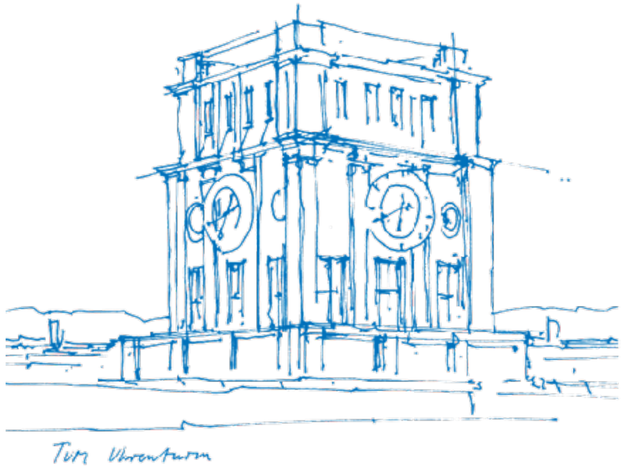
Computer Vision III:
Introduction

Dr. Nikita Araslanov
17.10.2023

Content credit:
Prof. Dr. Laura Leal-Taixé
https://dvl.in.tum.de



The team

Lectures

Dr. Nikita Araslanov

Exercises

M.Sc. Dominik Muhle

M.Sc. Regine Hardwig

- Lectures and exercises will be recorded (live-streamed, if possible);
- Completing assignments is strongly encouraged.

Course logistics

- Theory: 11 lectures (Hörsaal 1 "Interims II")
  - every Tuesday 16-18PM;
  - **note:** no lecture on 07.11.
- Exercises (MI HS 1):
  - Every Thursday 8-10AM. First slot: this Thursday (19.10)!

Lecture topics

1. Introduction
2. Object detection 1
3. Object detection 2
4. Single object tracking
5. Multiple object tracking
6. Semantic segmentation
7. Instance & panoptic segmentation
8. Video object segmentation
9. Transformers
10. Semi-supervised DST
11. Unsupervised DST

Content credit: Prof. Dr. Laura Leal-Taixé
Dynamic Vision and Learning Group
https://dvl.in.tum.de/

Moodle

- Have a question or looking for something? Check Moodle first!
  - Sign up in TUM online for access: https://www.moodle.tum.de/
  - Ask content questions online so others benefit
  - Don’t post solutions
  - Public announcements (e.g. regarding exam)

Emails & slides

- All material will be uploaded on Moodle.
- Questions regarding the lecture content, exercises etc., **use Moodle**!
- You can also reach us over email:

  cv3-ws23@vision.in.tum.de

- Emails to the individual addresses (exceptional cases).

What this course is

- **High-level (semantic)** computer vision:
  - object detection;
  - image segmentation;
  - object tracking;
  - etc...
- This is in contrast to **low-level** computer vision
  - not concerned with semantics, but with image structure/formation.
  - e.g. depth, optical flow estimation;

Semantic: “relating to meaning in language”, e.g.:
- **What** is in the image?
- **Where** is it?

What this course is not

- An introduction to deep learning –
  - take “Introduction to Deep Learning” (IN2346) [Every semester!]
- A practical project course –
  - take e.g., “Geometric Scene Understanding” (IN4346) [SoSe]
- An introduction to 3D computer vision –
  - take “Computer Vision 2: Multiple View Geometry” (IN2228) [SoSe]

Ideally, this is not your first encounter with image processing!

What is computer vision?

- First defined in the 60s
- “Mimic the human visual system”
- Centre block of artificial intelligence

What is computer vision?

- Project MAC (est. 1963) – CSAIL today

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence Group July 7, 1966
Vision Memo. No. 100.

THE SUMMER VISION PROJECT

Seymour Papert

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition".

CSAIL: The Stata Center (Boston)

Summer project: current status

- One of the core areas of AI research
- >10K yearly conference attendees

First quiz: What is h5-index/h5-median?

Compute vision

Machine Learning

	Publication	h5-index	h5-median
1.	Nature	467	707
2.	The New England Journal of Medicine	439	876
3.	Science	424	665
4.	IEEE/CVF Conference on Computer Vision and Pattern Recognition	422	681
5.	The Lancet	368	688
6.	Nature Communications	349	456
7.	Advanced Materials	326	415
8.	Cell	316	503
9.	Neural Information Processing Systems	309	503
10.	International Conference on Learning Representations	303	563

(Source: scholar.google.com. Accessed: 16.10.2023)

A visual system prototype

- Where do we start?
- CS approach: think of input/output
- Input: an image
- Output: depends on the goal

Semantic scene understanding

What does it mean to “understand the scene”?

(Cityscapes, Cordts et. al., 2016)

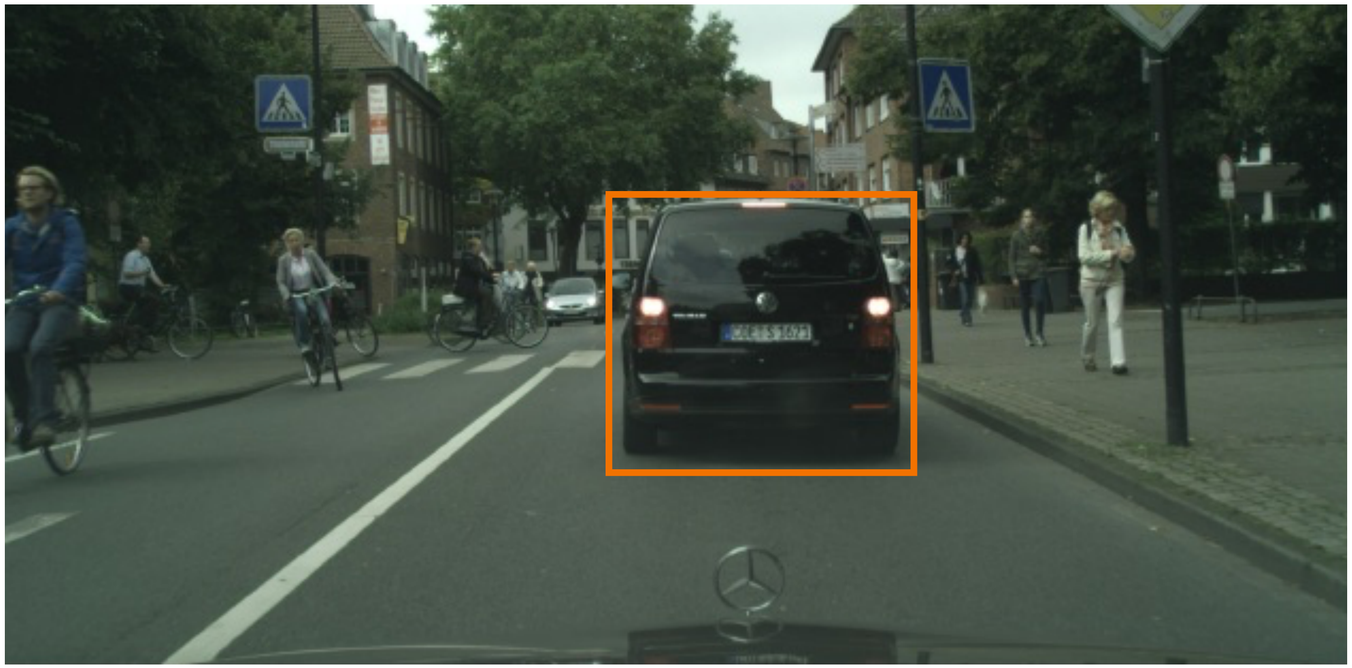
Semantic scene understanding

“There is a car” (classification)

(Cityscapes, Cordts et. al., 2016)

Semantic scene understanding

"That's a car" (object detection)



(Cityscapes, Cordts et. al., 2016)

Semantic scene understanding

➡ Semantic segmentation

(Cityscapes, Cordts et. al., 2016)

Semantic scene understanding

➡ Instance segmentation

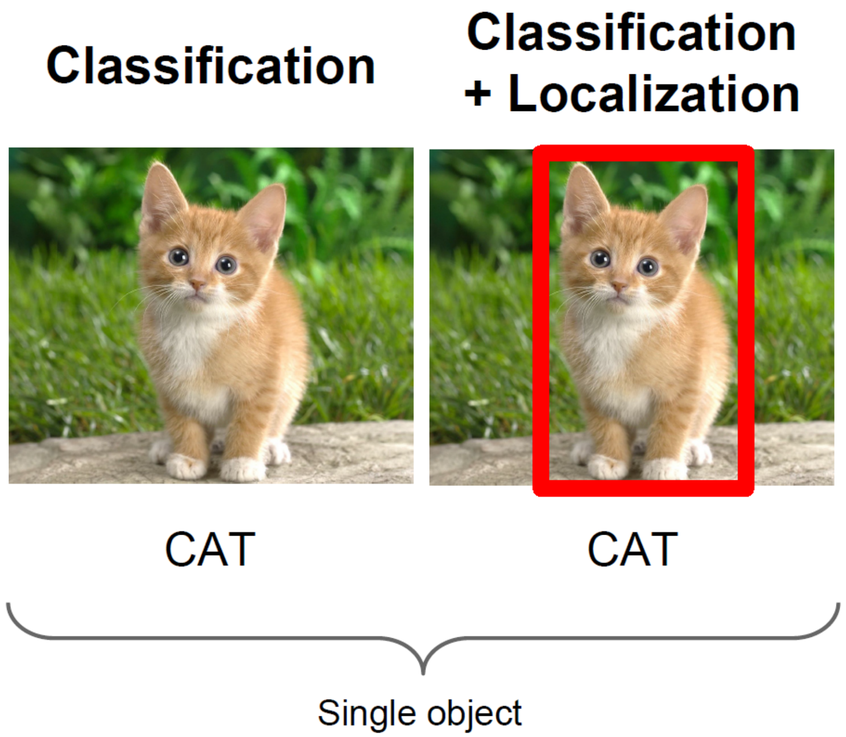
(Cityscapes, Cordts et. al., 2016)

Semantic scene understanding

- Dense tracking – temporally consistent segmentation

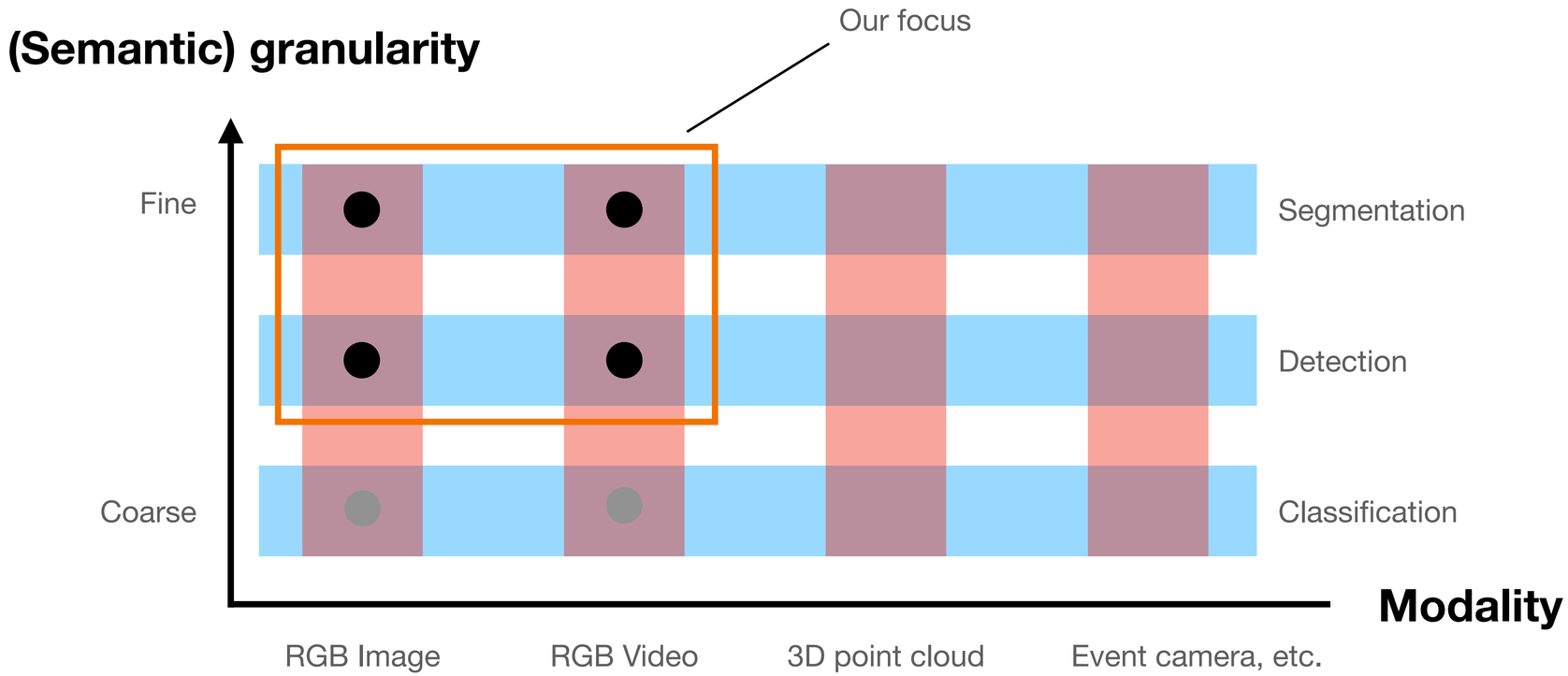
(Cityscapes, Cordts et. al., 2016)

What is in the image?



(Credit: Li, Karpathy and Johnson)

This course in two dimensions



Understanding an image

- Different representations depending on the granularity
  - Detection (bounding box – coarse description)
  - Semantic segmentation (pixel-level)
  - Instance segmentation (e.g. “person 1”, “person 2”)

Understanding a video

Why use the temporal domain?

- Motion analysis, multi-view reasoning;
- A smoothness assumption: no abrupt changes between frames.

...and technical challenges:

- High computational demand.
- A lot of redundancy.
- Occlusions, multiple objects moving and interacting.

A visual system prototype

- Regardless of the formulation, let's face it:
  - It's challenging
- Are we making any progress?
  - Yes!

Progress: now and then

2000s

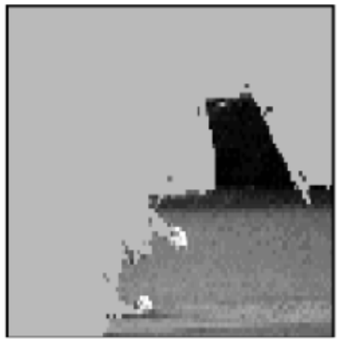
N-Cut

(Shi and Malik, 2000)

Input

Segment 1

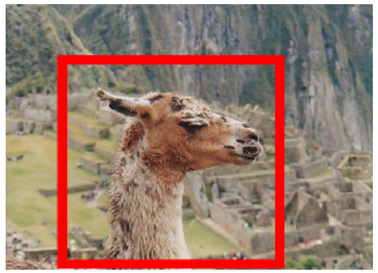
Segment 2



Segment 3

GrabCut

(Rother et al., 2004)



Input

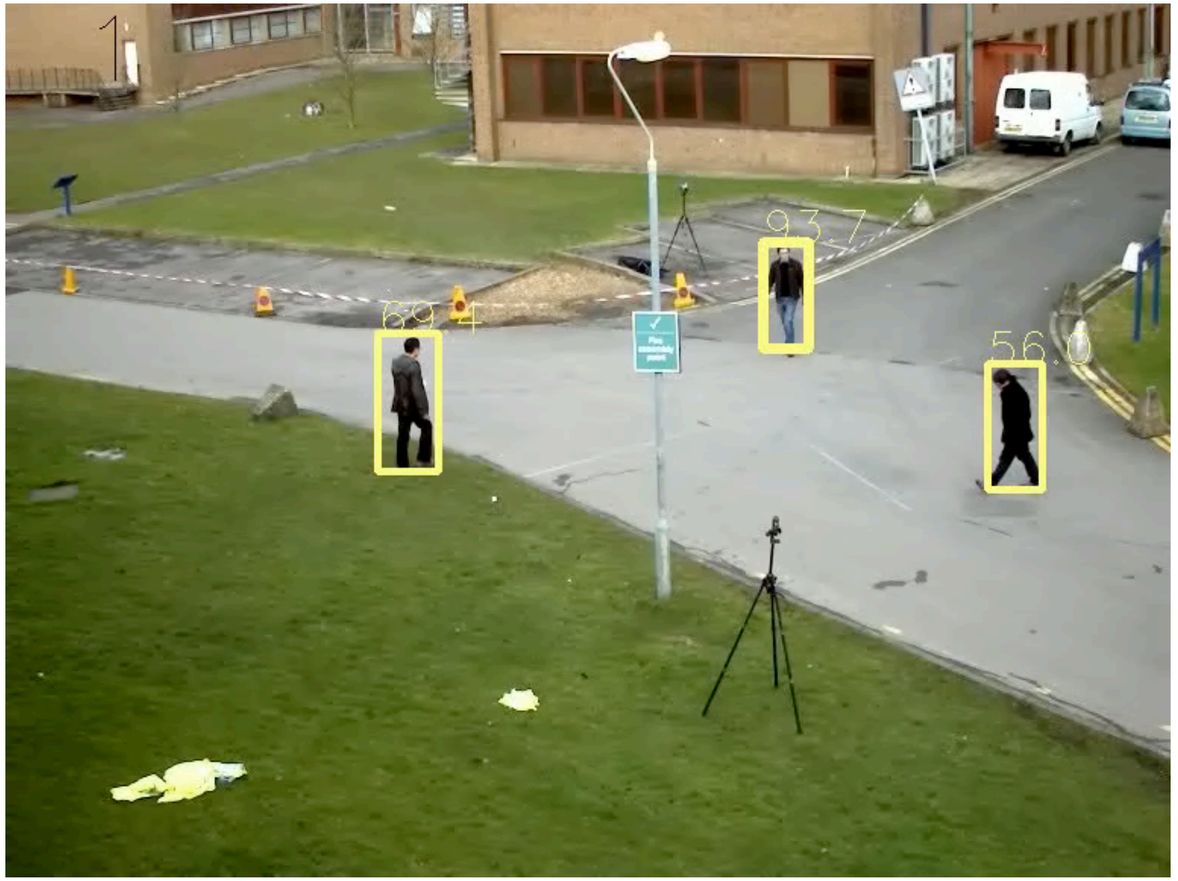
Output

Input

Output

Progress: now and then

Late 2000s



(Credit: https://motchallenge.net)

Progress: now and then

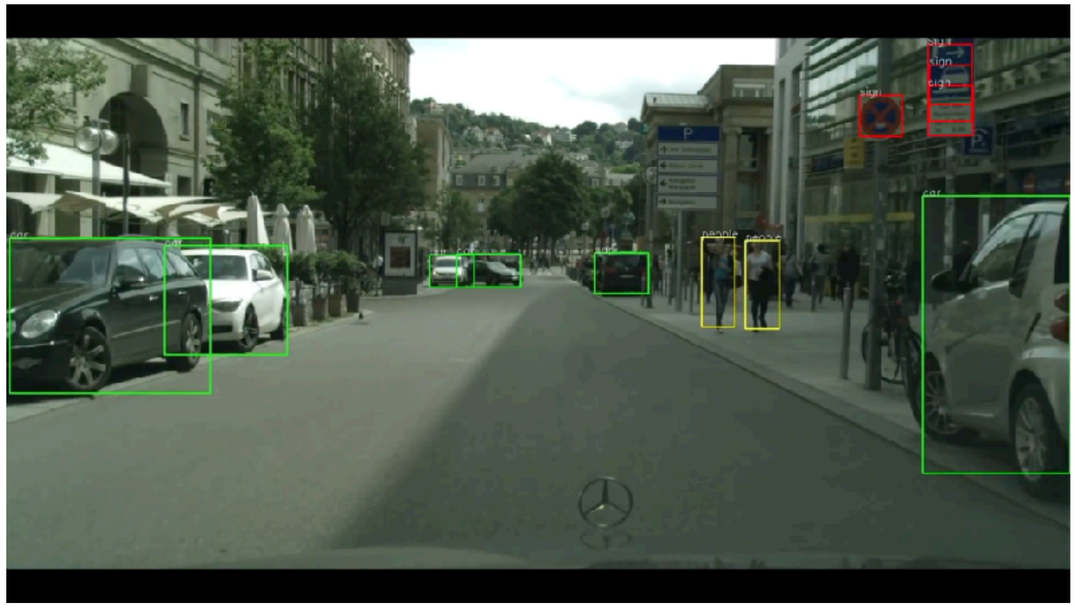
2010s

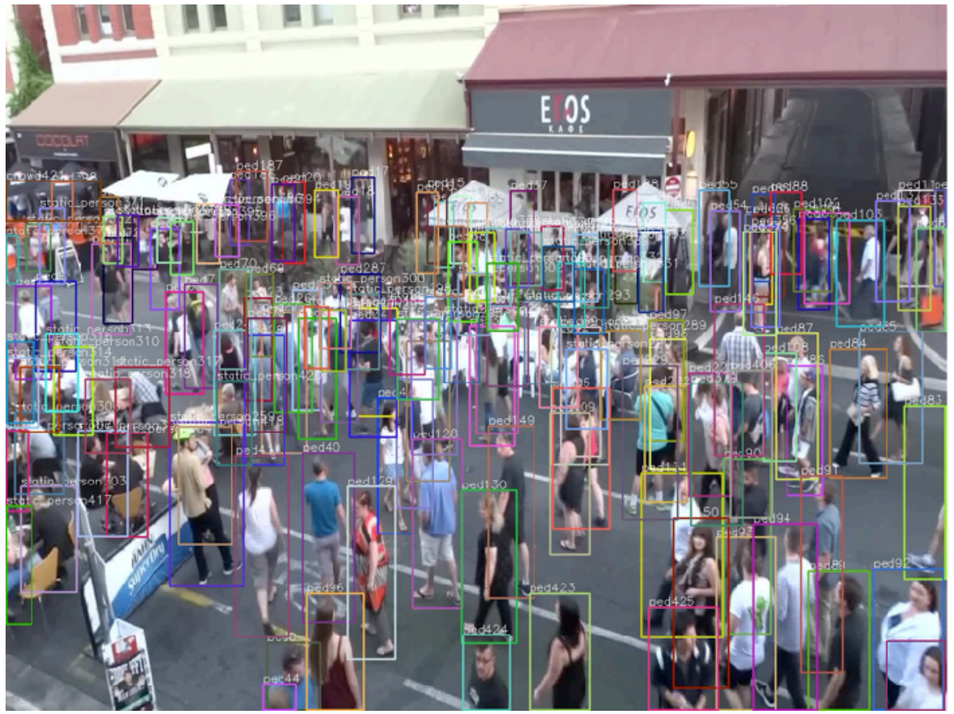
Mask R-CNN

(He et al., 2017)

Progress: now and then

2020s





Progress: now and then

The state of the art today: Semi-supervised learning

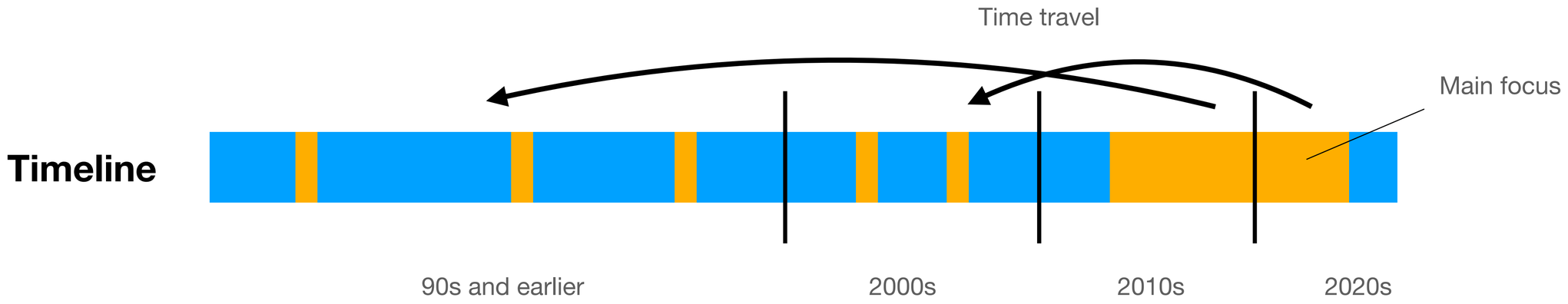
Segment Anything **(Kirillov et al., 2023)**

Progress: now and then

The state of the art today: Semi-supervised learning

Segment Anything **(Kirillov et al., 2023)**

Progress: Now and then



Some architectures and concepts

- R-CNN, Fast R-CNN and Faster R-CNN (2-stage object detection)
- YOLO, SSD, RetinaNet (1-stage object detection)
- Siamese networks (online tracking)
- Message Passing Networks (offline tracking)
- Mask R–CNN, UPSNet (panoptic segmentation)
- Deformable/atrous convolutions
- Graph neural networks (GNNs)
- Vision Transformers (ViT), DETR (object detection), SAM
- Contrastive learning

What we've learned so far

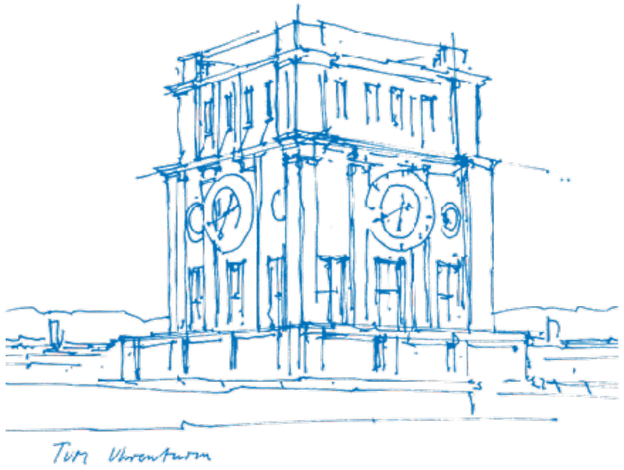
- **Two modalities** of interest:
  - Images and videos.
- **Two tasks** of interest:
  - Detection and segmentation.

- Focus on **deep learning** pipelines.
  - **Main objective**: Understanding how and why they work (often using analytical models and the historical context).

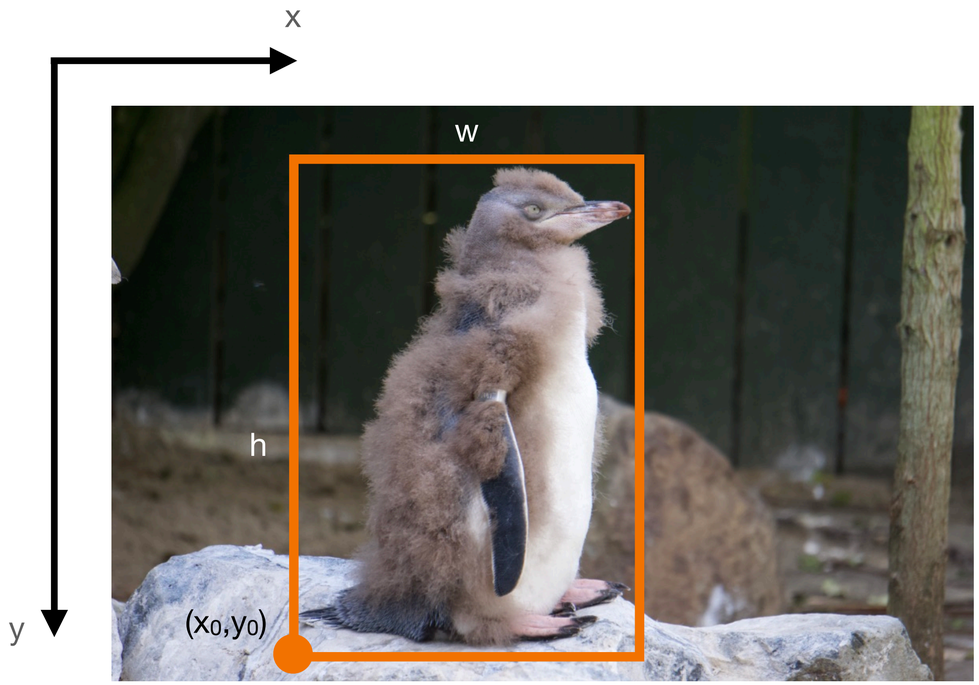
Computer Vision III:
Object detection

Dr. Nikita Araslanov
17.10.2023

Content credit:
Prof. Dr. Laura Leal-Taixé
https://dvl.in.tum.de



Object detection



Quiz #1: How many parameters define a bounding box?

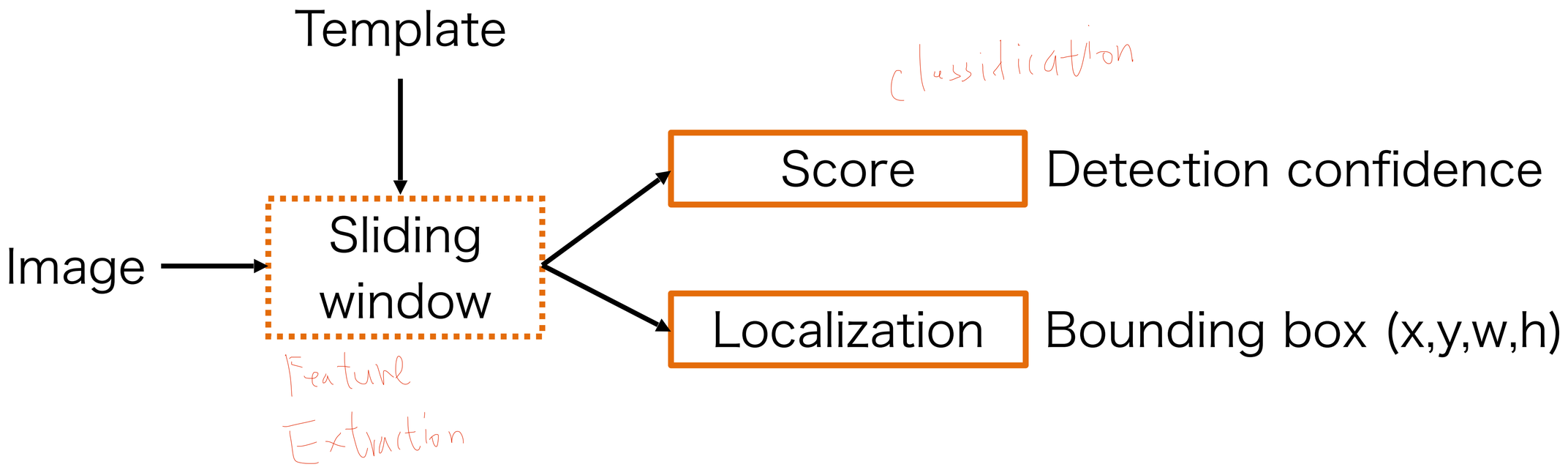
4 (x_0,y_0,h,w)

Quiz #2: Is such parametarisation unique?

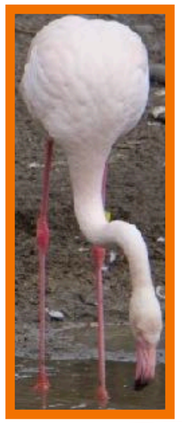
No.

Types of object detectors

- One-stage detectors (“old” times):



Template matching with sliding window



Template

Template matching with sliding window

For every position you measure the distance (or correlation) between the template and the image region:

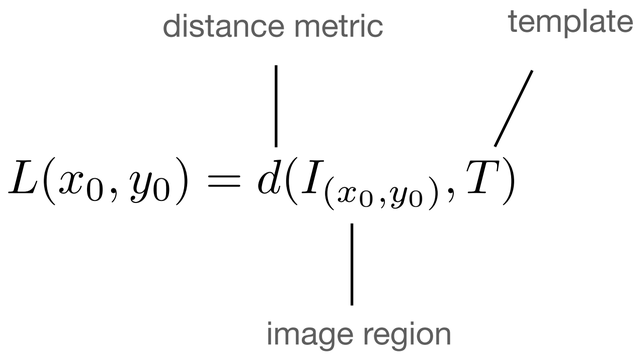
$$L(x_0, y_0) = d(I_{(x_0, y_0)}, T)$$

distance metric

template

image region

Measuring template similarity



- Sum of squared distances (SSD), or mean squared error (MSE):

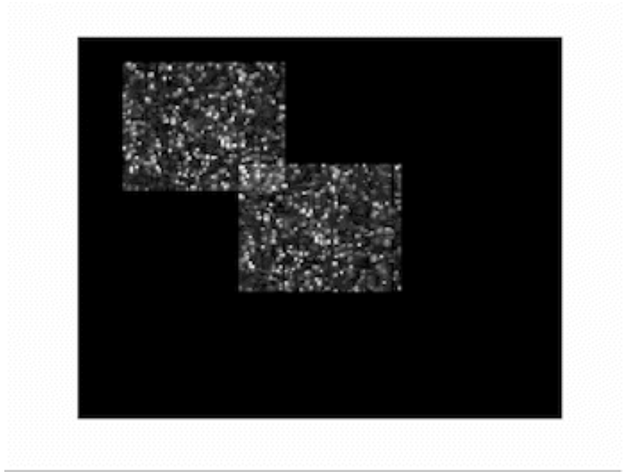
$$d(I_{(x_0,y_0)}, T) = \frac{1}{n} \sum_{x,y} (I_{(x_0,y_0)}(x, y) - T(x, y))^2$$

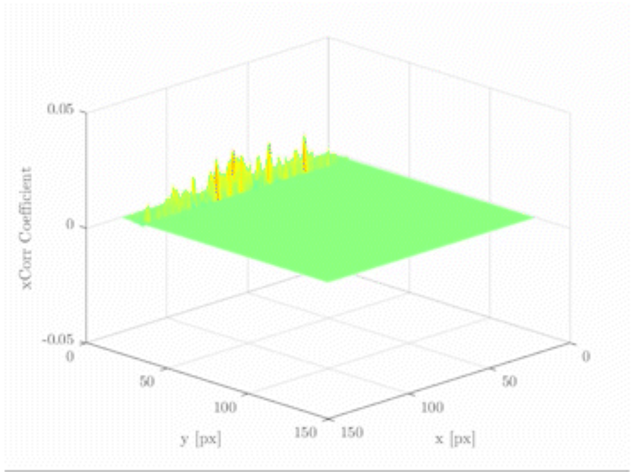
Template matching distance

- Normalised cross-correlation (NCC): $$d(I_{(x_0,y_0)}, T) = \frac{1}{n} \sum_{x,y} \frac{1}{\sigma_I \sigma_T} I_{(x_0,y_0)}(x,y) T(x,y)$$

- Zero-normalised cross-correlation (ZNCC):

$$d(I_{(x_0,y_0)}, T) = \frac{1}{n} \sum_{x,y} \frac{1}{\sigma_I \sigma_T} (I_{(x_0,y_0)}(x,y) - \mu_I)(T(x,y) - \mu_T)$$





Quiz: Can I just swap SSD with NCC in my code?

Source: https://en.wikipedia.org/wiki/Cross-correlation

Template matching with sliding window

For every position you measure the distance (or correlation) between the template and the image region:

$$L(x_0, y_0) = d(I_{(x_0,y_0)}, T)$$

distance metric

template

image region

Template matching with sliding window

For every position you measure the distance (or correlation) between the template and the image region:

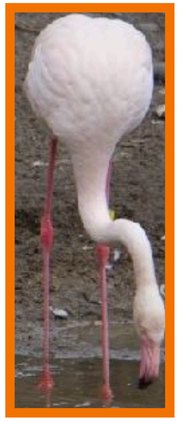
$$L(x_0, y_0) = d(I_{(x_0,y_0)}, T)$$

distance metric

template

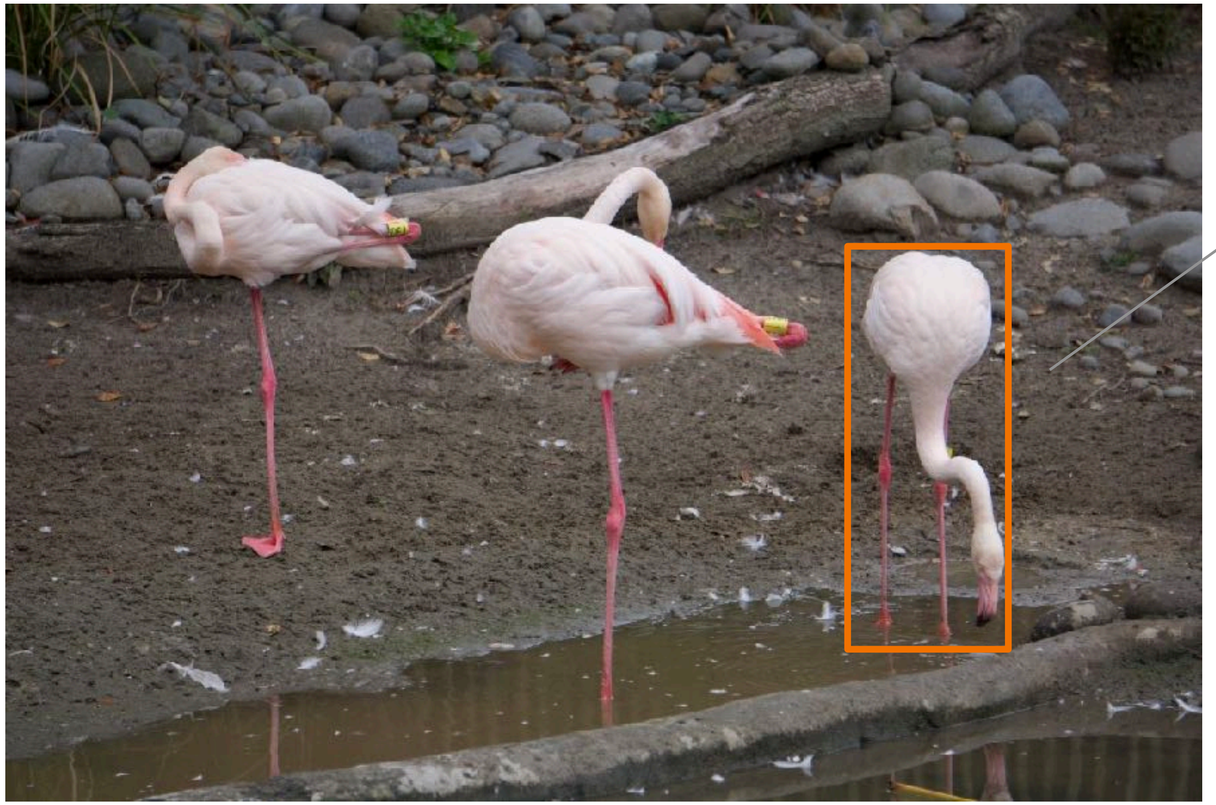
image region

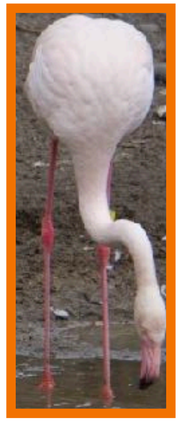
Template matching with sliding window



LOW correlation

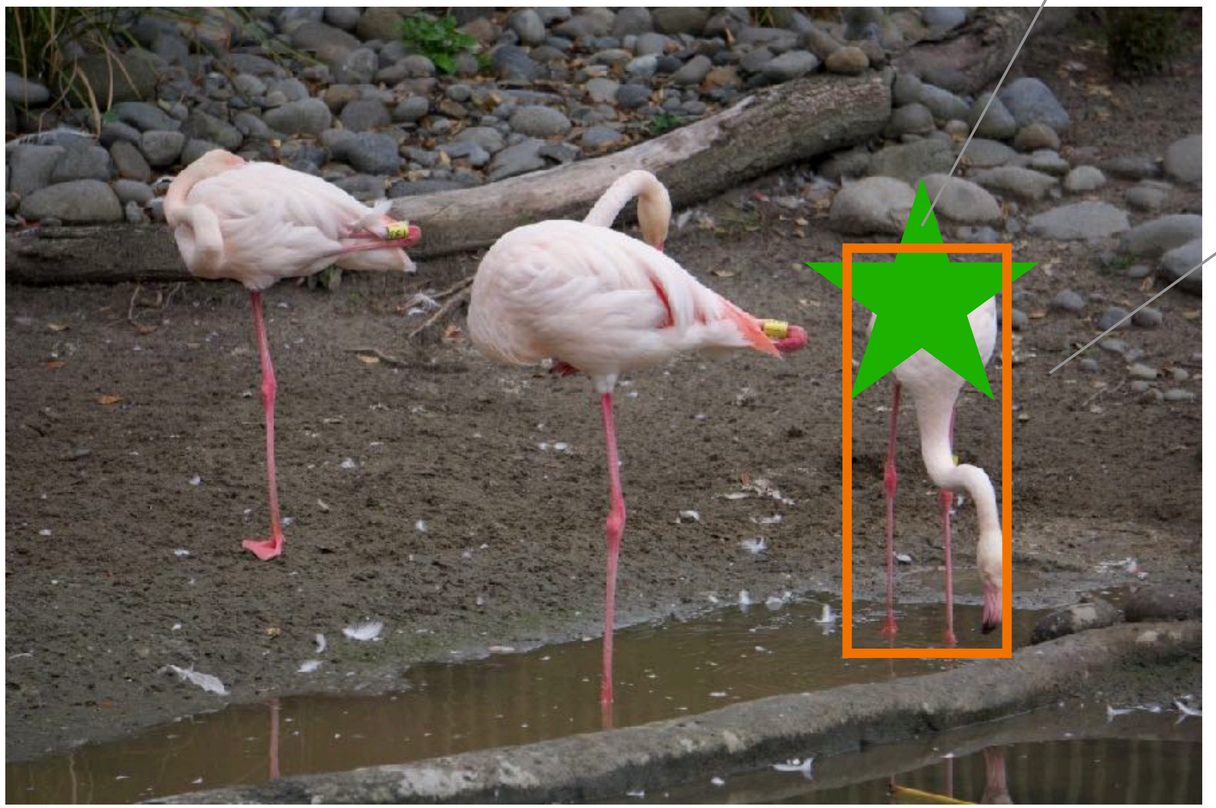
Template matching with sliding window

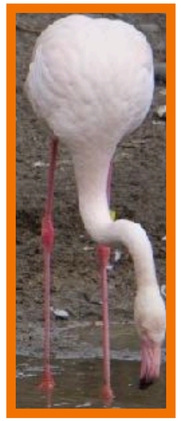




HIGH correlation

Template matching with sliding window





HIGH correlation

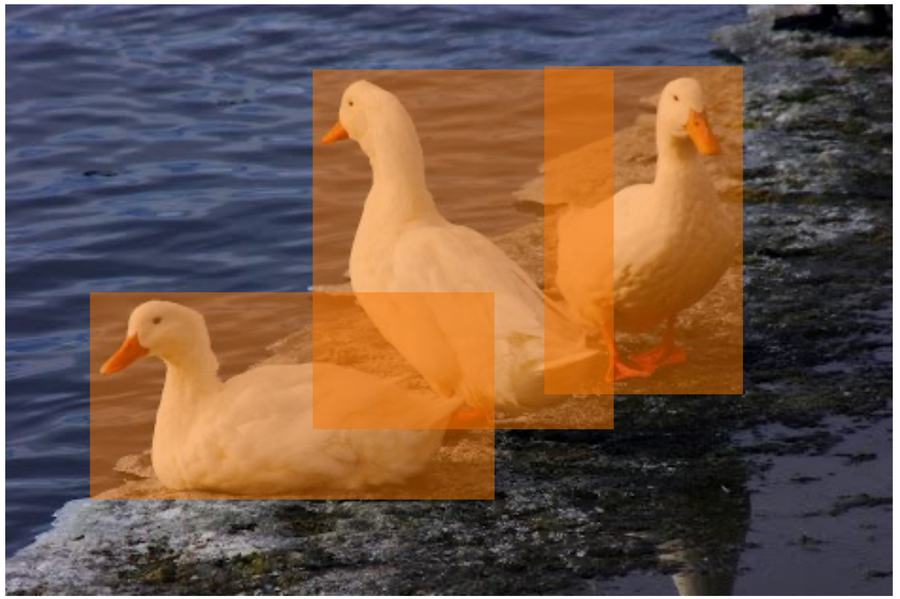
Template matching: disadvantages

- (Self-)occlusions (e.g. due to pose changes)

- Changes in appearance

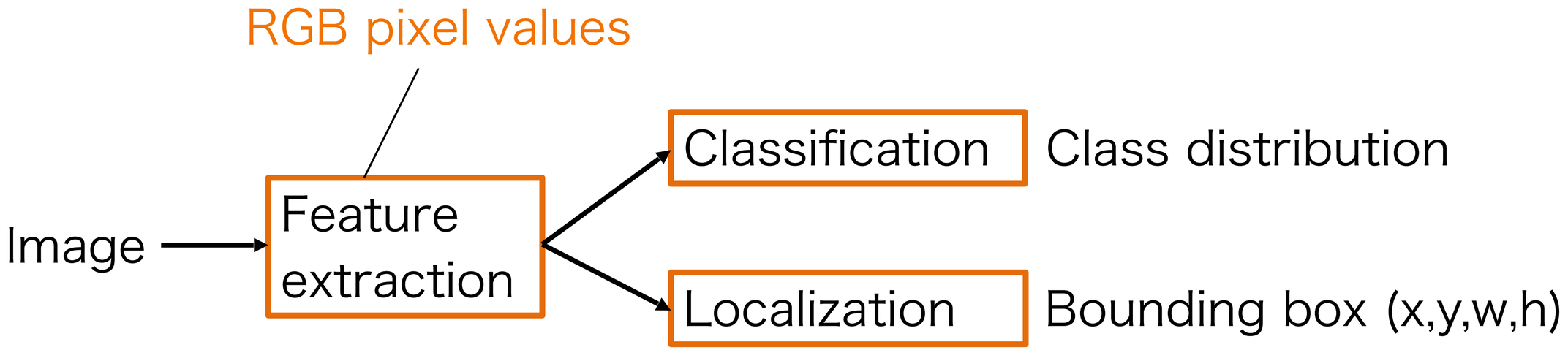
Template matching: disadvantages

- Unknown position, scale and aspect ratio
  - brute-force search (inefficient)



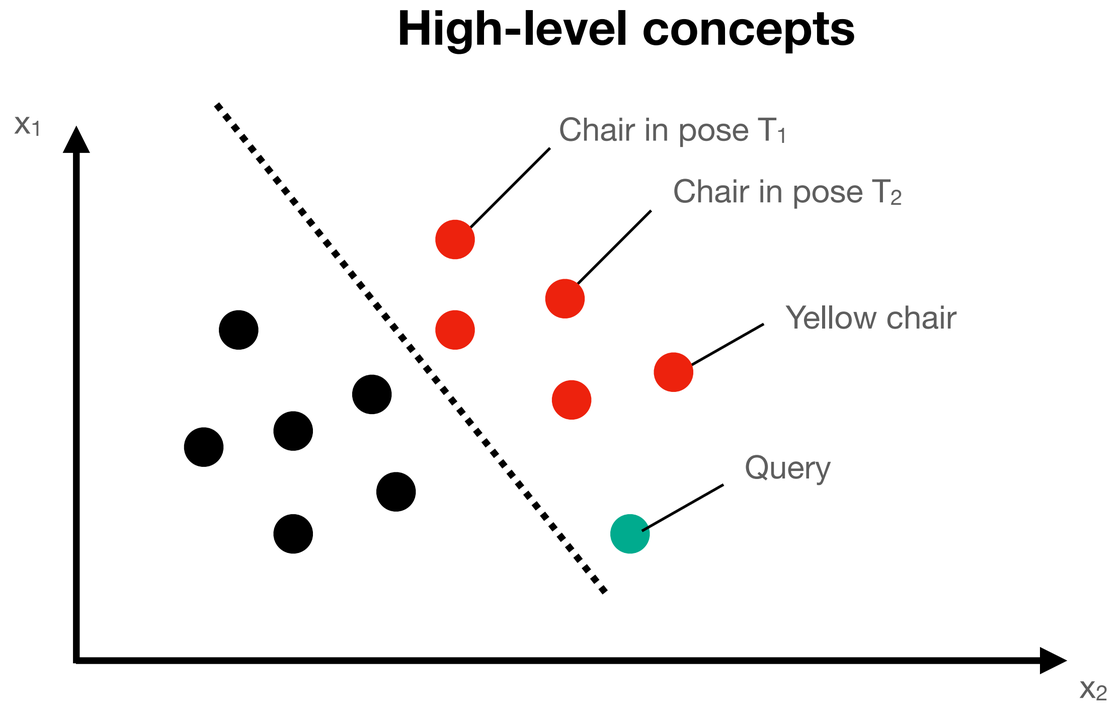
Types of object detectors

- One-stage detectors (new):



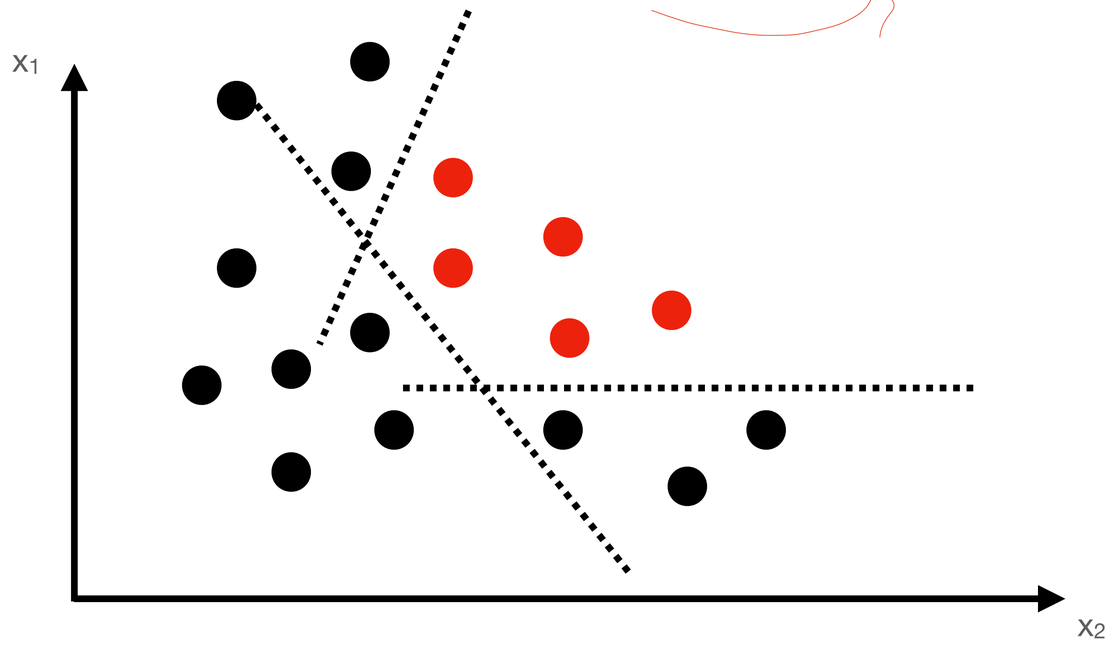
Feature-based detection

Idea: Learn feature based classifiers invariant to natural object changes

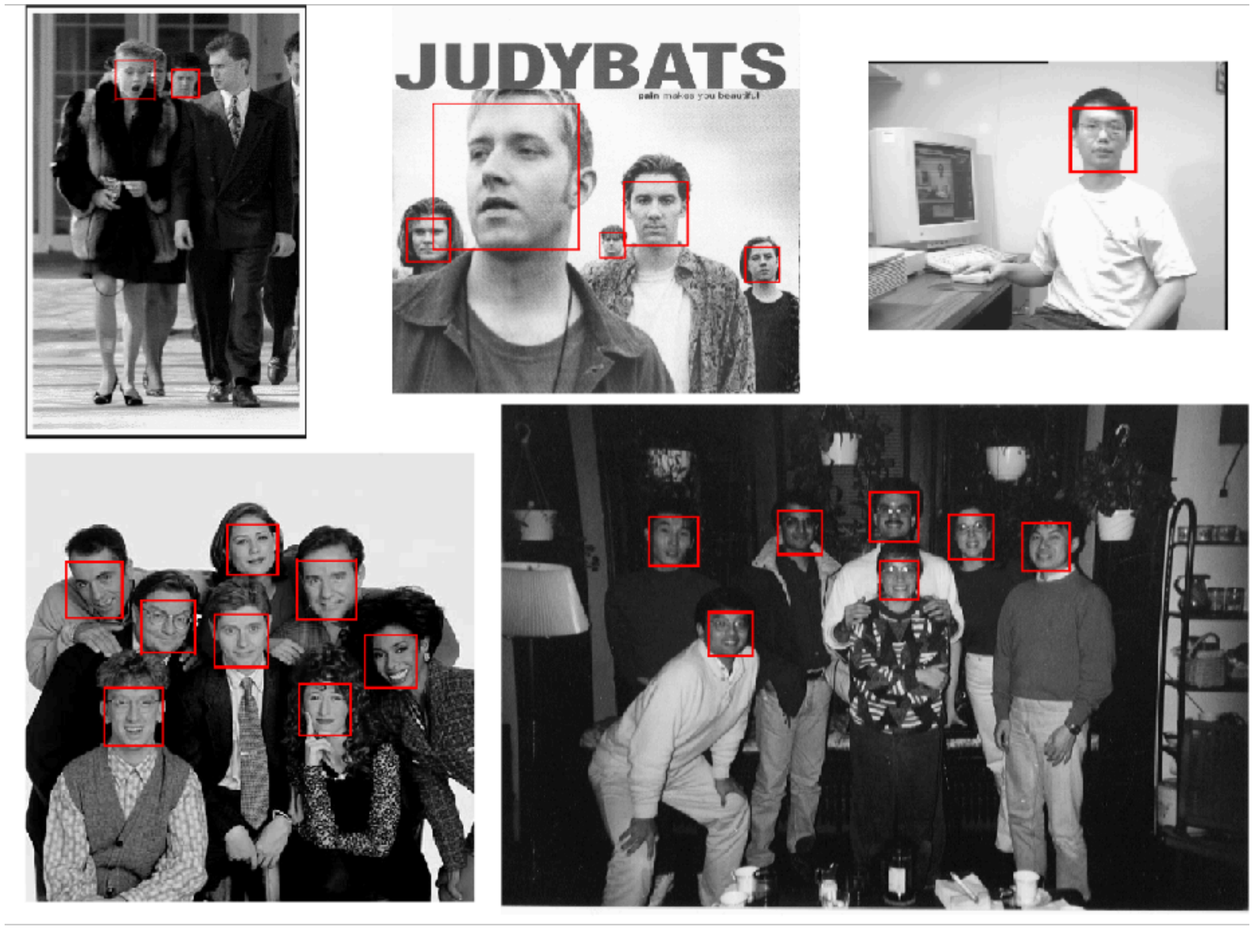


Feature-based detection

- Features are not always linearly separable
- Learning multiple weak learners to build a strong classifier



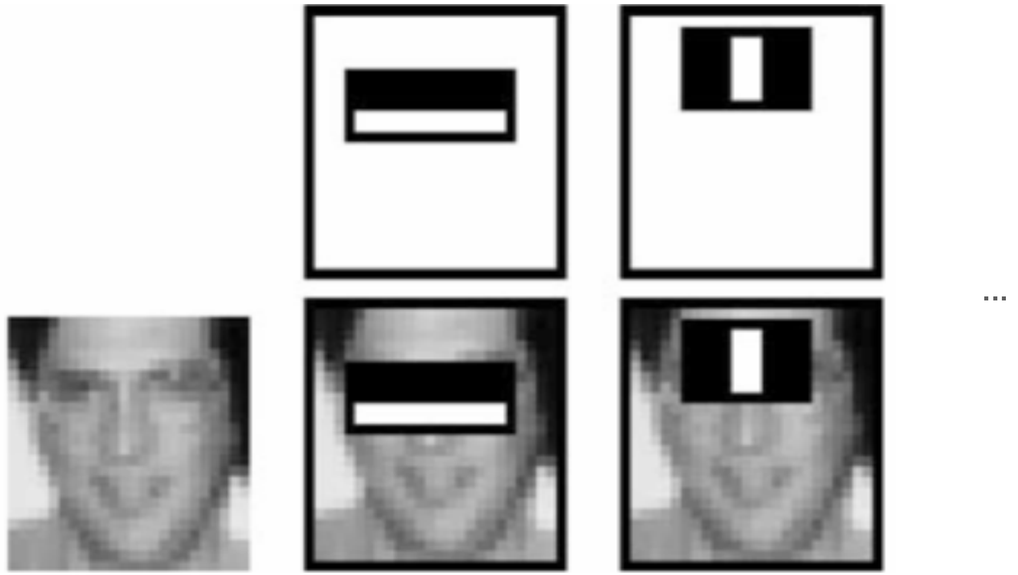
Viola-Jones detector



Viola and Jones. Rapid object detection using a boosted cascade of simple features. CVPR 2001.

Viola-Jones detector

Haar-like features



...

$h_1(\cdot)$ $h_2(\cdot)$

“weak” learners

Viola and Jones. Rapid object detection using a boosted cascade of simple features. CVPR 2001.

Viola-Jones detector

Given data (x_i, y_i):

1. Define a set of Haar-like features

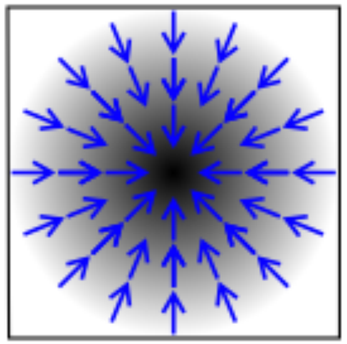
AdaBoost

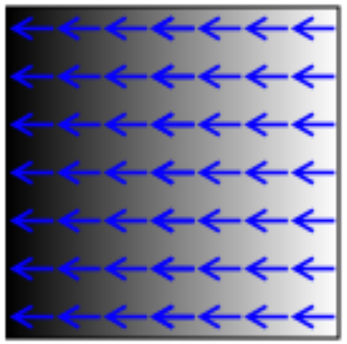
2. Find a weak classifier with the lowest error across the dataset
3. Save the weak classifier and update the priority of the data samples
4. Repeat Steps 2-3 N times

- Our final classifier is the linear combination of all weak learners

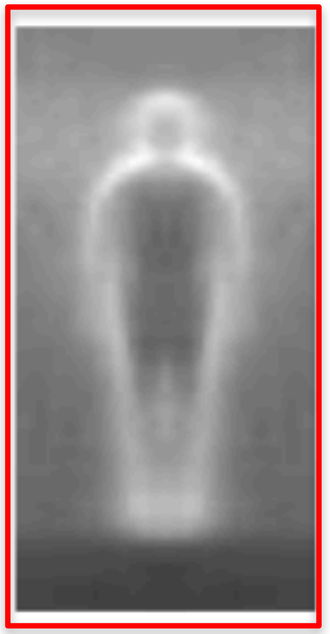
Viola and Jones. Rapid object detection using a boosted cascade of simple features. CVPR 2001.

Histogram of Oriented Gradients





Quiz:
Is “black” lower
or higher than “white”?

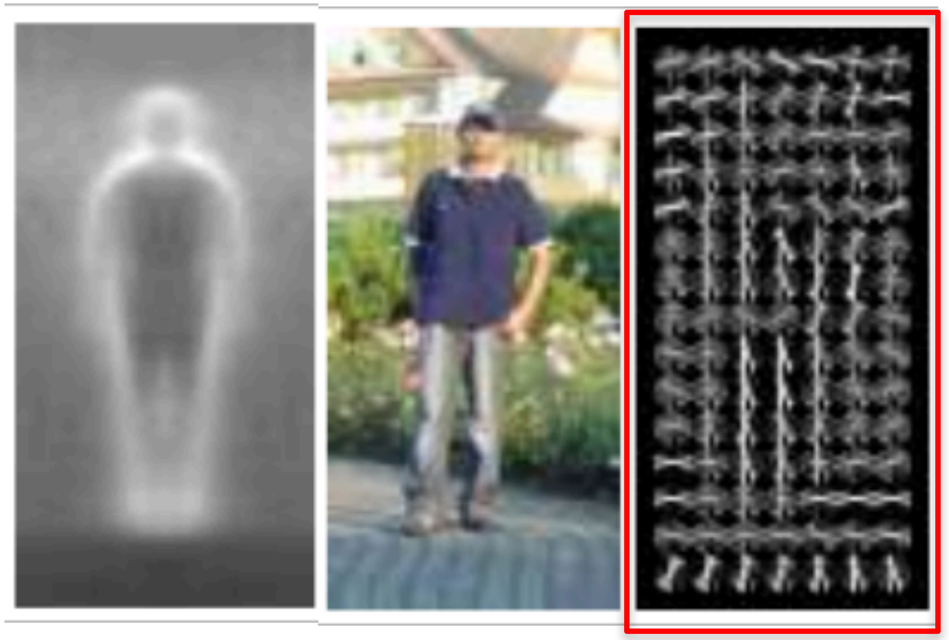


Gradient: blue arrows show the gradient, i.e., the direction of greatest change of the image.

Average gradient image over training samples → gradients provide shape information.

Dalal and Triggs. Histogram of oriented gradients for human detection. CVPR 2005.

Histogram of Oriented Gradients



HOG descriptor → Histogram of oriented gradients.

Compute gradients in dense grids, compute gradients and create a histogram based on gradient direction.

Dalal and Triggs. Histogram of oriented gradients for human detection. CVPR 2005.

Histogram of Oriented Gradients

Step 1: Choose your training set of images that contain the object you want to detect.

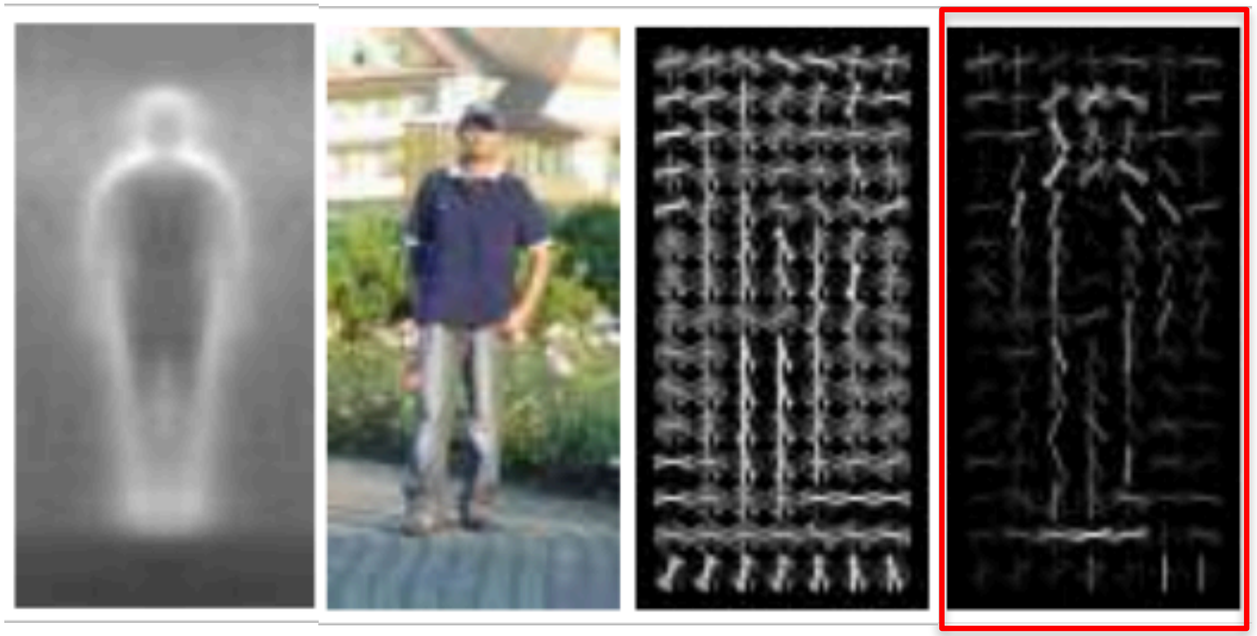
Step 2: Choose a set of images that do NOT contain that object.

Step 3: Extract HOG features from both sets.

Step 4: Train an SVM classifier on the two sets to detect whether a feature vector represents the object of interest or not (0/1 classification).

Dalal and Triggs. Histogram of oriented gradients for human detection. CVPR 2005.

Histogram of Oriented Gradients

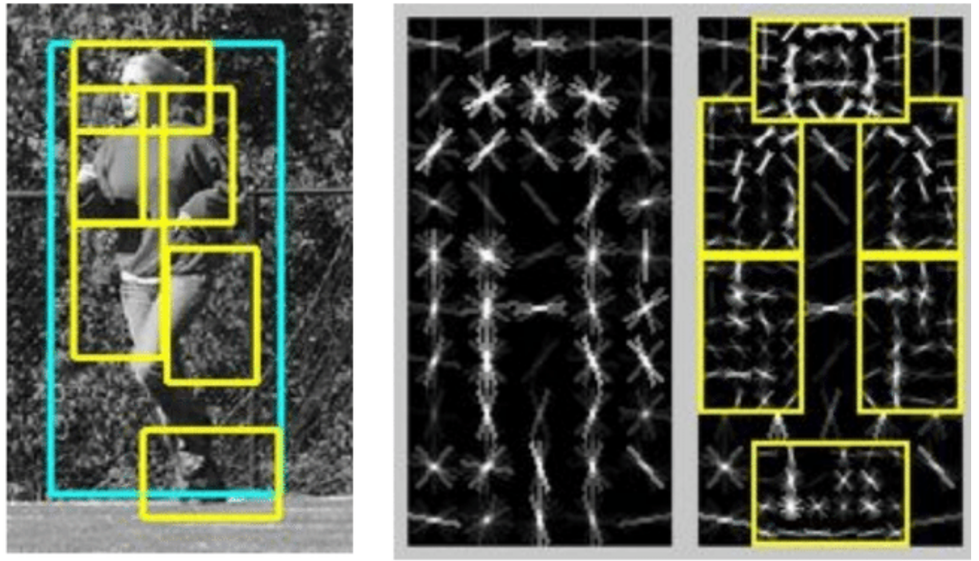


HOG features weighted by the positive SVM weights – the ones used for the pedestrian object classifier.

Dalal and Triggs. Histogram of oriented gradients for human detection. CVPR 2005.

Deformable Part Model

- Many objects are not rigid.
- Bottom-up approach:
  - detect body parts
  - detect “person” if the body parts are in correct arrangement

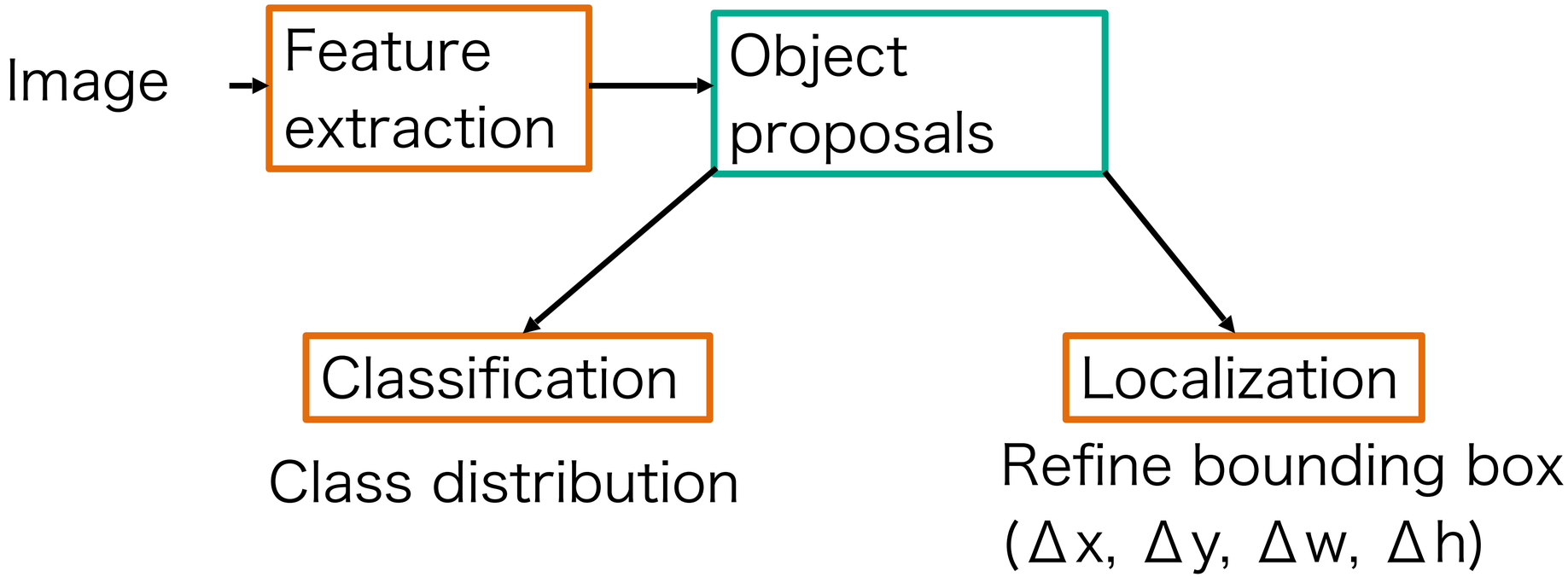


Felzenszwalb et al. A discriminatively trained, multiscale, deformable part model. CVPR 2008.

- Note: The amount of work for each RoI may grow significantly.
- Does it make sense to run it for every sliding window?

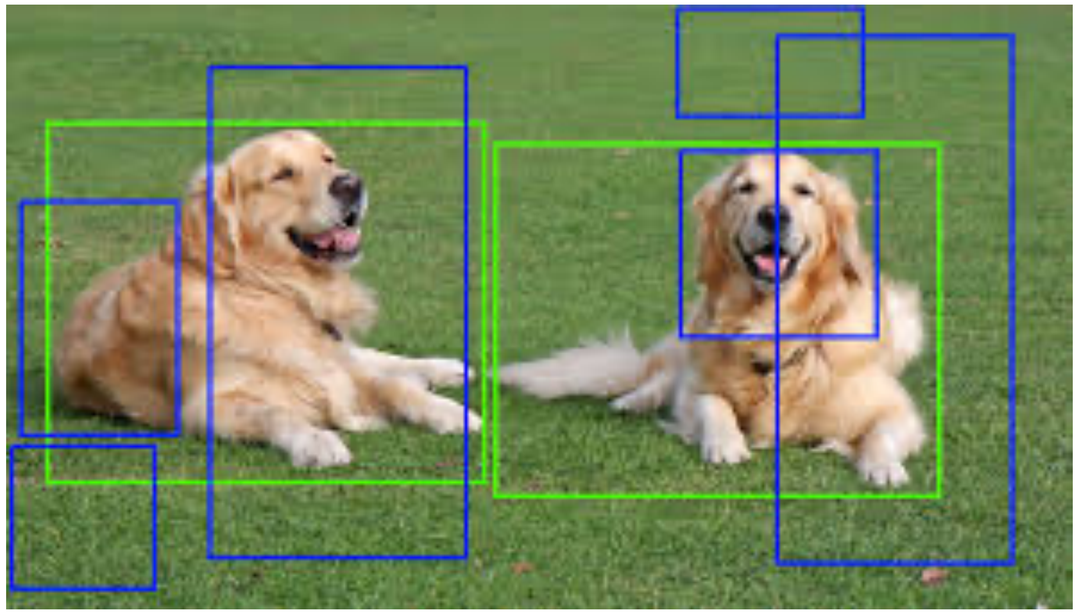
Types of object detectors

- Two-stage detectors



What defines an object?

- We need a generic, **class-agnostic** objectness measure: how likely it is for an image region to contain an object

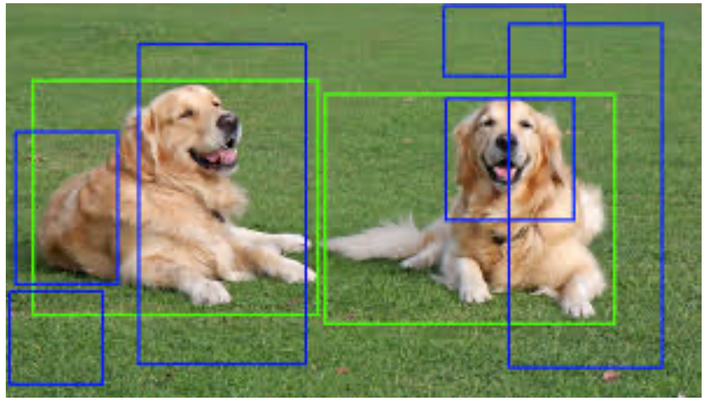


Very likely to be an object

Maybe it is an object

What defines an object?

- We need a generic, **class-agnostic** objectness measure: how likely it is for an image region to contain an object
- Using this measure yields a number of candidate **object proposals** or **regions of interest (RoI)** where to focus.



+ classifier

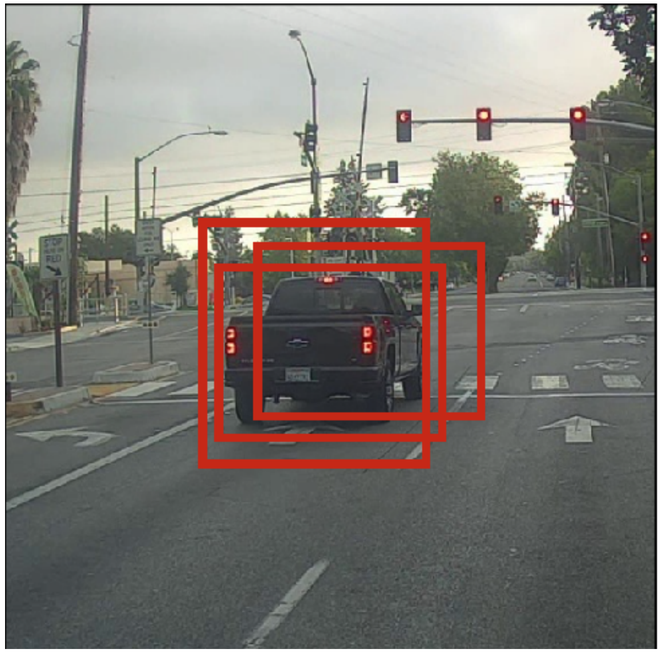
Object proposal methods

- Selective search: van de Sande et al. Segmentation as selective search for object recognition. ICCV 2011.
  - Using class-agnostic segmentation at multiple scales.

- Edge boxes: Zitnick and Dollar. Edge boxes: Locating object proposals from edges. ECCV 2014.
  - Bounding boxes that wholly enclose detected contours.

Do we want all proposals?

- Many boxes trying to explain one object
- We need a method to keep only the “best” boxes



Non-Maximum Suppression (NMS)

- Many boxes trying to explain one object
- We need a method to keep only the “best” boxes

Non-Maximum Suppression (NMS)

Algorithm 1 Non-Max Suppression

```
1:  procedure NMS(B,c)
2:      B_nms ← ∅
3:      for b_i ∈ B do
4:          discard ← False
5:          for b_j ∈ B do
6:              if same(b_i, b_j) > λ_nms then
7:                  if score(c, b_j) > score(c, b_i) then
8:                      discard ← True
9:          if not discard then
10:             B_nms ← B_nms ∪ b_i
11:     return B_nms
```

Start with anchor box i

For another box j

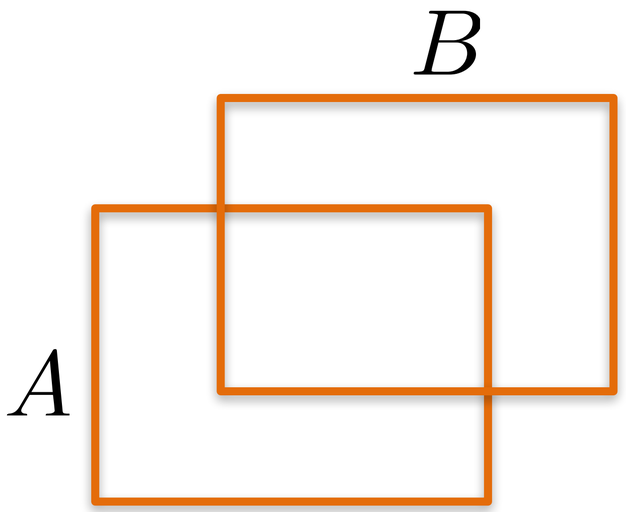
If they overlap

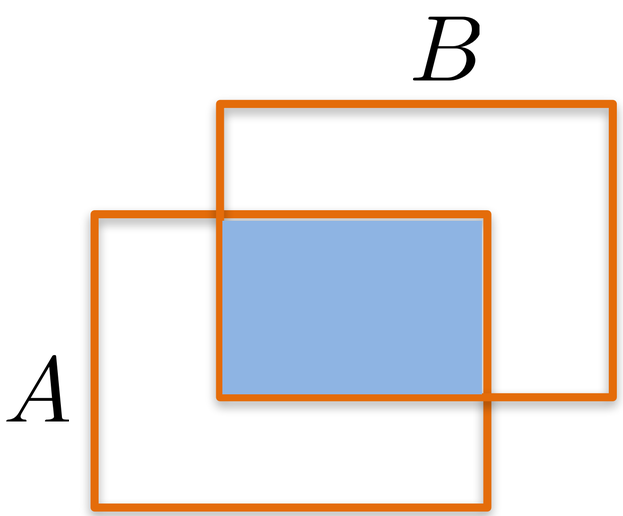
Discard box i if the score is lower than the score of j

Region overlap

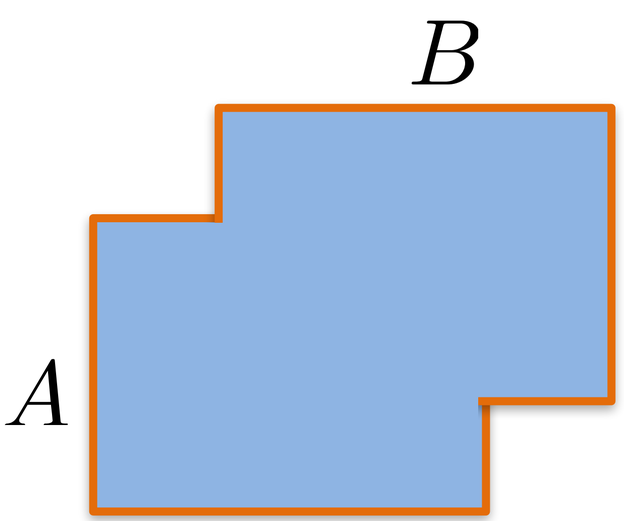
- We measure region overlap with the **Intersection over Union (IoU)** or **Jaccard Index**:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$





Intersection



Union

Region overlap

Algorithm 1 Non-Max Suppression

```
1:  procedure NMS(B,c)
2:      B_nms ← ∅
3:      for b_i ∈ B do
4:          discard ← False
5:          for b_j ∈ B do
6:              if same(b_i, b_j) > λ_nms then
7:                  if score(c, b_j) > score(c, b_i) then
8:                      discard ← True
9:          if not discard then
10:             B_nms ← B_nms ∪ b_i
11:     return B_nms
```

Start with anchor box i

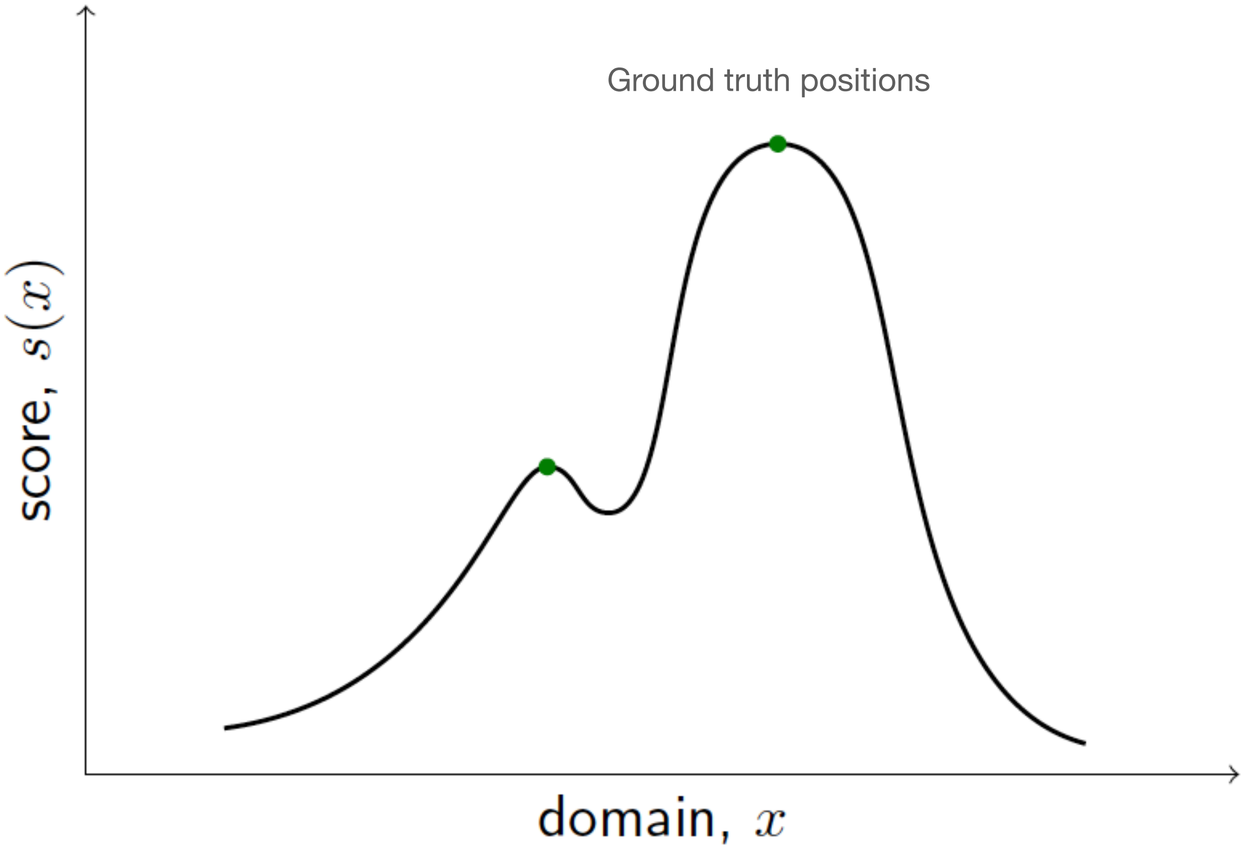
For another box j

If they overlap

Discard box i if the score is lower than the score of j

NMS: The problem

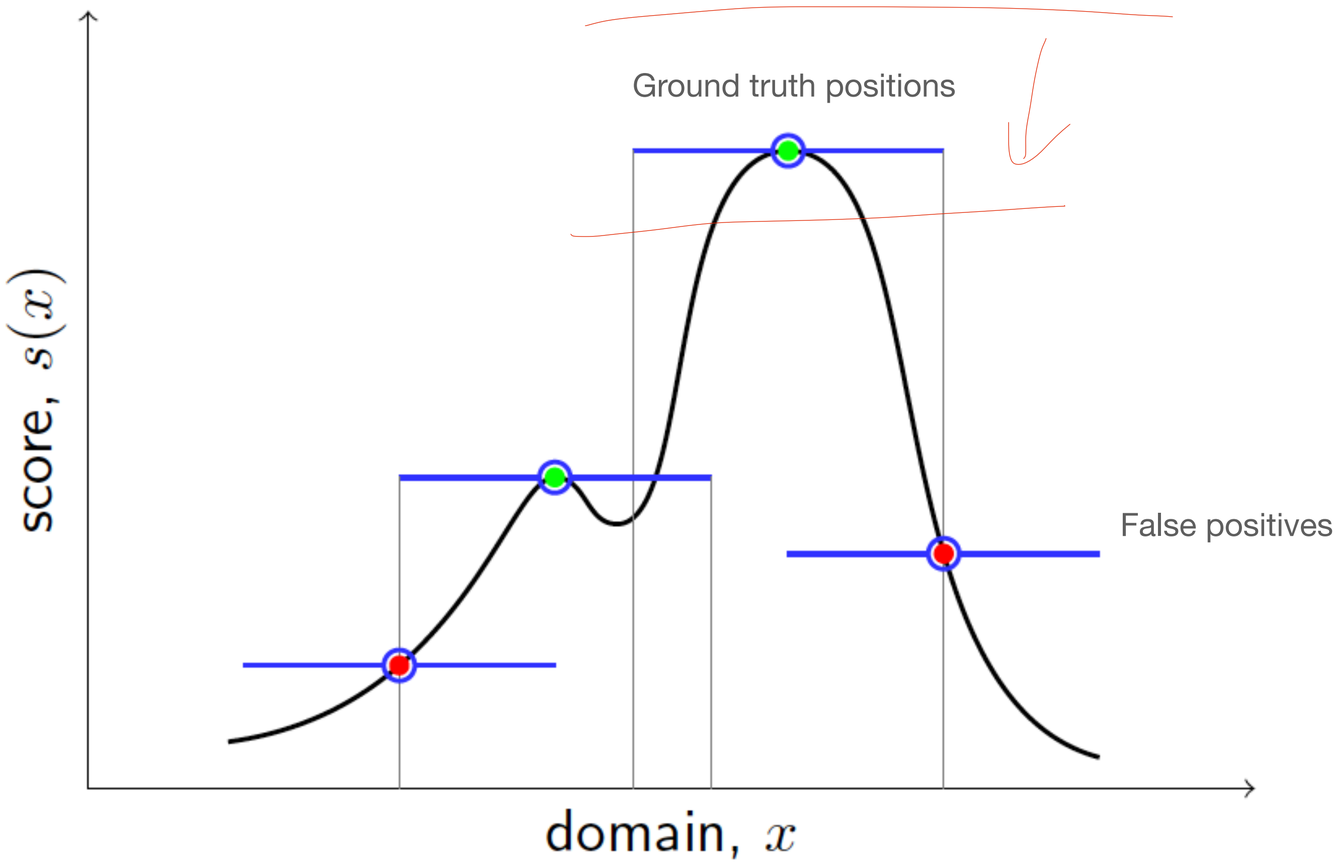
- Consider a 1D example:



Hosang, Benenson and Schiele. A ConvNet for Non-Maximum Suppression. GCPR 2015.

NMS: The problem

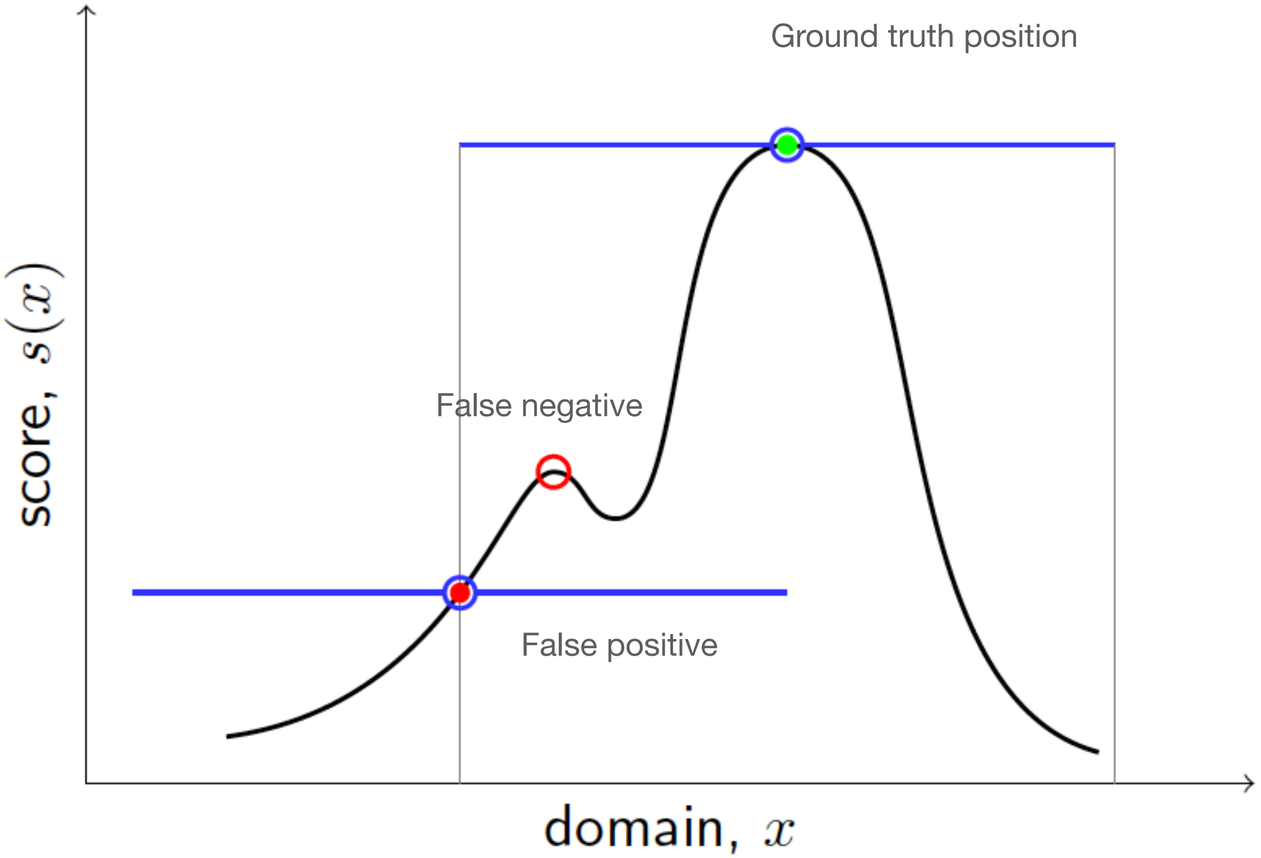
- Choosing a high threshold – more false positives, low precision



Hosang, Benenson and Schiele. A ConvNet for Non-Maximum Suppression. GCPR 2015.

NMS: The problem

- Choosing a low threshold – more false negatives, low recall

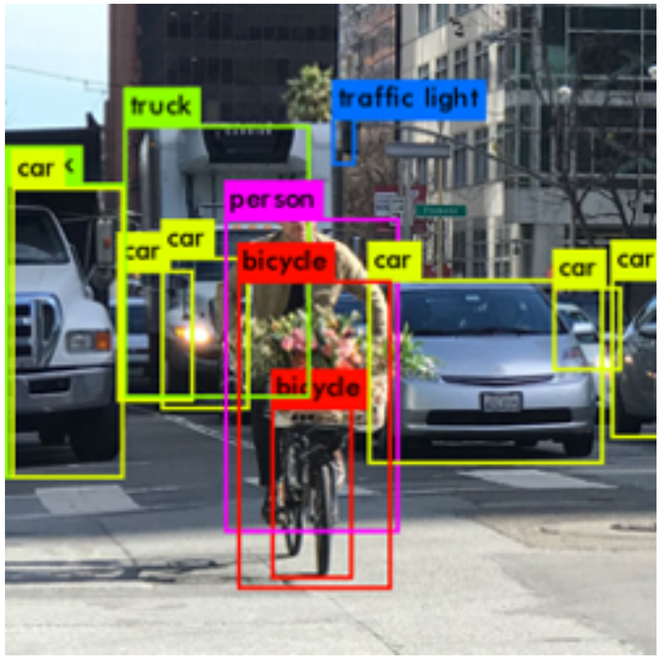


Non-Maximum Suppression (NMS)

- NMS will be used at test time. Most detection methods (even Deep Learning ones) use NMS!

Towards real-world object detection

Can we design features that work everywhere?



Probably not, but we can learn them from data.

↓

Deep Learning (starting next week!)

Deep object detectors – next week(s)

- One-stage detectors
  - YOLO, SSD, RetinaNet
  - CenterNet, CornerNet, ExtremeNet
- Two-stage detectors
  - R-CNN, Fast R-CNN, Faster R-CNN
  - SPP-Net, R-FCN, FPN

What we've learned today

- Course logistics
  - Questions? Use Moodle!
  - First exercise session: This Thursday (19.10)!
- (Old-school) object detection:
  - A sliding window approach. Metrics: SSD, NCC, ZNCC.
  - Feature-based detection: Viola-Jones (Haar features), HoGs.
  - Object proposals and NMS.

Computer Vision III:
Introduction

Dr. Nikita Araslanov
17.10.2023

Content credit:
Prof. Dr. Laura Leal-Taixé
https://dvl.in.tum.de