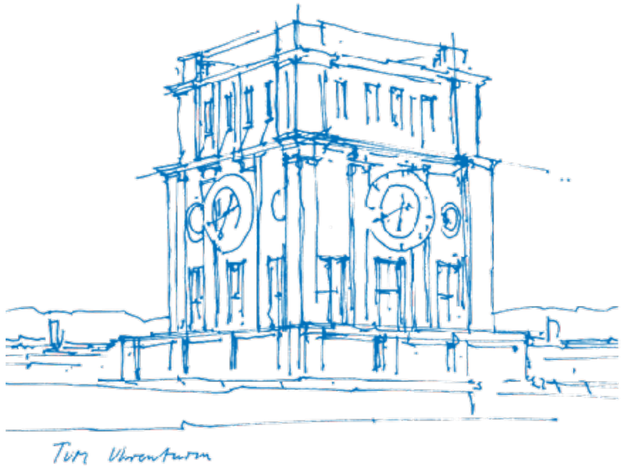
Computer Vision III:

Instance segmentation

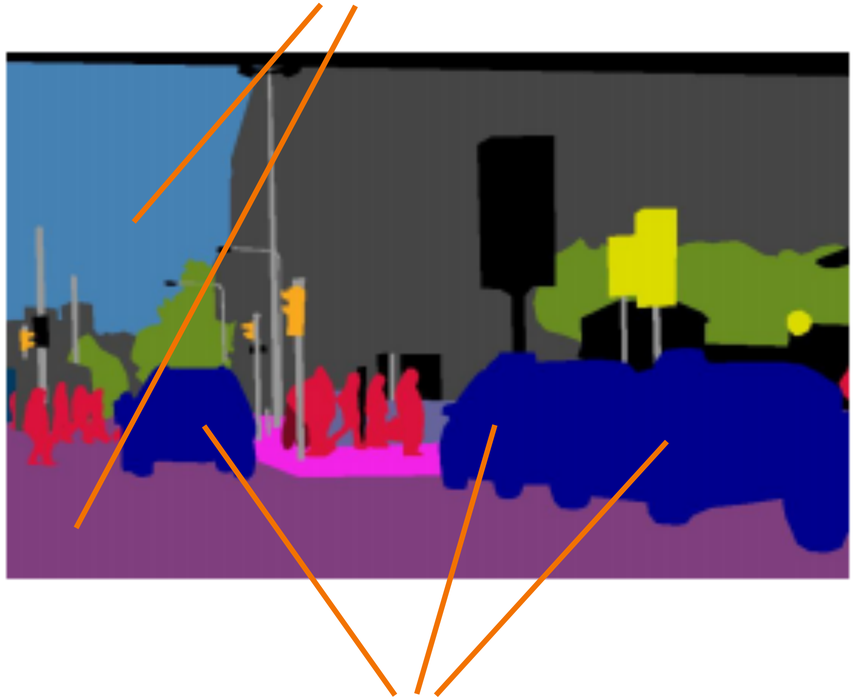
Nikita Araslanov
13.12.2022

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



Semantic segmentation

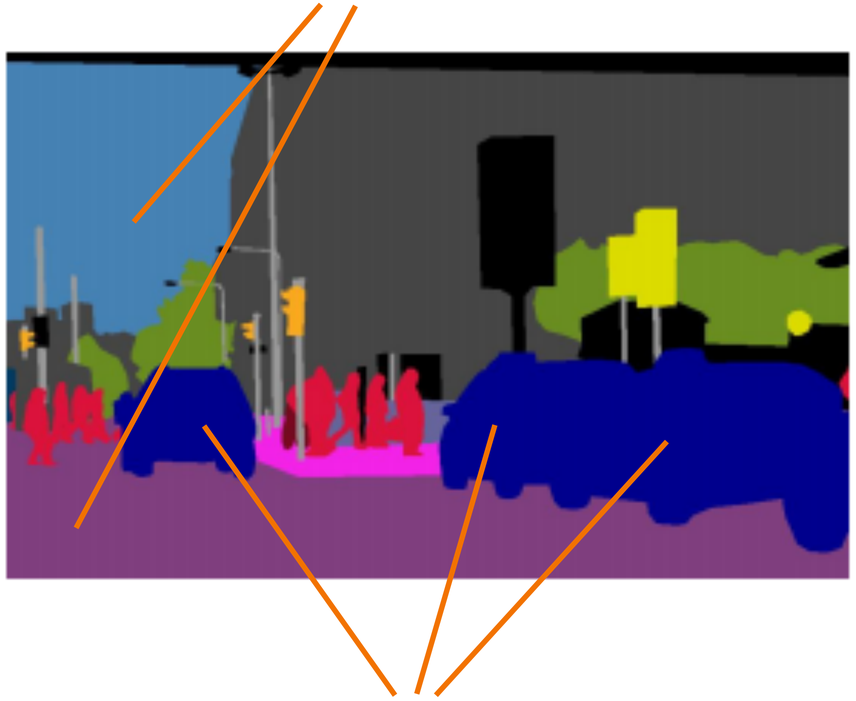
Label every pixel, including the background (sky, grass, road)



Does not differentiate between the pixels from objects (instances) of the same class

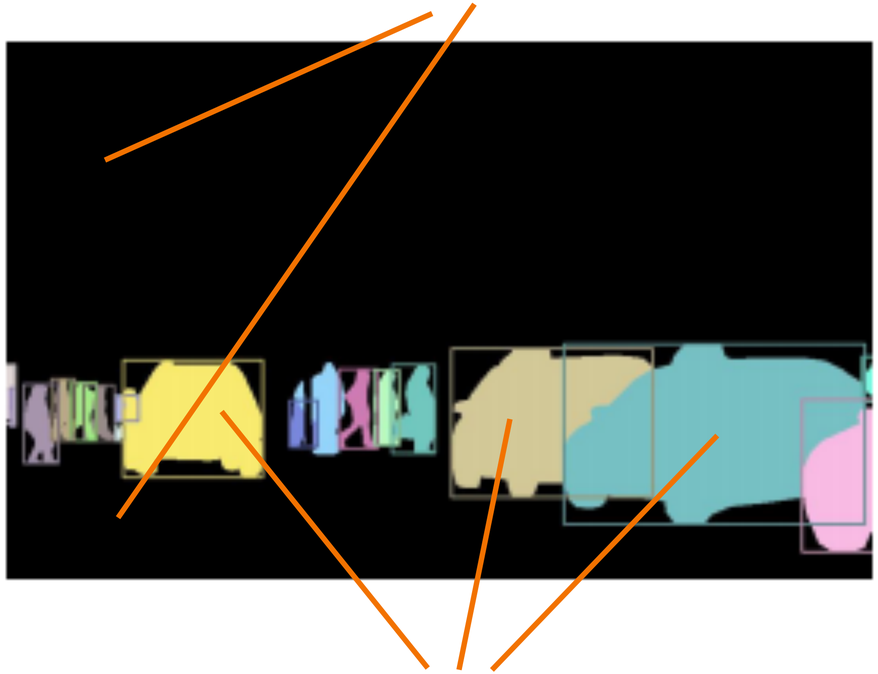
Instance segmentation

Label every pixel, including the background (sky, grass, road)



Does not differentiate between the pixels from objects (instances) of the same class

Do not label pixels coming from uncountable objects (“stuff”), e.g. “sky”, “grass”, “road”

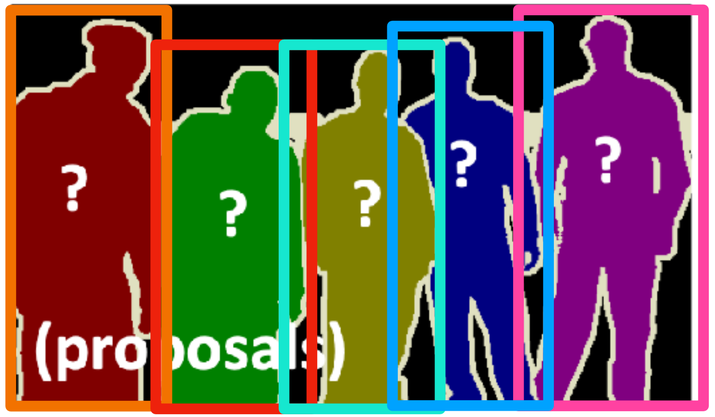


Differentiates between the pixels coming from instances of the same class

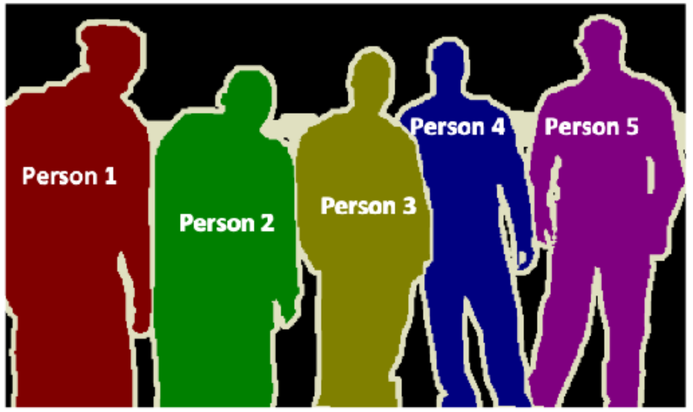
Instance segmentation methods

Proposal-based

1. Proposals
(e.g. bounding boxes)

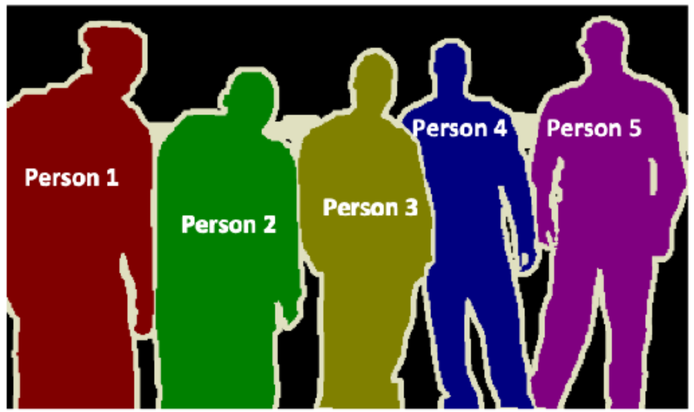


2. Segment and classify



Proposal-free

1. Semantic segmentation (optional)

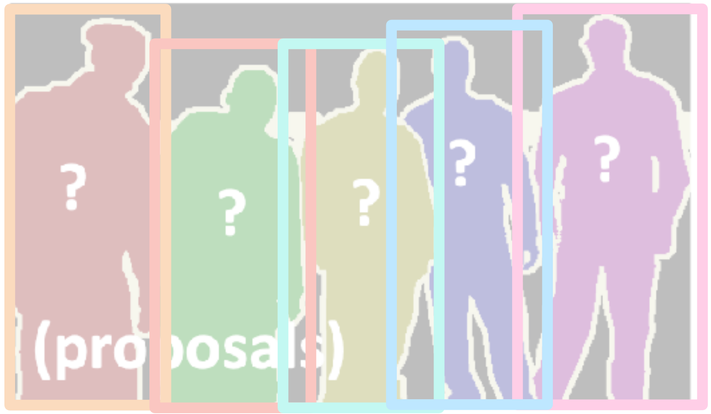


2. Group pixels into instances

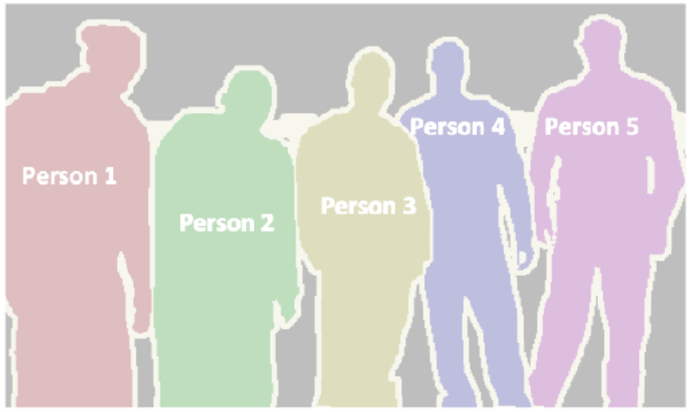
Instance segmentation methods

Proposal-based

1. Proposals
(e.g. bounding boxes)

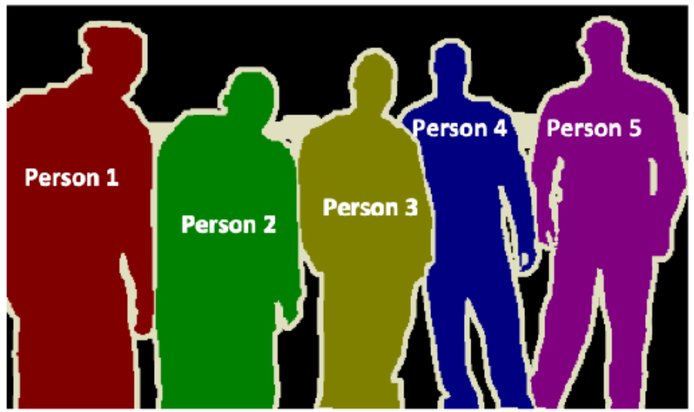


2. Segment and classify



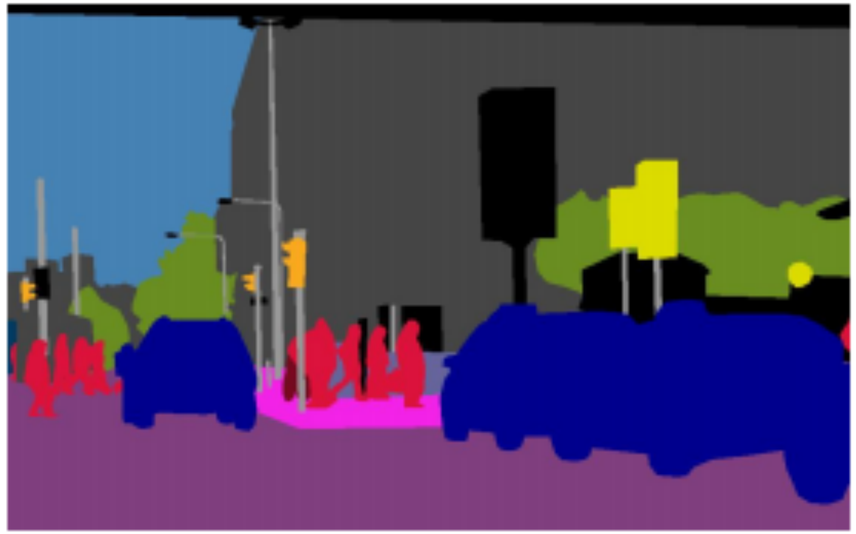
Proposal-free

1. Semantic segmentation (optional)



2. Group pixels into instances

Proposal-free methods

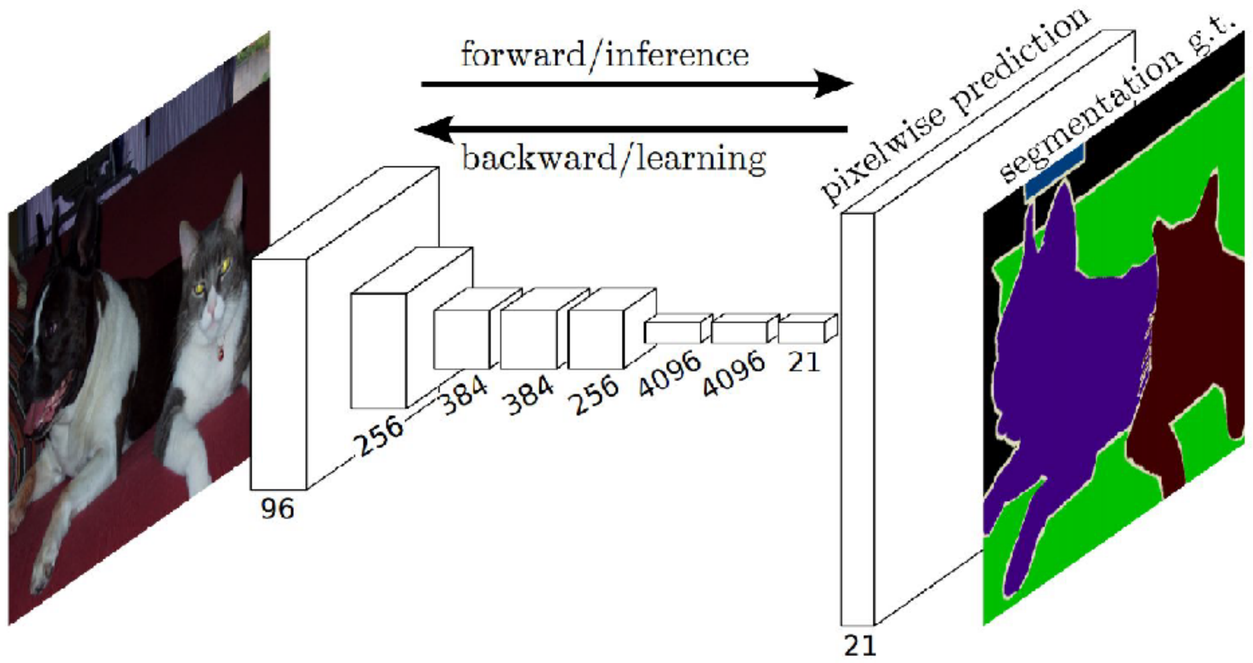


A semantic map

We already know how to obtain this!

Why proposal-free?

- Fully Convolutional Networks for Semantic Segmentation



Long et al., (2015)

Proposal-free methods

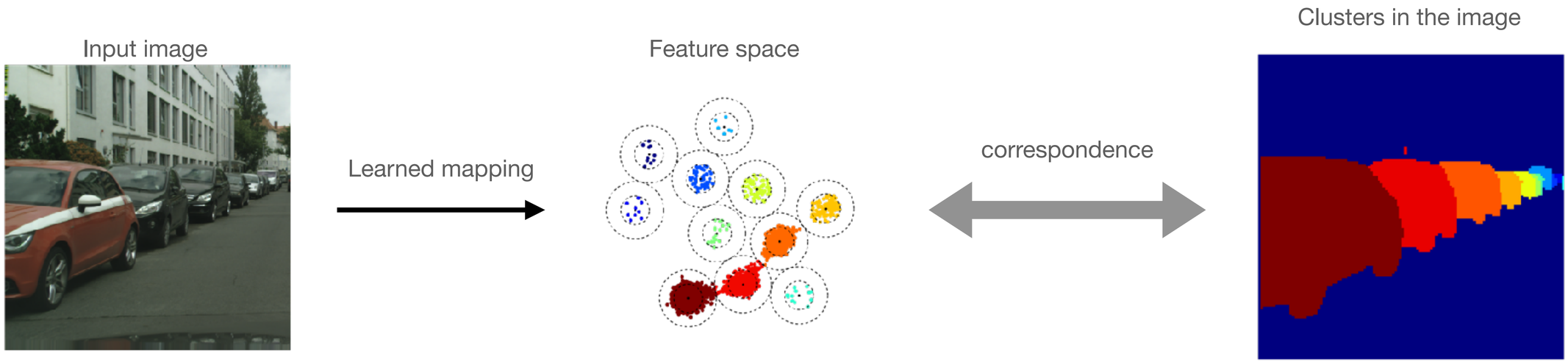
- Silberman et al. “Instance Segmentation of Indoor Scenes using a Coverage Loss” (2012).
- Liang et al. “Proposal-free Network for Instance-level Object Segmentation“ (2015).
- De Brabandere et al. “Semantic Instance Segmentation with a Discriminative Loss Function” (2017).
- Kirillov et al. „InstanceCut: from Edges to Instances with MultiCut“ (2017).
- Bai and Urtasun “Deep Watershed Transform for Instance Segmentation“ (2017).

Proposal-free methods

- Silberman et al. “Instance Segmentation of Indoor Scenes using a Coverage Loss” (2012).
- Liang et al. “Proposal-free Network for Instance-level Object Segmentation“ (2015).
- **De Brabandere et al. “Semantic Instance Segmentation with a Discriminative Loss Function” (2017).**
- Kirillov et al. „InstanceCut: from Edges to Instances with MultiCut“ (2017).
- Bai and Urtasun “Deep Watershed Transform for Instance Segmentation“ (2017).

Instances via clustering

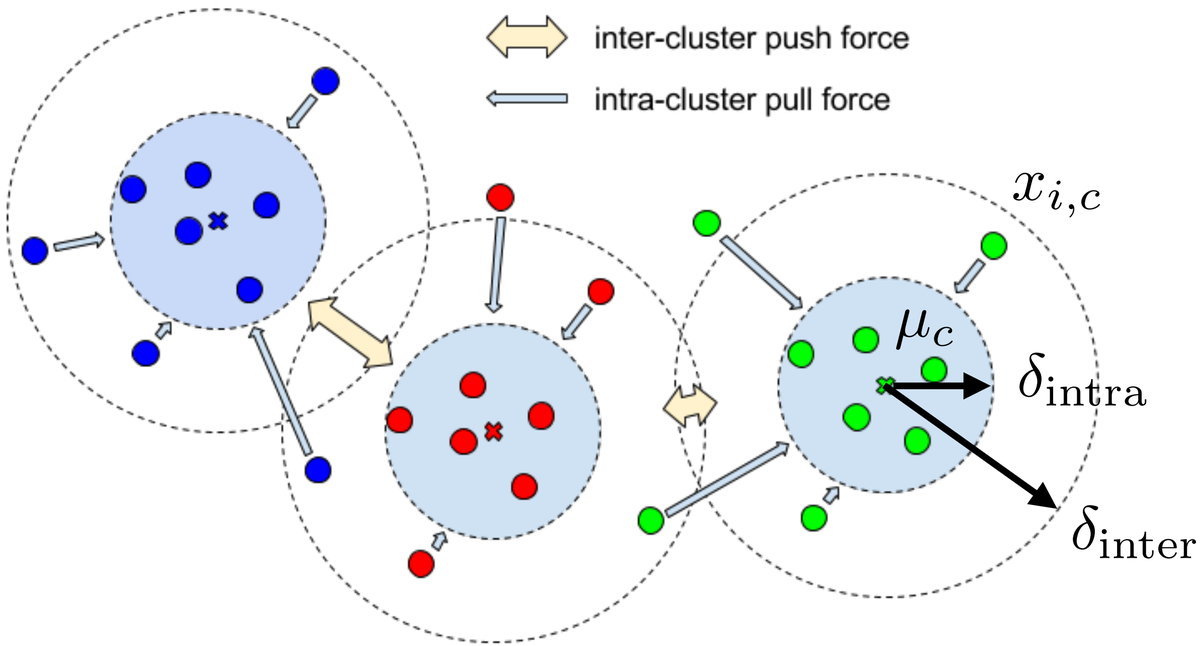
- Instance as a cluster in feature space:



De Brabandere et al. „Semantic Instance Segmentation with a Discriminative Loss Function“ (2017).

Instances via clustering

- Recall metric learning and hinge loss:



Intra-cluster term:

$$\max(0, \|\mu_c - x_{i,c}\|_2 - \delta_{\text{intra}})$$

Inter-cluster term:

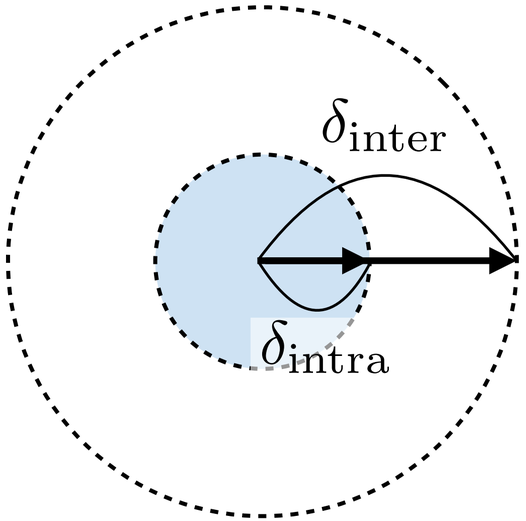
We don't want to intersect classes.

$$\max(0, 2\delta_{\text{inter}} - \|\mu_c - \mu_{c'}\|_2)$$

De Brabandere et al. „Semantic Instance Segmentation with a Discriminative Loss Function“ (2017).

Instances via clustering

- The learned metric space can be used for clustering.
- How to actually perform the clustering?
- We can set: $\delta_{\text{inter}} > 2\delta_{\text{intra}}$
- After training, any embedding is within distance $2\delta_{\text{intra}}$ to all embeddings of the same cluster (ideally).



Instances via clustering

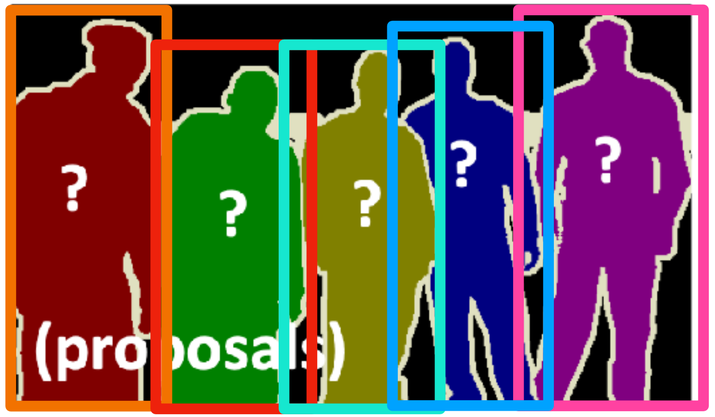
- Test-time strategy:
  1. Select unlabelled pixel.
  2. Find (unlabelled) neighbours.
  3. Assign the pixels to a new cluster.
  4. Repeat until all pixels are labelled.

- More robust: Mean shift (Comaniciu et al., 2002)

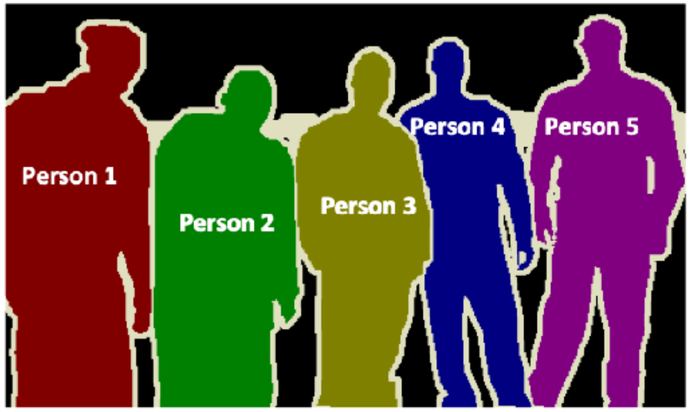
Instance segmentation methods

Proposal-based

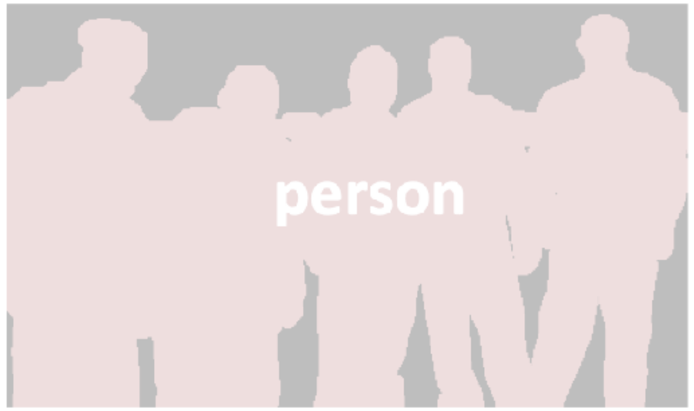
1. Proposals
(e.g. bounding boxes)



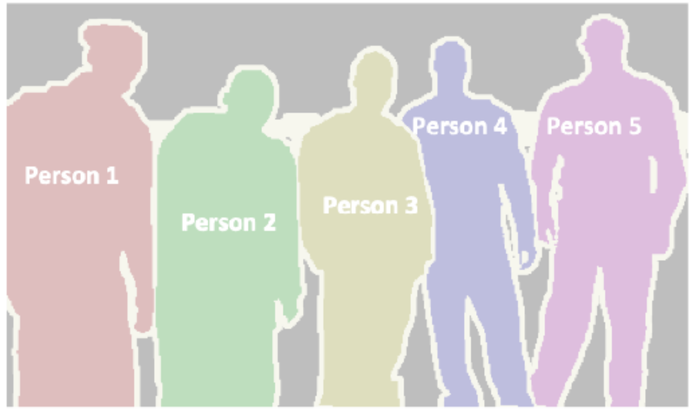
2. Segment and classify



Proposal-free



1. Semantic segmentation (optional)

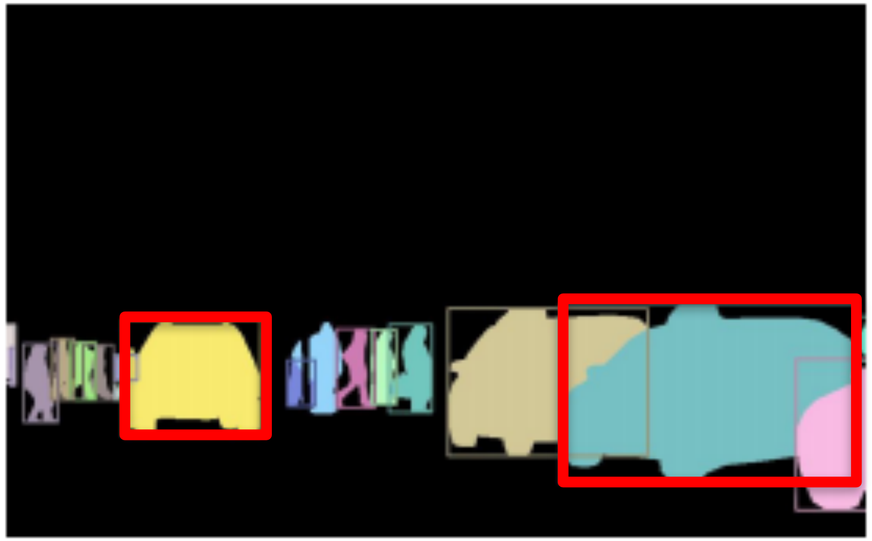


2. Group pixels into instances

Proposal-based methods

Bounding boxes...

We already know how to obtain those!



Proposal-based methods

- B. Hariharan et al. “Simultaneous Detection and Segmentation“. ECCV 2014
  - Follow-up work: B. Hariharan et al. “Hypercolumns for Object Segmentation and Fine-grained Localization ”. CVPR 2015

- Dai et al. „Instance-aware Semantic Segmentation via Multi-task Network Cascades“. CVPR 2016
  - Previous work: Dai et al. “Convolutional Feature Masking for Joint Object and Stuff Segmentation“. CVPR 2015

Proposal-based methods

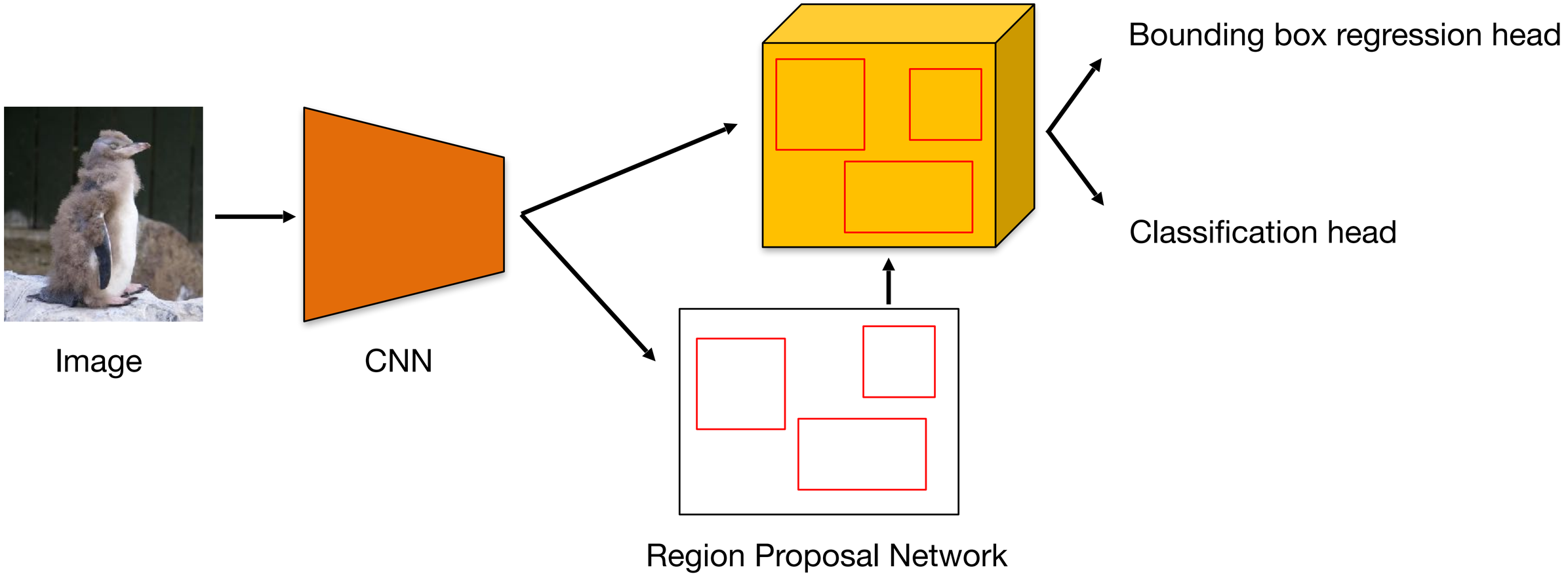
- Can we extend our best object detection to instance segmentation?

- Start with Faster R-CNN
  - add another head → “mask head”
  - **Mask R-CNN**

Mask R-CNN

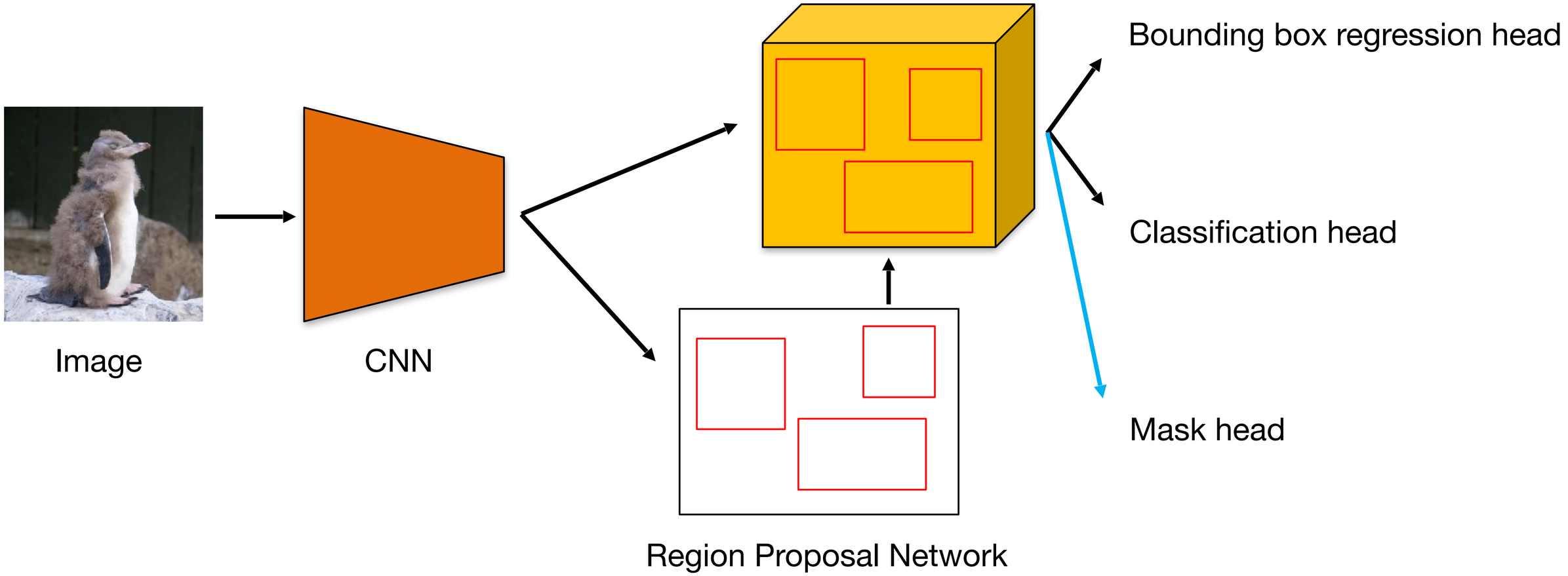
What is Mask R-CNN?

- Starting from the Faster R-CNN architecture

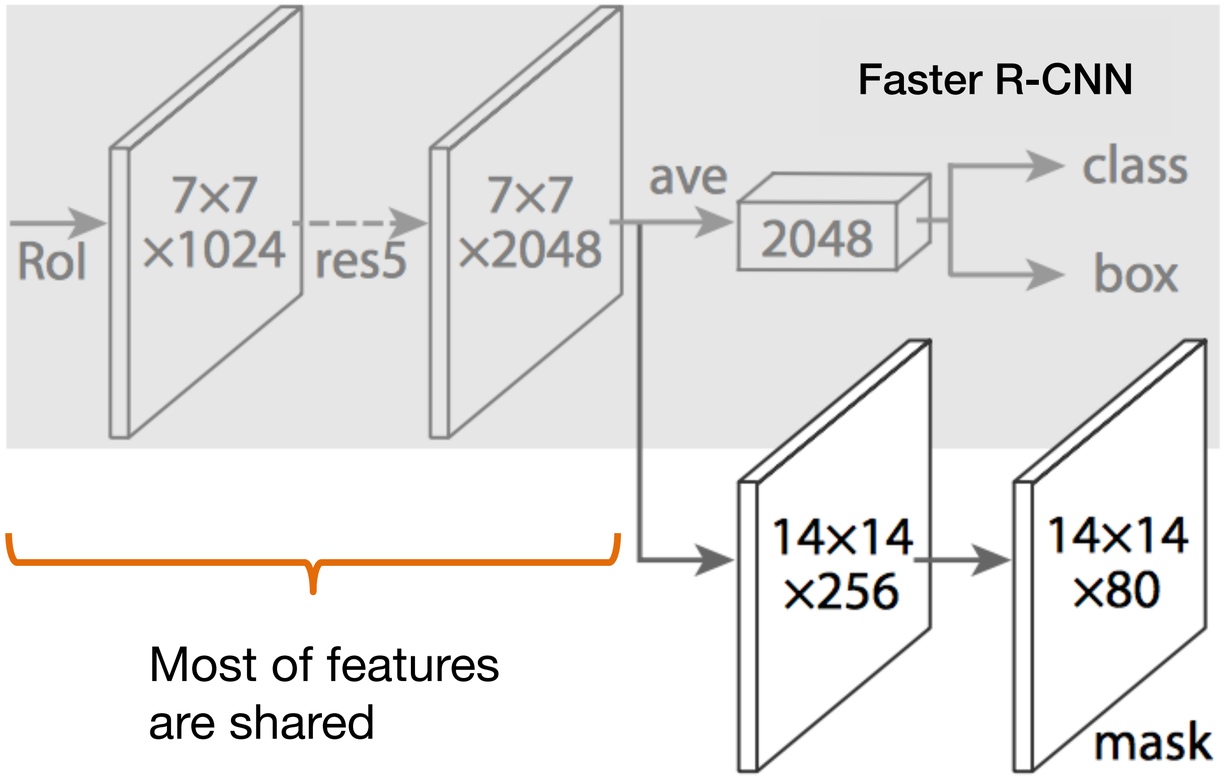


What is Mask R-CNN?

- Starting from the Faster R-CNN architecture



What is Mask R-CNN?

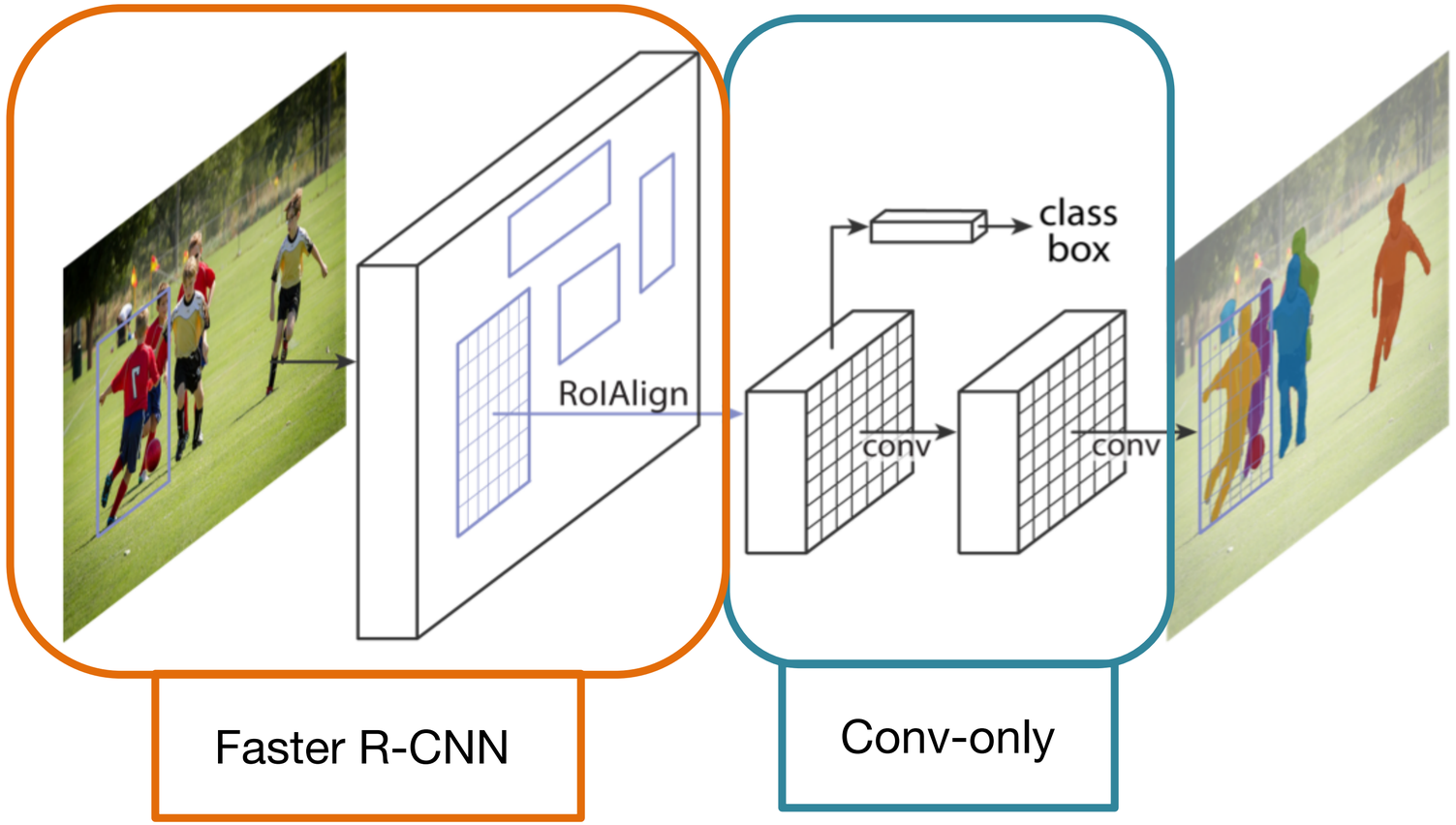


Object recognition head

Segmentation head

What is Mask R-CNN?

- Faster R-CNN + mask head for segmentation

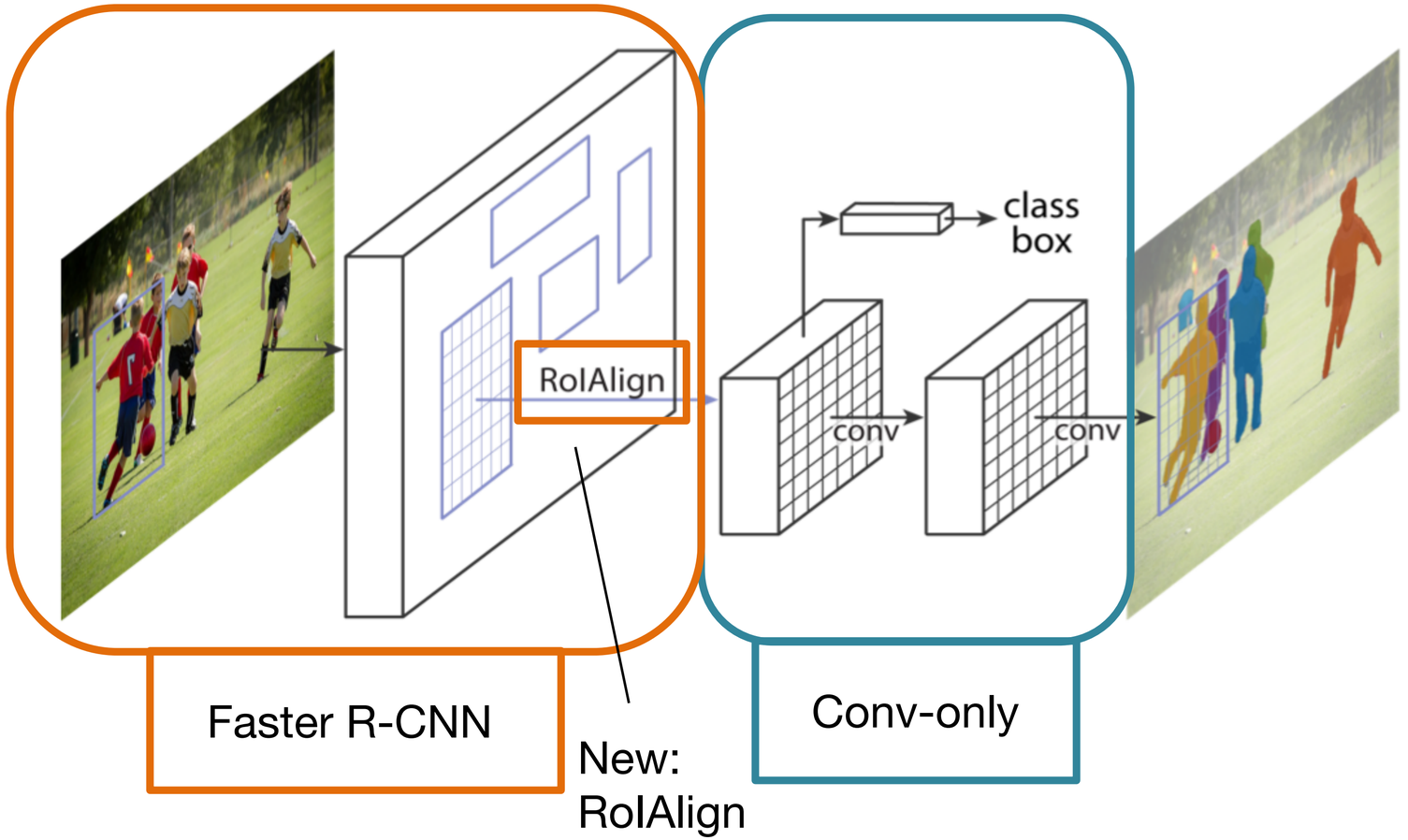


+ mask loss:
cross-entropy per pixel

He at al. “Mask R-CNN” ICCV 2017

What is Mask R-CNN?

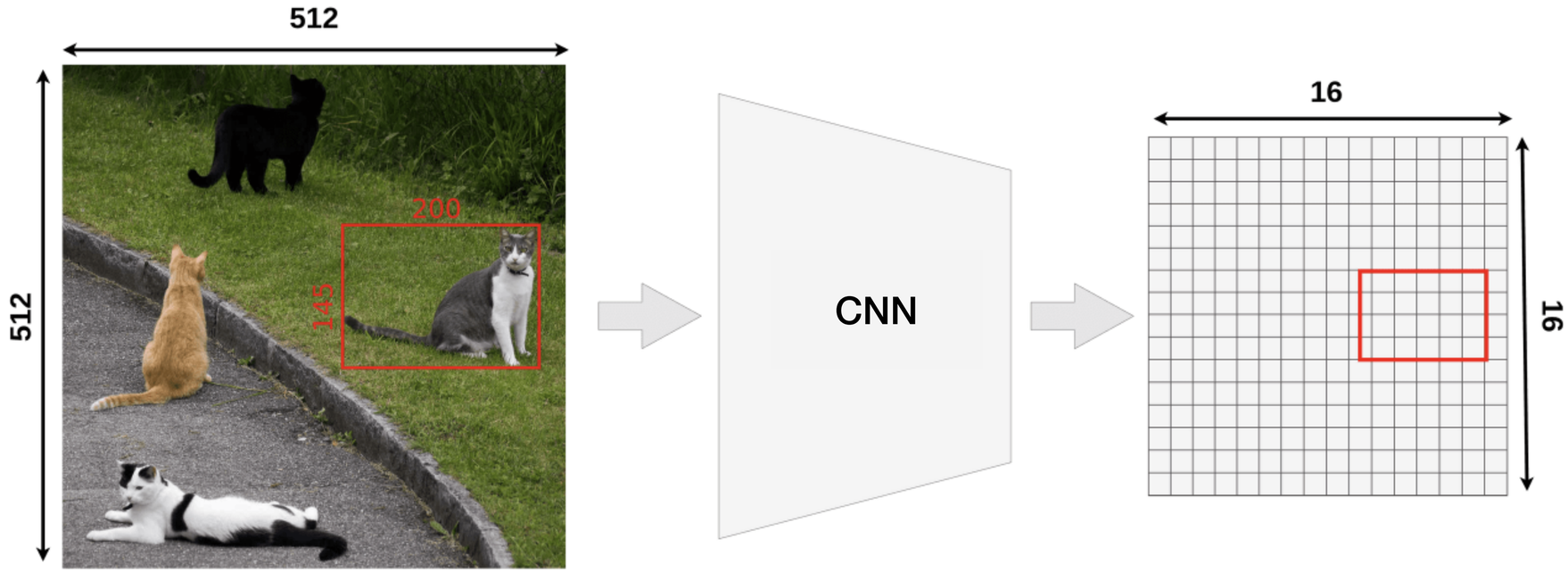
- Faster R-CNN + mask head for segmentation



+ mask loss:
cross-entropy per pixel

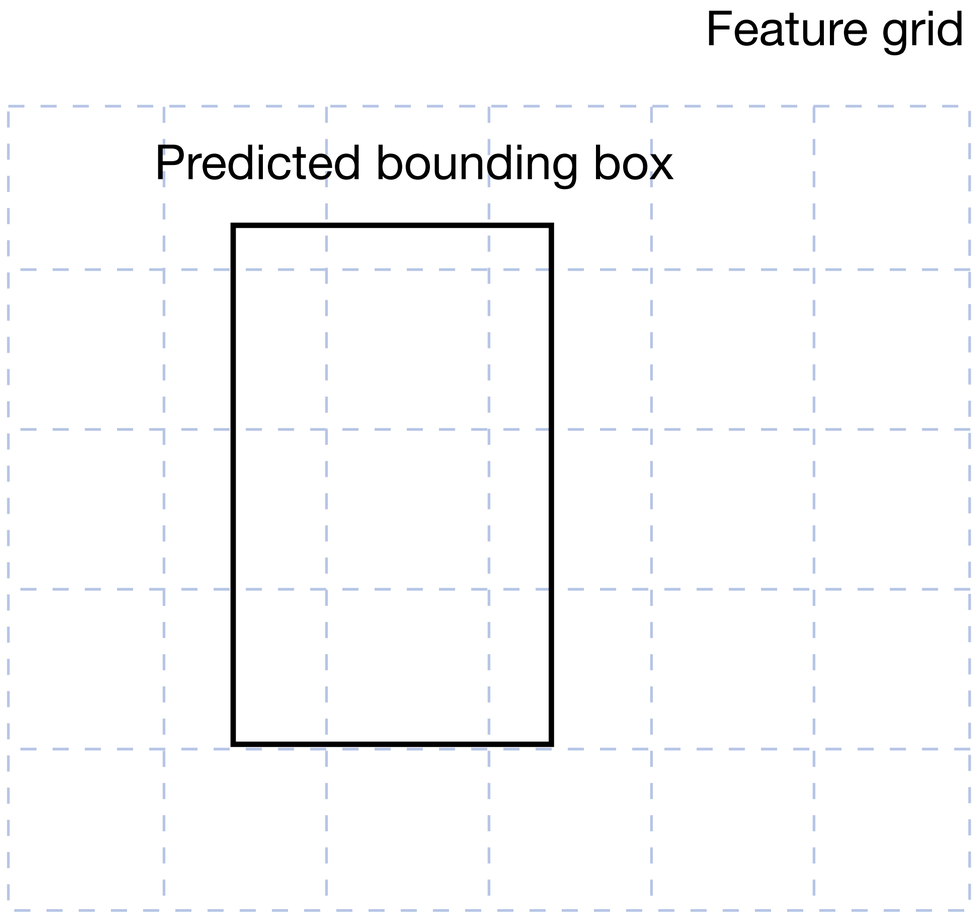
He at al. “Mask R-CNN” ICCV 2017

Recall RoIPool



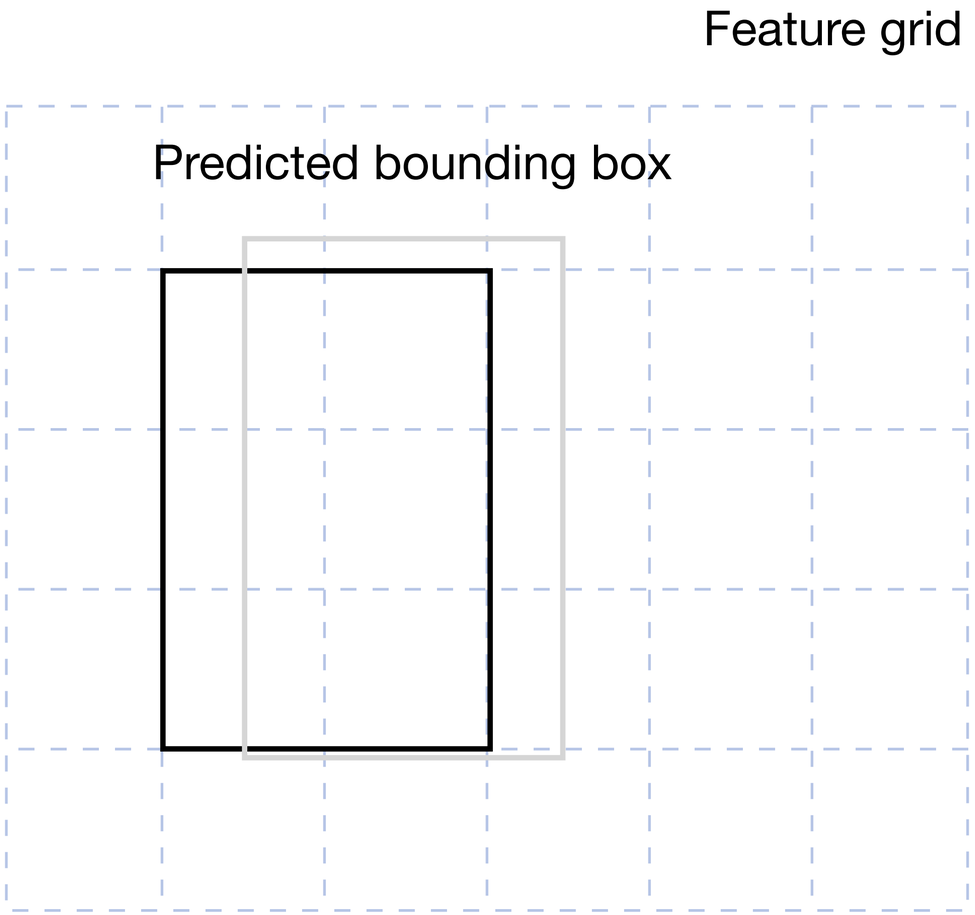
Model feature mapping process

RoIPool



Two quantisations:

RoIPool

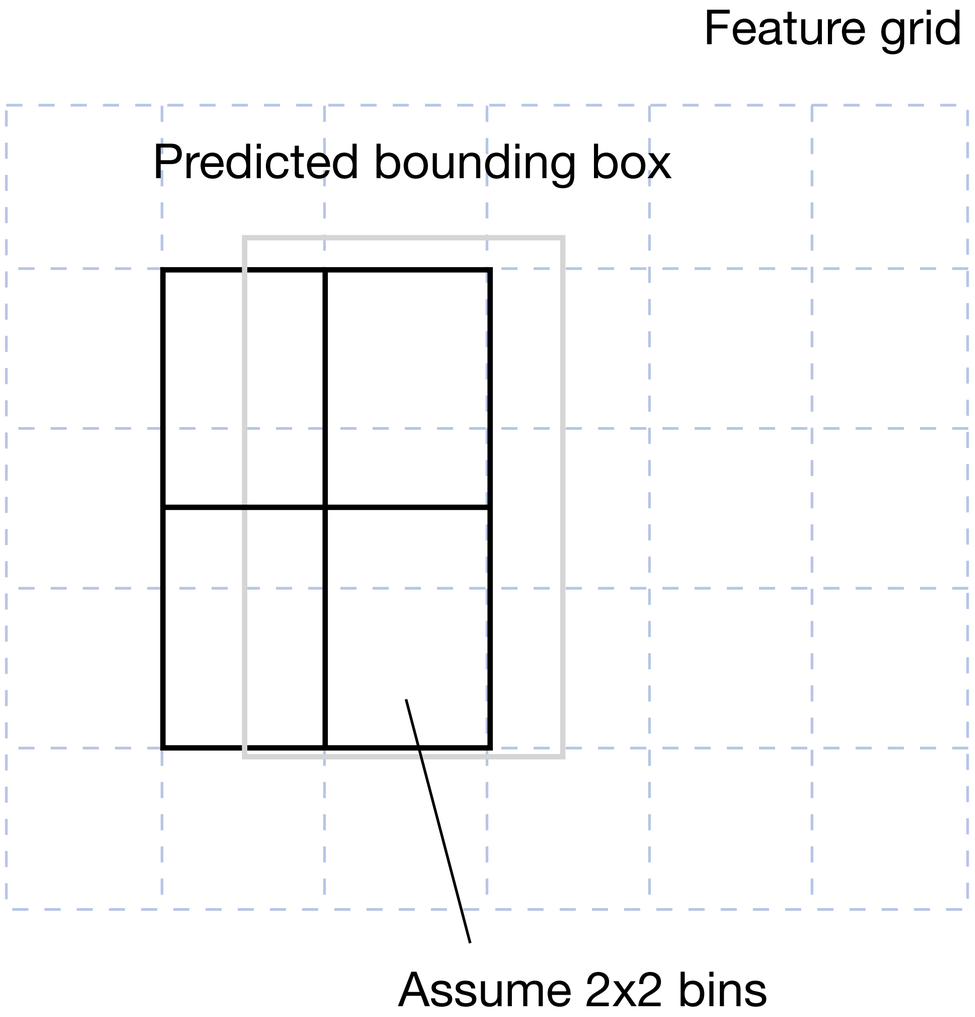


Two quantisations:

1. Bounding box alignment

float bounding box to integer bounding box

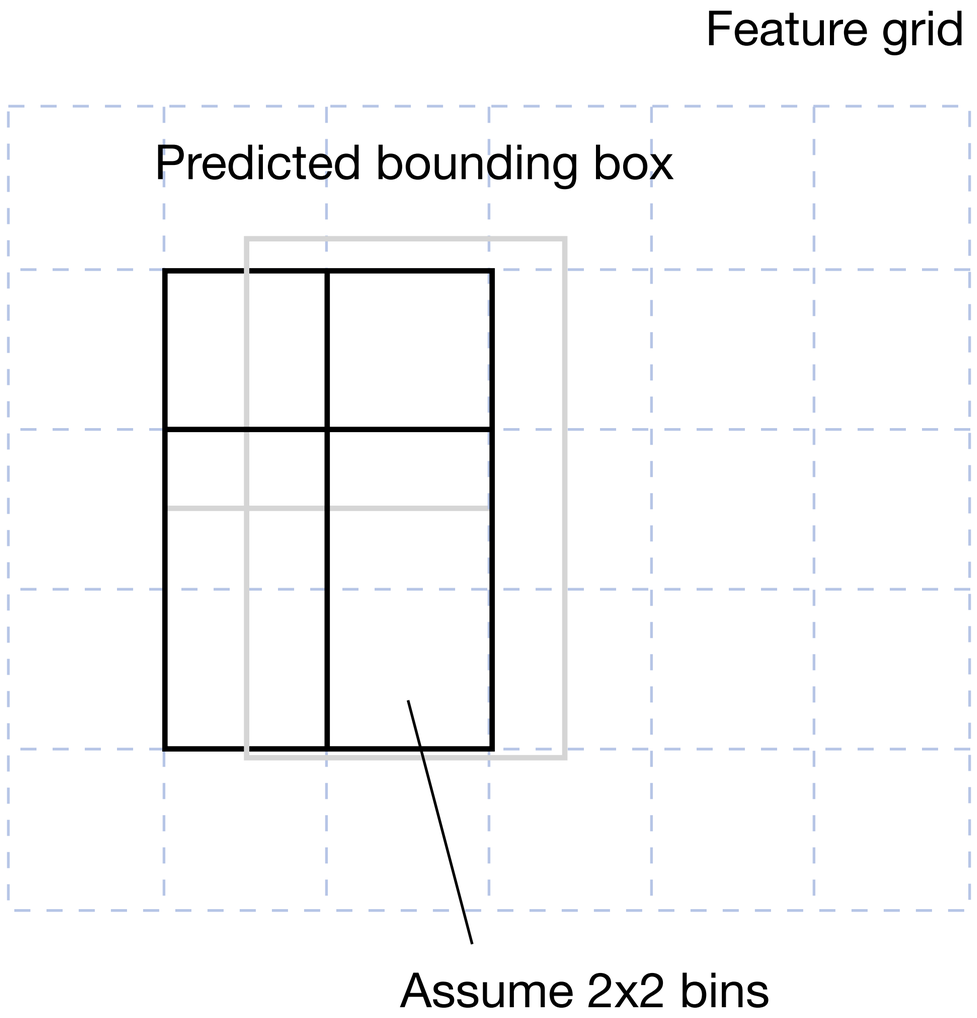
RoIPool



Two quantisations:

1. Bounding box alignment

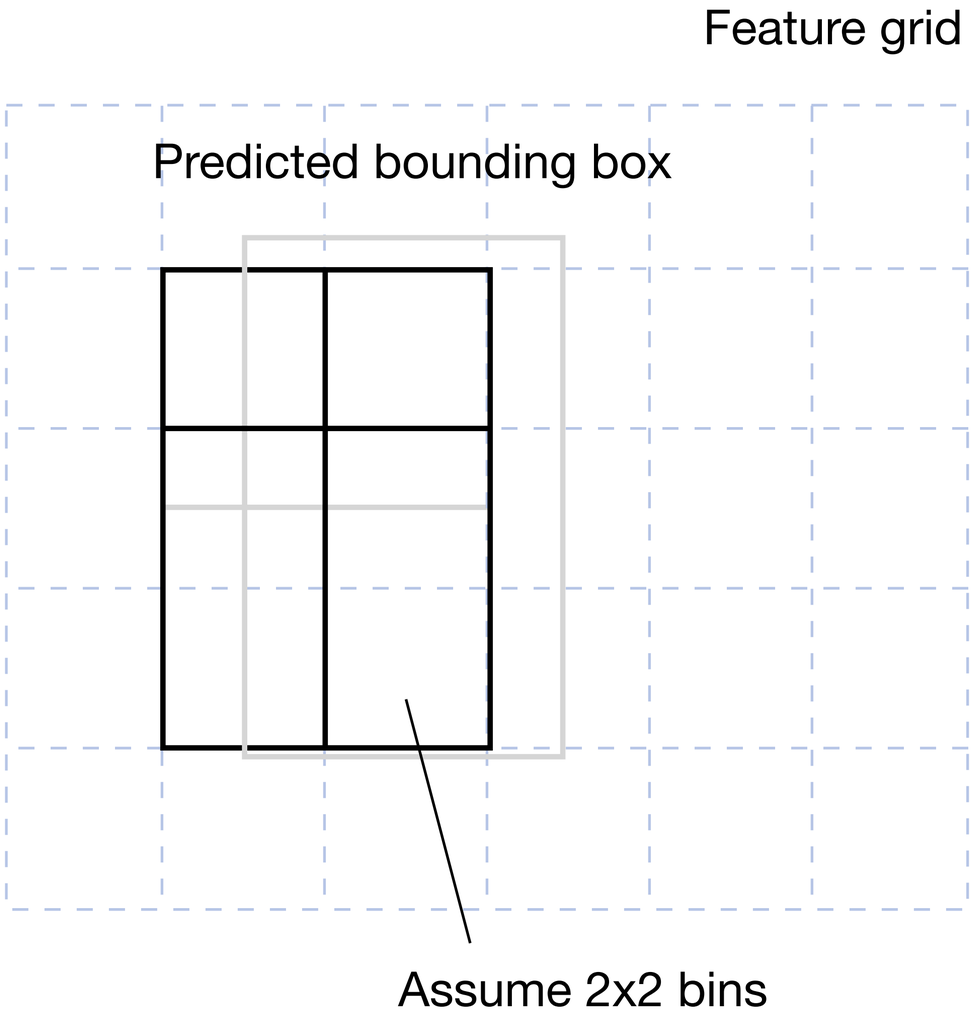
RoIPool



Two quantisations:

1. Bounding box alignment
2. Bin alignment

RoIPool



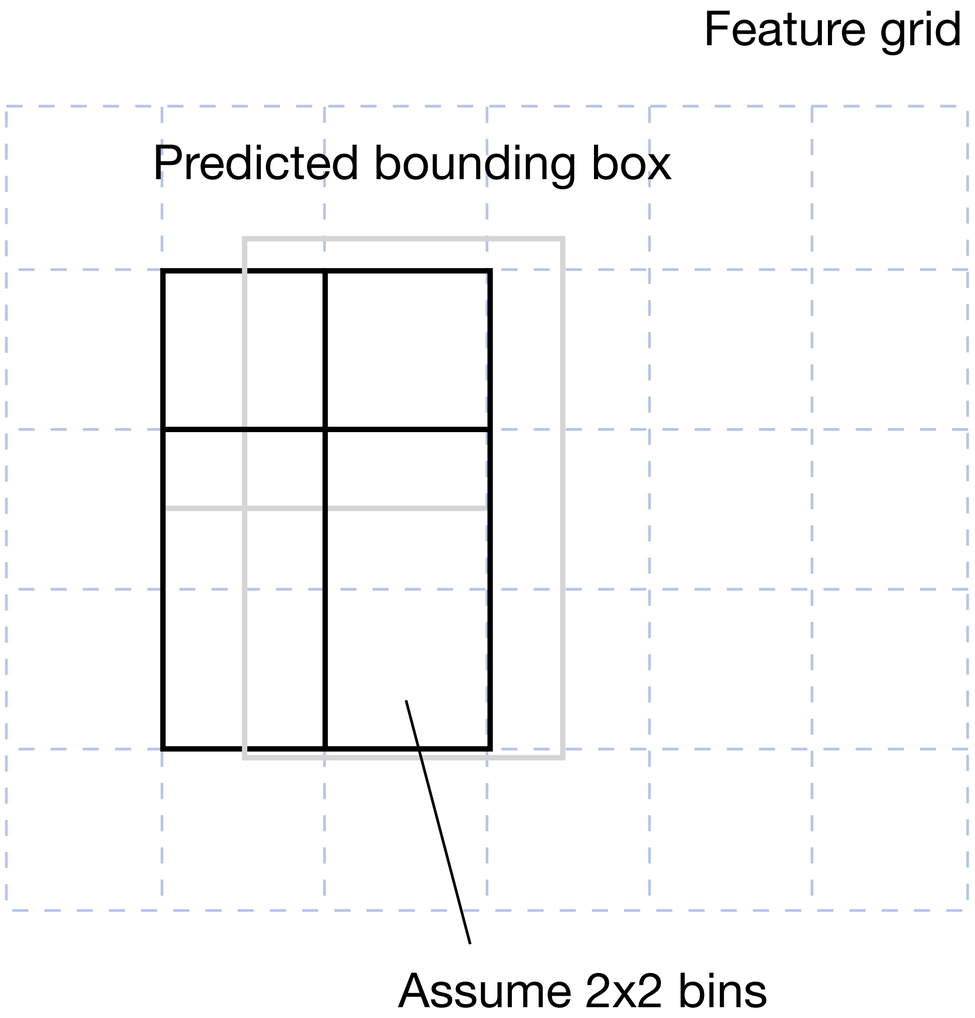
Two quantisations:

1. Bounding box alignment
2. Bin alignment

Pooling within each bin

(max or average)

RoIPool



Two quantisations:

1. Bounding box alignment
2. Bin alignment

Pooling within each bin

(max or average)

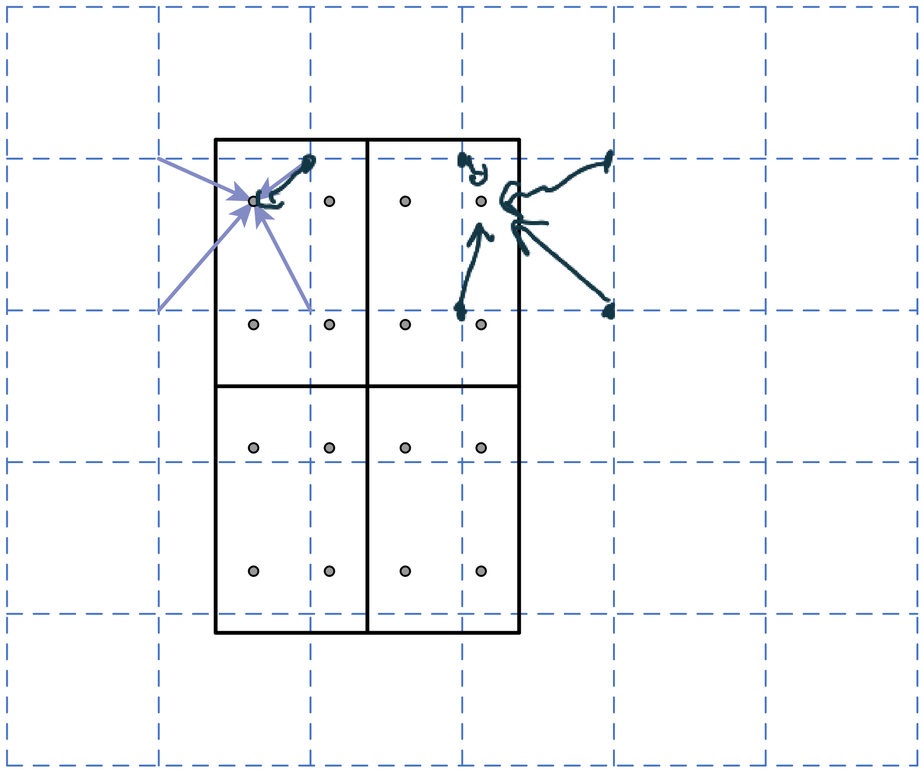
Works well for object detection

(we use this for classification only)

RoIPool vs. RoIAlign

- We need accurate localisation for mask prediction
- RoIPool is inaccurate due to two quanitsations
- Better alternative: RoIAlign

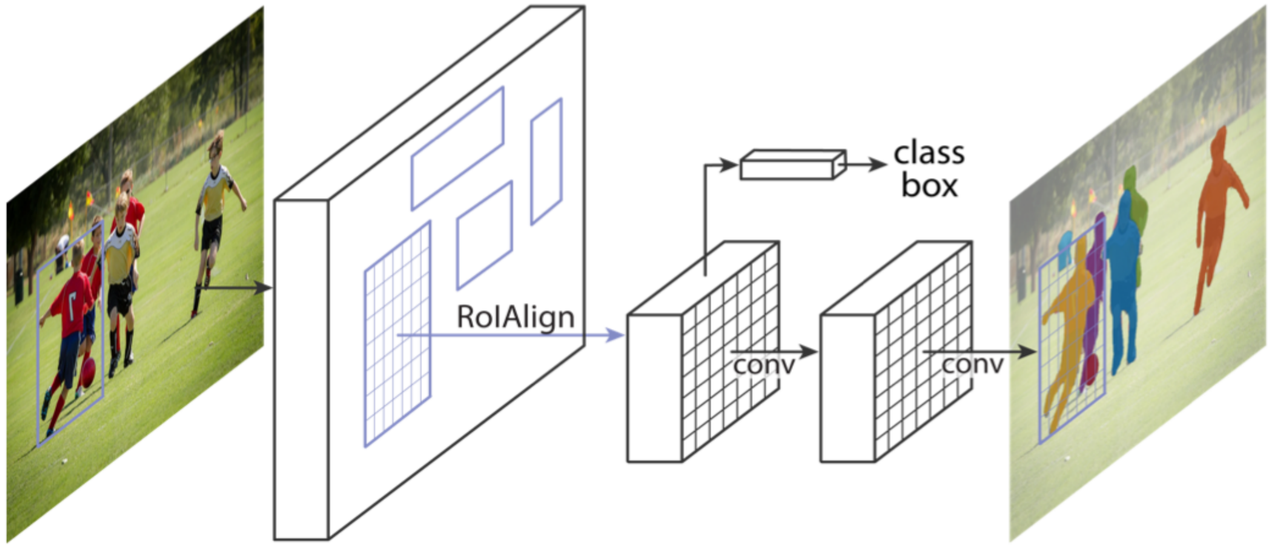
RoIAlign



- No quantisation;
- Define 4 regularly placed sampling points within each bin;
- Compute feature values with bilinear interpolation.
- Aggregate each bin as before (max or average pooling)

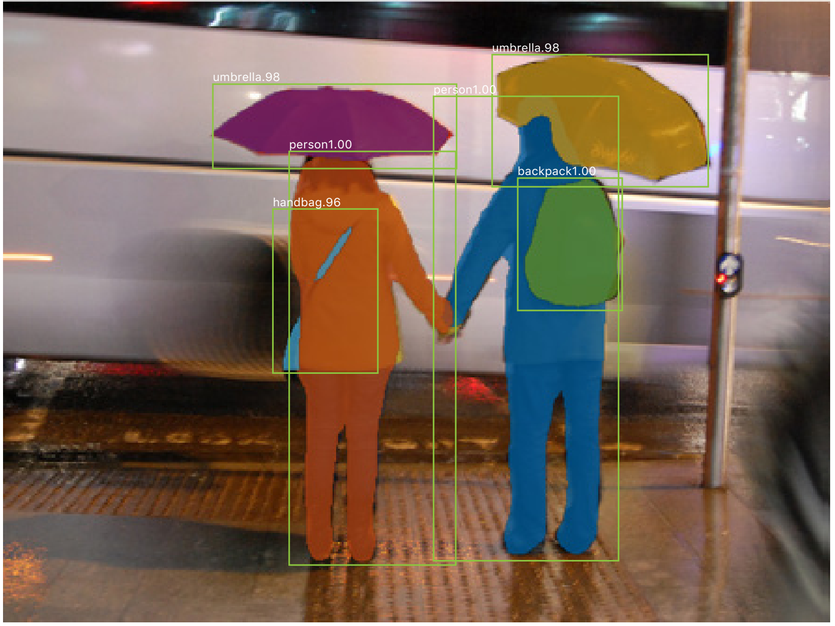
What is Mask R-CNN?

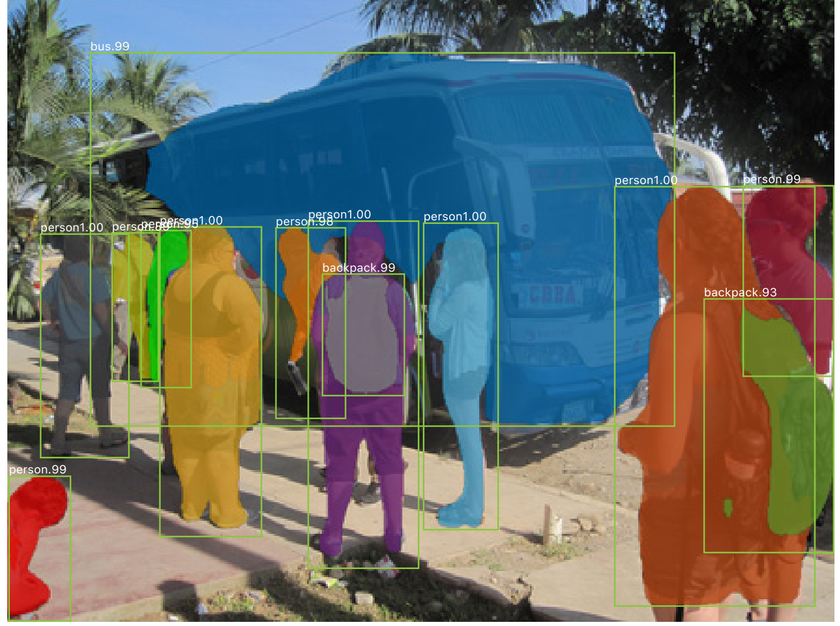
- Seemingly incremental improvements
- Simple design overall
- Best paper award (CVPR 2017)

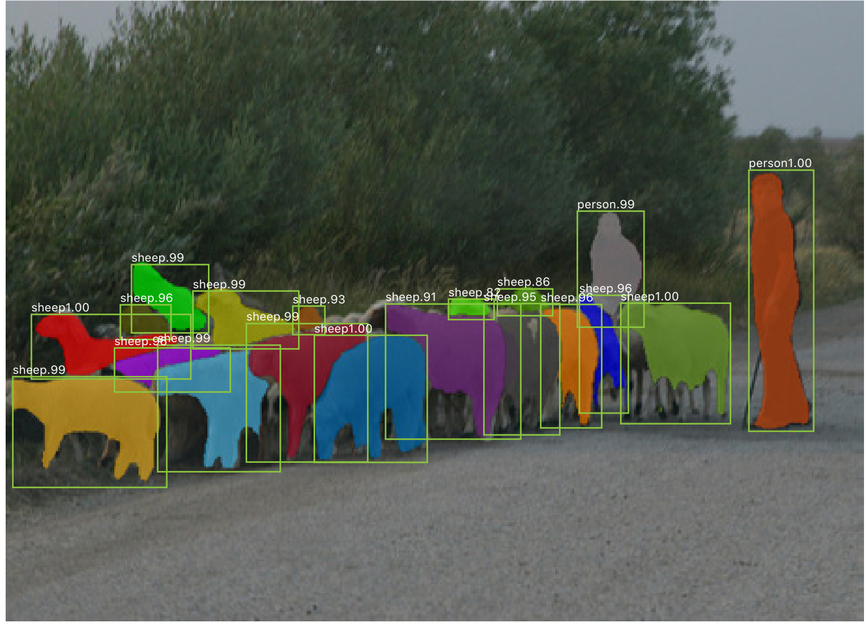


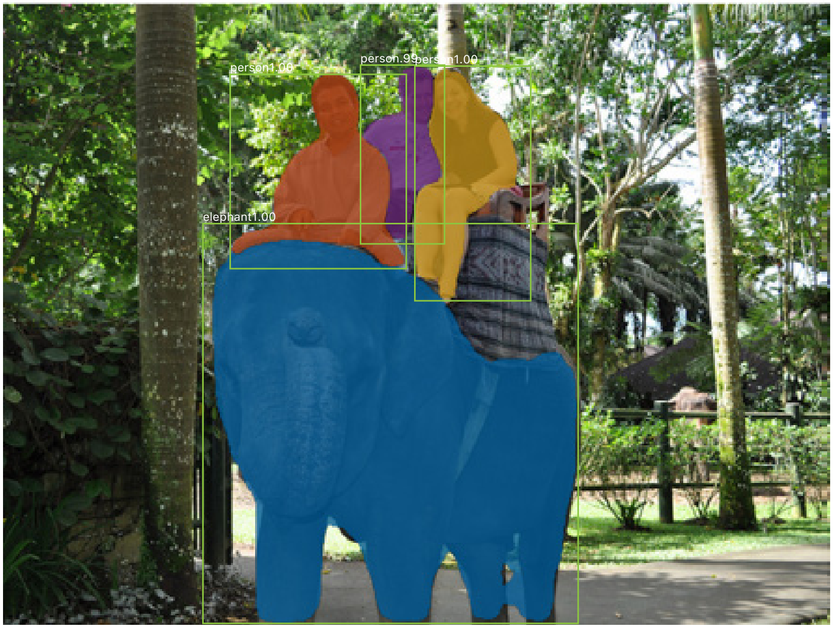
He at al. “Mask R-CNN” ICCV 2017

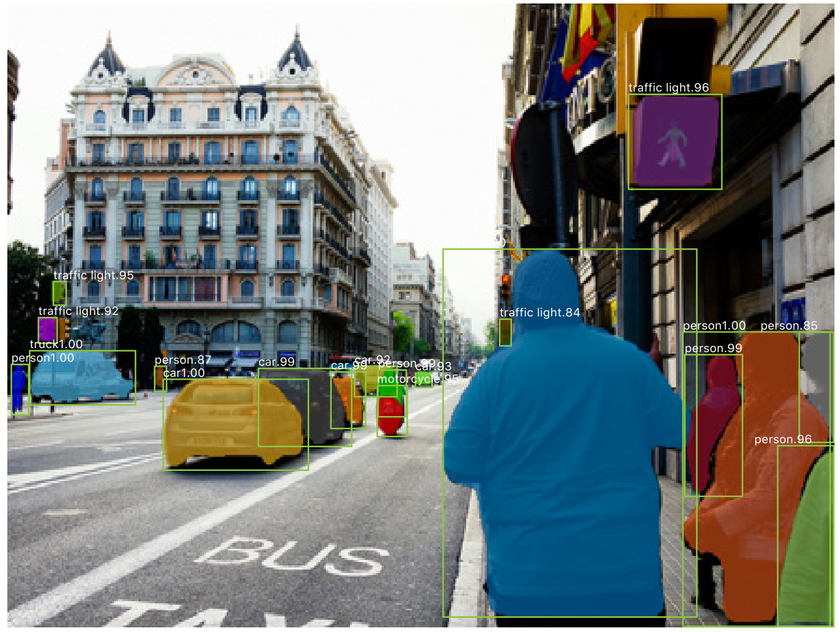
Mask R-CNN: Qualitative results

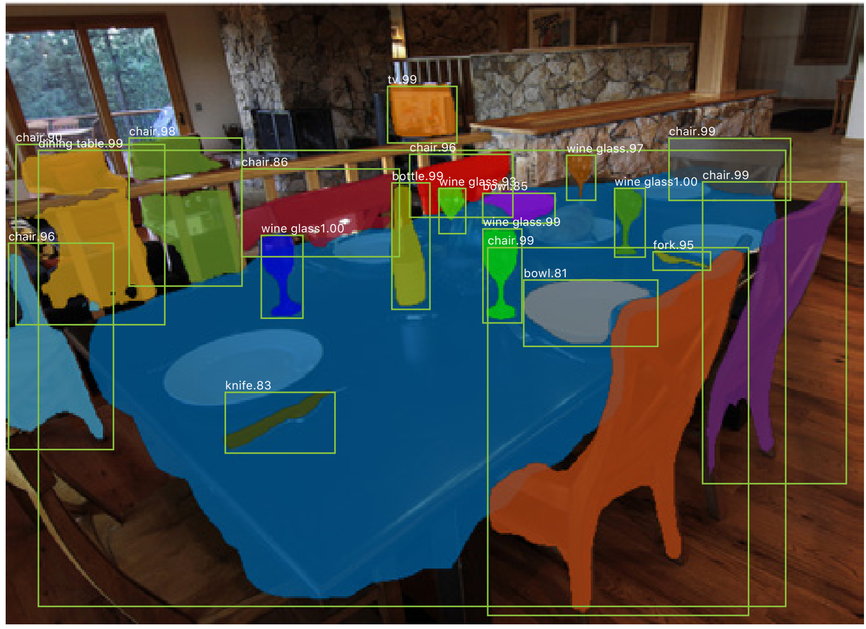












Mask R-CNN: Qualitative results

Mask R-CNN: Improvements

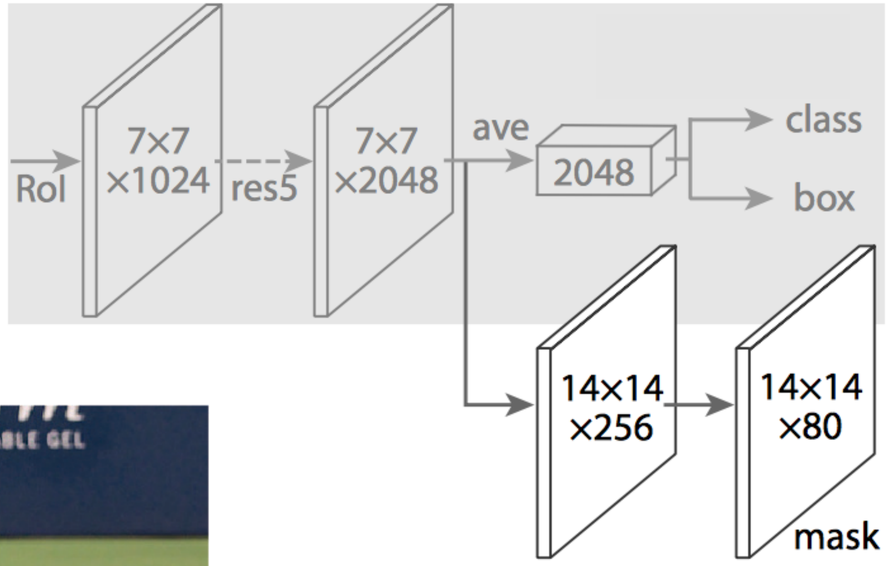
- Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020).
- Huang et al., “Mask Scoring R-CNN” (2019).
- Liu et al., “Path Aggregation Network for Instance Segmentation” (2018).
- Cai and Vasconcelos. “Cascade R-CNN: High Quality Object Detection and Instance Segmentation” (2019)

Mask R-CNN: Improvements

- **Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020).**
- Huang et al., “Mask Scoring R-CNN” (2019).
- Liu et al., “Path Aggregation Network for Instance Segmentation” (2018).
- Cai and Vasconcelos. “Cascade R-CNN: High Quality Object Detection and Instance Segmentation” (2019)

Mask R-CNN + PontRend

- Problem: low mask resolution

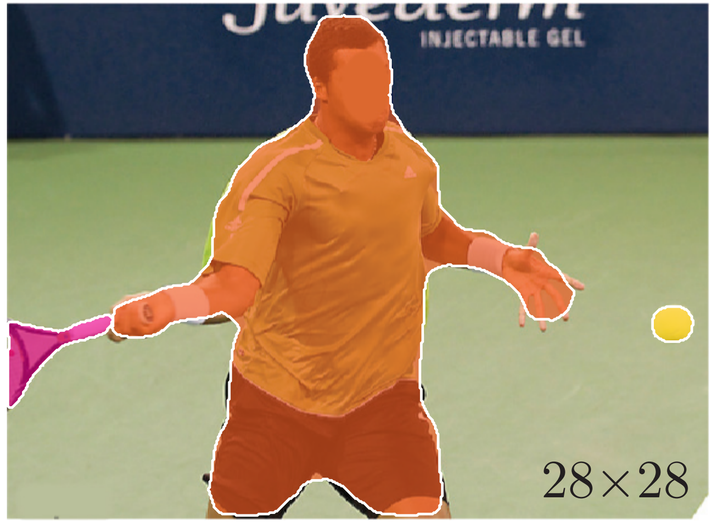


- Example:

28x28

Mask head prediction

Bilinear upsampling

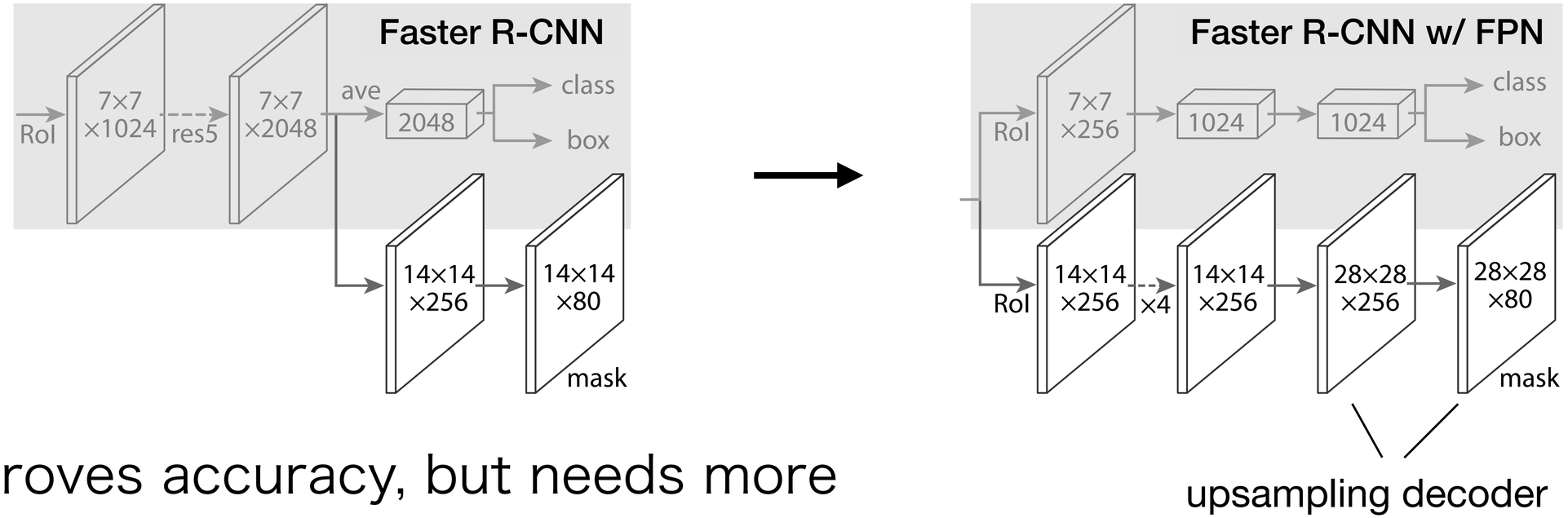


Upsampled mask

Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020)

Mask R-CNN + PontRend

- Why not equip mask head with a decoder?
  - Recall feature upsampling from previous lecture

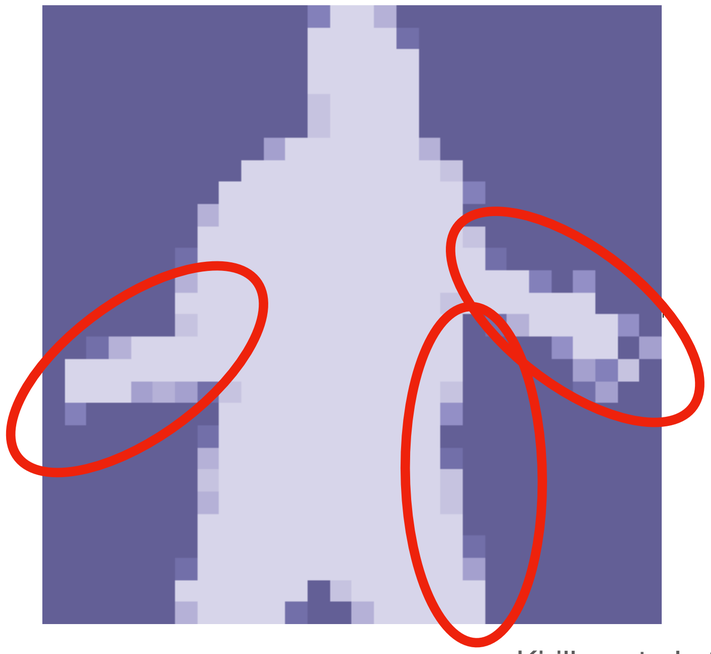


- Improves accuracy, but needs more parameters and computation

Mask R-CNN + PontRend

- Where is bilinear upsampling problematic?
  - Fine details mostly at the boundaries.

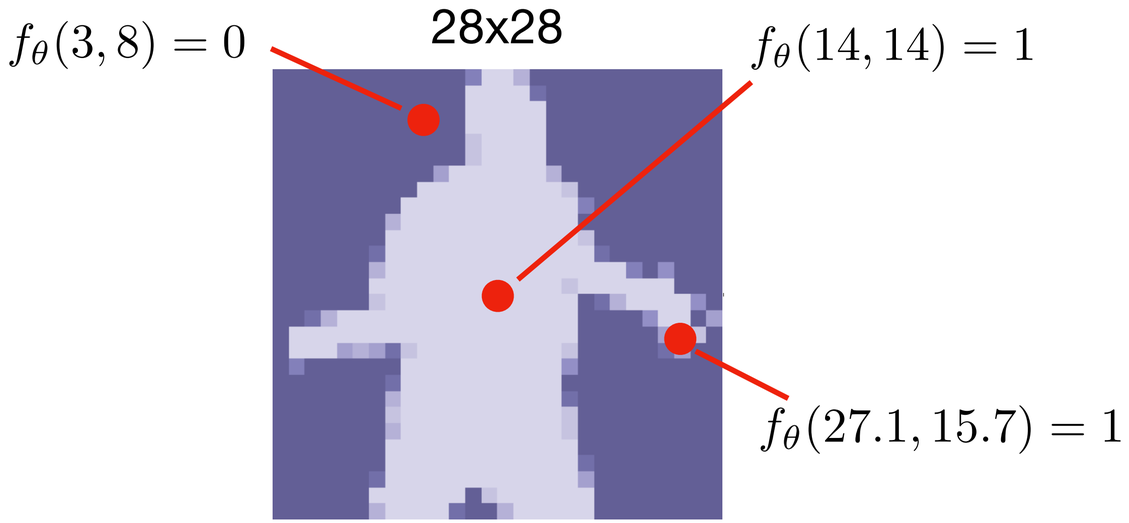
28x28



Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020)

Implicit neural representation

- Mask head is an example of mapping a discrete signal representation (e.g. a pixel) to a desired value (e.g. binary mask);
- Instead, we parameterise the mask as a continuous function that maps the signal domain (e.g. (x,y) coordinate):

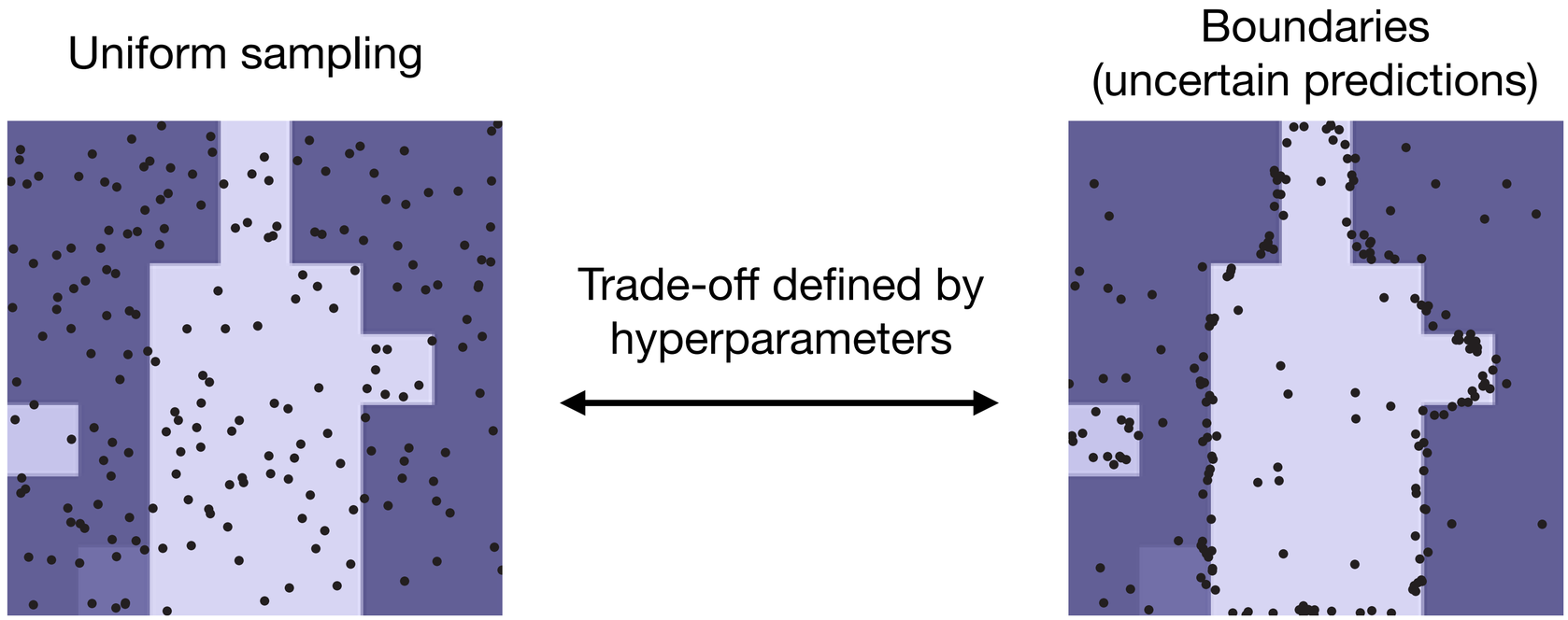


- $f_\theta(x, y)$ is an example of an implicit (neural) representation;
- Why is it useful here?
- We can query fractional coordinates.

Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020)

Implicit neural representation

- Idea:
  - Train implicit mask representation by focusing on the boundaries;



Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020)

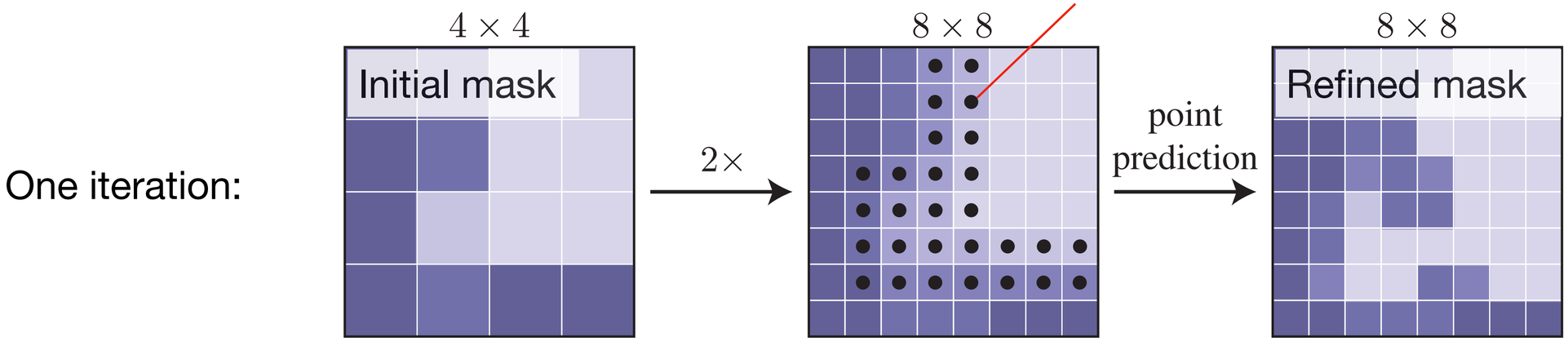
Implicit neural representation

- Idea:
  - Train implicit mask representation by focusing on the boundaries;
  - Test time: Use the learned implicit mapping to refine boundaries.

Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020)

Implicit neural representation

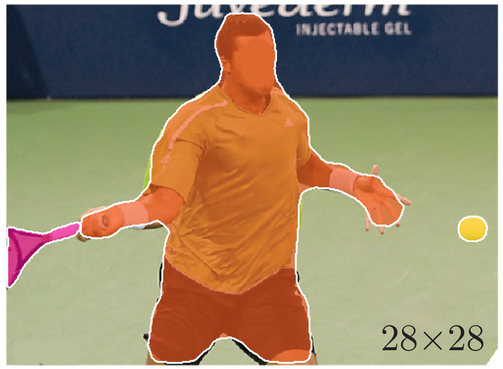
- Idea:
  - Train implicit mask representation by focusing on the boundaries;
  - Test time: Use the learned implicit mapping to refine boundaries.
- Adaptive subdivision step (test time):

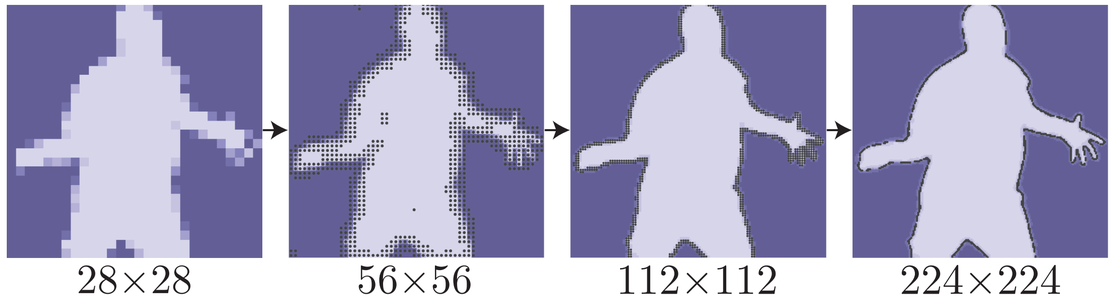


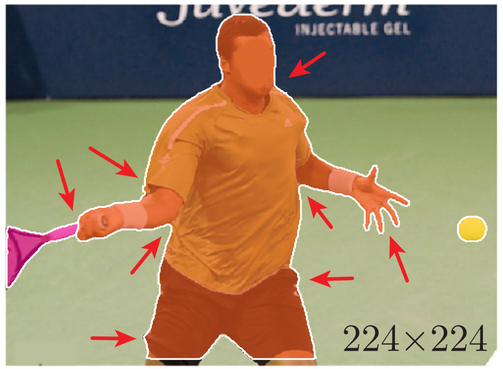
Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020)

Implicit neural representation

- Idea:
  - Train implicit mask representation by focusing on the boundaries;
  - Test time: Use the learned implicit mapping to refine boundaries.
- Adaptive subdivision step (test time):







Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020)

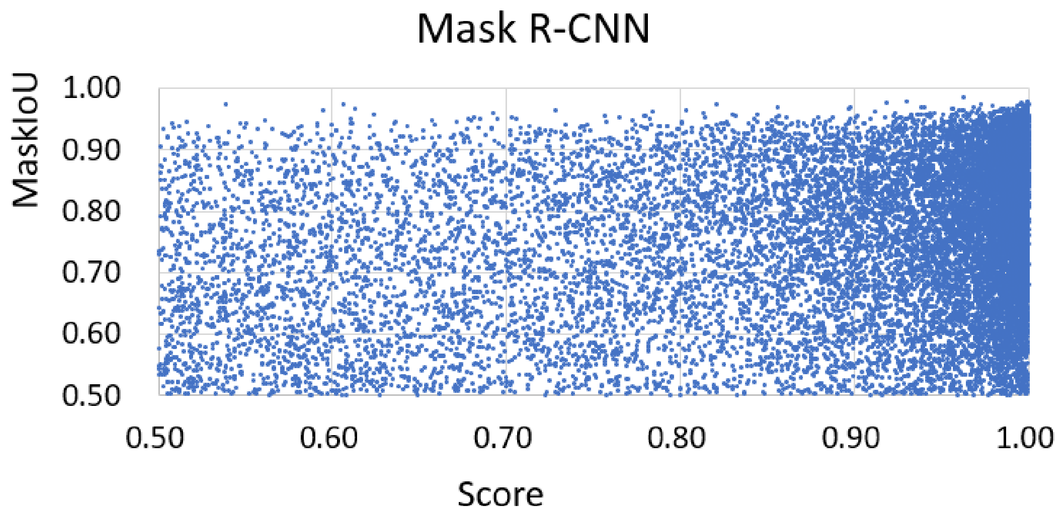
PointRend: Qualitative results

Mask R-CNN: Improvements

- Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020).
- **Huang et al., “Mask Scoring R-CNN” (2019).**
- Liu et al., “Path Aggregation Network for Instance Segmentation” (2018).
- Cai and Vasconcelos. “Cascade R-CNN: High Quality Object Detection and Instance Segmentation” (2019)

Mask Scoring R-CNN

- Recall how we obtain the confidence in Mask R-CNN?
- We still use the (box) classification head for that.
- It indicates only the bounding box quality.



Mask Scoring R-CNN

- Recall how we obtain the confidence in Mask R-CNN?
- We still use the (box) classification head for that.
- It indicates only the bounding box quality.

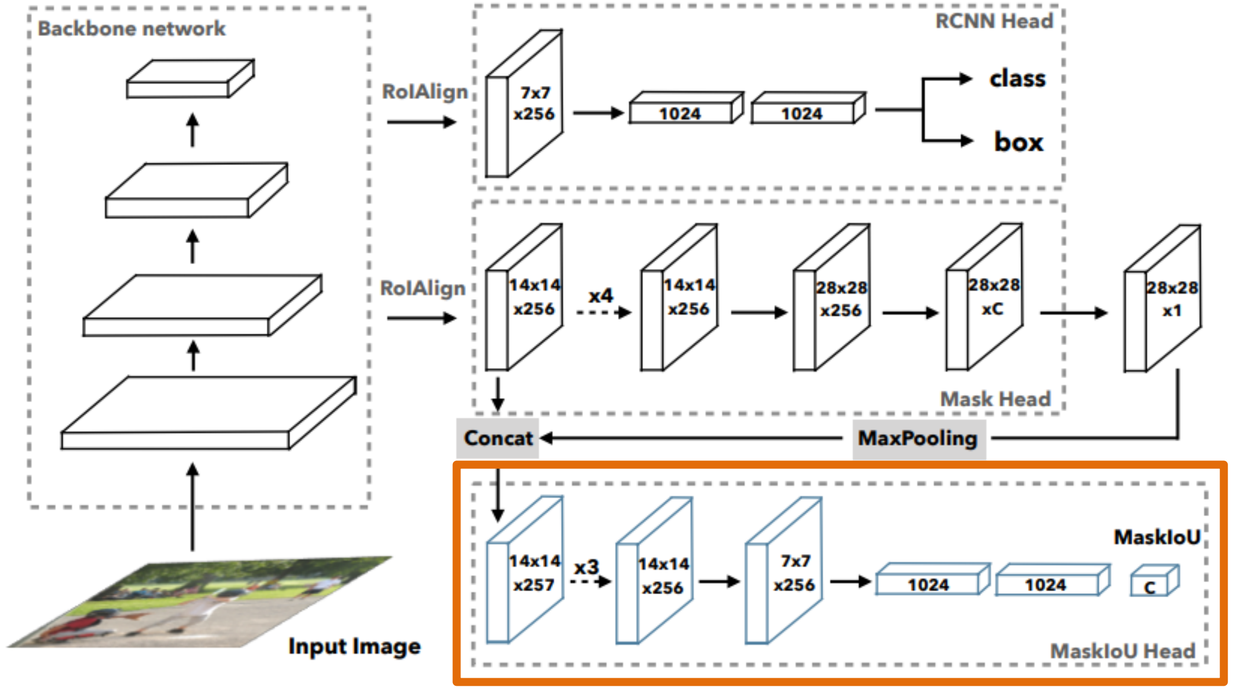
- We need a confidence score indicating **mask** quality.
- QUIZ: Why do we need it?

Mask Scoring R-CNN

Mask Score R-CNN idea: Learn to predict mask IoU w.r.t. ground truth

- Add a regression head:

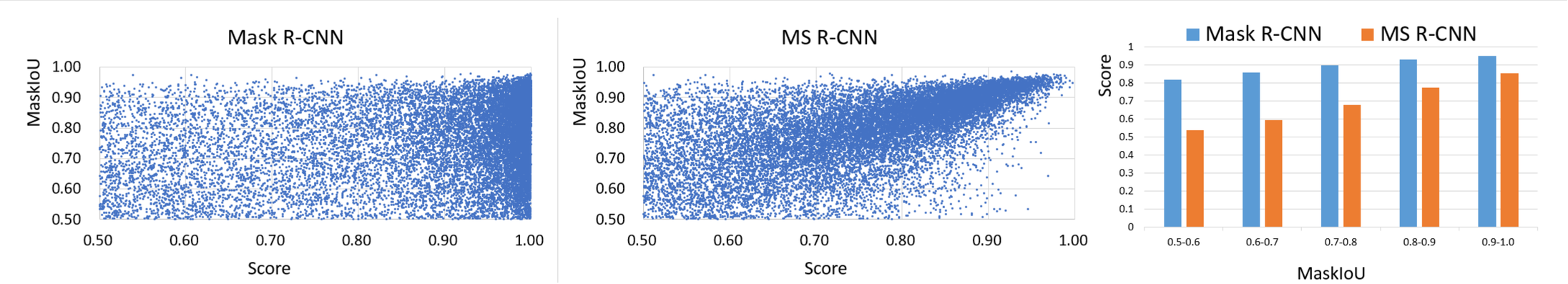
- Use RPN proposal for training.
- To compute IoU, binarise prediction with a threshold.
- Uses L2 loss.



Huang et al., “Mask Scoring R-CNN”, CVPR 2019

Mask Scoring R-CNN

Mask confidence scores become much more informative:



Huang et al., “Mask Scoring R-CNN”, CVPR 2019

Mask confidence score

- Mask confidence scores become much more informative:

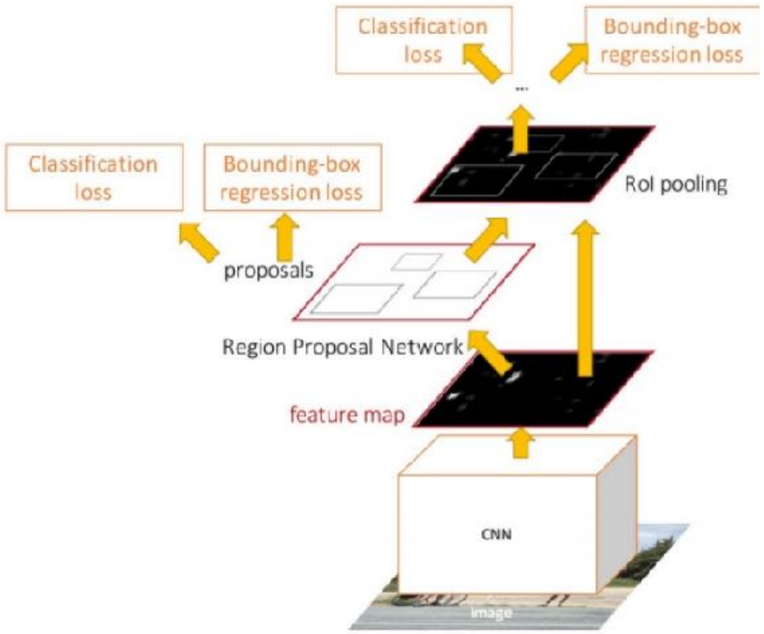
- Translates to higher mask accuracy by ~10% relative improvement.

Huang et al., “Mask Scoring R-CNN”, CVPR 2019

One-stage instance segmentation

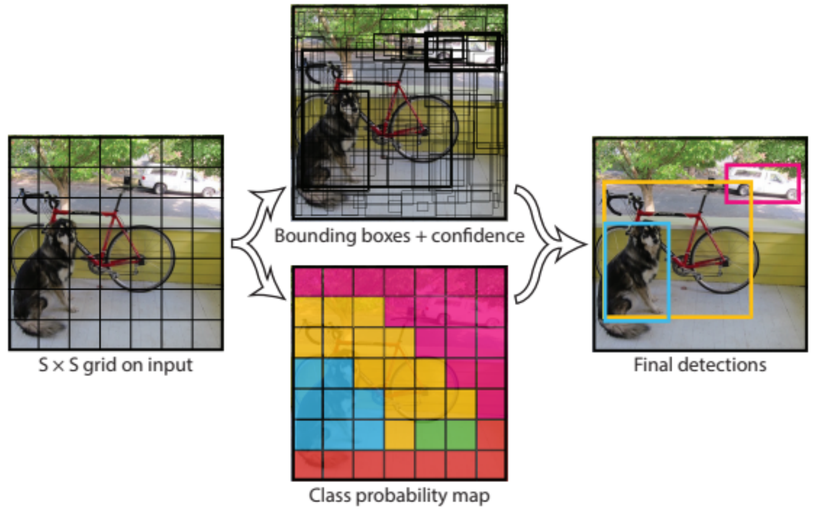
One-stage vs. two-stage detectors

Faster R-CNN



Slower, but more accurate

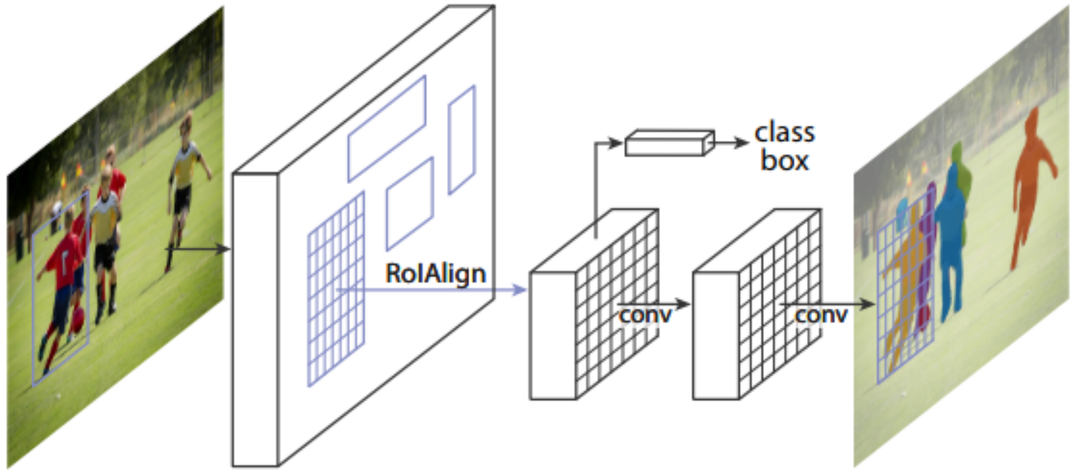
YOLO



Faster, but less accurate

One-stage instance segmentation?

Mask R-CNN



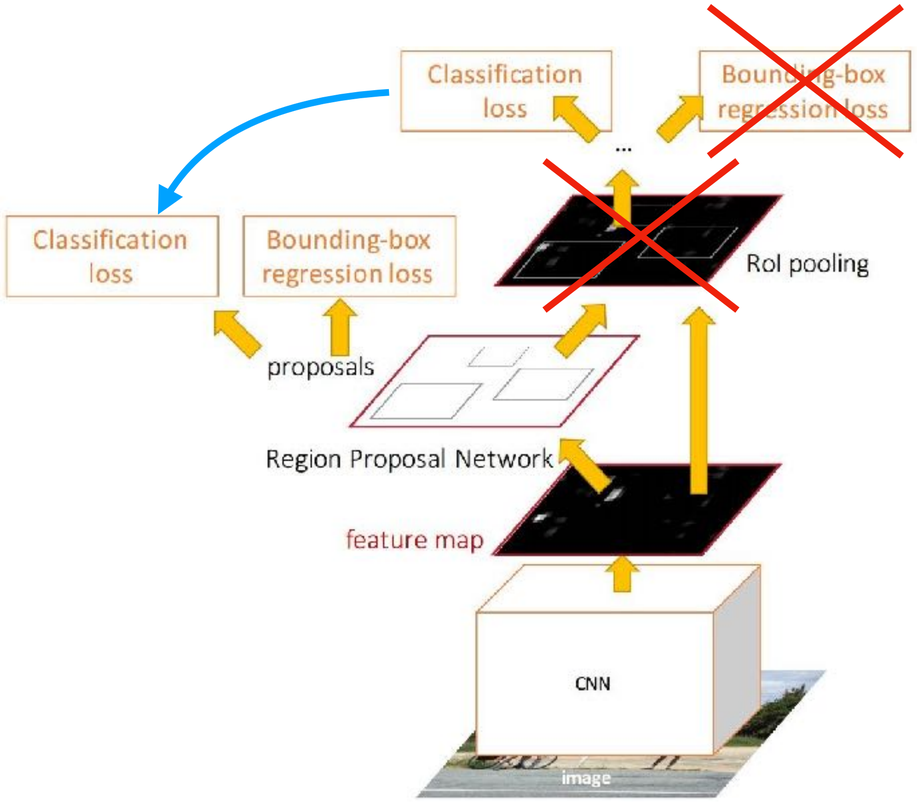
Slower, but more accurate

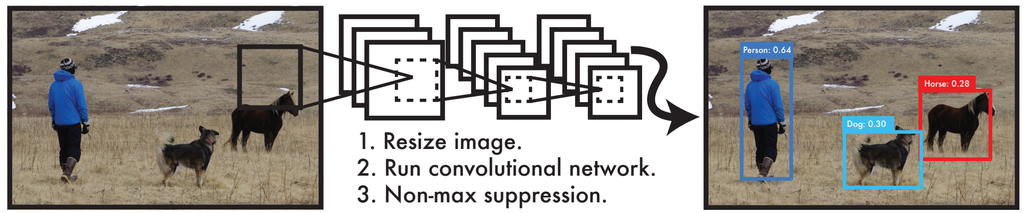
?

Faster, but less accurate

Recall YOLO

- Removing RoIPool from two-stage detectors:

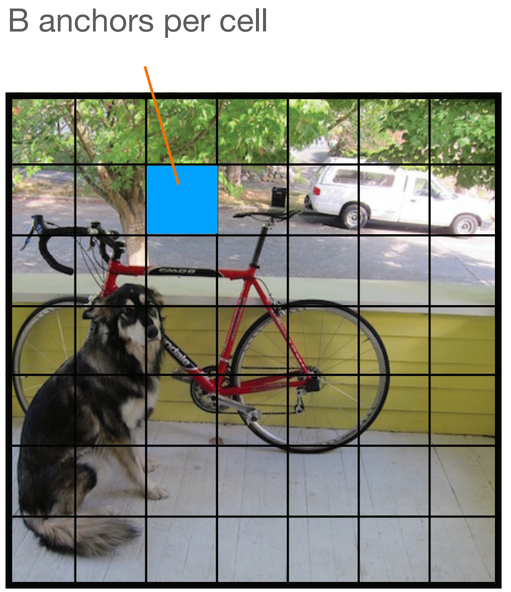




Redmon et al, “You only look once: Unified real-time object detection”, CVPR 2016.

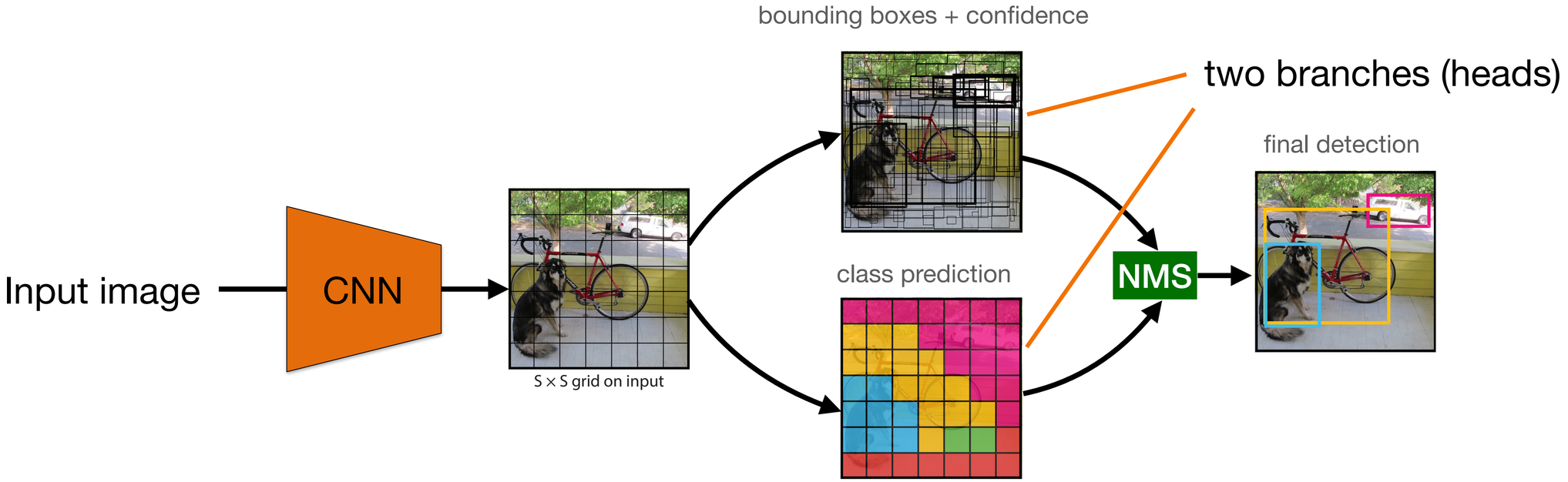
Recall YOLO

- Define a coarse grid ($S \times S$);
- Associate B anchors to each cell;
- Each anchor is defined by
  - localisation (x, y, w, h);
  - a confidence value (object / no object);
  - and a class distribution over C classes.



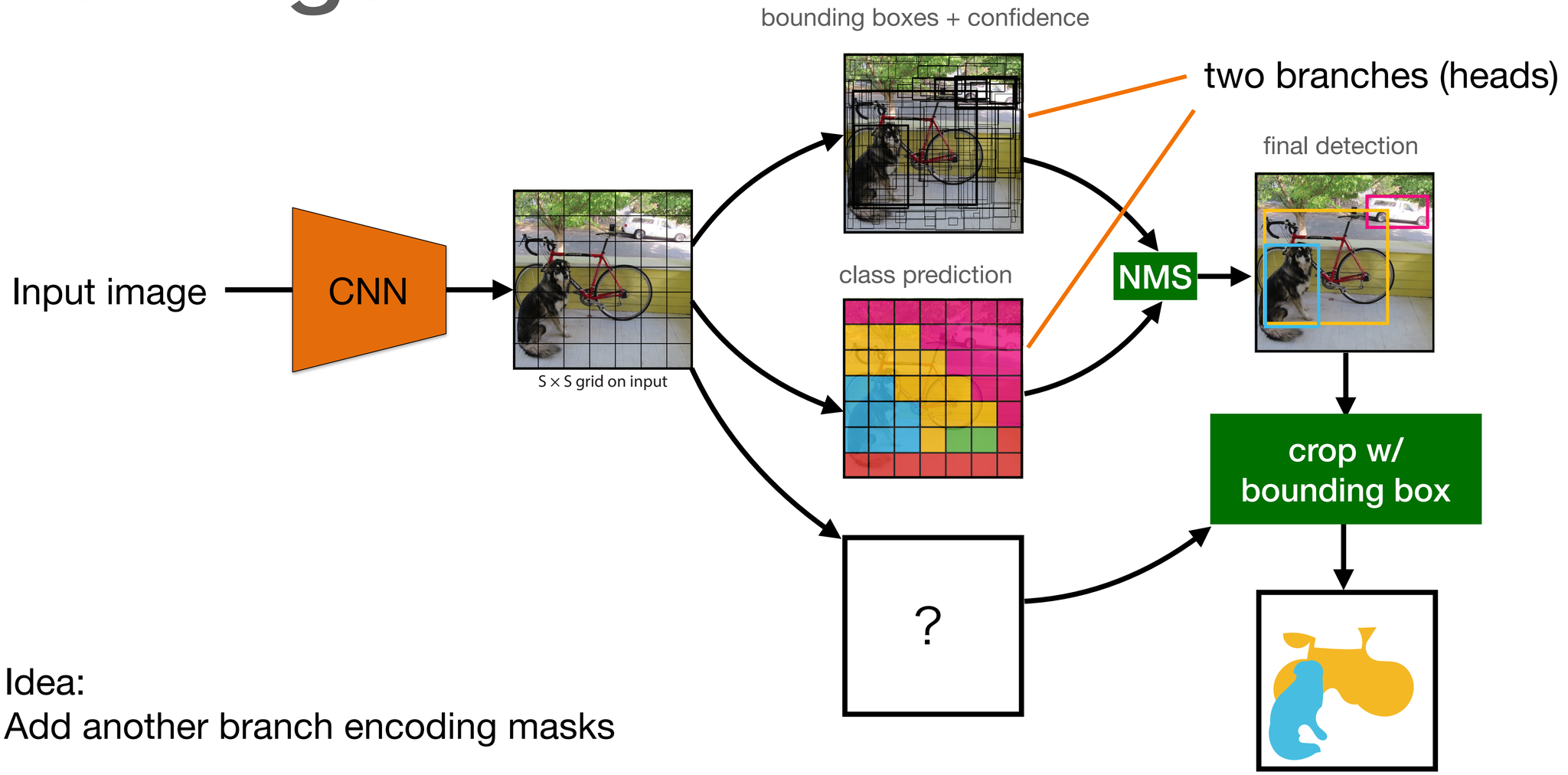
Redmon et al, “You only look once: Unified real-time object detection”, CVPR 2016.

Recall YOLO



Redmon et al, “You only look once: Unified real-time object detection”, CVPR 2016.

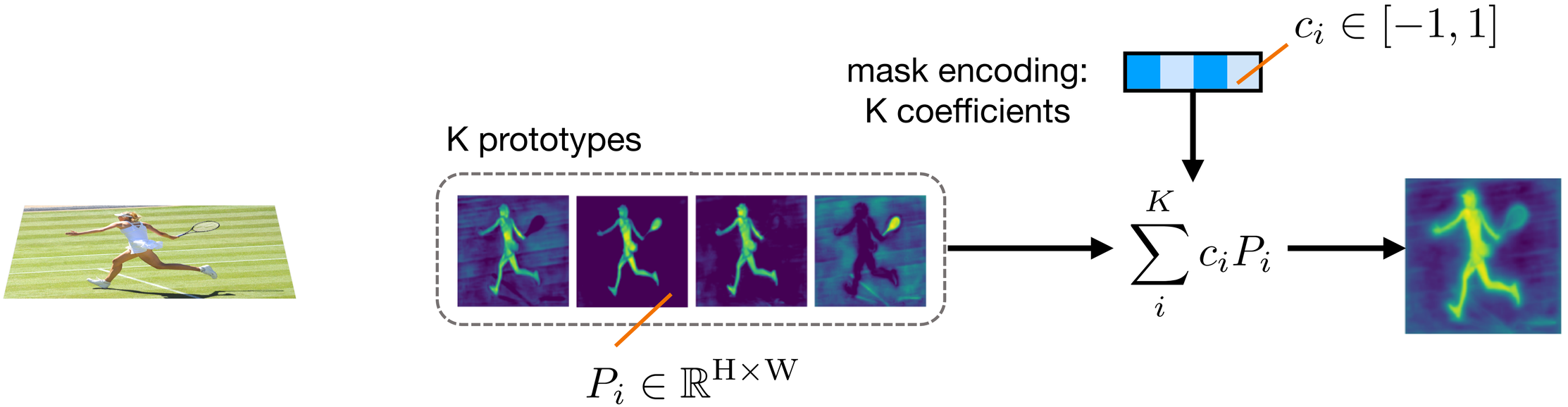
One-stage IS



Idea:
Add another branch encoding masks

YOLACT

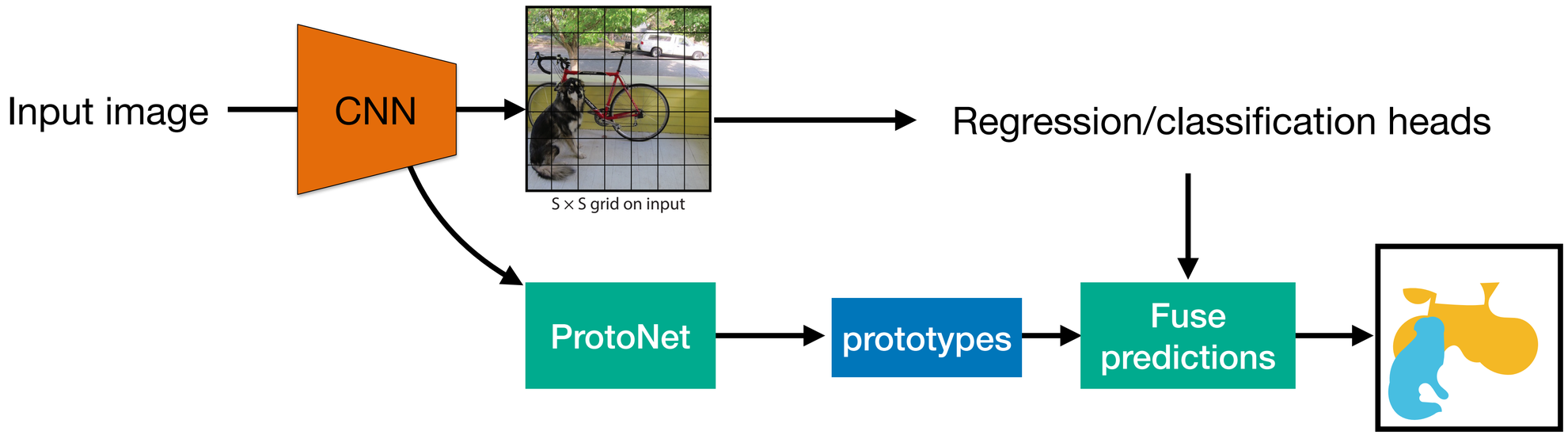
- How to encode masks?
  - YOLACT: A linear combination of “prototypes”:



D. Bolya et al. “YOLACT: Real-time Instance Segmentation”. ICCV 2019

YOLACT

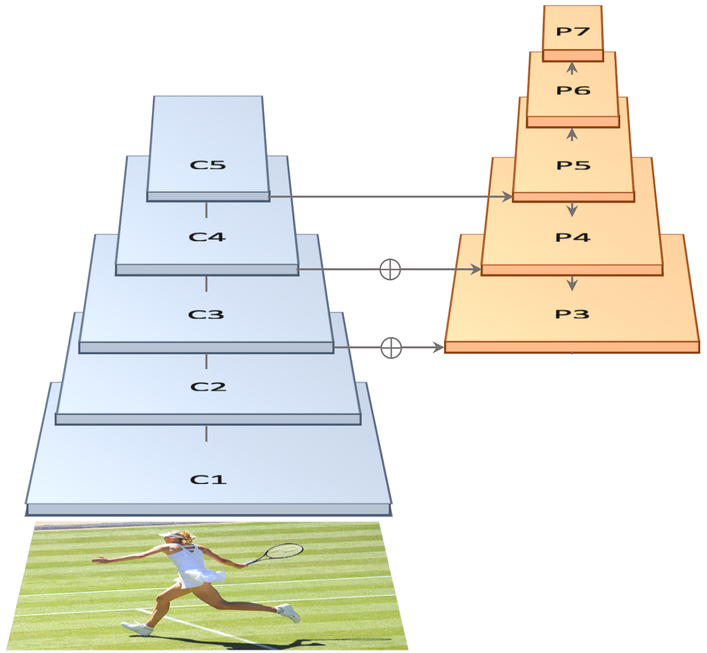
- Where to obtain the prototypes?
- Learn concurrently with a fully convolutional network!



D. Bolya et al. “YOLACT: Real-time Instance Segmentation”. ICCV 2019

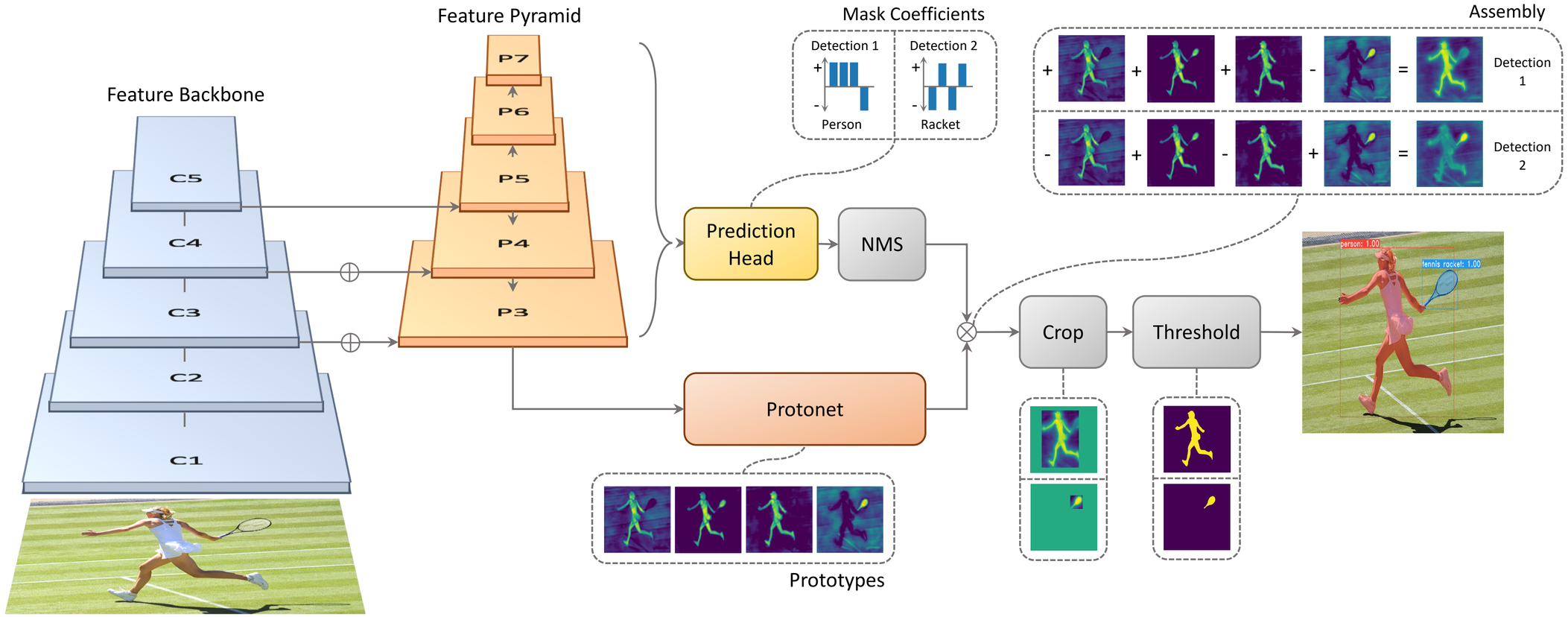
YOLACT

- Recall that one-stage frameworks are less robust to scale variation.
- We can equip the backbone with a FPN:



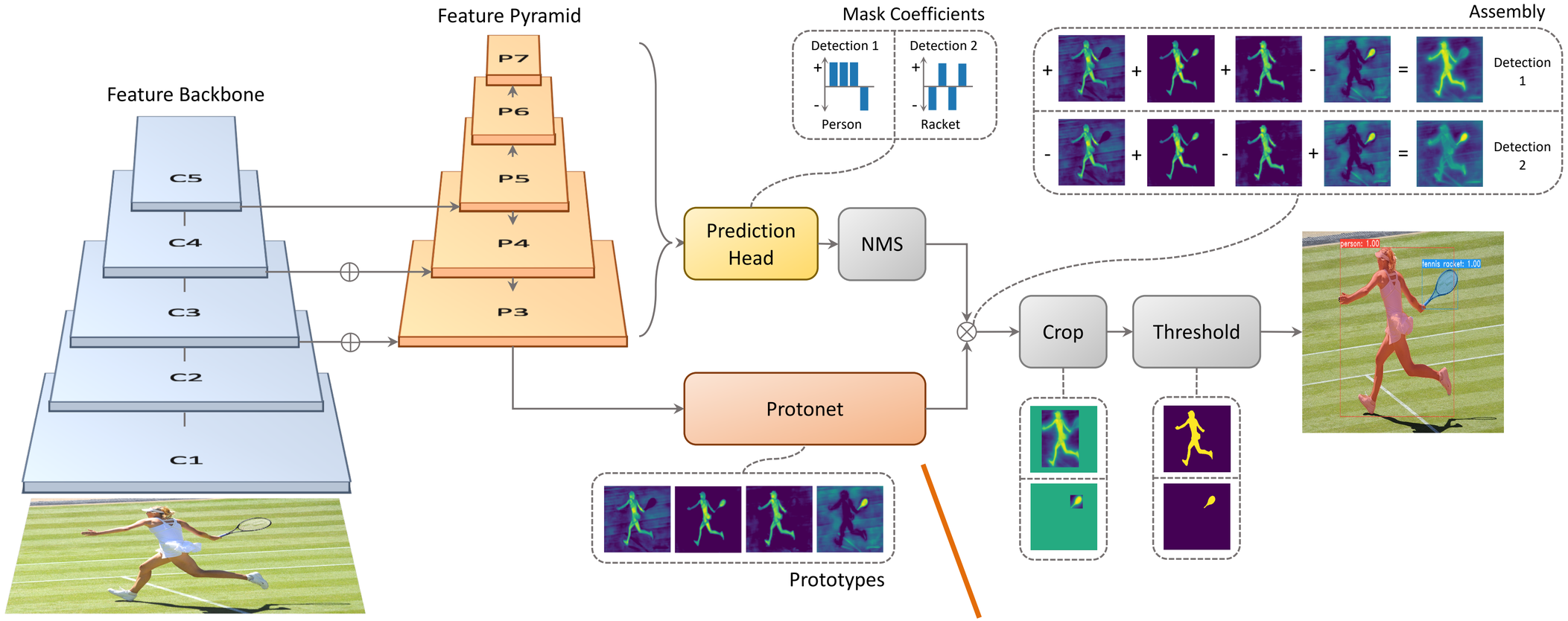
D. Bolya et al. “YOLACT: Real-time Instance Segmentation”. ICCV 2019

YOLACT: Overview



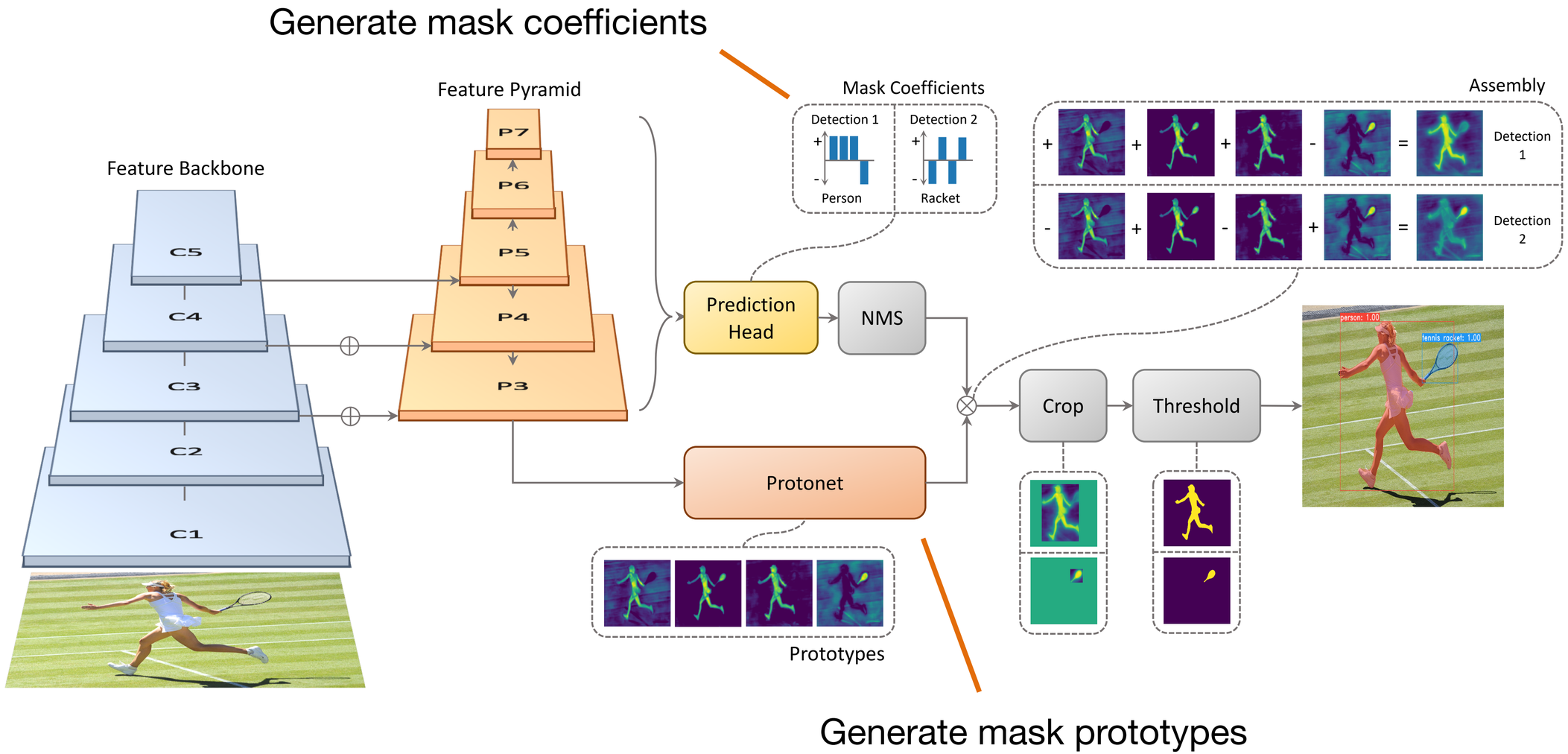
D. Bolya et al. “YOLACT: Real-time Instance Segmentation”. ICCV 2019

YOLACT: Overview



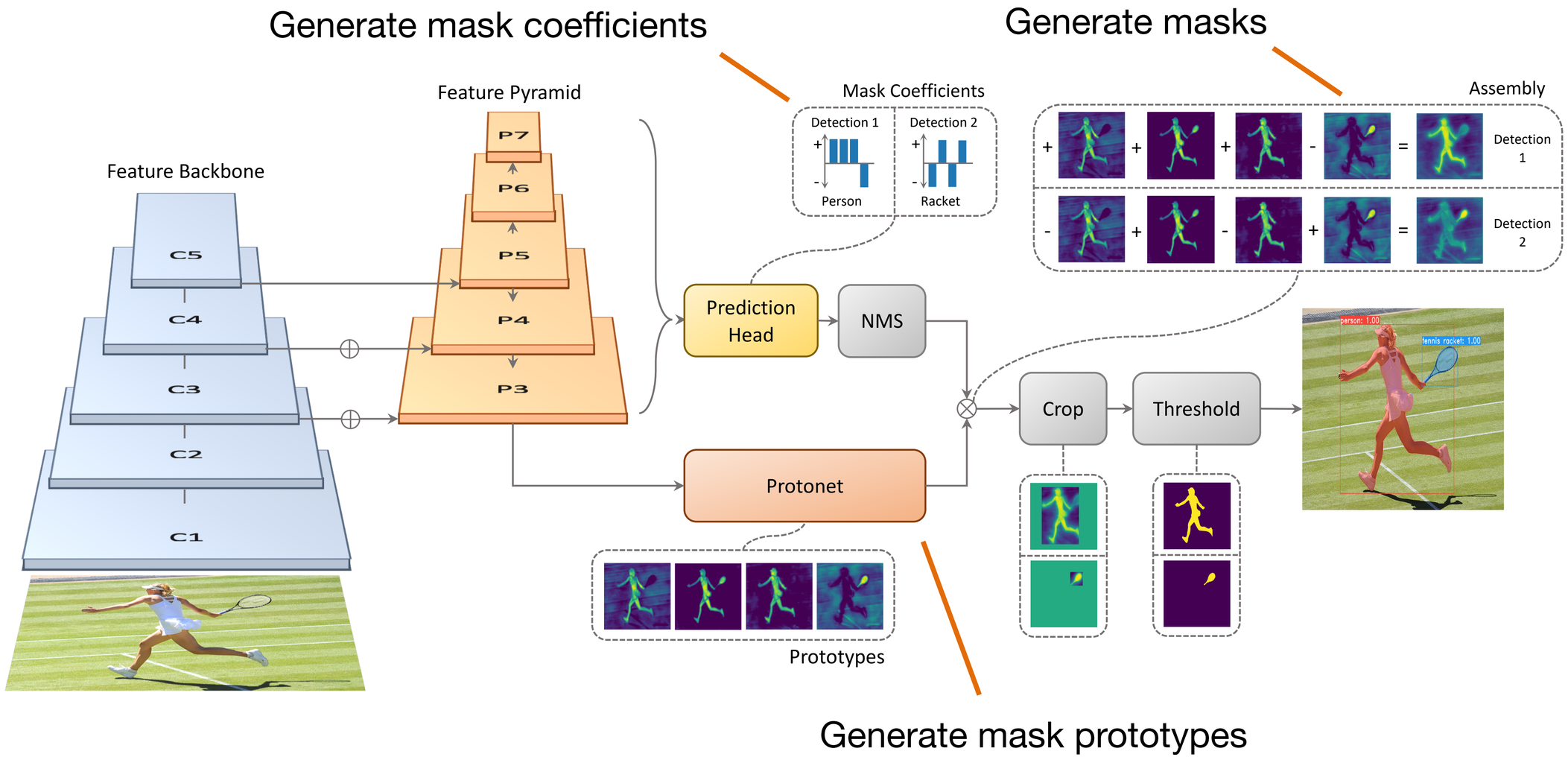
D. Bolya et al. “YOLACT: Real-time Instance Segmentation”. ICCV 2019

YOLACT: Overview



D. Bolya et al. “YOLACT: Real-time Instance Segmentation”. ICCV 2019

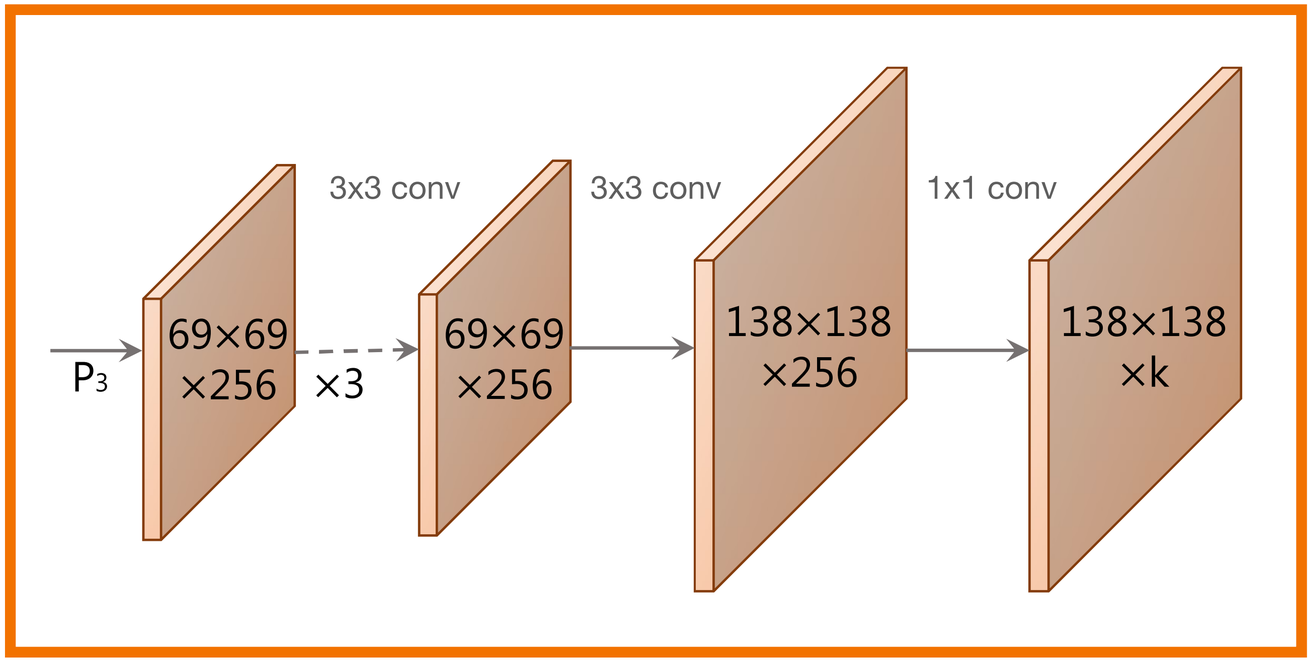
YOLACT: Overview

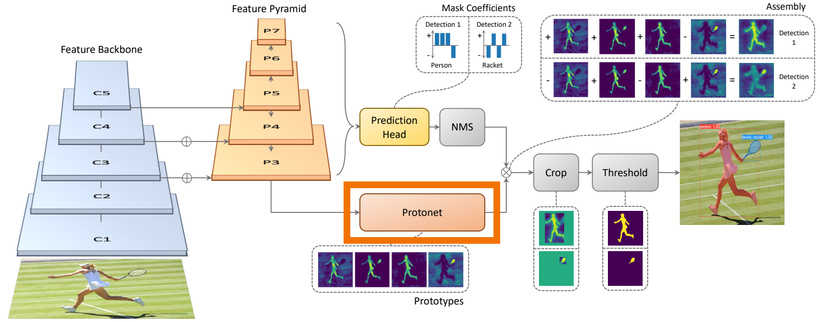


D. Bolya et al. “YOLACT: Real-time Instance Segmentation”. ICCV 2019

YOLACT: Protonet

Fully convolutional network:



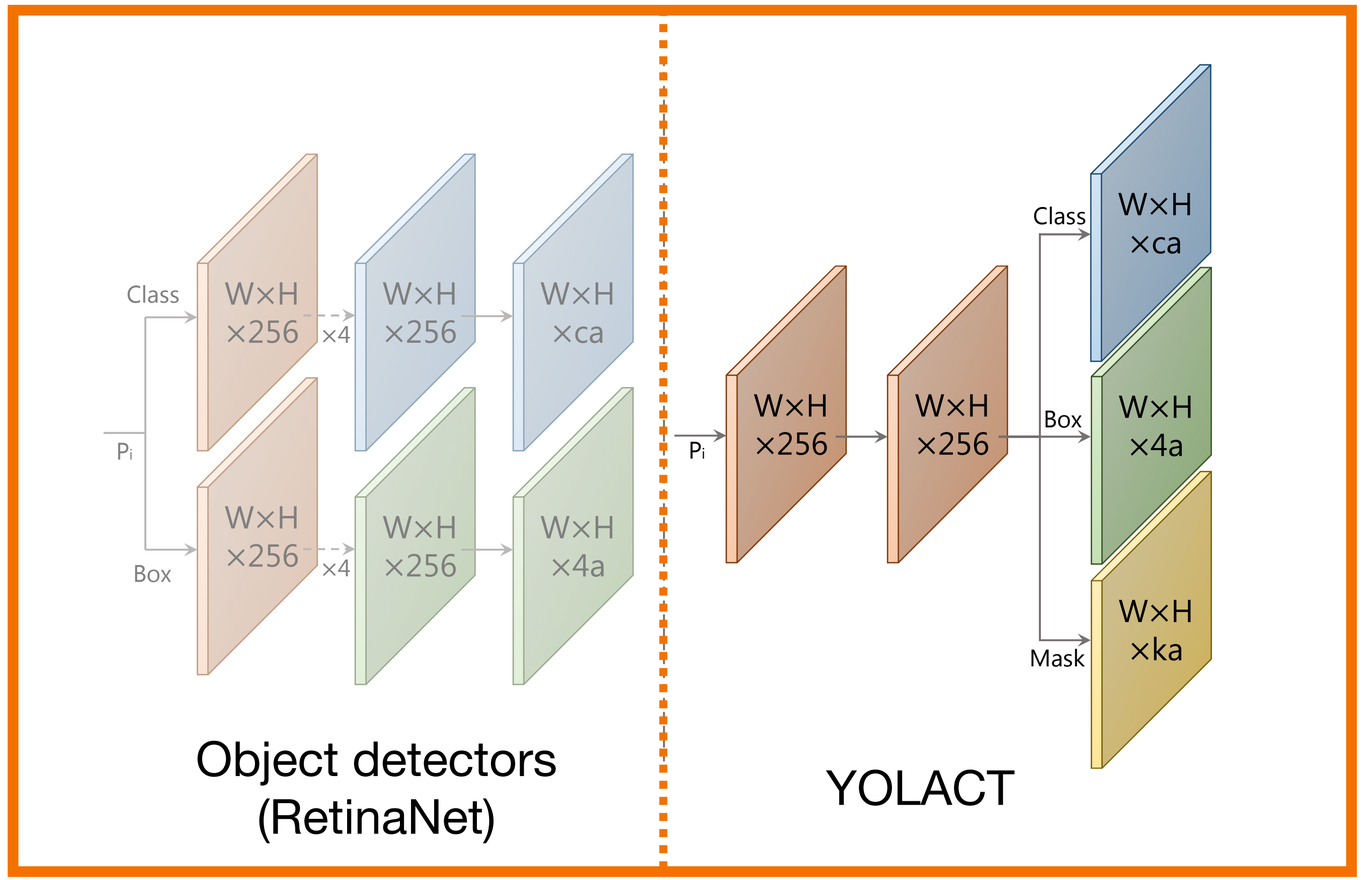


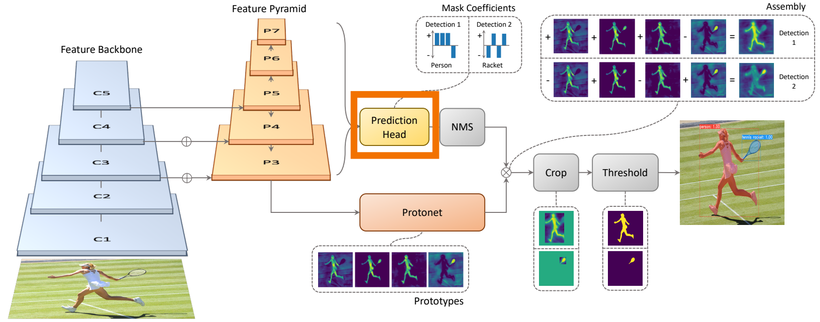
No loss function is applied on the prototypes.

The prototypes “emerge” from mask supervision in the last layer of the whole pipeline.

YOLACT: Mask coefficients

Fully convolutional network:





Predict one class per anchor box

Predict the regression per anchor box

Predict k coefficients (one per prototype mask) per anchor

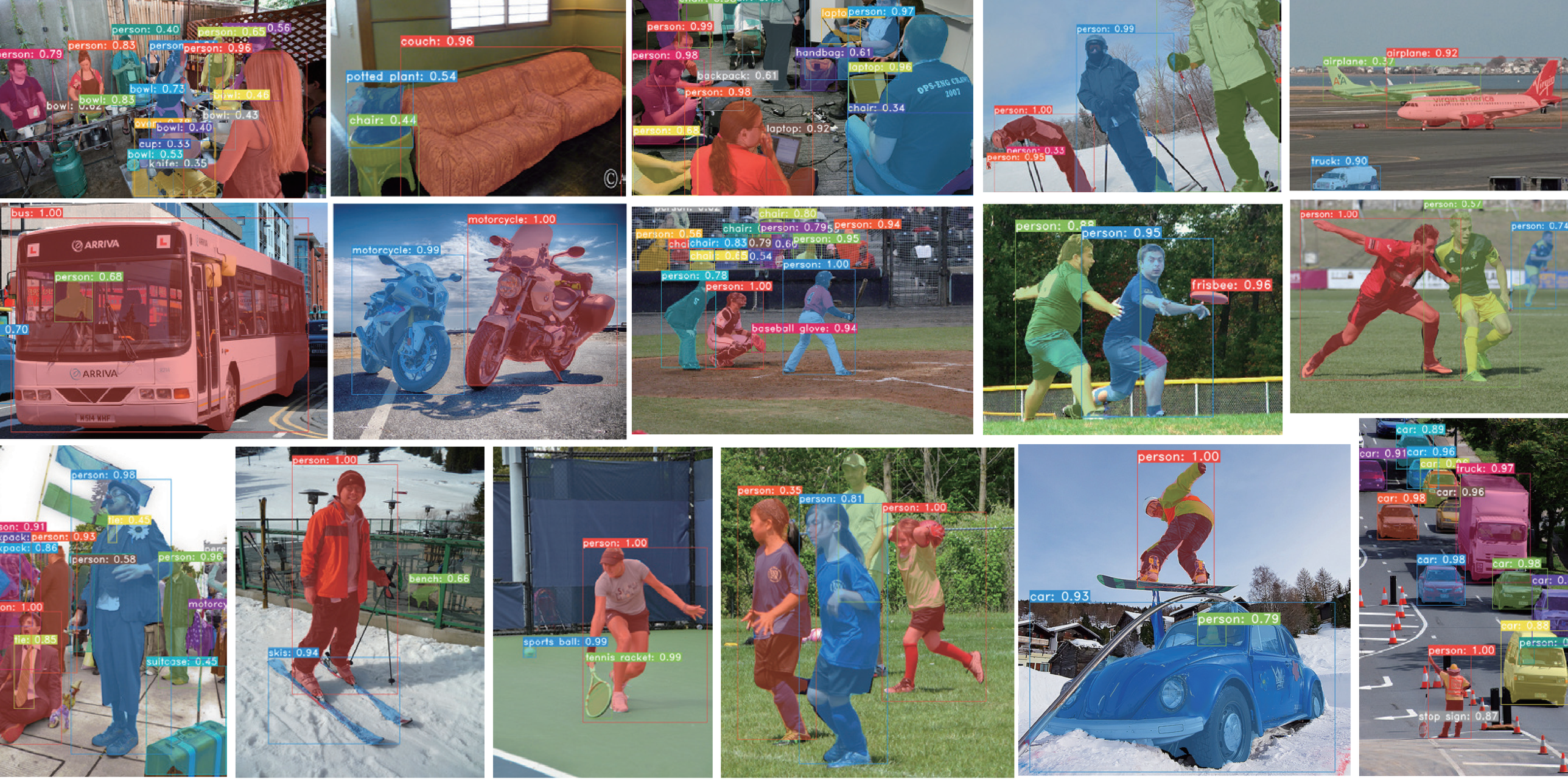
YOLACT: Loss functions

Standard detection losses:

- bounding box regression loss
- bounding box classification loss

Mask loss:

- binary cross-entropy (per pixel)

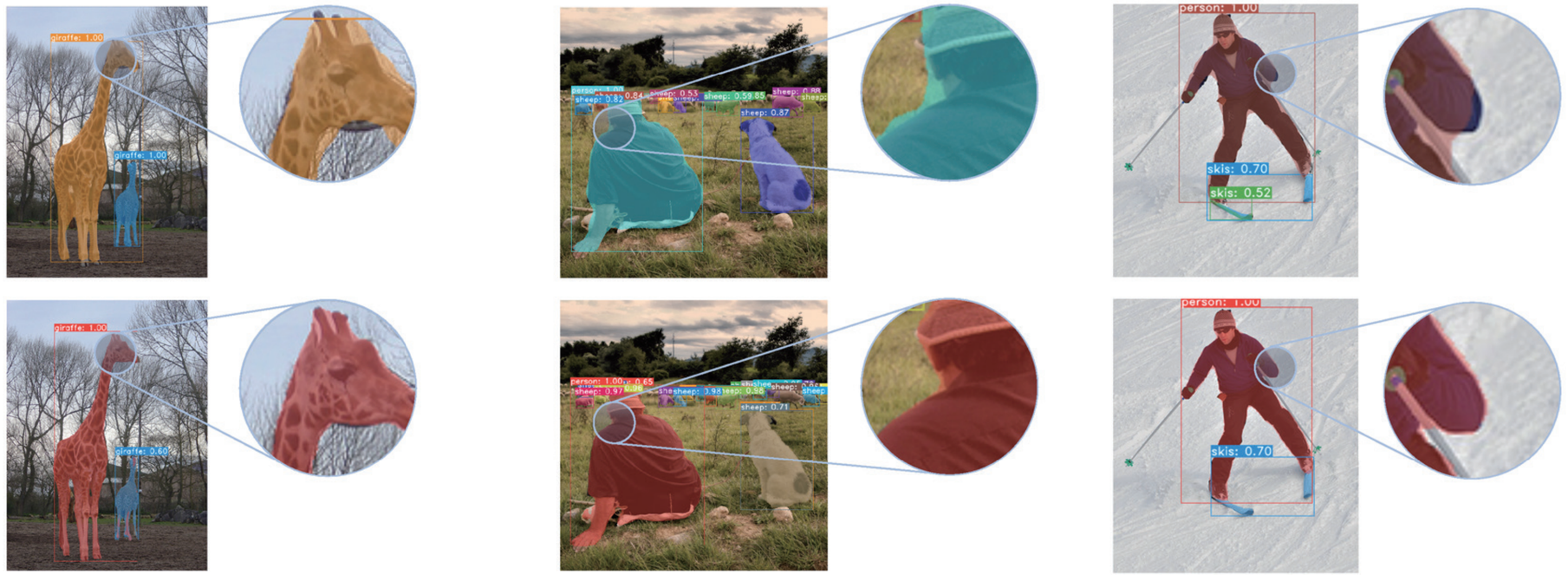


YOLACT: Qualitative results

YOLACT: Qualitative results

Mask R-CNN

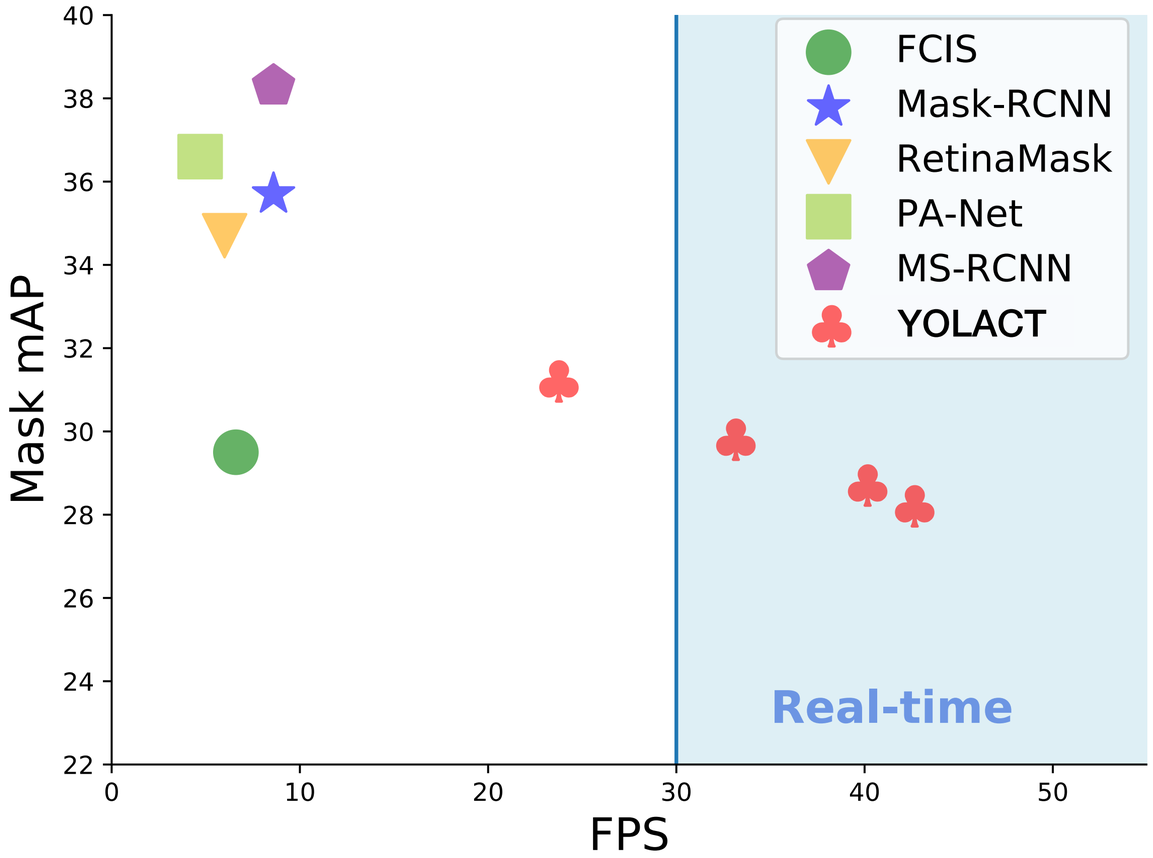
Ours



For large objects, the quality of the masks is even better than those of two-stage detectors (Quiz: Why?)

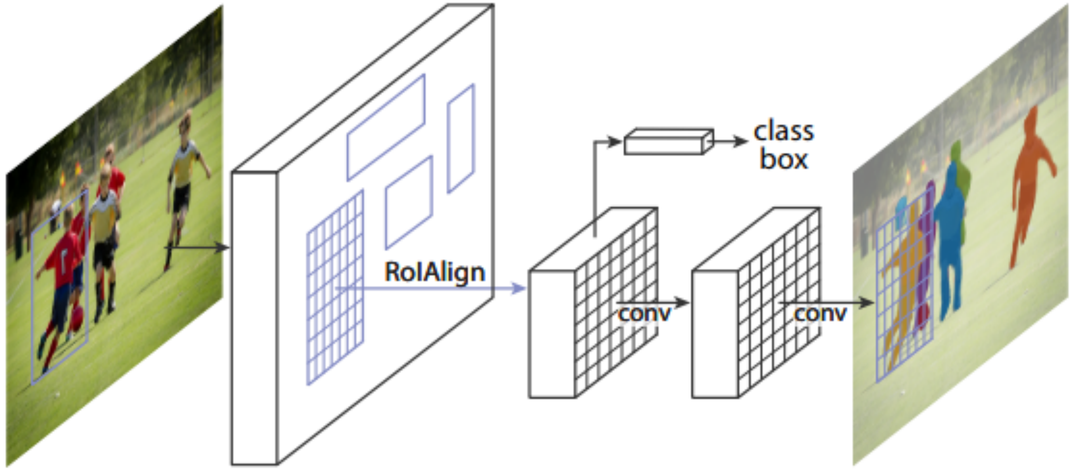
Real-time efficiency

Main advantage: very fast



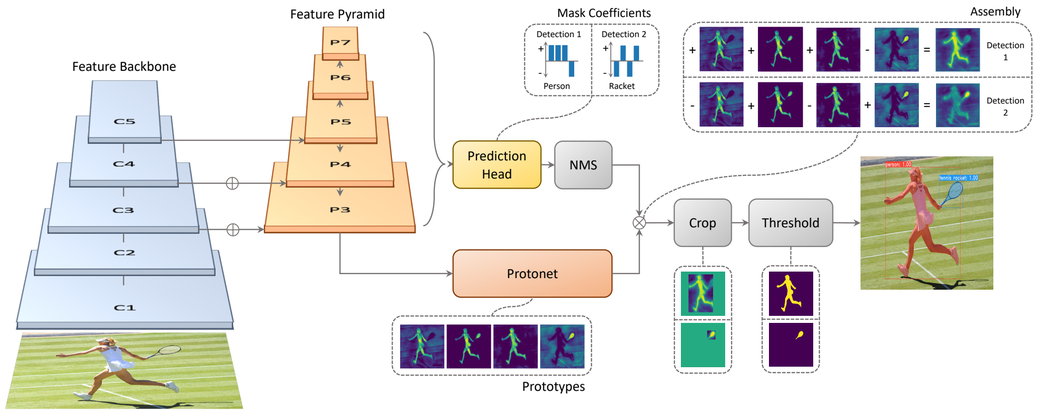
One-stage vs two-stage instance segmenters

Mask R-CNN



Slower, but more accurate

YOLACT



Faster, but less accurate

One-stage IS

- Boyla et al. “YOLACT++: Better real-time instance segmentation” (2019).
- Chen et al. “TensorMask: A Foundation for Dense Object Segmentation” (2019).
- Chen et al. “BlendMask: Top-Down Meets Bottom-Up for Instance Segmentation” (2020).
- Lee et al. “CenterMask : Real-Time Anchor-Free Instance Segmentation” (2020).
- Xie et al. “PolarMask: Single Shot Instance Segmentation with Polar Representation” (2020).
- Wang et al. “SOLO: Segmenting Objects by Locations” (2020).
- Wang et al. “SOLOv2: Dynamic and Fast Instance Segmentation” (2020).

One-stage IS

- Boyla et al. “YOLACT++: Better real-time instance segmentation” (2019).
- Chen et al. “TensorMask: A Foundation for Dense Object Segmentation” (2019).
- Chen et al. “BlendMask: Top-Down Meets Bottom-Up for Instance Segmentation” (2020).
- Lee et al. “CenterMask : Real-Time Anchor-Free Instance Segmentation” (2020).
- Xie et al. “PolarMask: Single Shot Instance Segmentation with Polar Representation” (2020).
- Wang et al. “SOLO: Segmenting Objects by Locations” (2020).
- **Wang et al. “SOLOv2: Dynamic and Fast Instance Segmentation” (2020).**

SOLOv2

- Recall semantic segmentation.
- The last layer is a 1x1 convolution – a linear classifier:

$$Y = KX$$

Pixelwise class scores
$[C \times HW]$

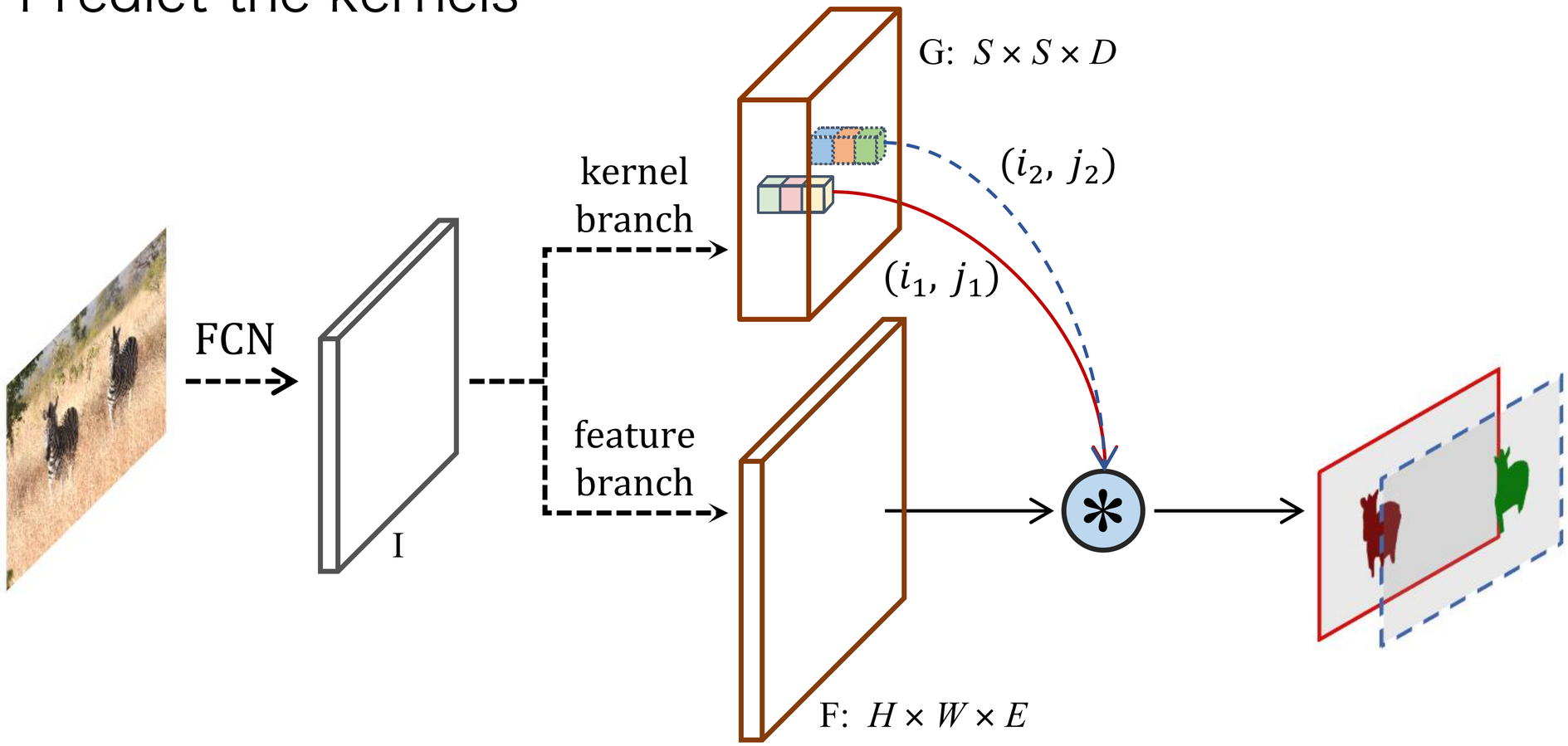
Layer parameters
(1x1 conv)
$[C \times D]$

Features
$[D \times HW]$

- Why not apply the same strategy to instance segmentation?
- Problem: The number of kernel cannot be fixed.

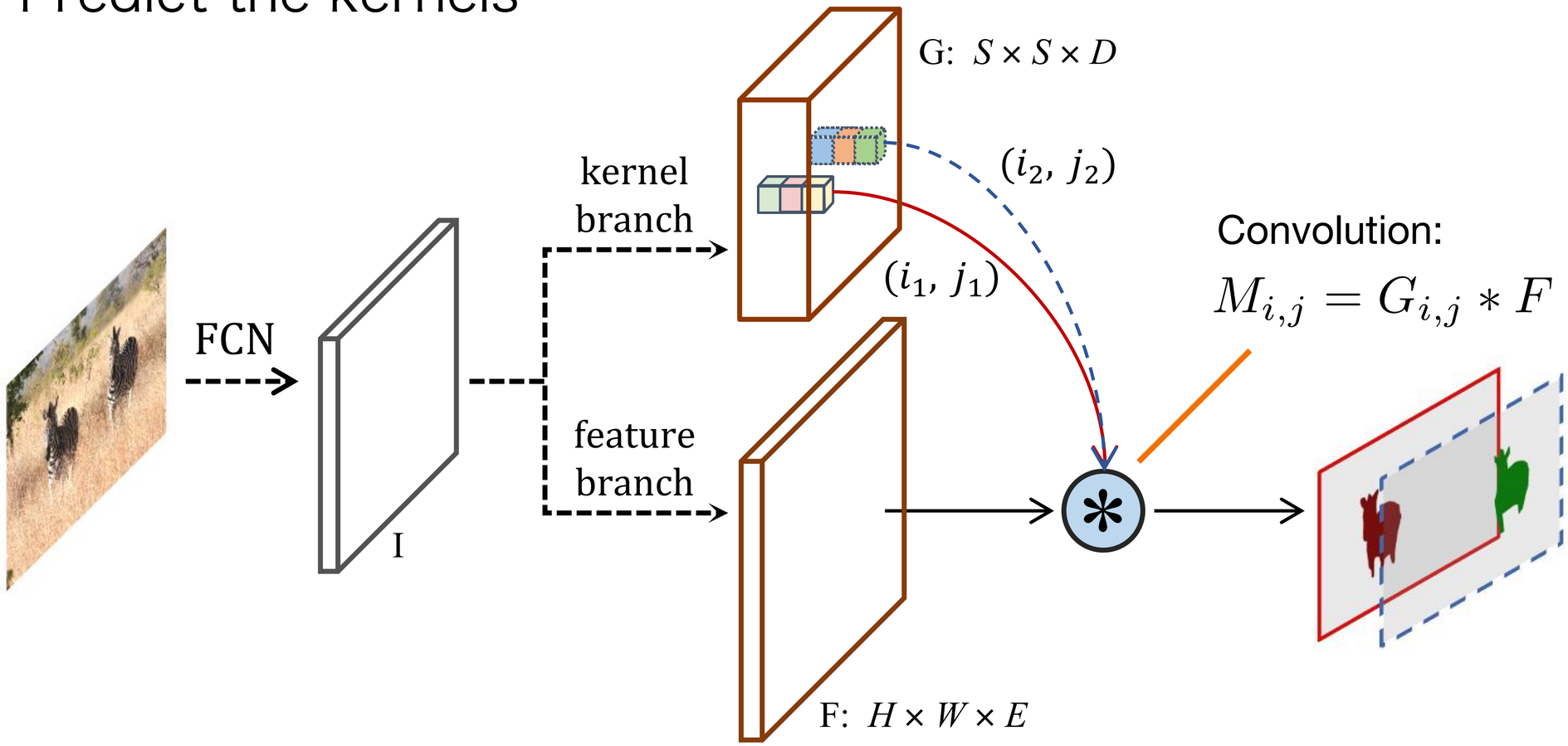
SOLOv2

- Idea: Predict the kernels



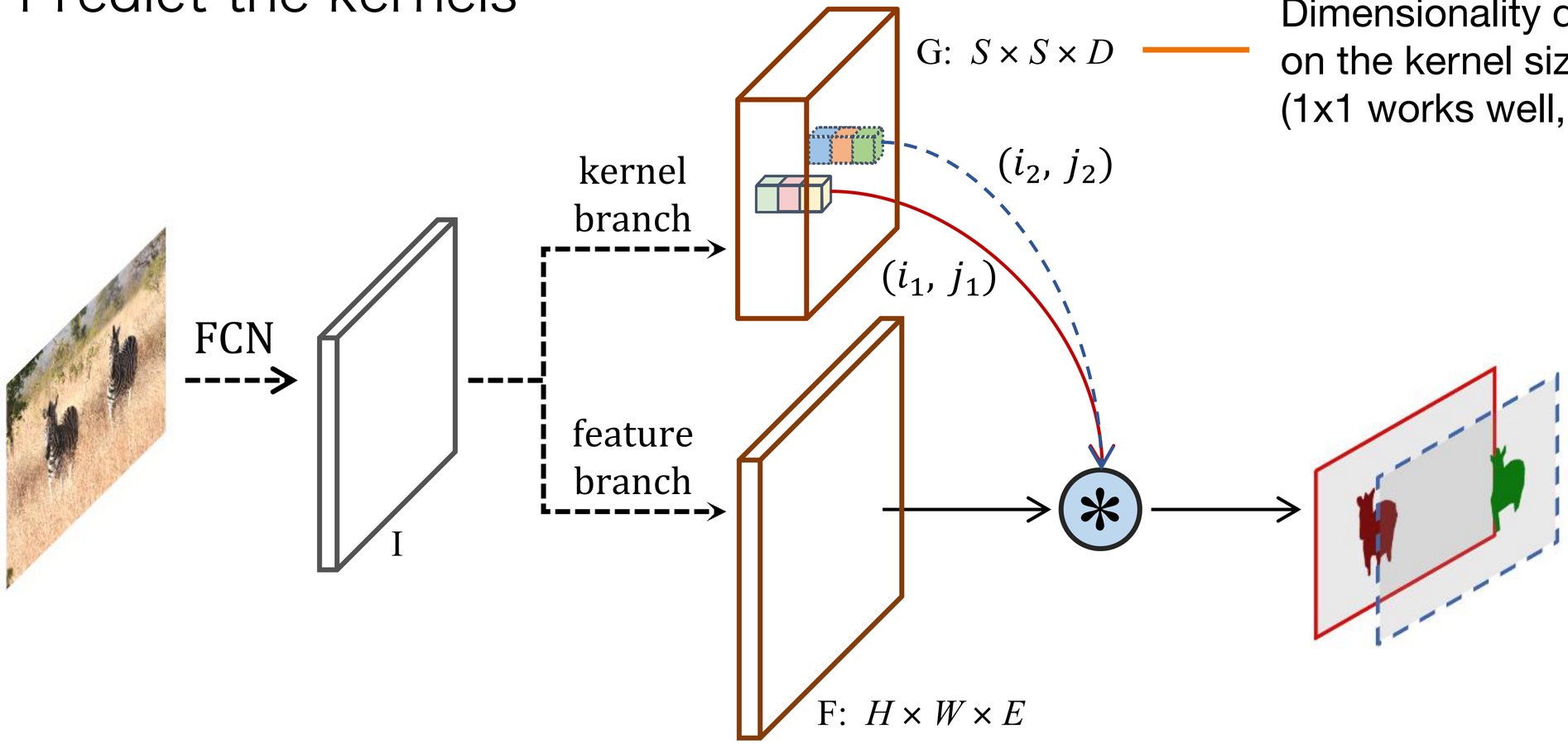
SOLOv2

- Idea: Predict the kernels



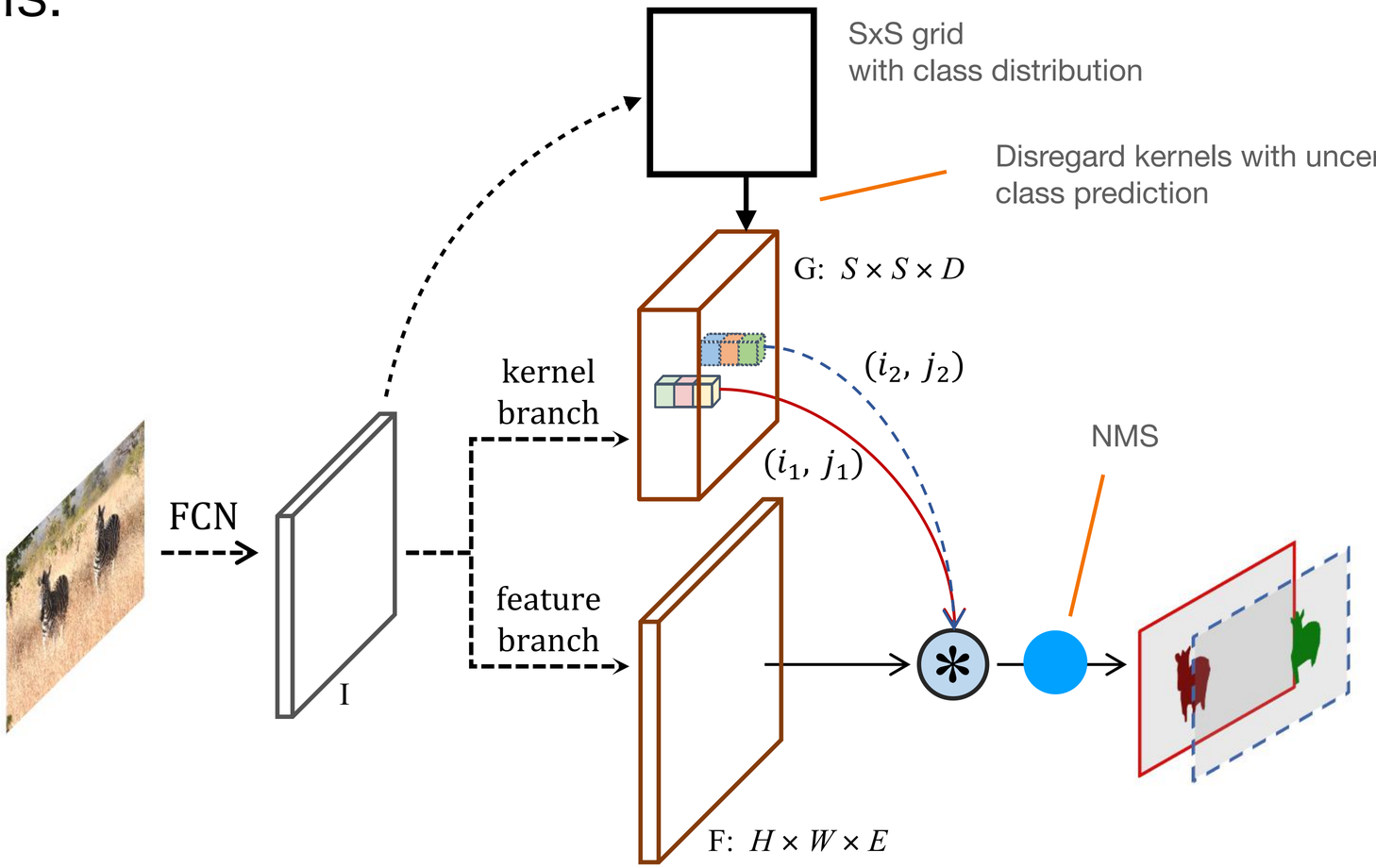
SOLOv2

- Idea: Predict the kernels



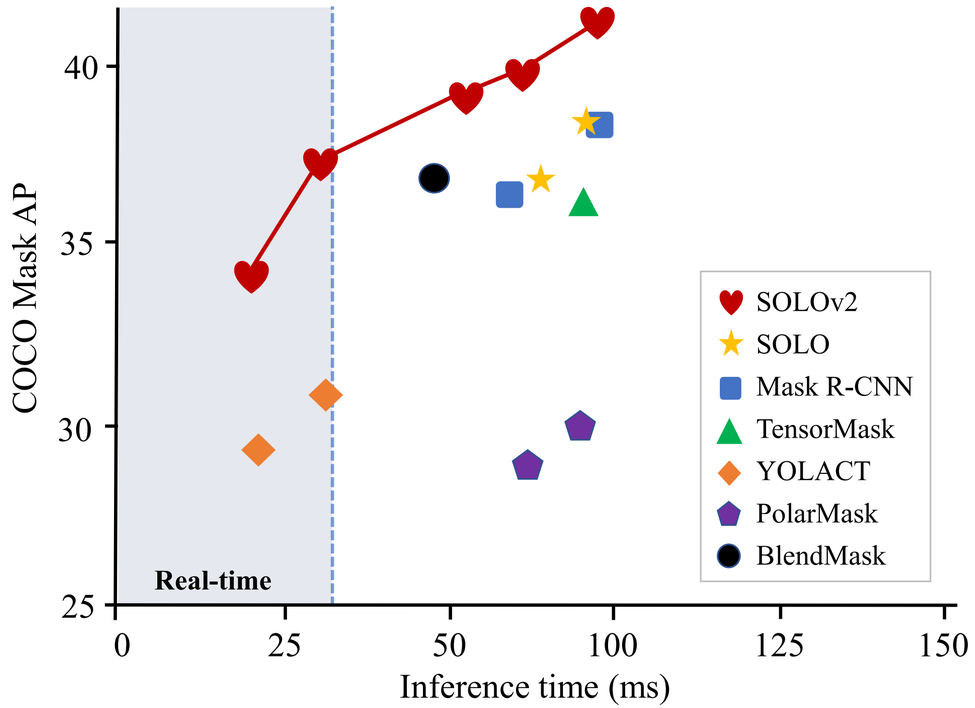
SOLOv2

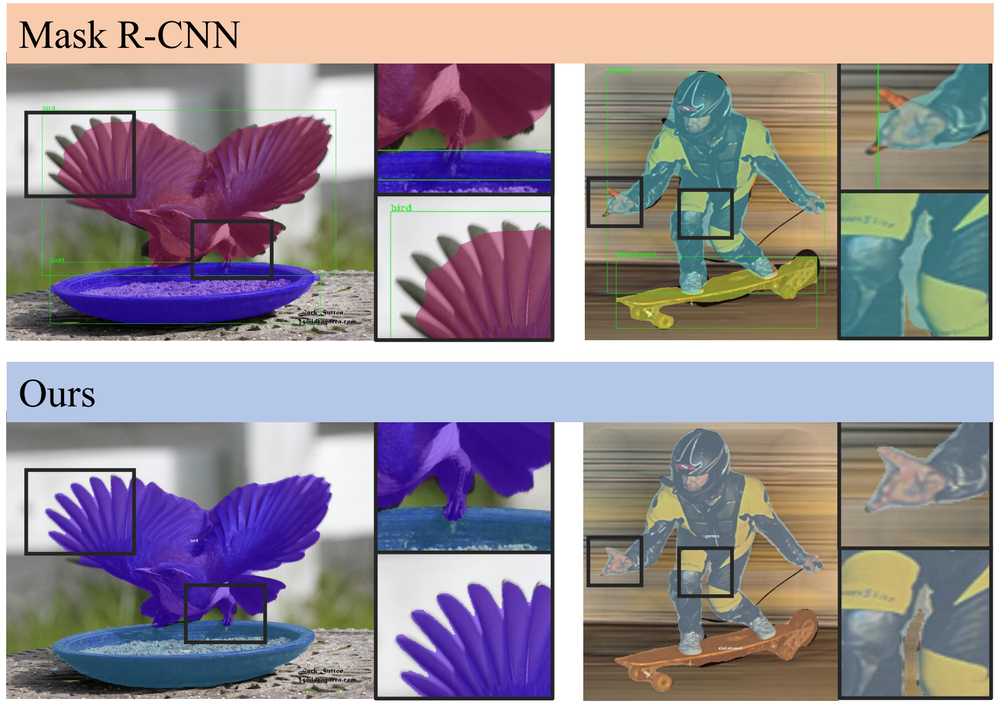
- A few details:



SOLOv2

- Conceptually very simple;
  - A natural extension from semantic segmentation models.
  - Fast and accurate:





Summary

- One-stage and two-stage instance segmentation methods offer accuracy vs. efficiency trade-off.
- Similar to our conclusions about object detectors:
  - Two-stage methods are more accurate (robust to scale variation), but less efficient;
  - One-stage methods are faster and have competitive accuracy.
    - Accurate segmentation of large-scale objects.

Recurrent methods

- Romera-Paredes & Torr “Recurrent instance segmentation” (2016).
- Ren & Zemel “End-to-end instance segmentation with recurrent attention” (2017).
- Araslanov et al. “Actor-critic Instance Segmentation” (2019).

- Conceptually interesting, but computation scales linearly with the number of instances in the image:
  - Can exploit the context of previous predictions;
  - Struggle in scenes with many objects.

Next lecture:
Panoptic segmentation