 2 | 1 | 0 | |
| | 1 | 5 | 6 | 4 | 1 | |
| | 2 | 6 | 8 | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 1 | 2 | 1 | 0 | |
| | 1 | 5 | 6 | 4 | 1 | |
| | 2 | 6 | 8 | 4 | 1 | |
| | 1 | 4 | 4 | 2 | 0 | |
| | 0 | 1 | 1 | 0 | 0 | |
| | | | | | | |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 1 | 2 | 1 | 0 | |
| | 1 | 5 | 6 | 4 | 1 | |
| | 2 | 6 | 8 | 4 | 1 | |
| | 1 | 4 | 4 | 2 | 0 | |
| | 0 | 1 | 1 | 0 | 0 | |
| | | | | | | |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | | | | | | |
| | 0 | 1 | 2 | 1 | 0 | |
| | 1 | 5 | 6 | 4 | 1 | |
| | 2 | 6 | 8 | 4 | 1 | |
| | 1 | 4 | 4 | 2 | 0 | |
| | 0 | 1 | 1 | 0 | 0 | |
| | | | | | | |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}\begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | | | | | |
| | 0 | 1 | 2 | 1 | 0 | |
| | 1 | 5 | 6 | 4 | 1 | |
| | 2 | 6 | 8 | 4 | 1 | |
| | 1 | 4 | 4 | 2 | 0 | |
| | 0 | 1 | 1 | 0 | 0 | |
| | | | | | | |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 2 | 1 | 0 | 0 |
| 0 | 1 | 5 | 6 | 4 | 1 | 0 |
| 0 | 2 | 6 | 8 | 4 | 1 | 0 |
| 0 | 1 | 4 | 4 | 2 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$I \otimes w$

# Sliding Dot Product

$$w = \frac{1}{5}$$

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

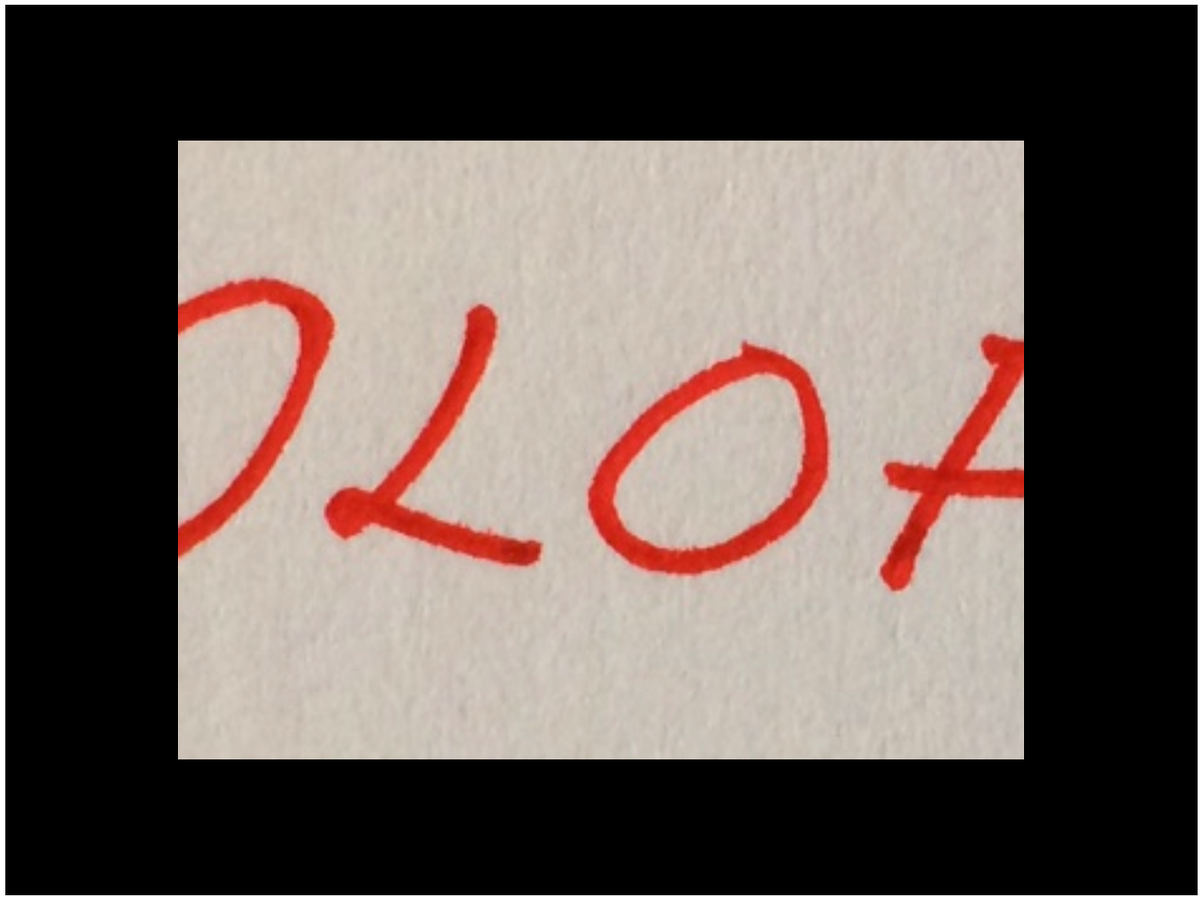
| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 5 | 10 | 5 | 0 | 0 |
| 0 | 0 | 10 | 10 | 5 | 0 | 0 |
| 0 | 0 | 5 | 5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

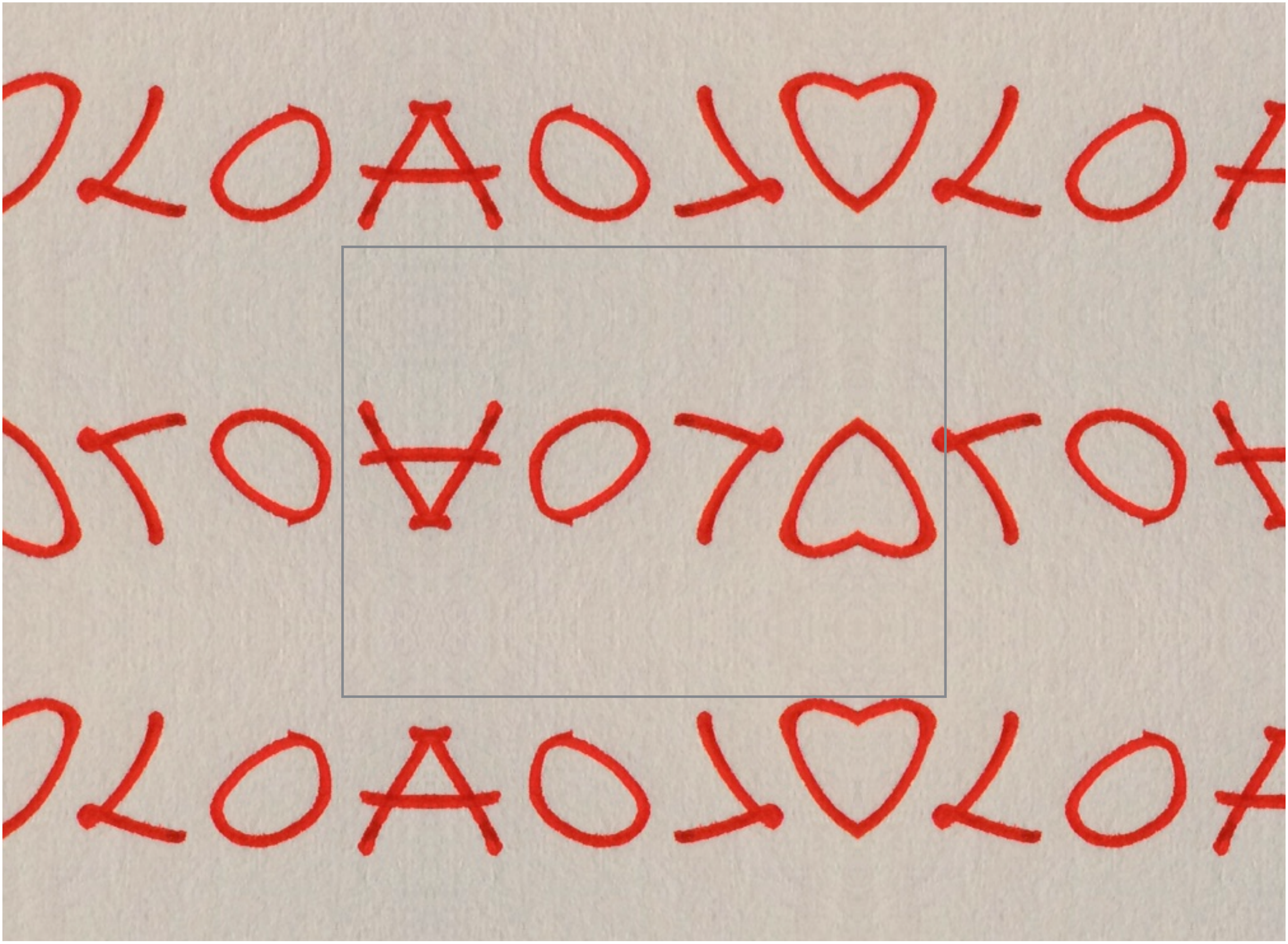
$I$

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

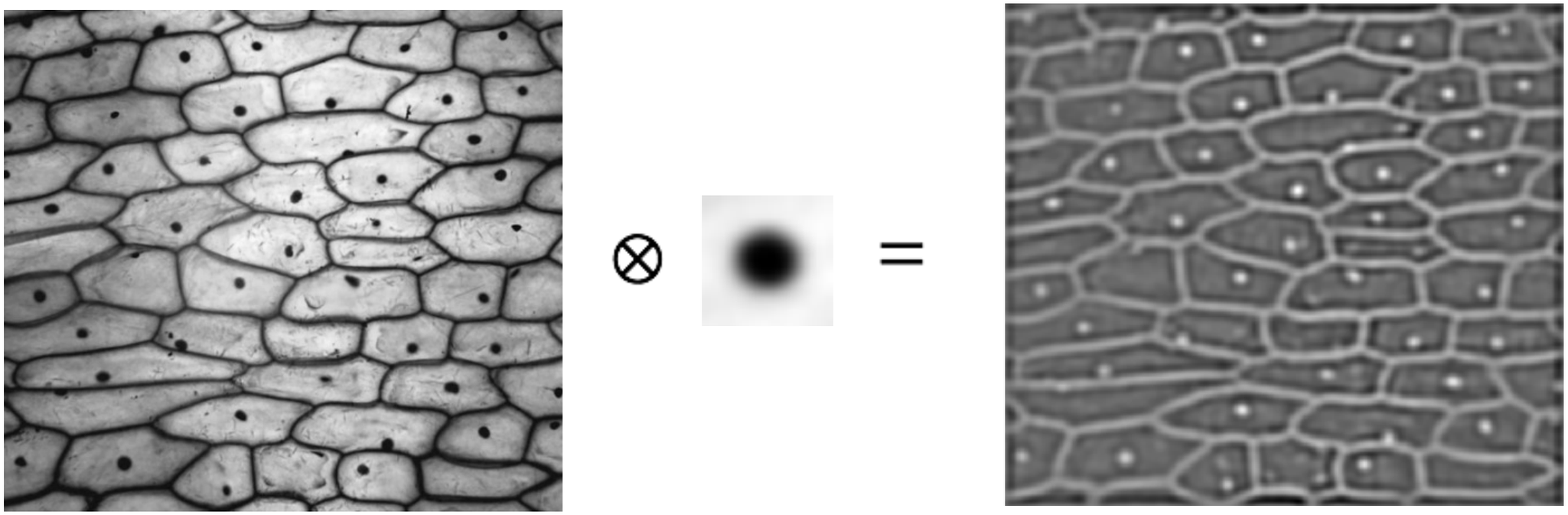
$I \otimes w > 4$

# Boundary Padding

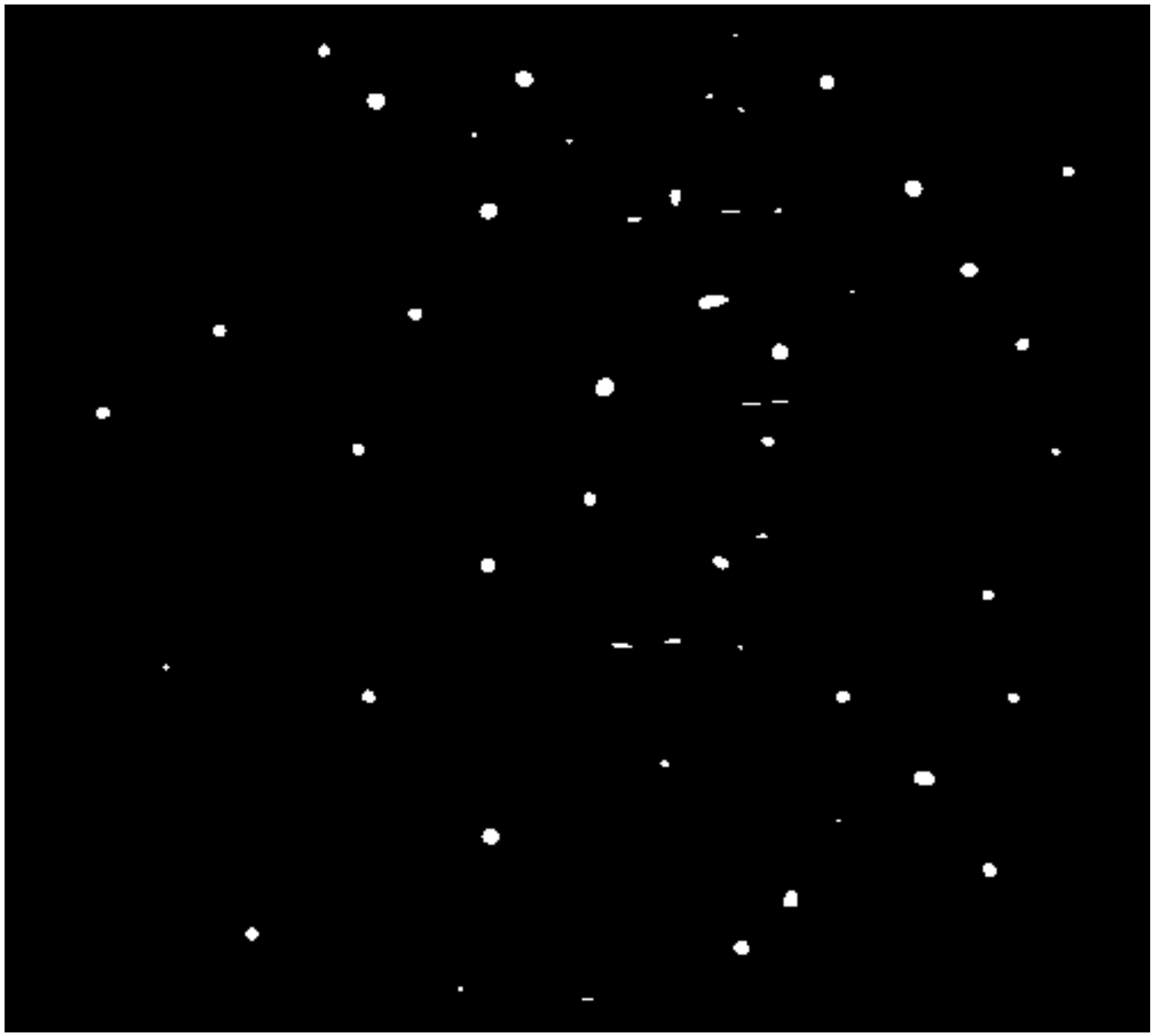




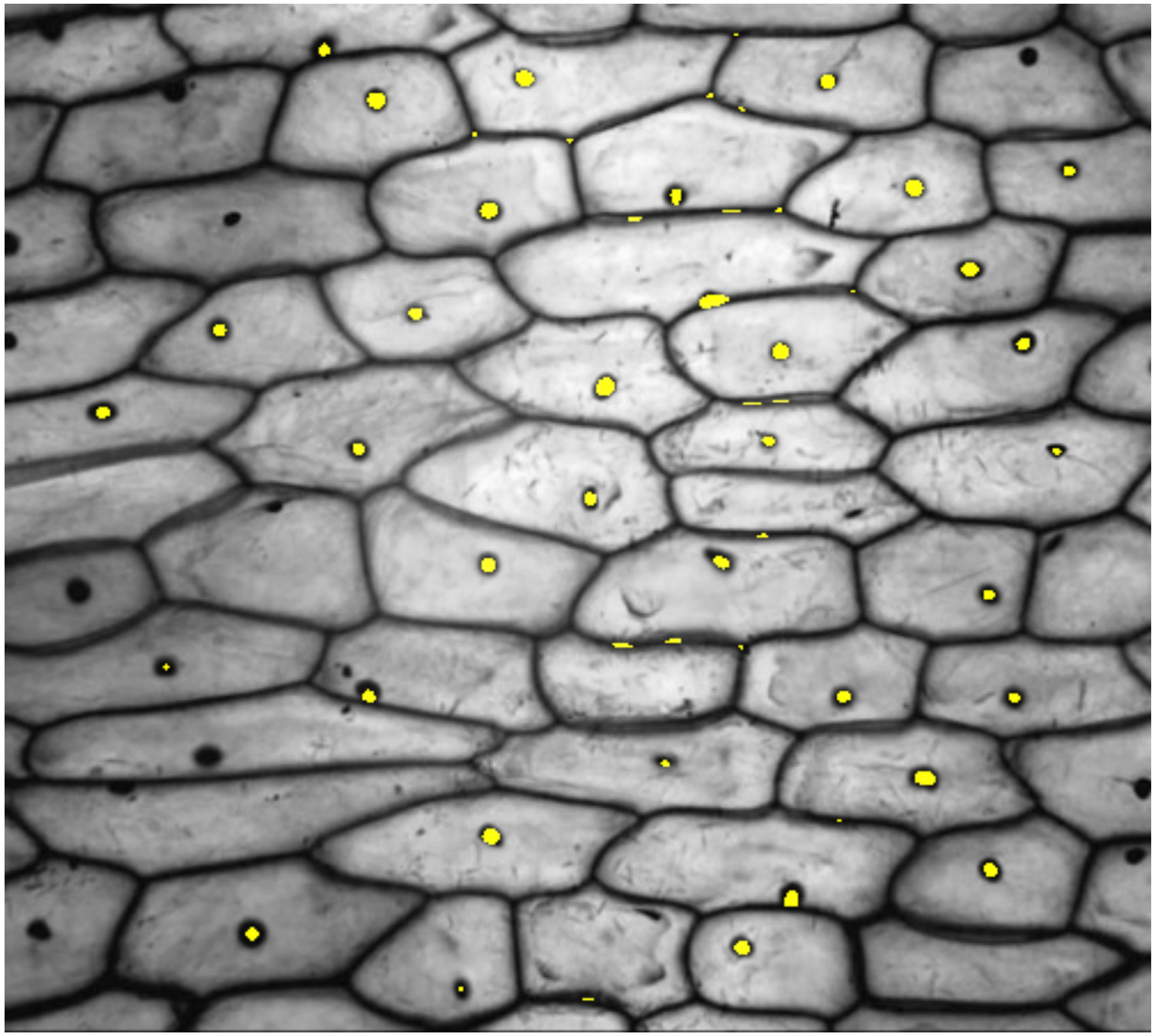
# Sliding Window Classification



# Per-Pixel Classification Results



# Per-Pixel Classification



# Maximum Filter / Non-Maximum Suppression

Is this pixel larger than its neighbors?

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 2 | 3 | 4 | 2 |
| 11 | 21 | 22 | 21 | 14 | 6 | 7 |
| 12 | 20 | 45 | 32 | 21 | 12 | 11 |
| 11 | 12 | 11 | 16 | 21 | 12 | 21 |
| 21 | 22 | 23 | 25 | 35 | 22 | 20 |
| 12 | 11 | 16 | 17 | 16 | 6 | 0 |
| 0 | 7 | 0 | 21 | 12 | 11 | 0 |

input

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

result

# Nonlinear Filters

Is this pixel larger than its neighbors?

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 2 | 3 | 4 | 2 |
| 11 | 21 | 22 | 21 | 14 | 6 | 7 |
| 12 | 20 | 45 | 32 | 21 | 12 | 11 |
| 11 | 12 | 11 | 16 | 21 | 12 | 21 |
| 21 | 22 | 23 | 25 | 35 | 22 | 20 |
| 12 | 11 | 16 | 17 | 16 | 6 | 0 |
| 0 | 7 | 0 | 21 | 12 | 11 | 0 |

input

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 0 | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

result

# Nonlinear Filters

Is this pixel larger than its neighbors?

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 2 | 3 | 4 | 2 |
| 11 | 21 | 22 | 21 | 14 | 6 | 7 |
| 12 | 20 | 45 | 32 | 21 | 12 | 11 |
| 11 | 12 | 11 | 16 | 21 | 12 | 21 |
| 21 | 22 | 23 | 25 | 35 | 22 | 20 |
| 12 | 11 | 16 | 17 | 16 | 6 | 0 |
| 0 | 7 | 0 | 21 | 12 | 11 | 0 |

input

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 1 | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

result

# Nonlinear Filters

Is this pixel larger than its neighbors?

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 2 | 3 | 4 | 2 |
| 11 | 21 | 22 | 21 | 14 | 6 | 7 |
| 12 | 20 | 45 | 32 | 21 | 12 | 11 |
| 11 | 12 | 11 | 16 | 21 | 12 | 21 |
| 21 | 22 | 23 | 25 | 35 | 22 | 20 |
| 12 | 11 | 16 | 17 | 16 | 6 | 0 |
| 0 | 7 | 0 | 21 | 12 | 11 | 0 |

input

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 1 | 0 | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

result

# Nonlinear Filters

Is this pixel larger than its neighbors?

| 1 | 2 | 1 | 2 | 3 | 4 | 2 |
|---|---|---|---|---|---|---|
| 11 | 21 | 22 | 21 | 14 | 6 | 7 |
| 12 | 20 | 45 | 32 | 21 | 12 | 11 |
| 11 | 12 | 11 | 16 | 21 | 12 | 21 |
| 21 | 22 | 23 | 25 | 35 | 22 | 20 |
| 12 | 11 | 16 | 17 | 16 | 6 | 0 |
| 0 | 7 | 0 | 21 | 12 | 11 | 0 |

input

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 1 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 1 | | |
| | | | | | | |
| | | | | | | |

result

# Nonlinear Filters

Is this pixel larger than its neighbors?

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 2 | 3 | 4 | 2 |
| 11 | 21 | 22 | 21 | 14 | 6 | 7 |
| 12 | 20 | 45 | 32 | 21 | 12 | 11 |
| 11 | 12 | 11 | 16 | 21 | 12 | 21 |
| 21 | 22 | 23 | 25 | 35 | 22 | 20 |
| 12 | 11 | 16 | 17 | 16 | 6 | 0 |
| 0 | 7 | 0 | 21 | 12 | 11 | 0 |

input

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 1 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 1 | 0 | |
| | 0 | 0 | 0 | 0 | 0 | |
| | | | | | | |

result

# Nonlinear Filters

Is this pixel larger than its neighbors?

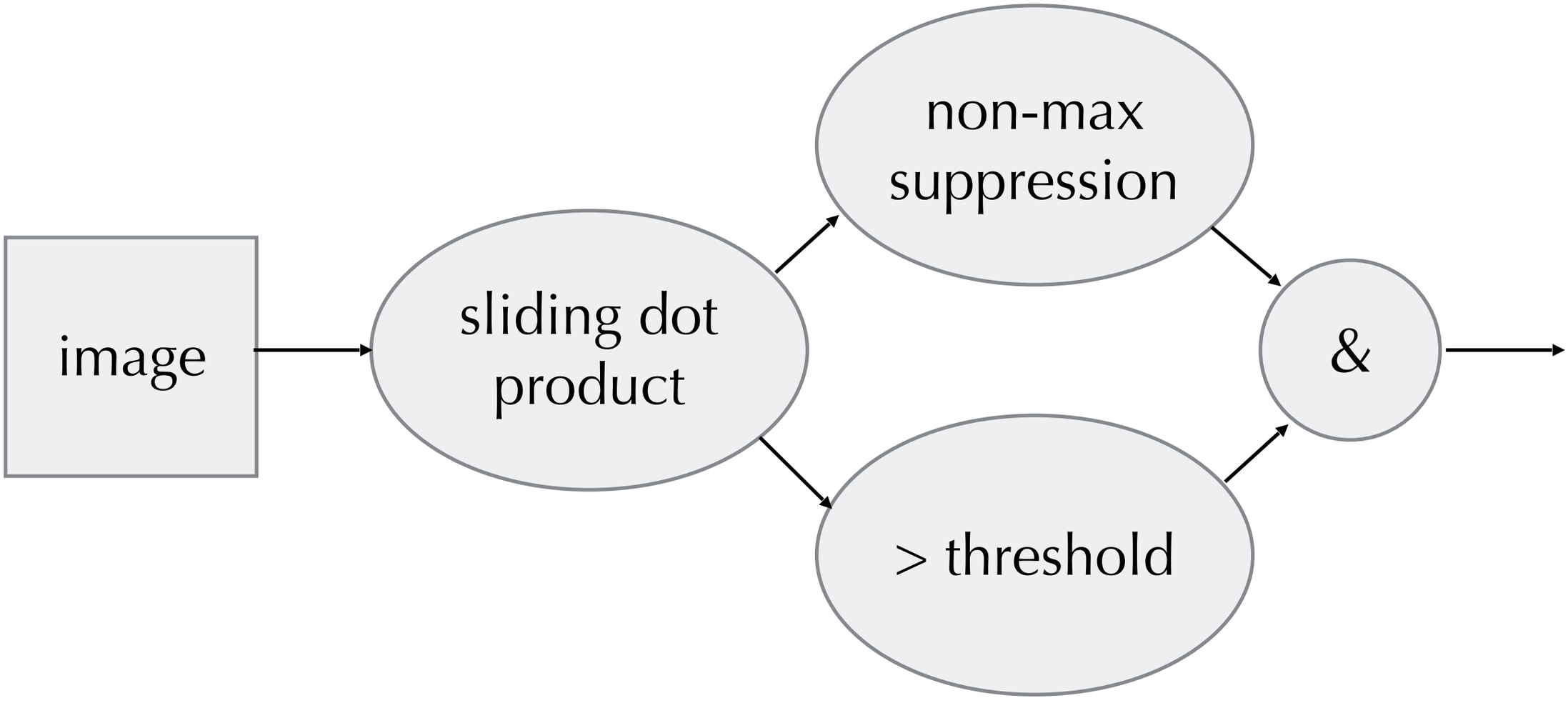
| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 2 | 3 | 4 | 2 |
| 11 | 21 | 22 | 21 | 14 | 6 | 7 |
| 12 | 20 | 45 | 32 | 21 | 12 | 11 |
| 11 | 12 | 11 | 16 | 21 | 12 | 21 |
| 21 | 22 | 23 | 25 | 35 | 22 | 20 |
| 12 | 11 | 16 | 17 | 16 | 6 | 0 |
| 0 | 7 | 0 | 21 | 12 | 11 | 0 |

input

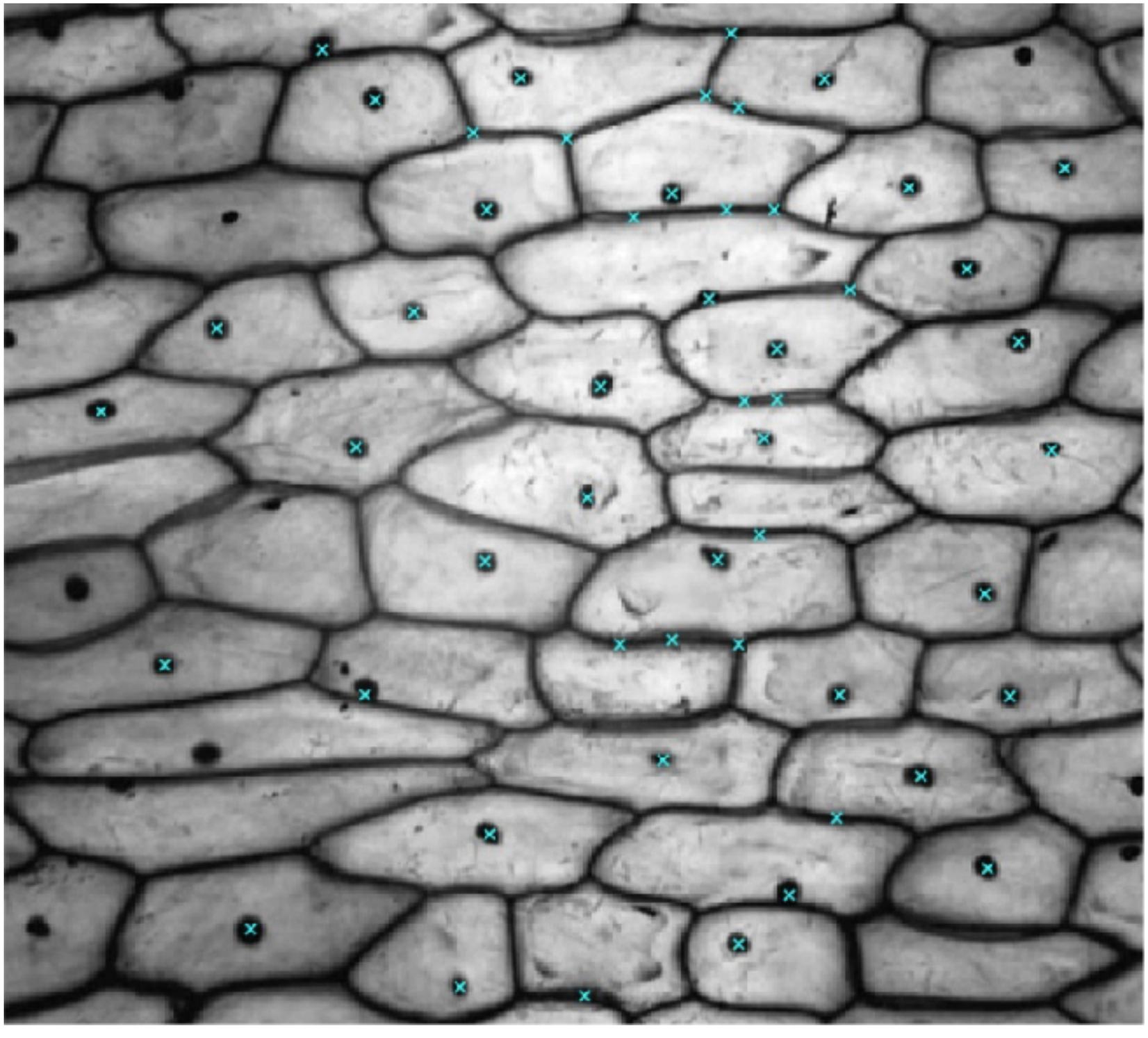
| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 1 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 1 | 0 | |
| | 0 | 0 | 0 | 0 | 0 | |
| | | | | | | |

result

# A Sliding-Window Detector



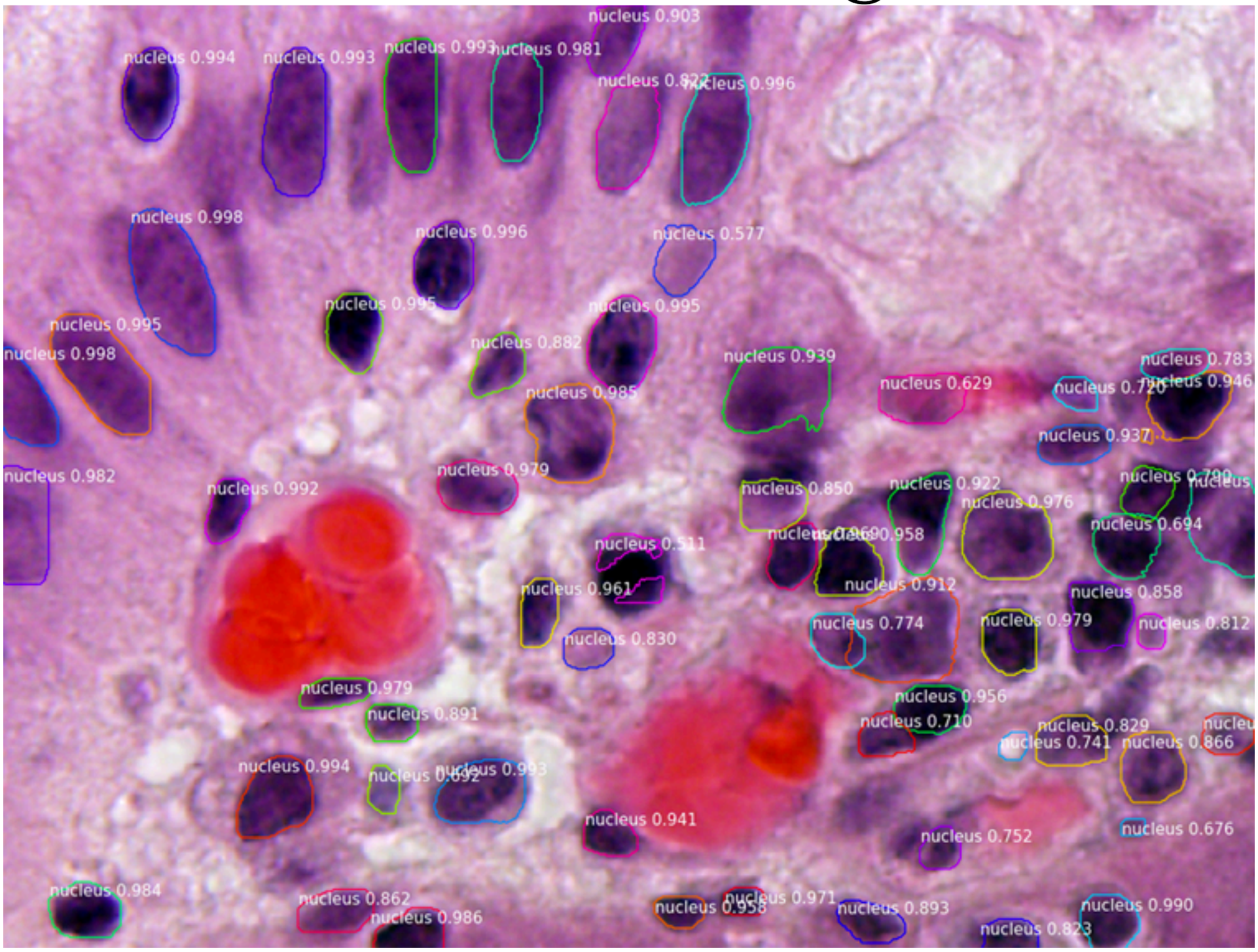
# Result



Count: 52

Manual: 48

# Instance Level Segmentation




[Result from apply Masked R-CNN]

# Lessons Learned

- Main lessons from this lecture
  - Images are vectors: Addition, subtraction, dot product
  - Linear filters: Mean, dot product-based
  - Nonlinear filters: Non-maximum suppression
  - Sliding window classification

# Lessons Learned

- Main lessons from this lecture
  - Images are vectors: Addition, subtraction, dot product
  - Linear filters: Mean, dot product-based
  - Nonlinear filters: Non-maximum suppression
  - Sliding window classification
- Next lecture: More filtering, gradients, multi-scale processing