Computer Vision III:

Image Segmentation 2

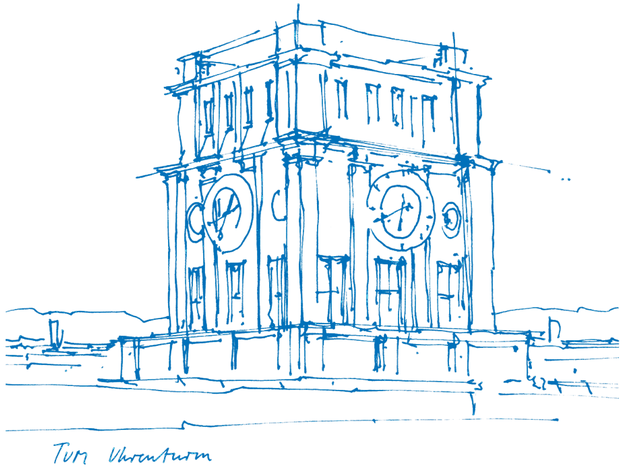
Dr. Nikita Araslanov

05.12.2023

Content credit:

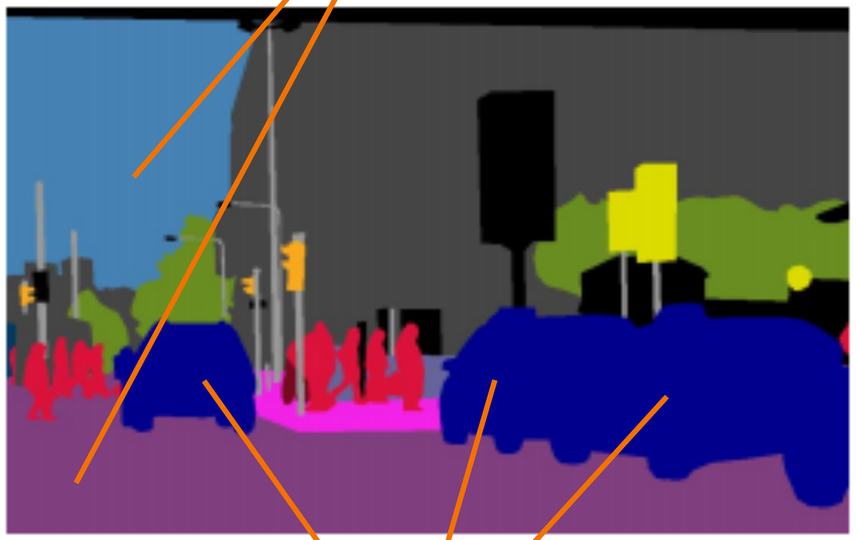
Prof. Laura Leal-Taixé

https://dvl.in.tum.de



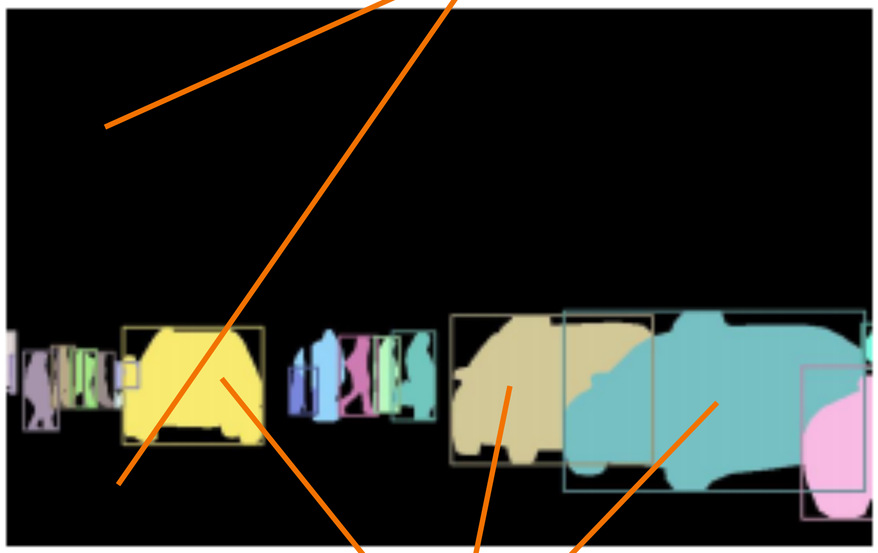
Instance segmentation

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”

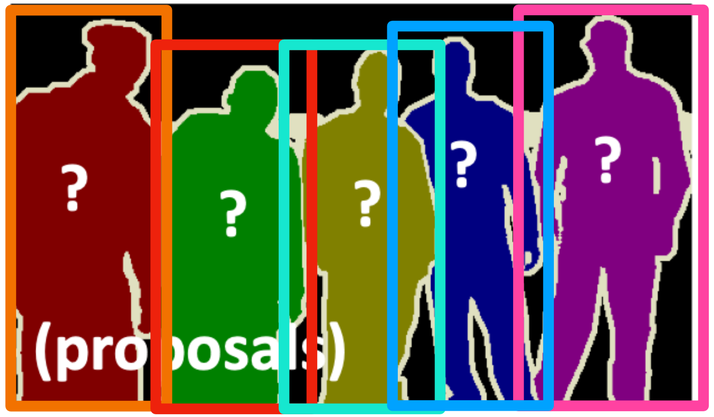


Differentiates between the pixels coming from instances of the same class

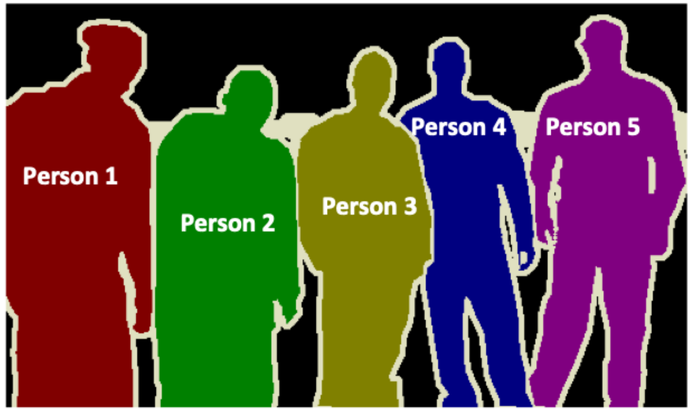
Instance segmentation methods

Proposal-based

1. Proposals
(e.g. bounding boxes)

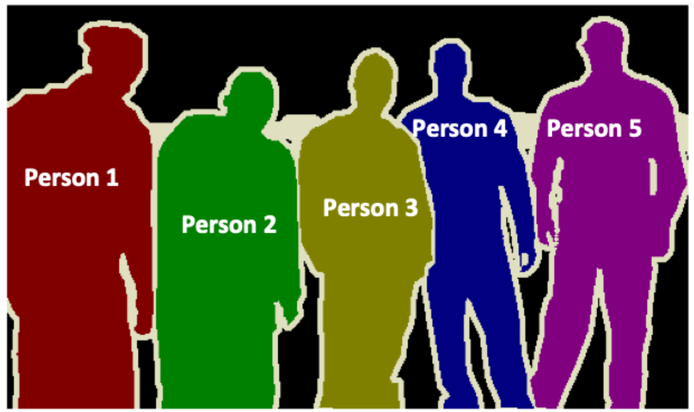


2. Segment and classify



Proposal-free

1. Semantic segmentation (optional)

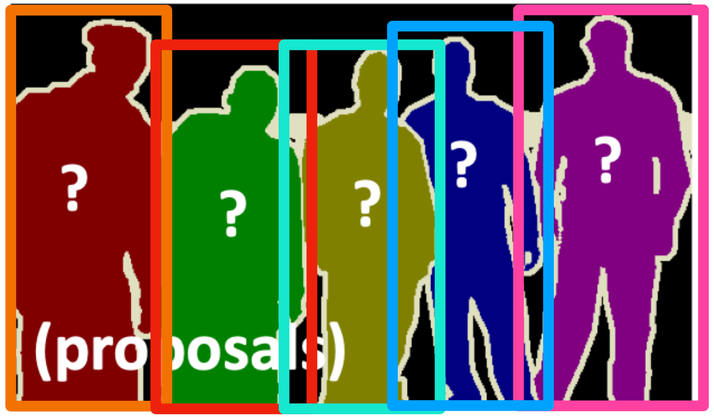


2. Group pixels into instances

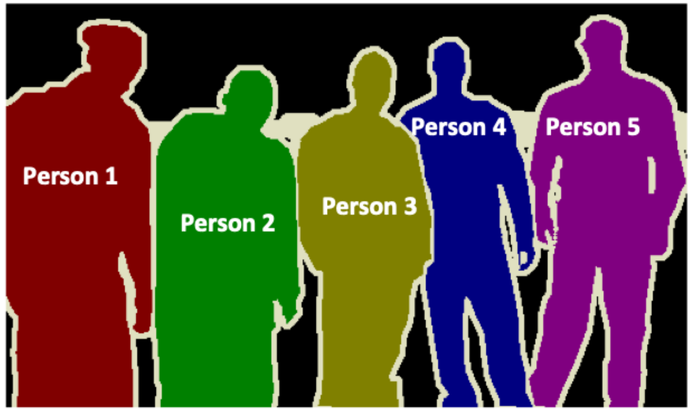
Instance segmentation methods

Proposal-based

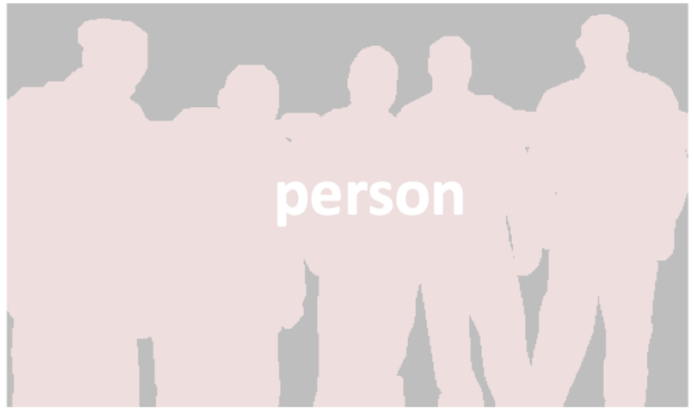
1. Proposals
(e.g. bounding boxes)



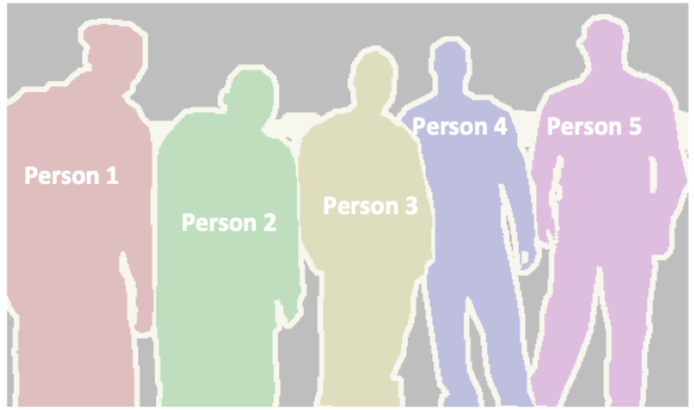
2. Segment and classify



Proposal-free



1. Semantic segmentation (optional)

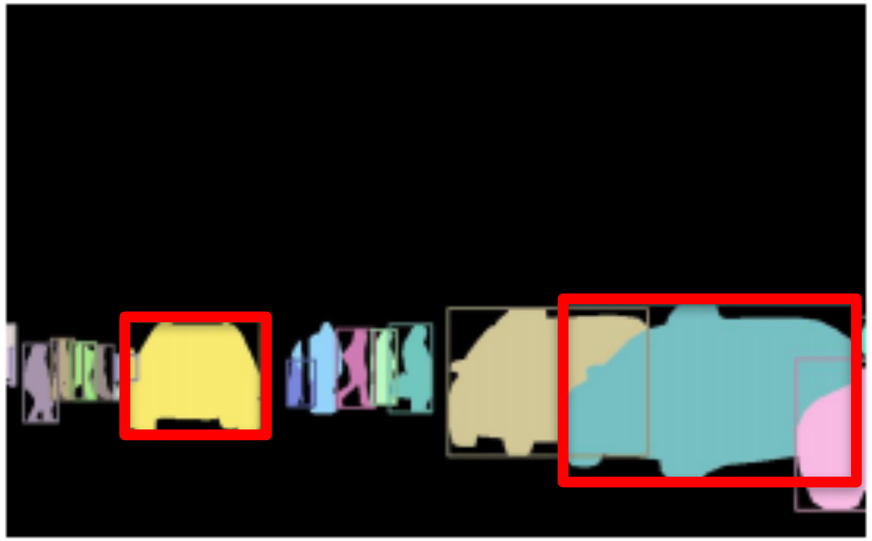


2. Group pixels into instances

Proposal-based methods

Bounding boxes...

We already know how to obtain those!

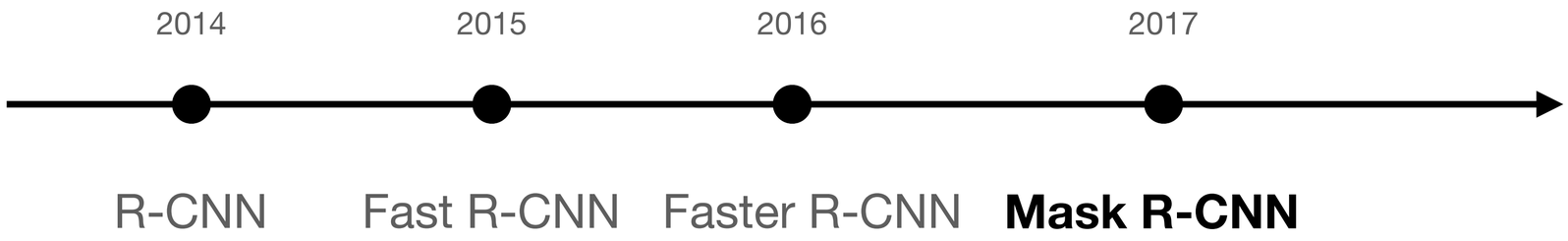


Proposal-based methods

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

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

R-CNN family



Mask R-CNN

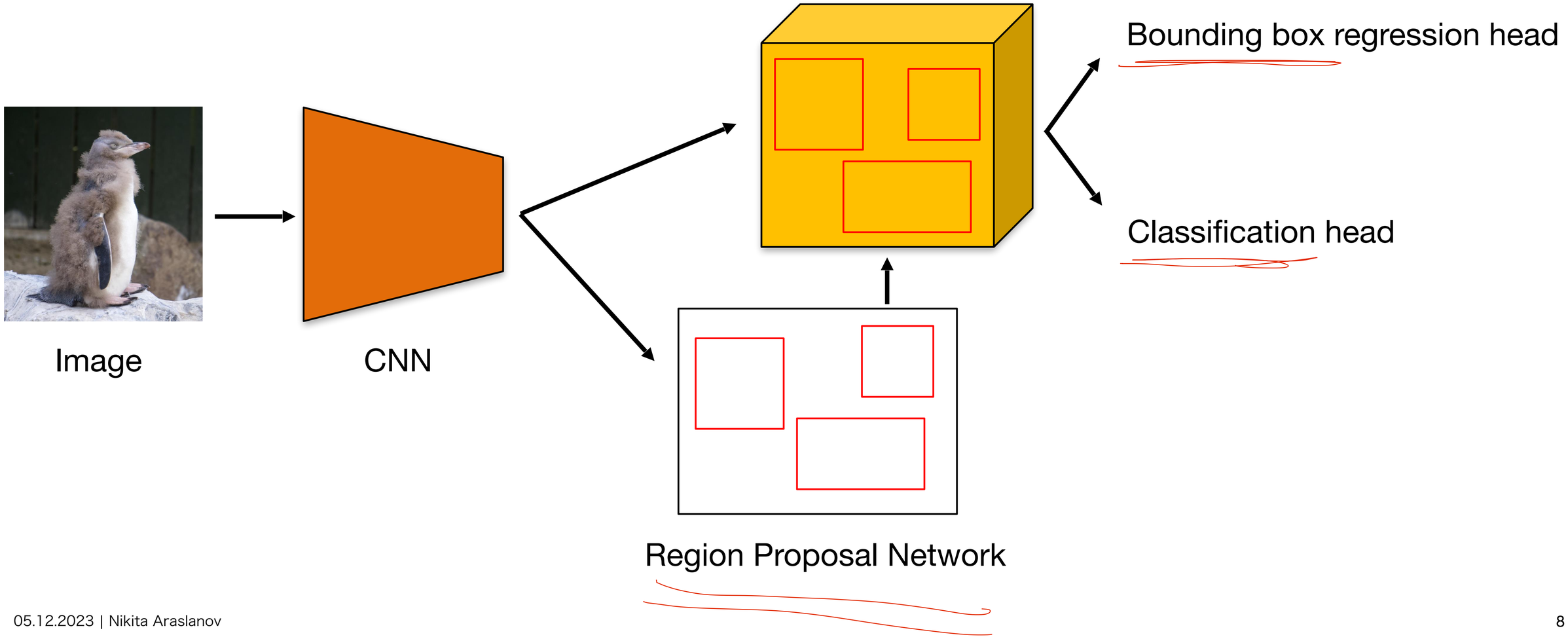
Kaiming He Georgia Gkioxari Piotr Dollár Ross Girshick

Facebook AI Research (FAIR)

(ICCV 2017)

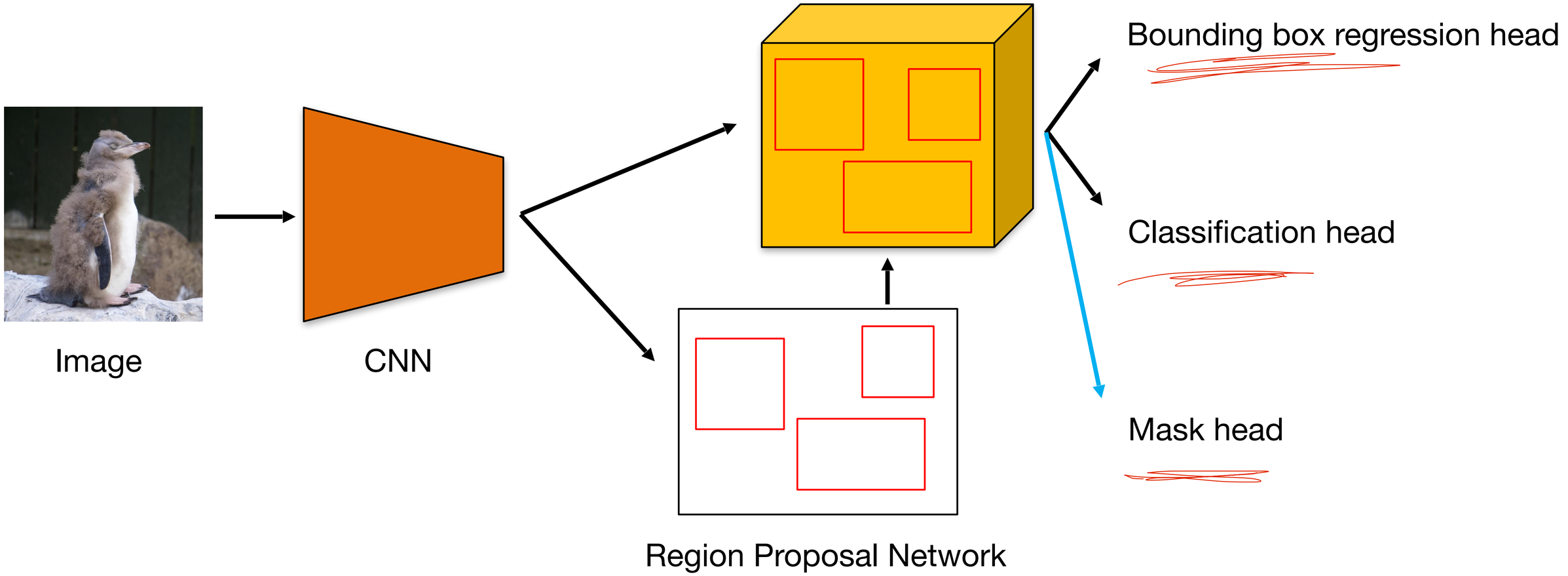
What is Mask R-CNN?

- Starting from the Faster R-CNN architecture

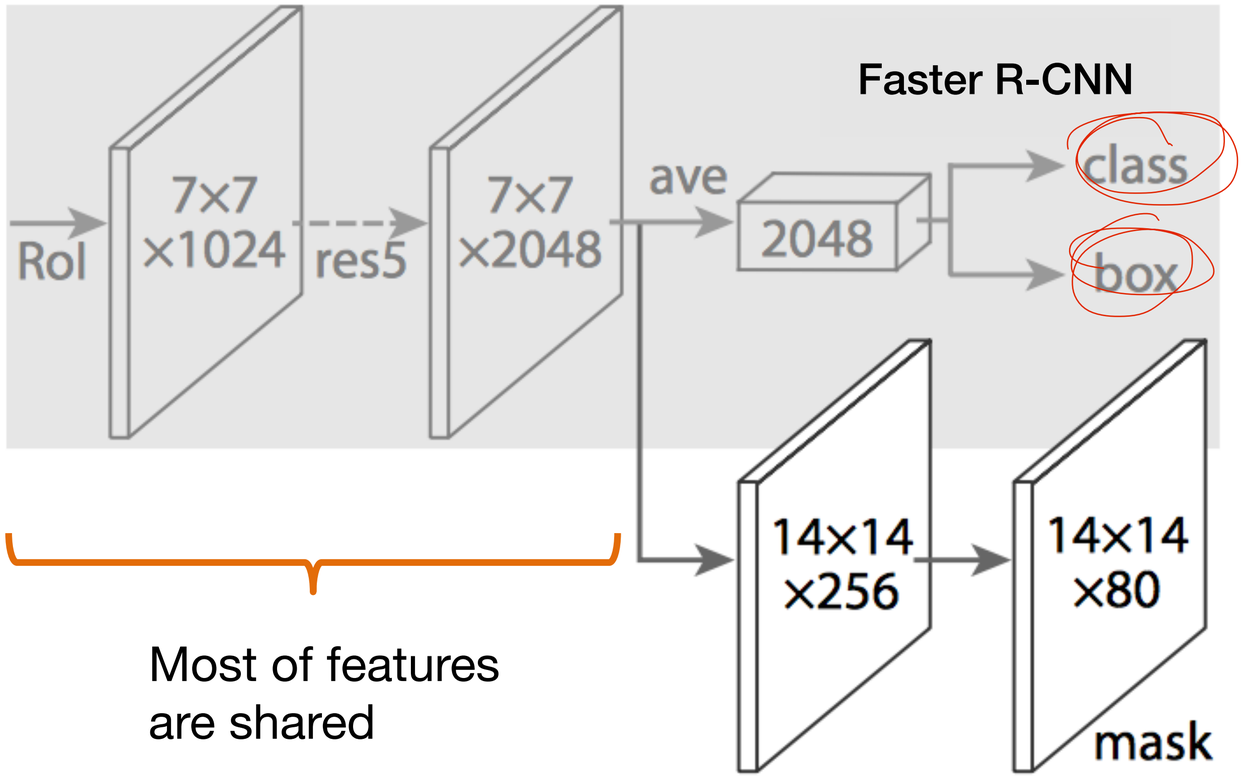


What is Mask R-CNN?

- Starting from the Faster R-CNN architecture



What is Mask R-CNN?

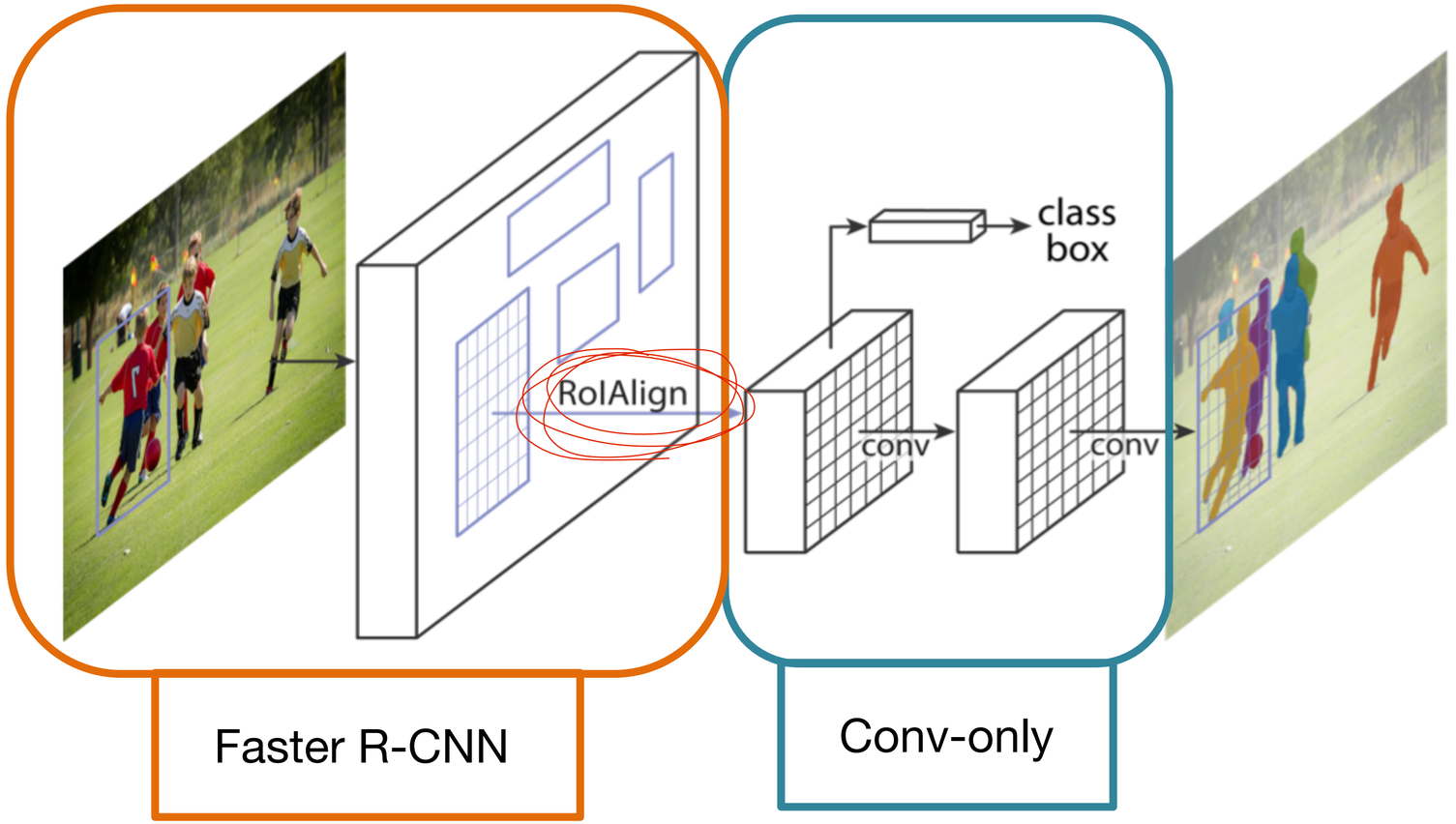


Object recognition head

Segmentation head

What is Mask R-CNN?

- Faster R-CNN + mask head for segmentation

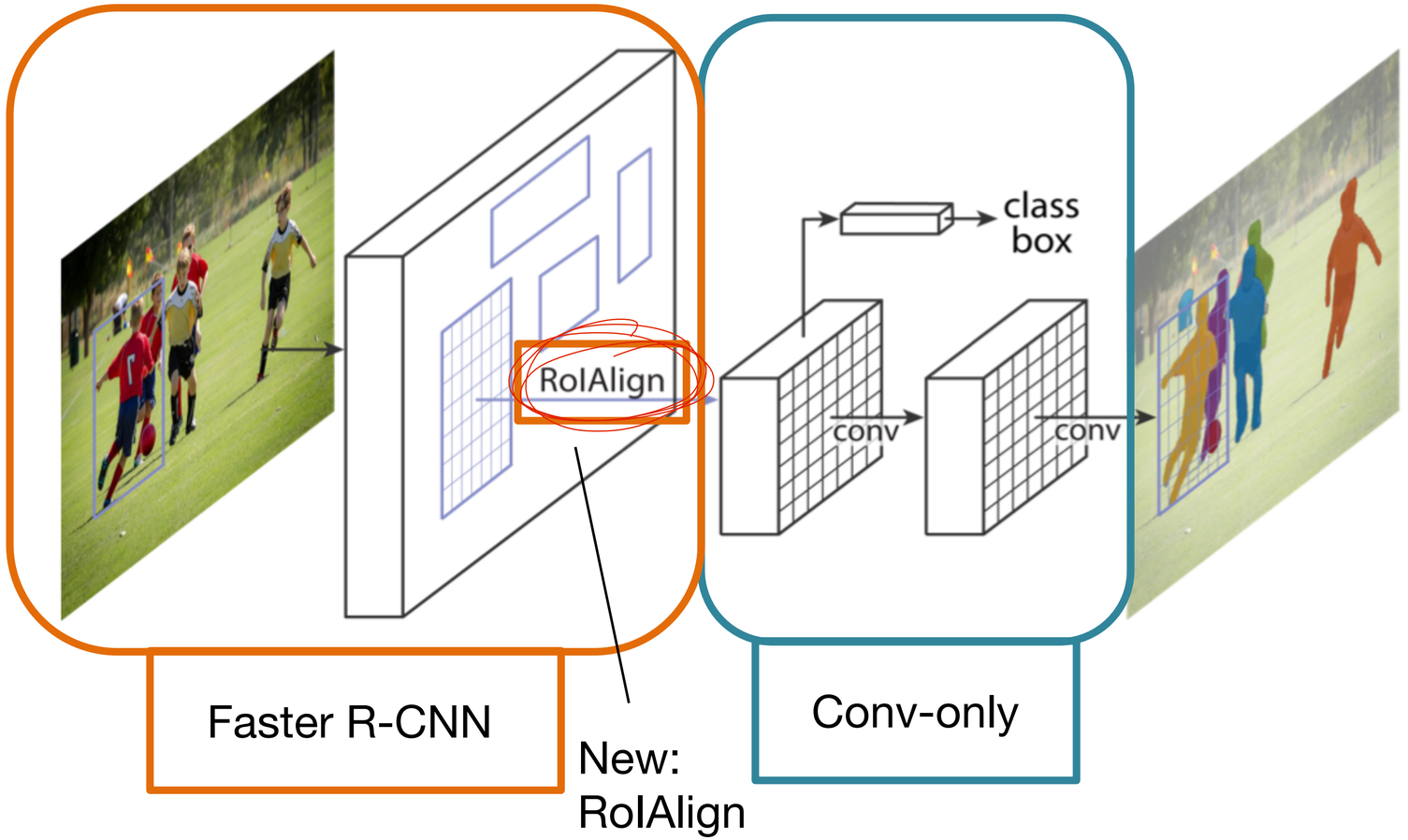


+ mask loss:
cross-entropy per pixel

He at al. “Mask R-CNN” ICCV 2017

What is Mask R-CNN?

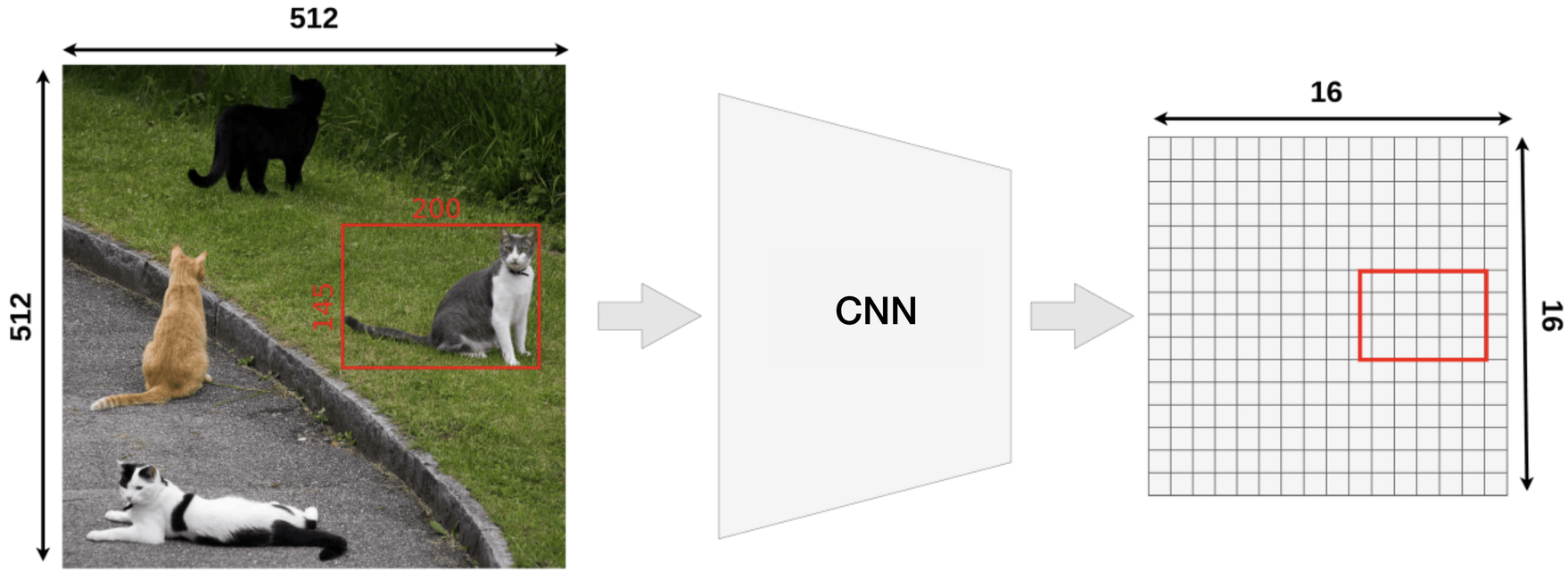
- Faster R-CNN + mask head for segmentation



+ mask loss:
cross-entropy per pixel

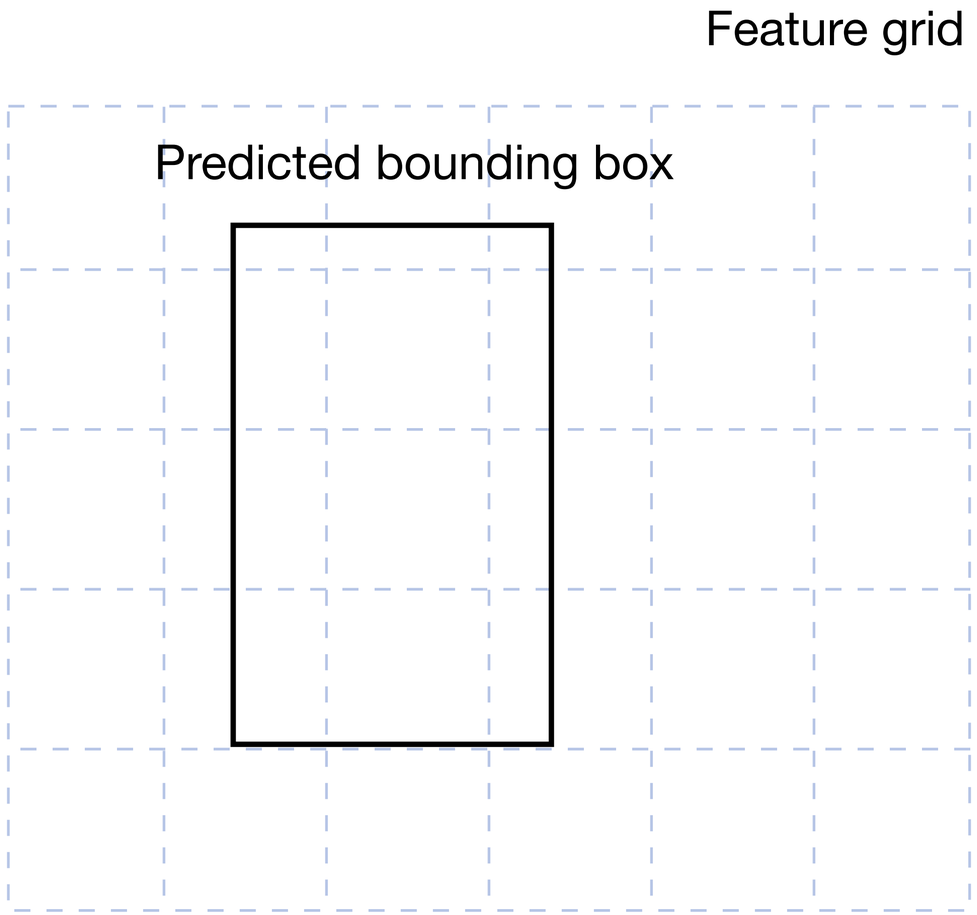
He at al. “Mask R-CNN” ICCV 2017

Recall RoIPool



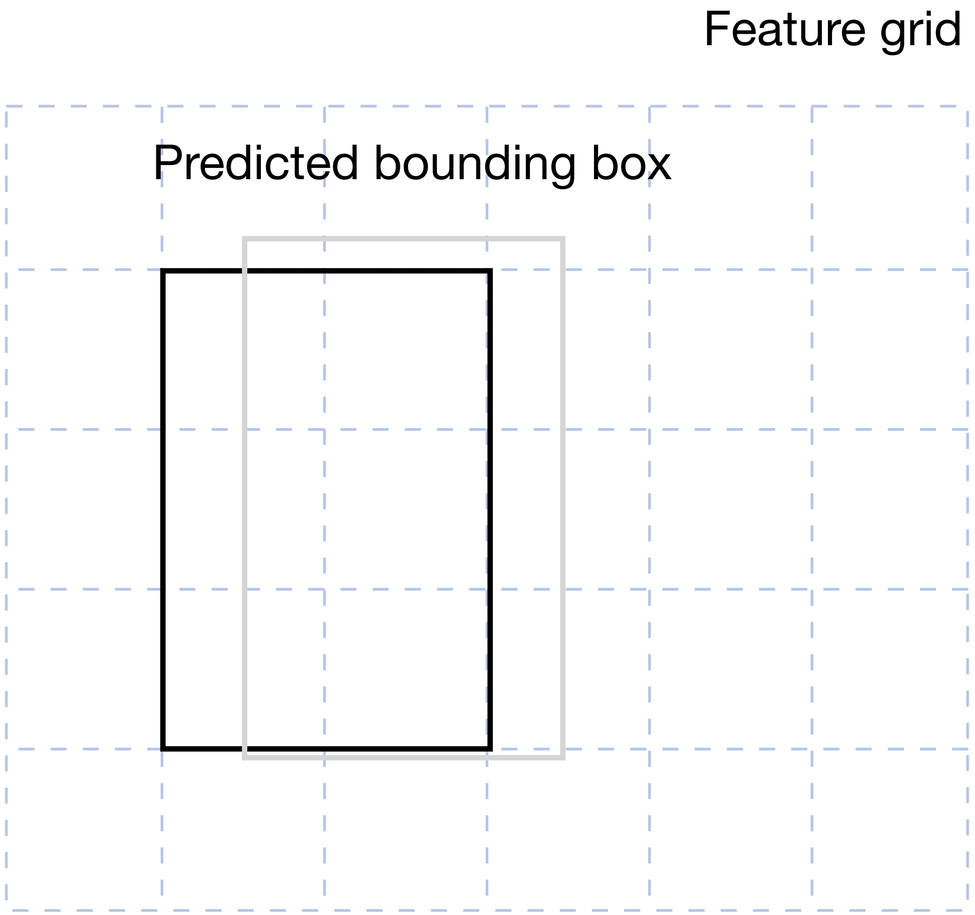
Model feature mapping process

RoIPool



Two quantisations:

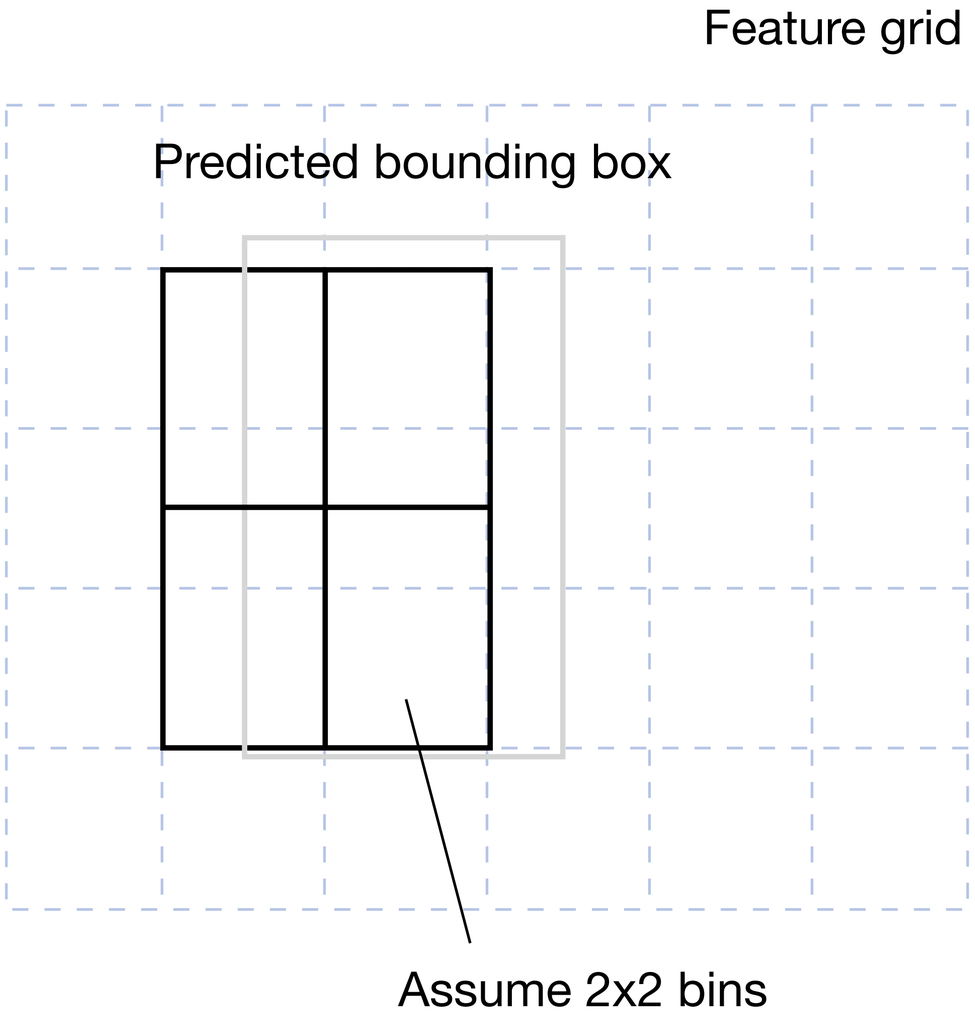
RoIPool



Two quantisations:

1. Bounding box alignment

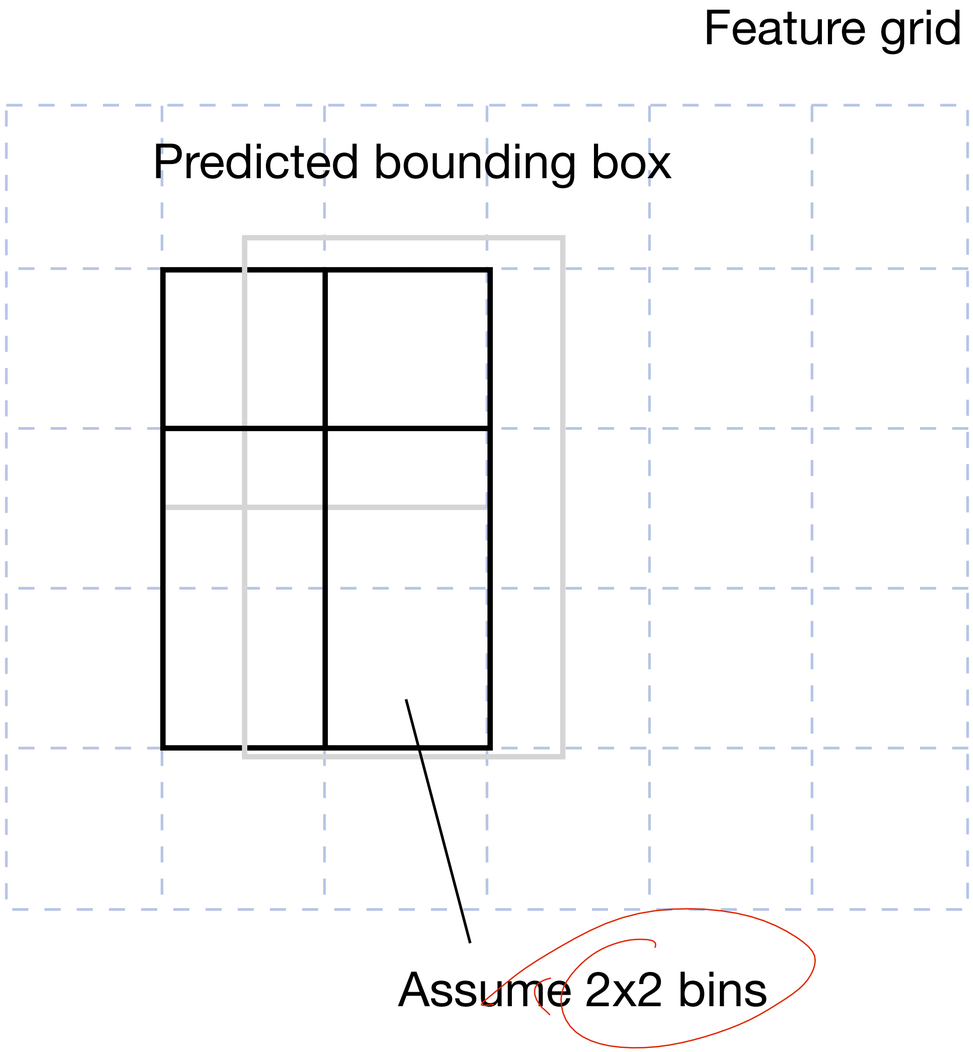
RoIPool



Two quantisations:

1. Bounding box alignment

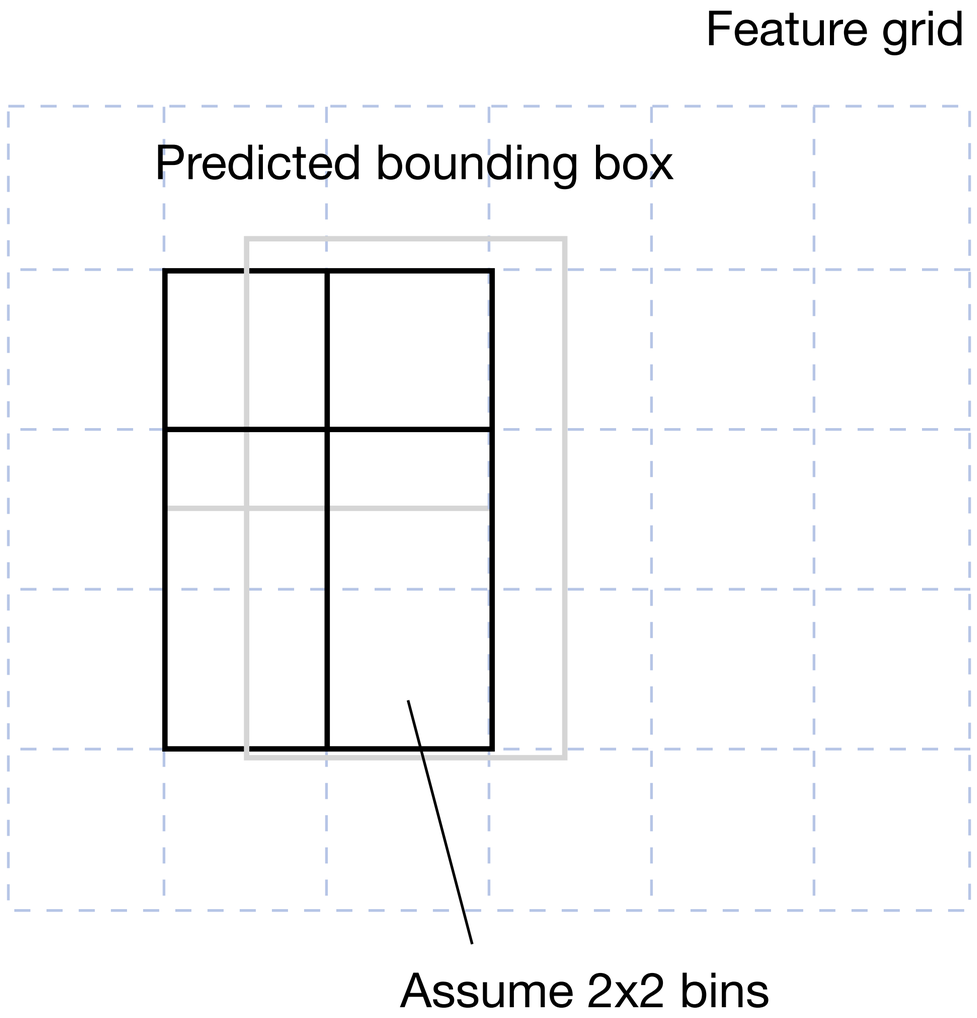
RoIPool



Two quantisations:

1. Bounding box alignment
2. Bin alignment

RoIPool



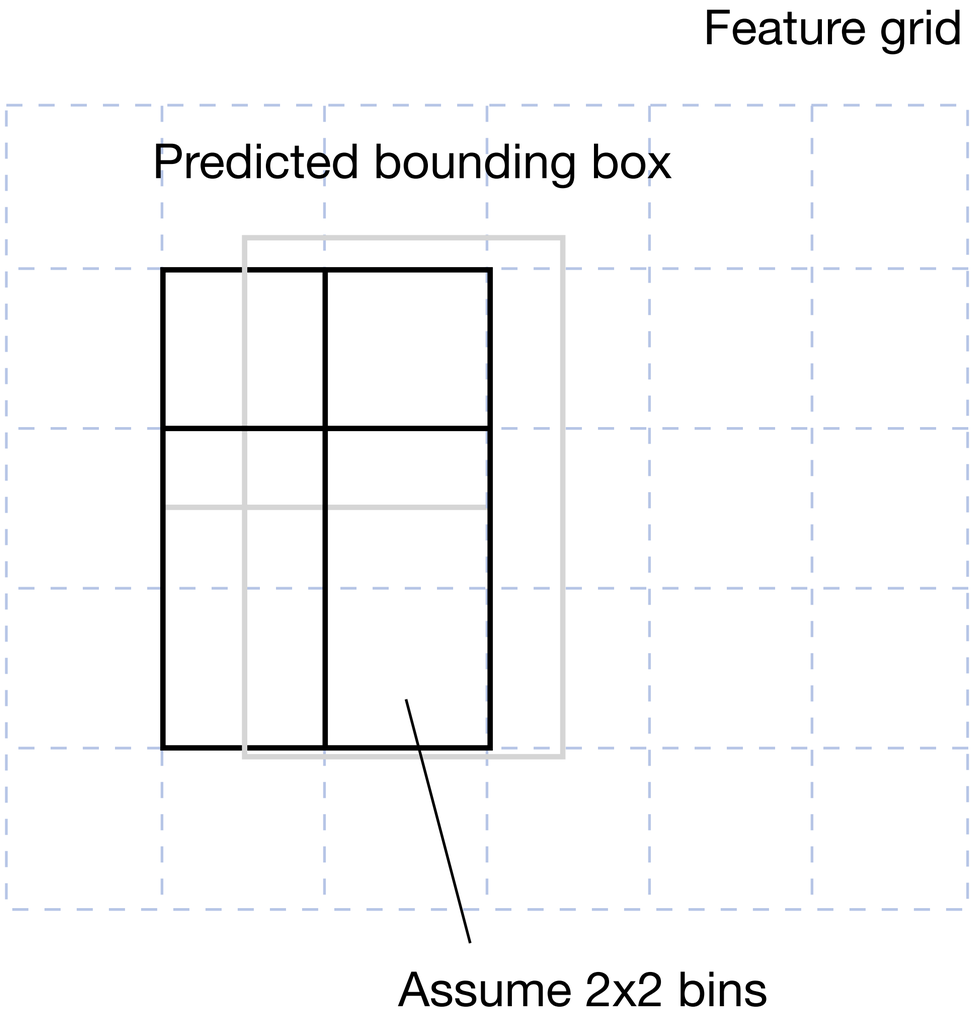
Two quantisations:

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)

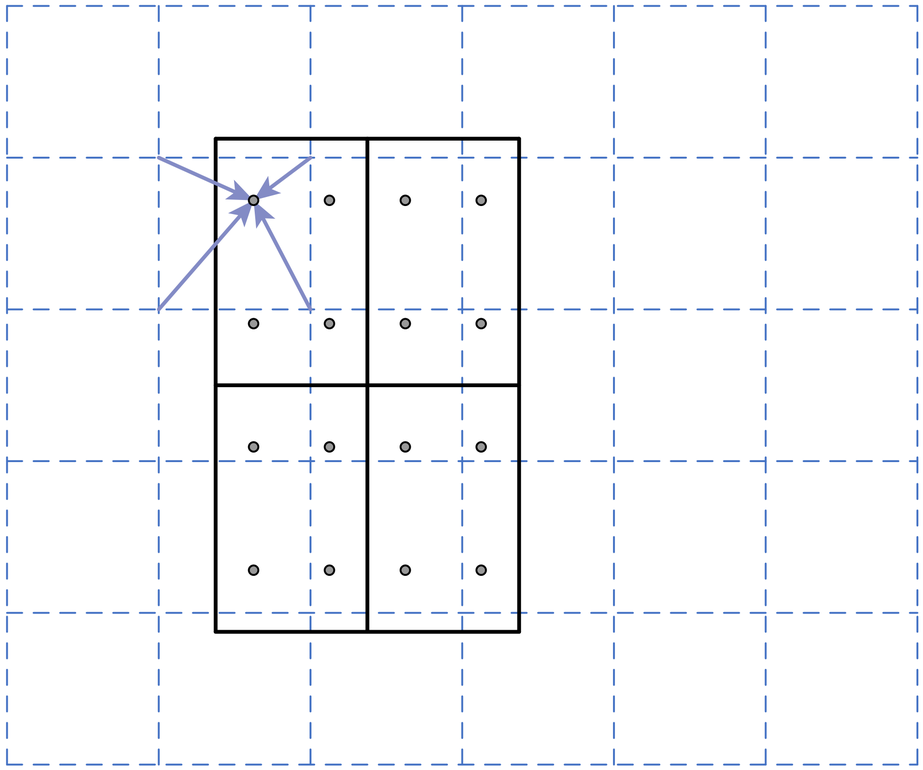
Works well for object detection

(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

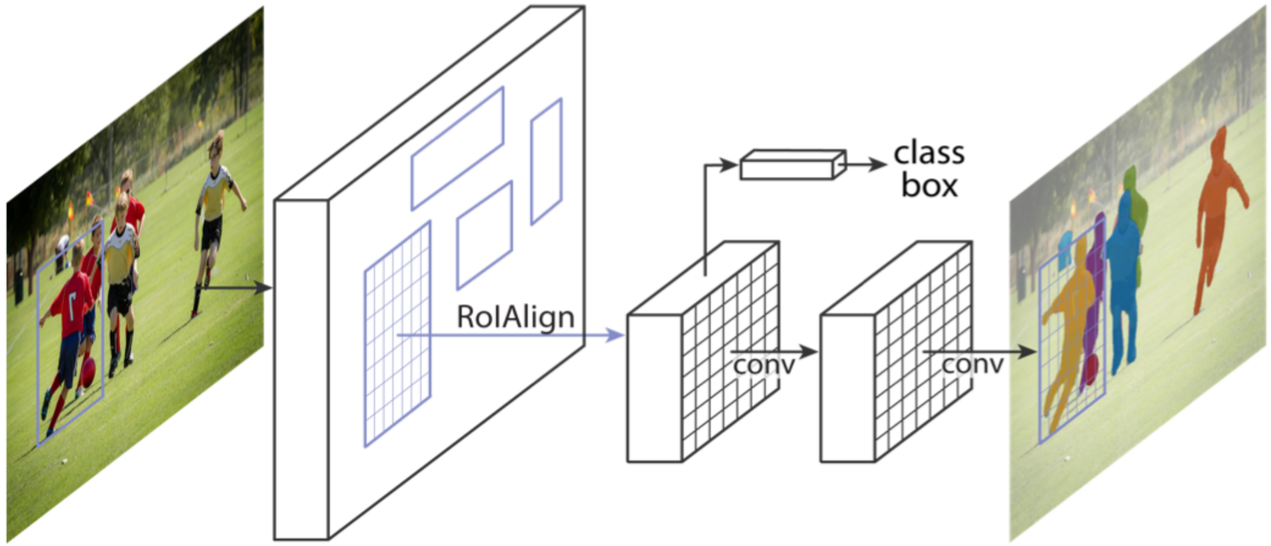
RoIAlign



- 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)

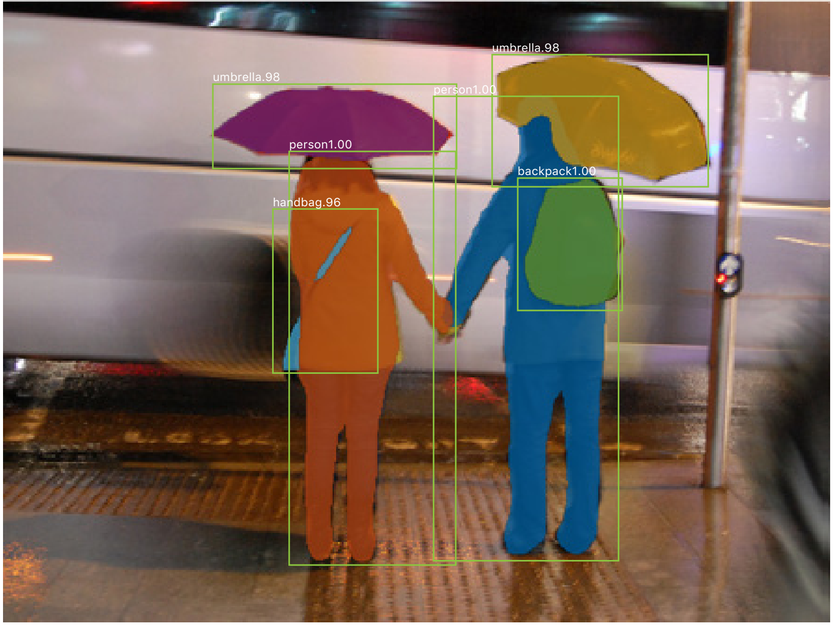
What is Mask R-CNN?

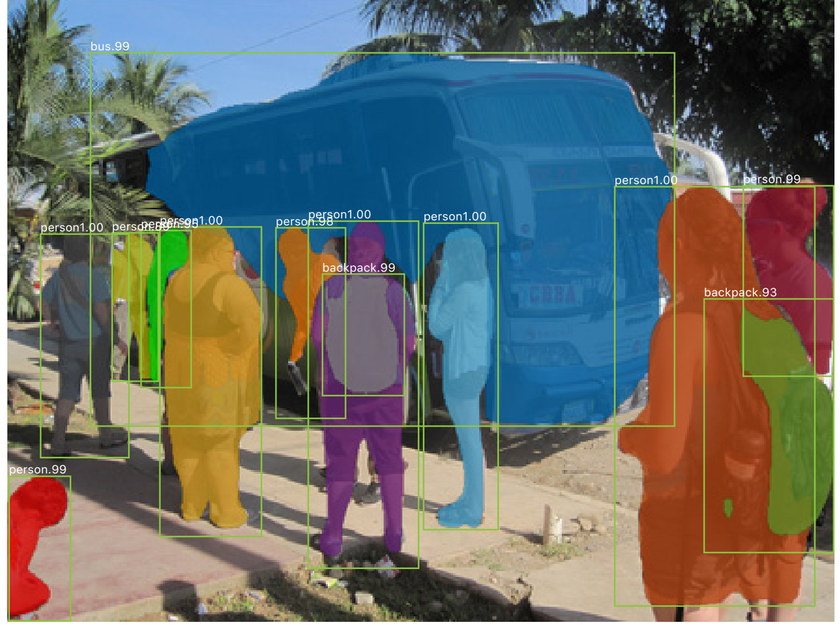
- Seemingly incremental improvements
- Simple design overall
- Best paper award (CVPR 2017)

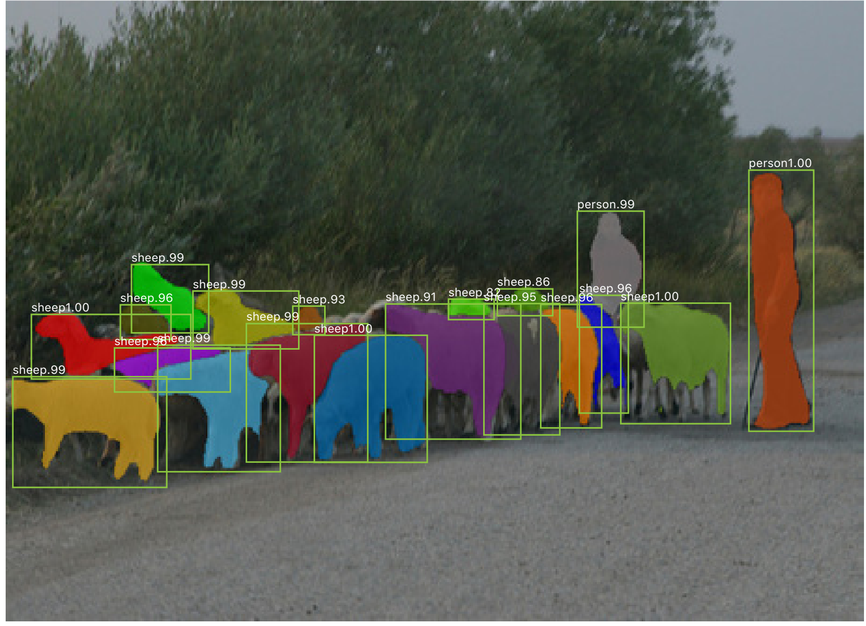


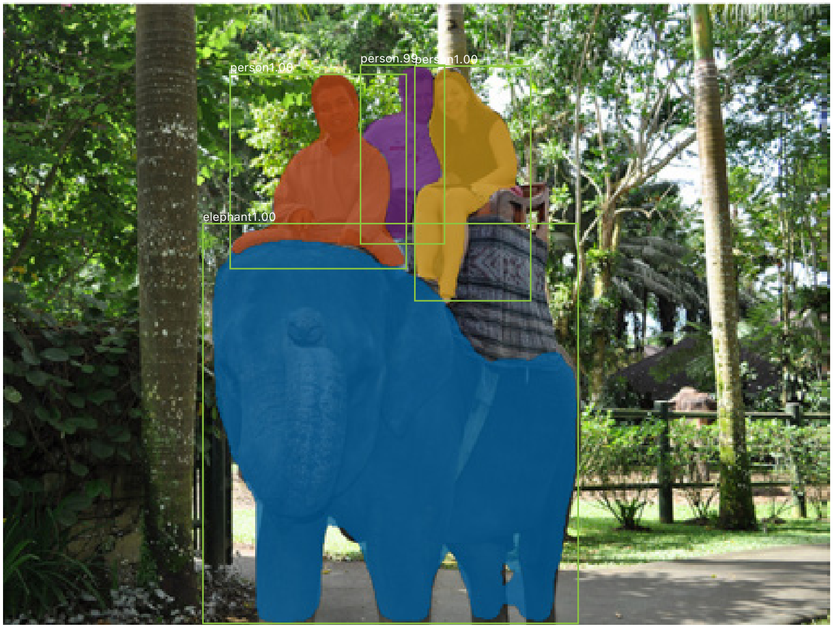
He at al. “Mask R-CNN” ICCV 2017

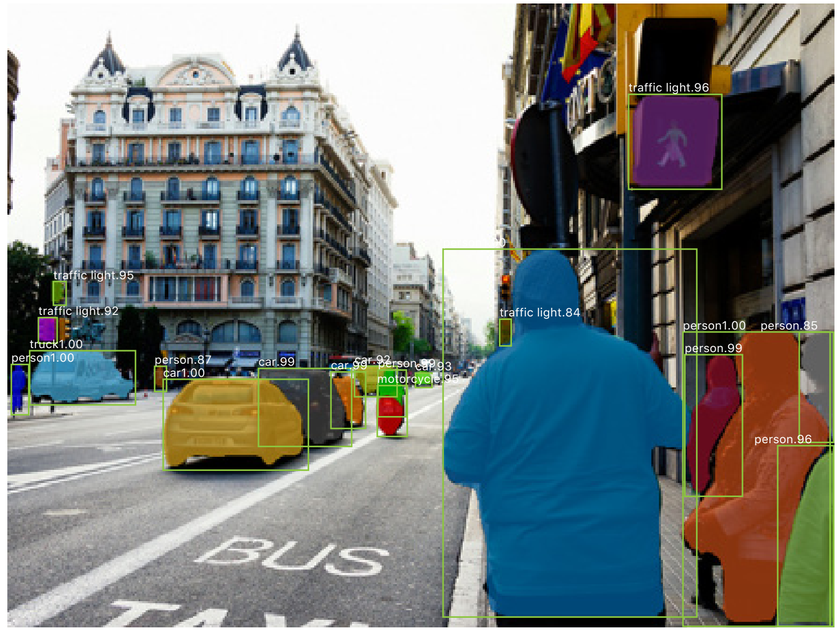
Mask R-CNN: Qualitative results

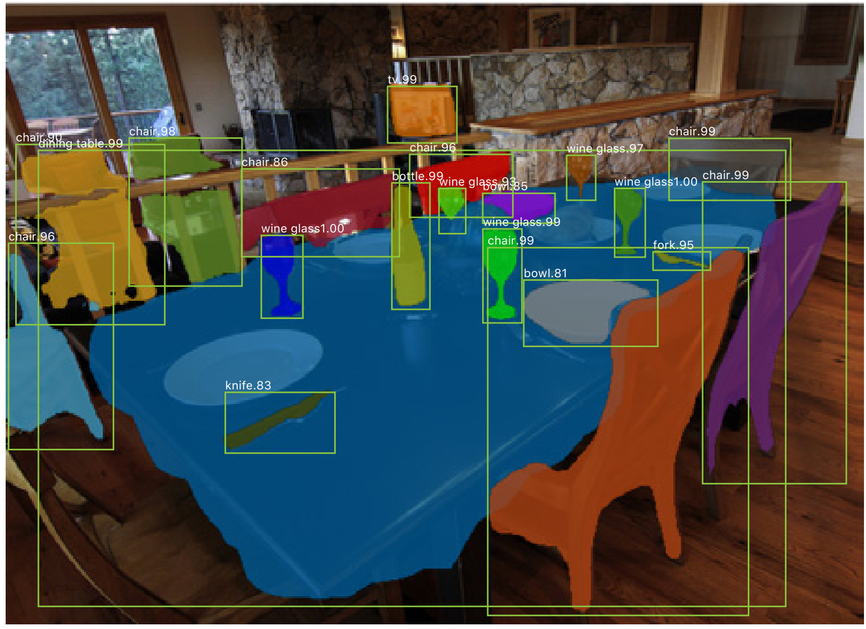












Mask R-CNN: Qualitative results

Mask R-CNN: Improvements

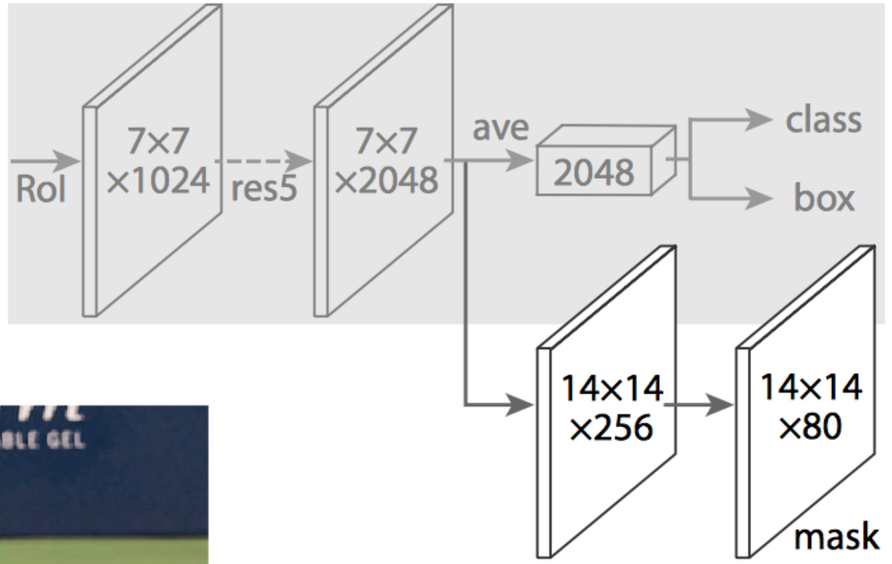
- Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020).
- Huang et al., “Mask Scoring R-CNN” (2019).
- Liu et al., “Path Aggregation Network for Instance Segmentation” (2018).
- Cai and Vasconcelos. “Cascade R-CNN: High Quality Object Detection and Instance Segmentation” (2019)

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

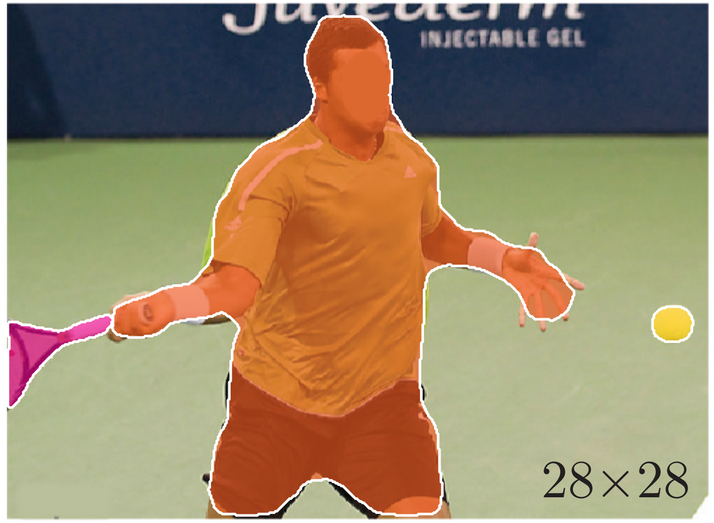


- Example:

28x28

Mask head prediction

Bilinear upsampling

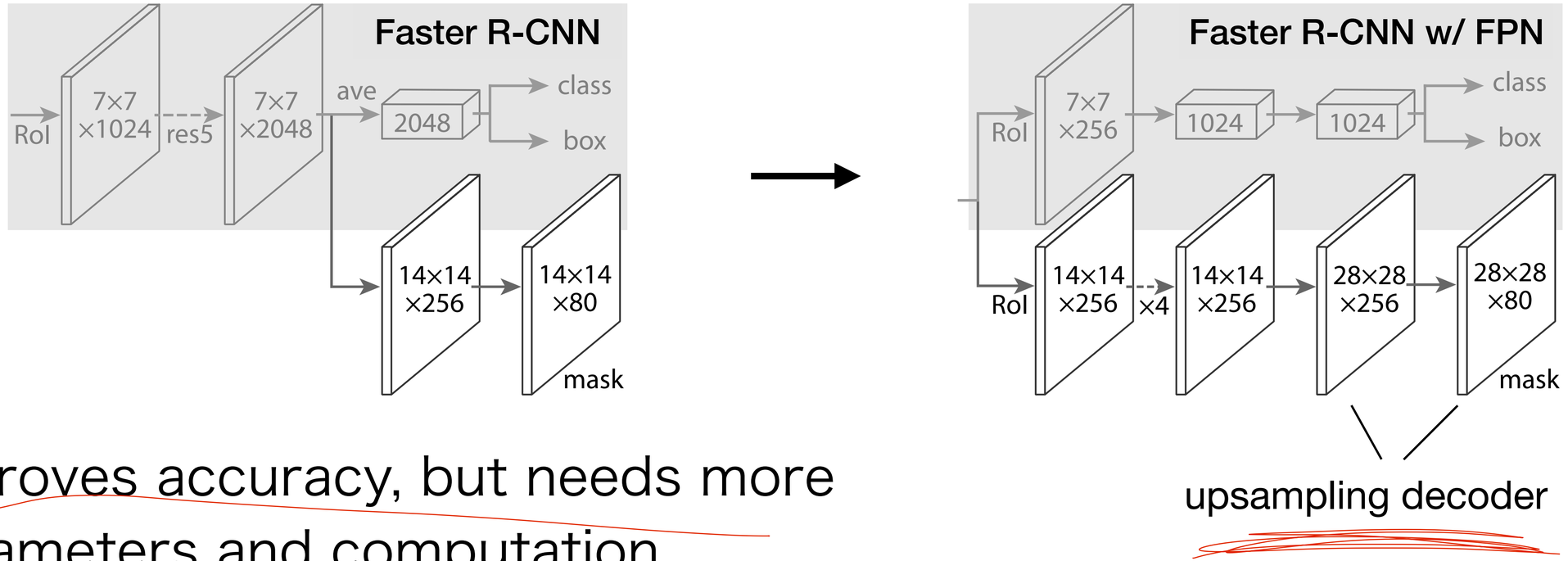


Upsampled mask

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

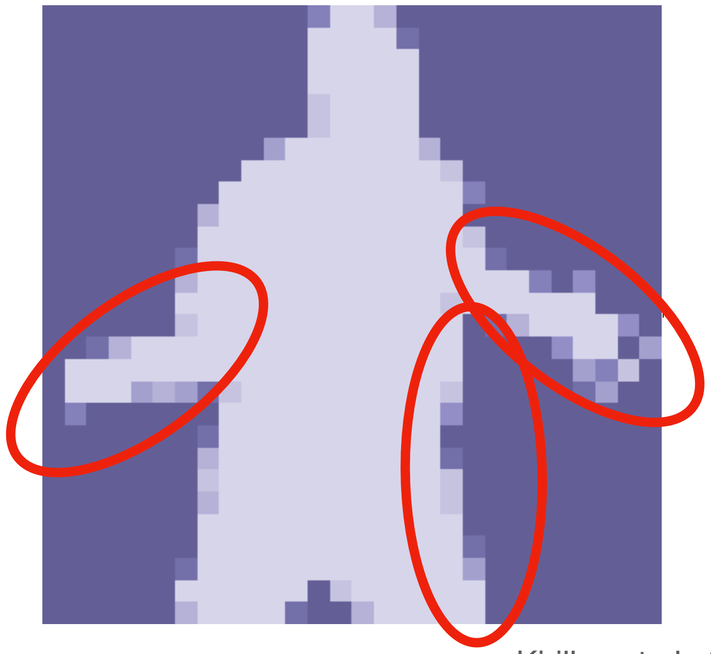


- Improves accuracy, but needs more parameters and computation

Mask R-CNN + PontRend

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

28x28

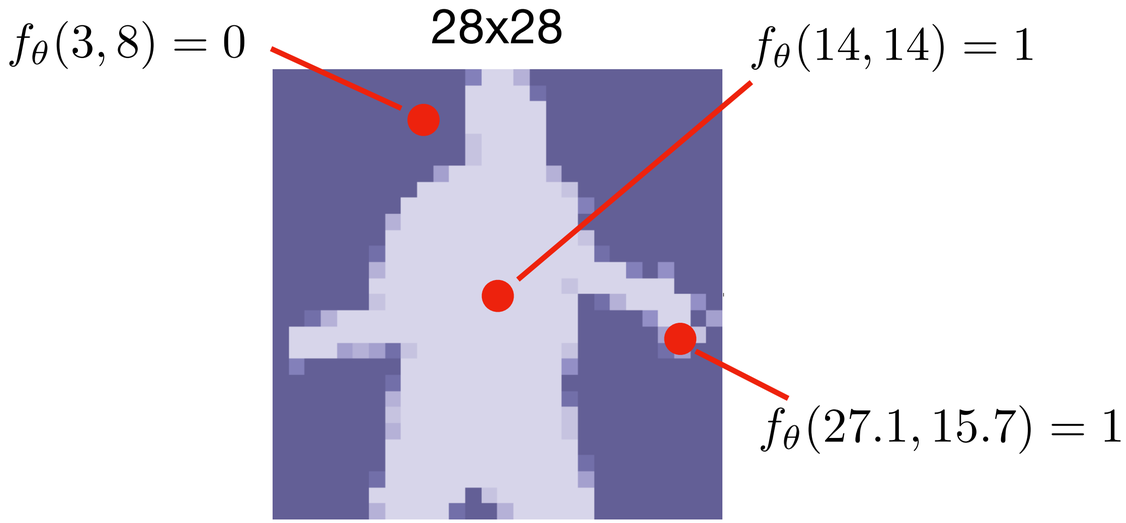


Kirillov et al., “PointRend: Image Segmentation as Rendering” (2020)

Coordinate-based neural representation

- 掩码头是将离散信号表示（如像素）映射到所需值（如二进制掩码）的一个示例；
- 相反，我们将掩码参数化为映射信号域（如 (x,y) 坐标）的连续函数：

- 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):

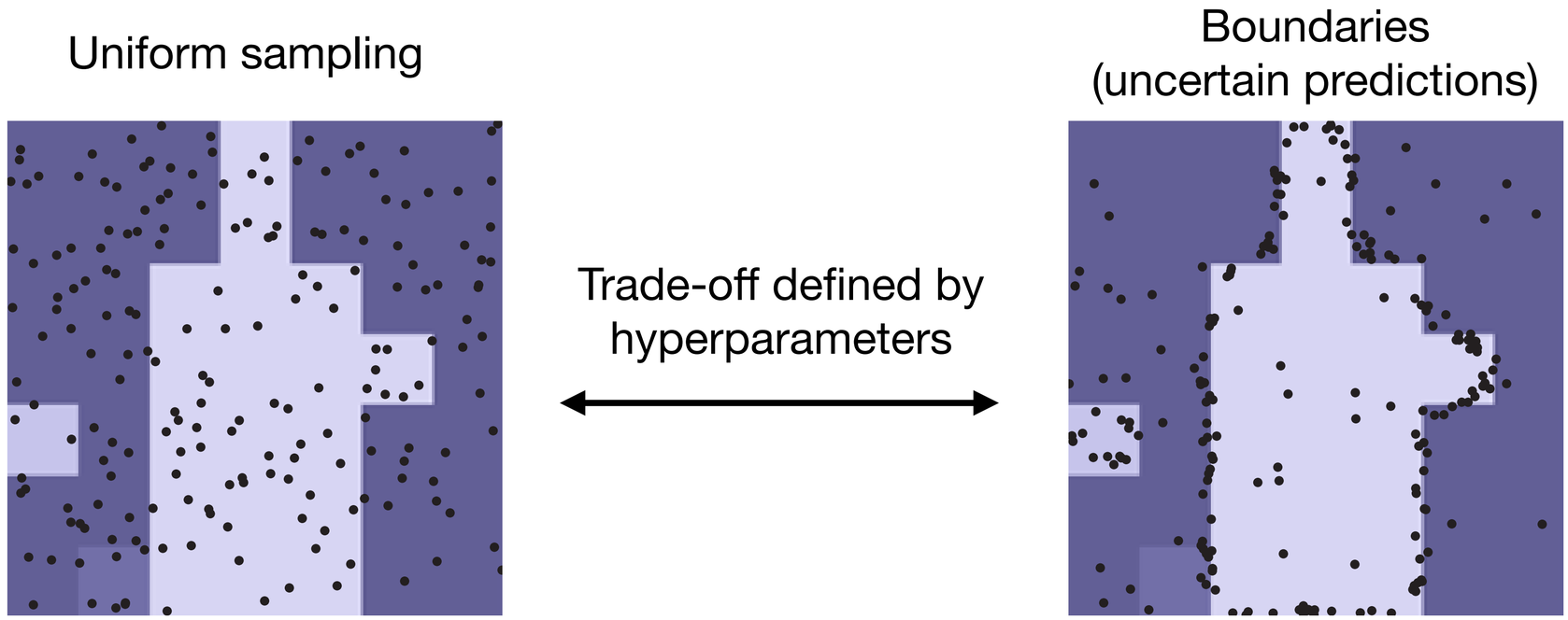


- $f_\theta(x, y)$ is an example of an coordinate-based net;
- Why is it useful here?
- We can query fractional coordinates.

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

Coordinate-based neural representation

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



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

Coordinate-based neural representation

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

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

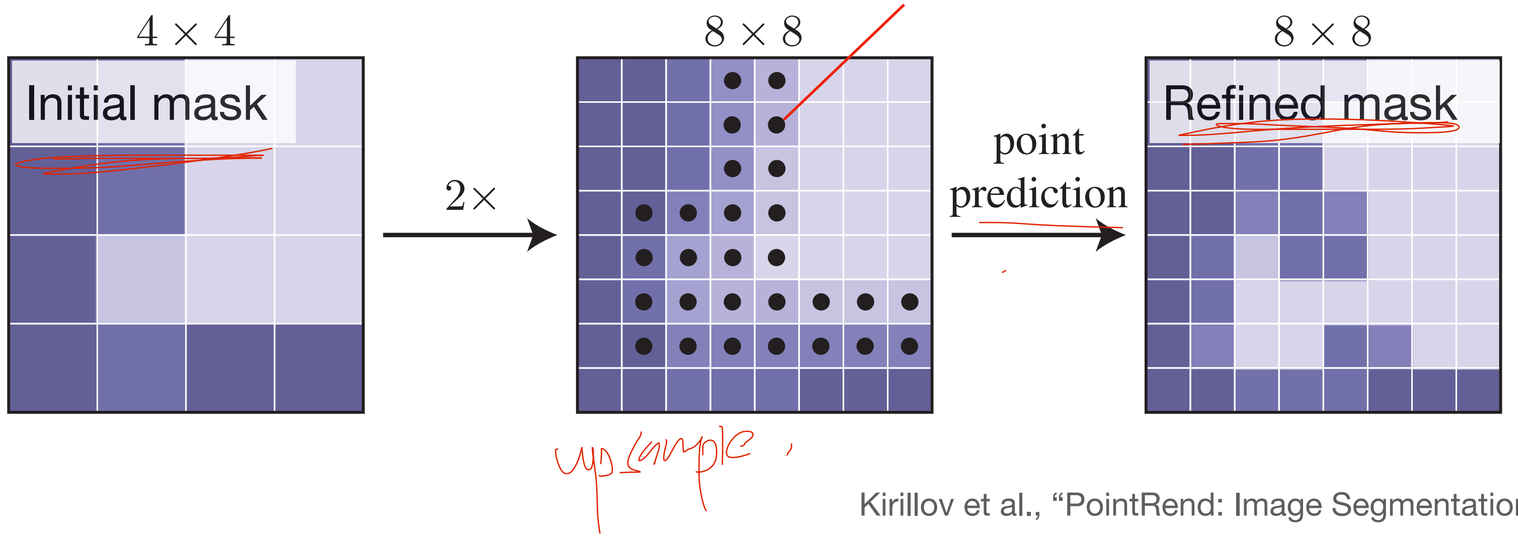
Coordinate-based neural representation

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

- Adaptive subdivision step (test time):

Queries to $f_\theta(x, y)$

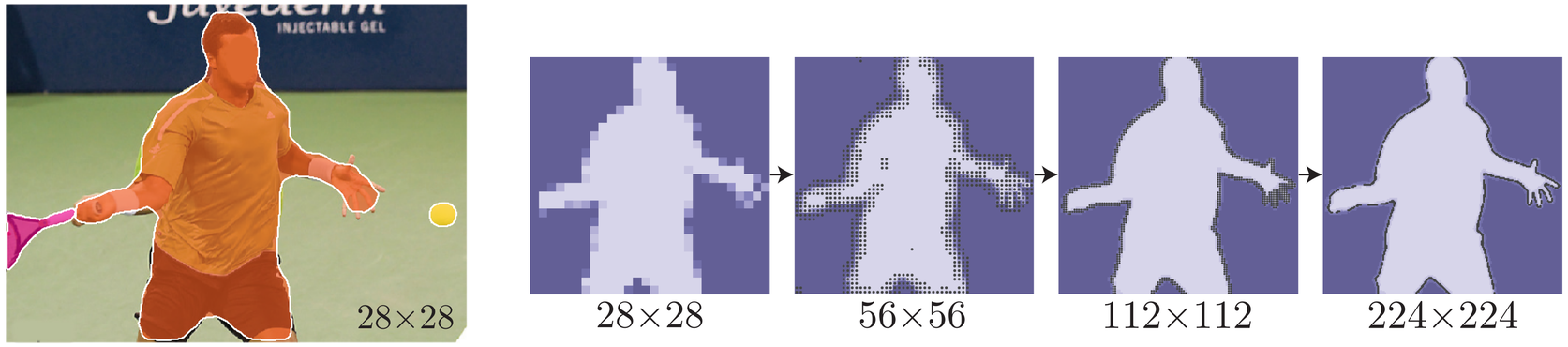
One iteration:



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

Coordinate-based neural representation

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



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

PointRend: Qualitative results

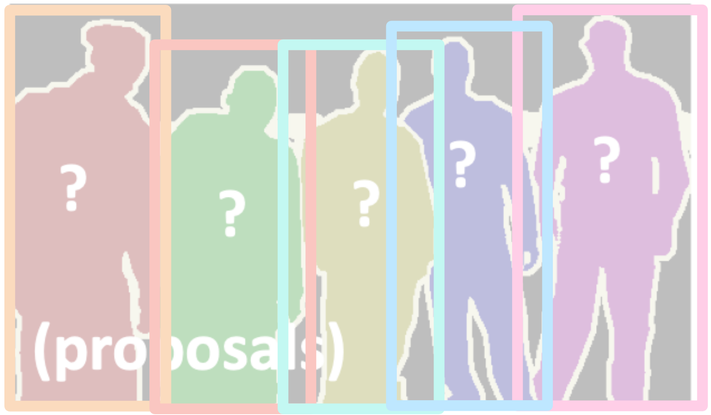
Coordinate-based neural representation

- Remarks:
  - In practice, compute a point-wise feature for each coordinate (Quiz: Why?).
  - We compute a point-wise feature with bilinear interpolation.
  - We can concatenate features sampled this way from multiple feature maps.
  - The point head f_θ is trained on these features, not the coordinates.

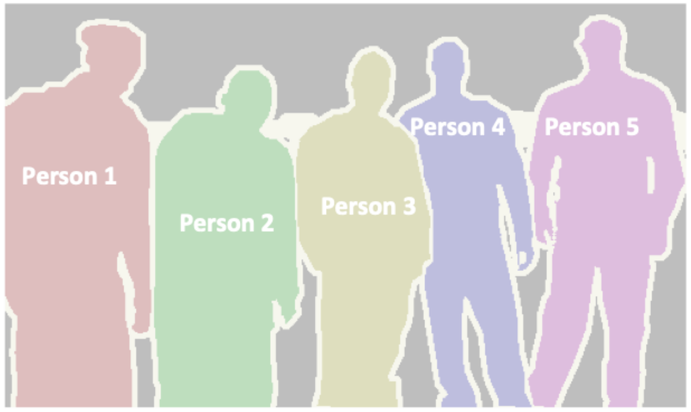
Instance segmentation methods

Proposal-based

1. Proposals
(e.g. bounding boxes)

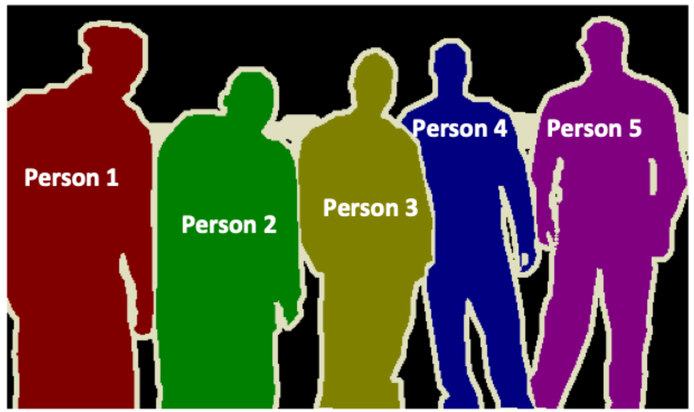


2. Segment and classify



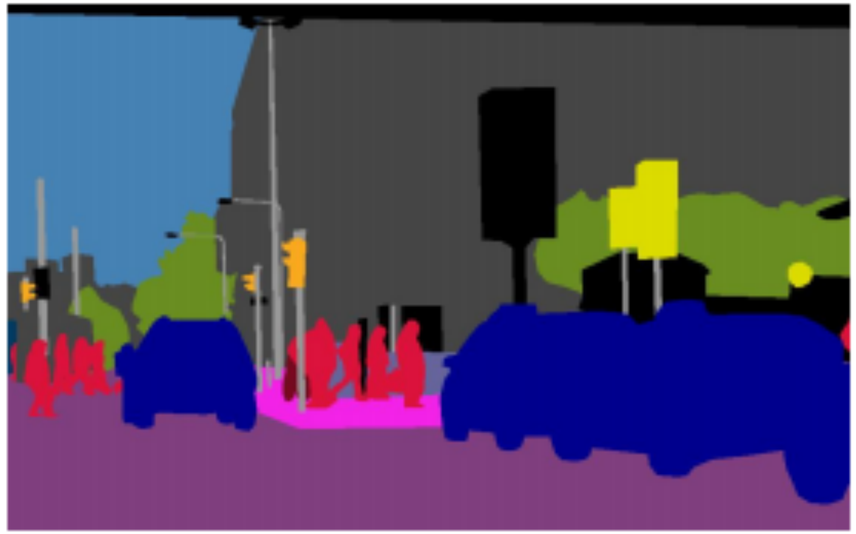
Proposal-free

1. Semantic segmentation (optional)



2. Group pixels into instances

Proposal-free methods

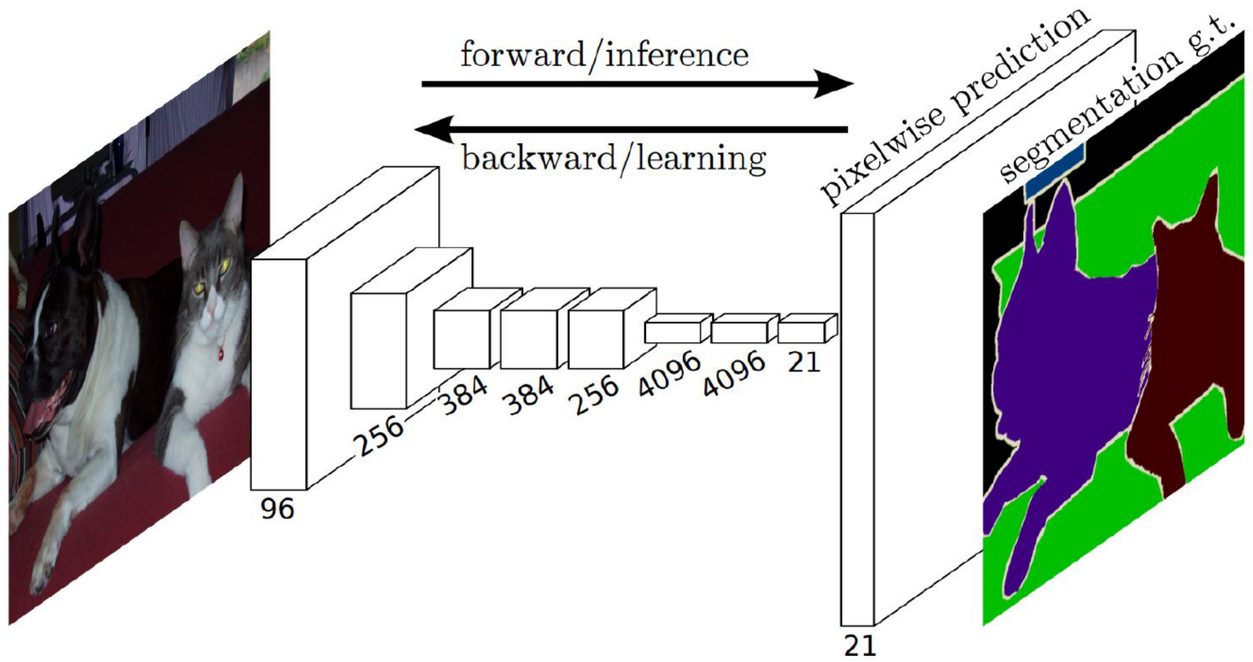


A semantic map

We already know how to obtain this!

Why proposal-free?

- Fully Convolutional Networks for Semantic Segmentation



Long et al., (2015)

SOLOv2

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

$$Y = KX$$

Pixelwise class scores

$[C \times HW]$

C class

Layer parameters
(1x1 conv)

$[C \times D]$

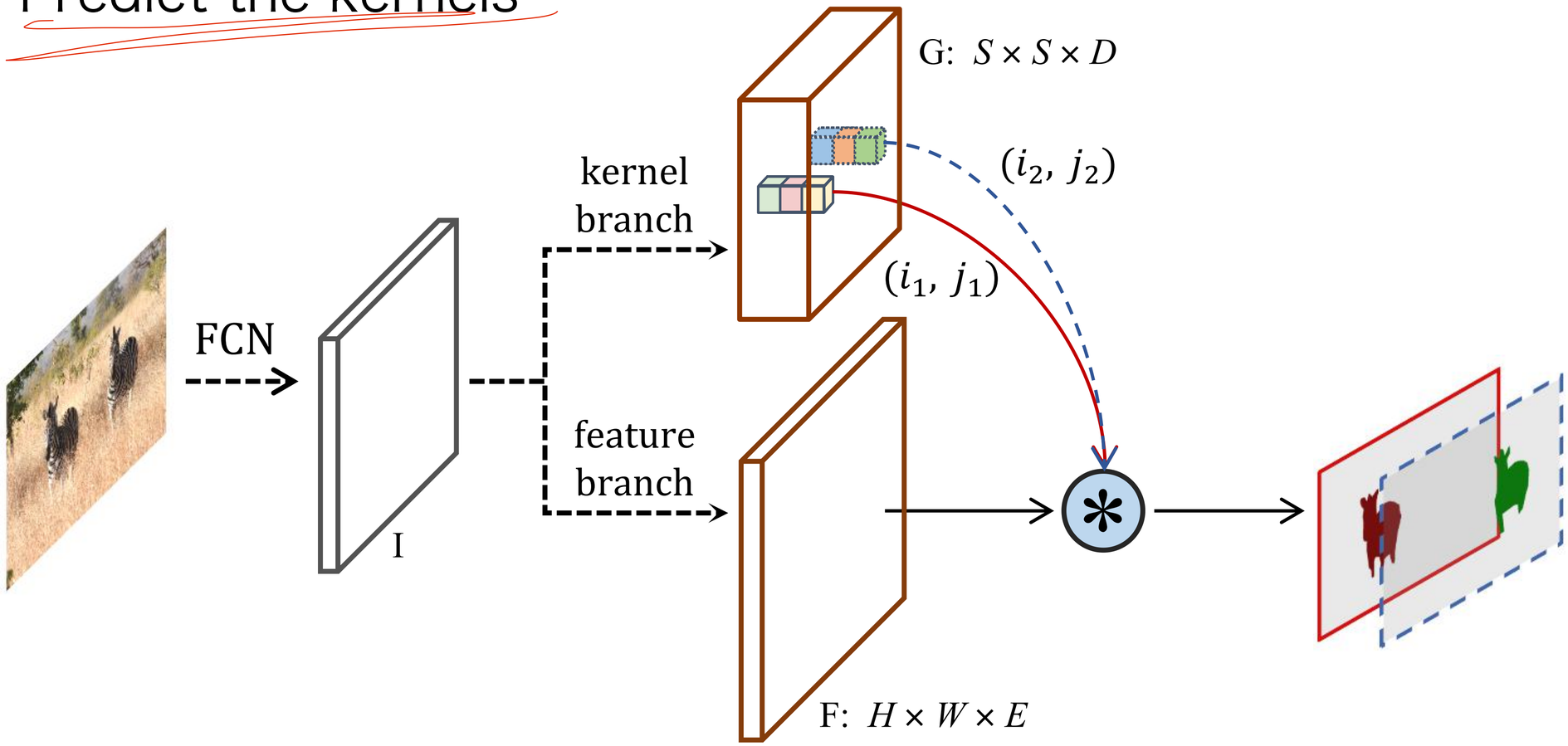
Features

$[D \times HW]$

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

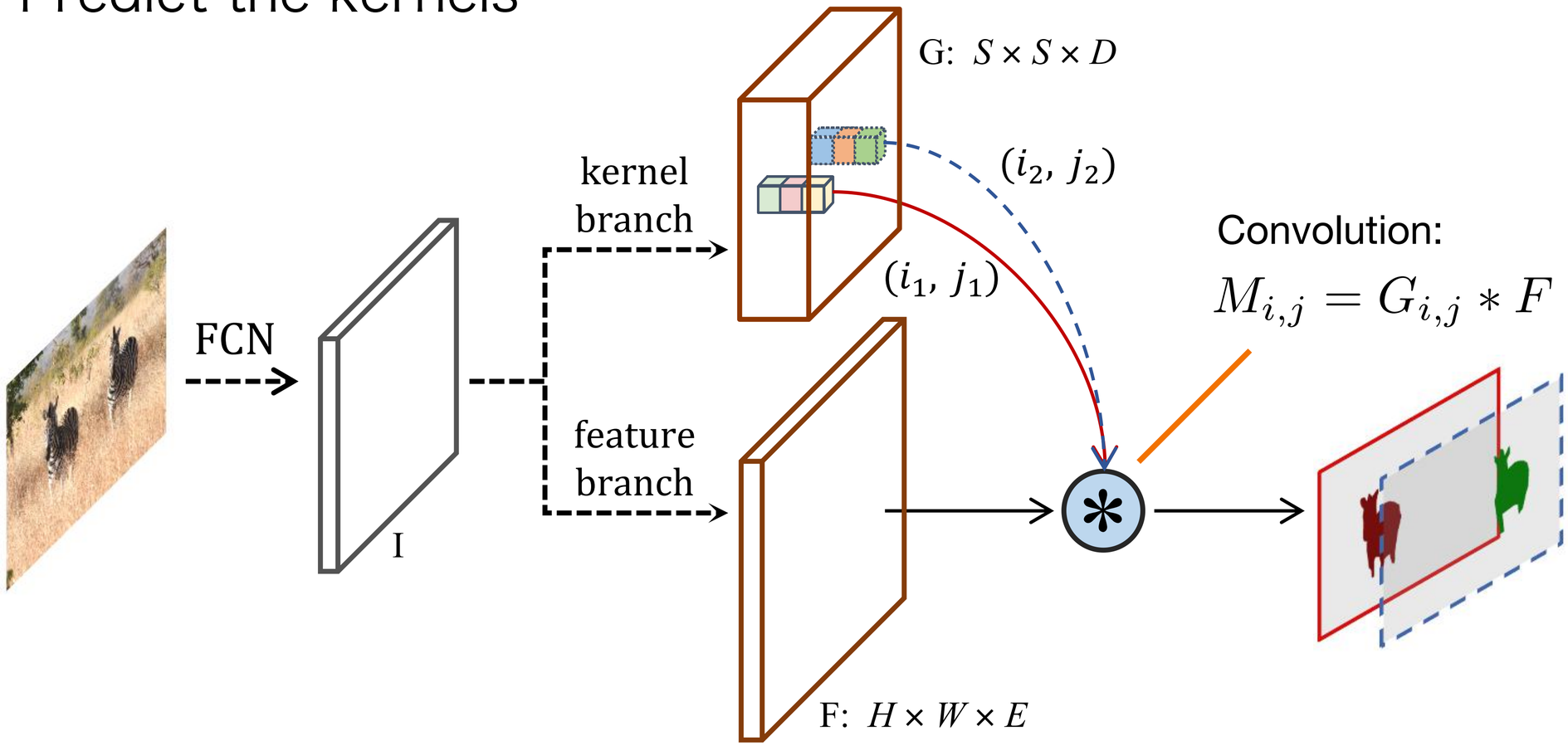
SOLOv2

- Idea: Predict the kernels



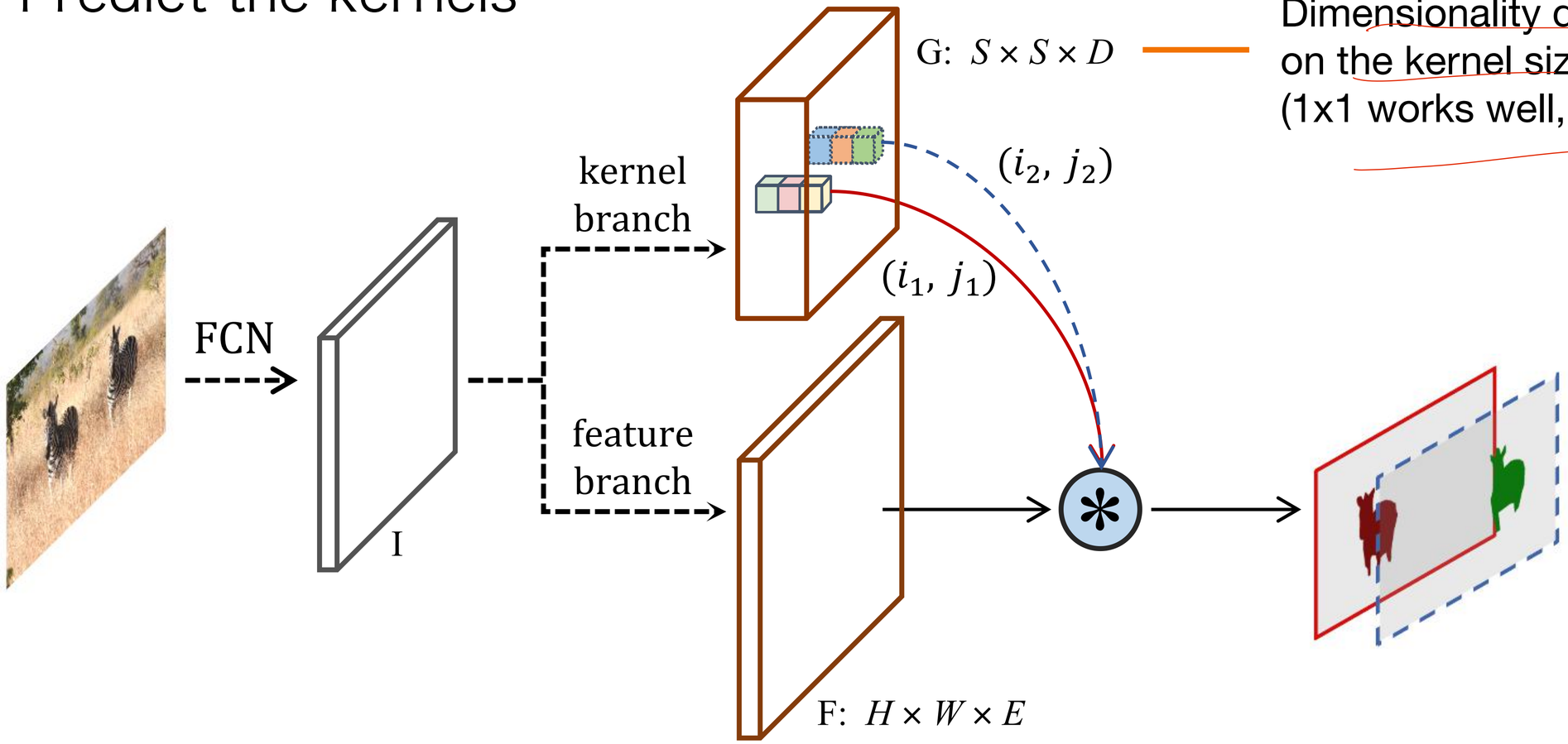
SOLOv2

- Idea: Predict the kernels



SOLOv2

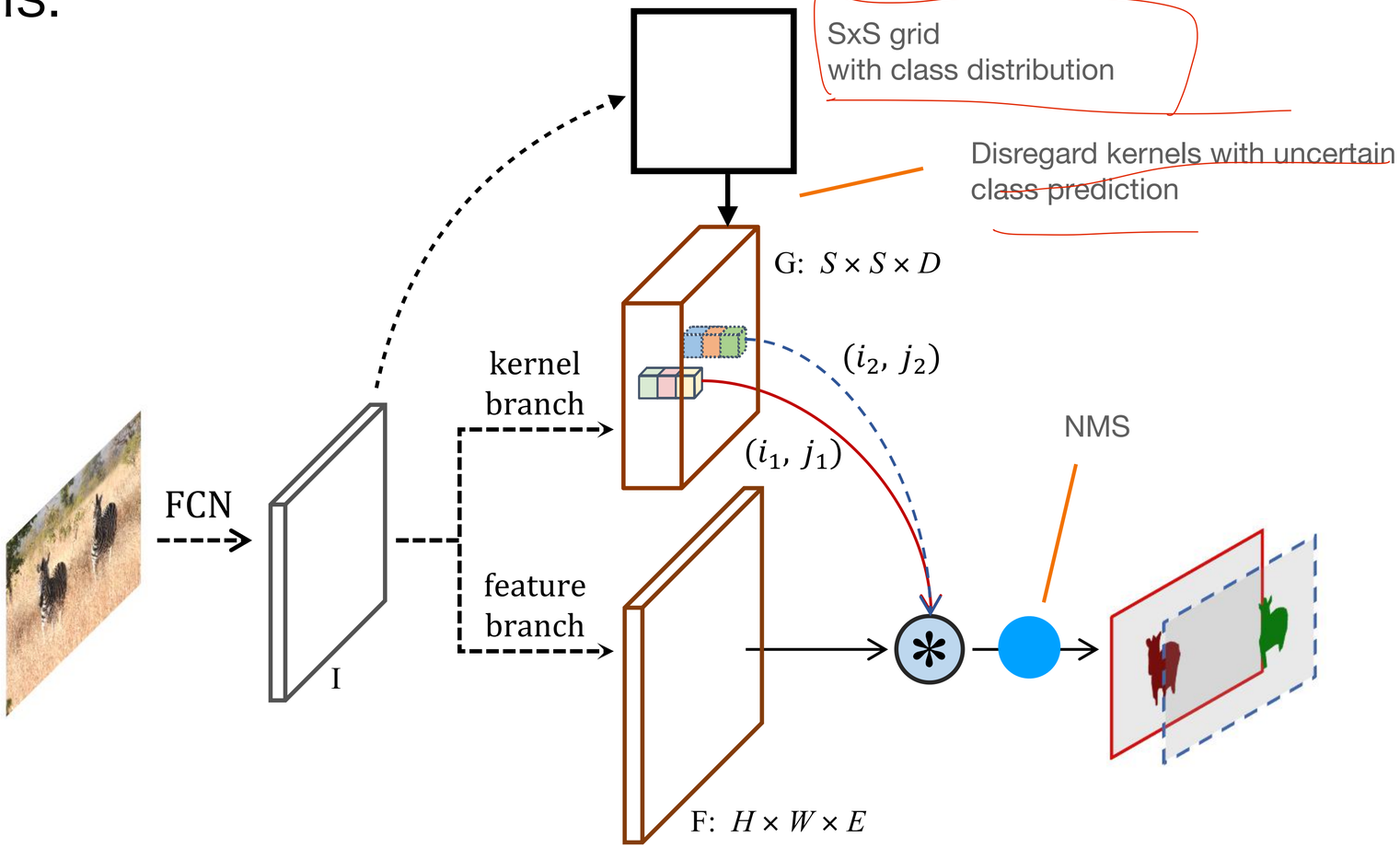
- Idea: Predict the kernels



Dimensionality depends on the kernel size (1x1 works well, so D = E)

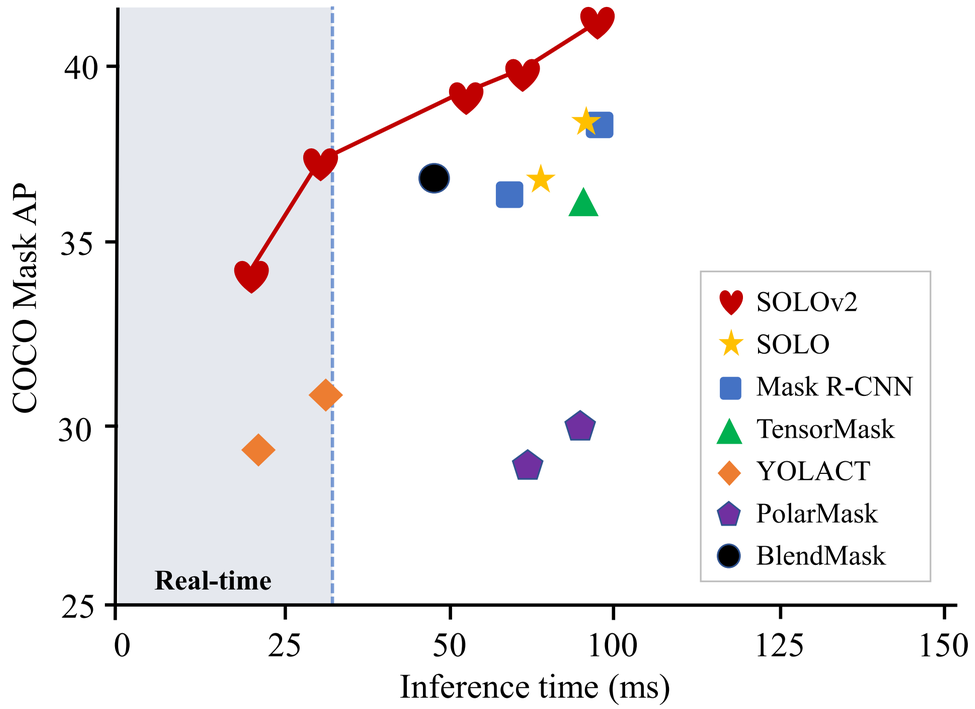
SOLOv2

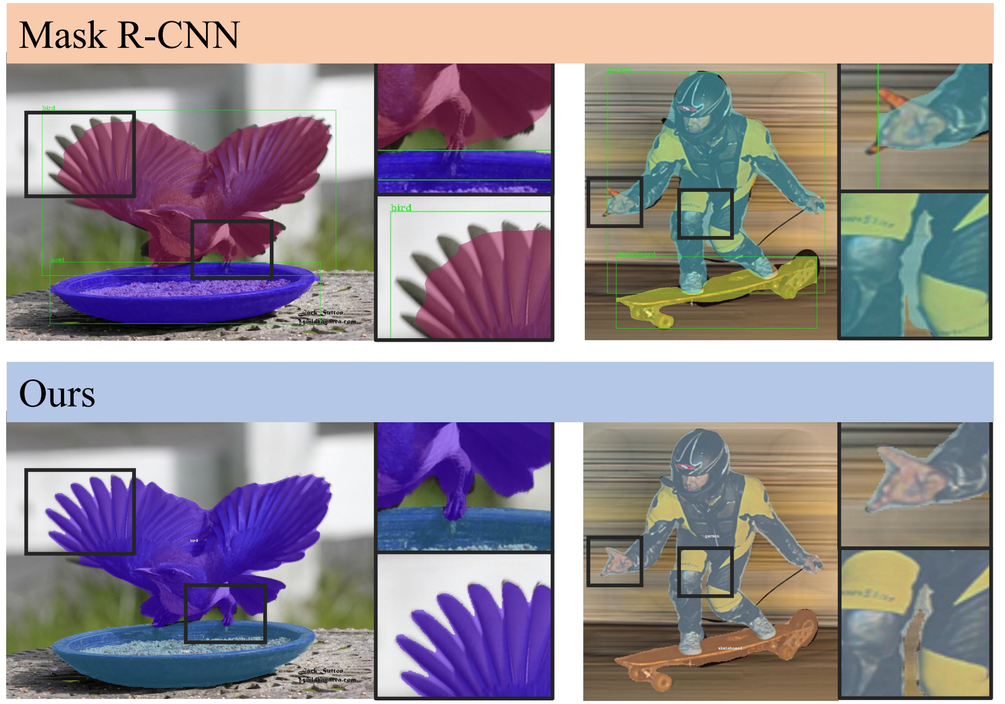
- A few details:



SOLOv2

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





Proposal-free methods

Graph- or cluster-based:

- 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

End-to-end deep nets:

- 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).

Proposal-free methods

Recurrent approaches:

- 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.

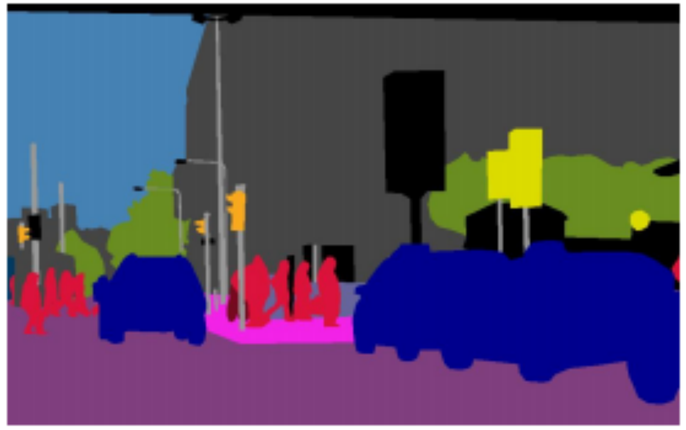
Instance segmentation: Summary

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

Panoptic segmentation

Panoptic segmentation

Semantic segmentation



(e.g. FCN, DeepLab)

+

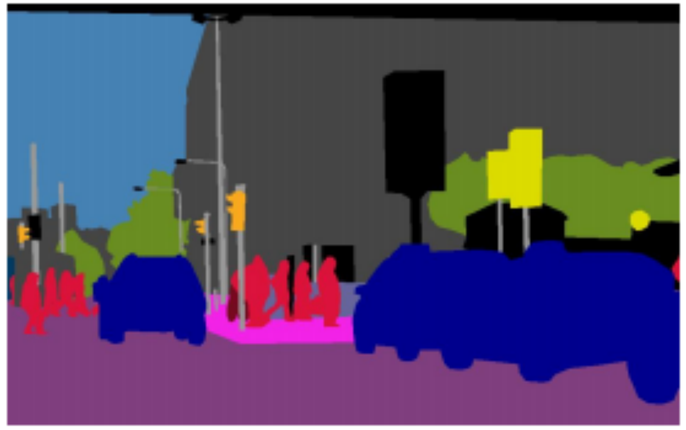
Instance segmentation

(e.g. Mask R-CNN)

Panoptic segmentation

label every pixel

Semantic segmentation



(e.g. FCN, DeepLab)

+

Instance segmentation

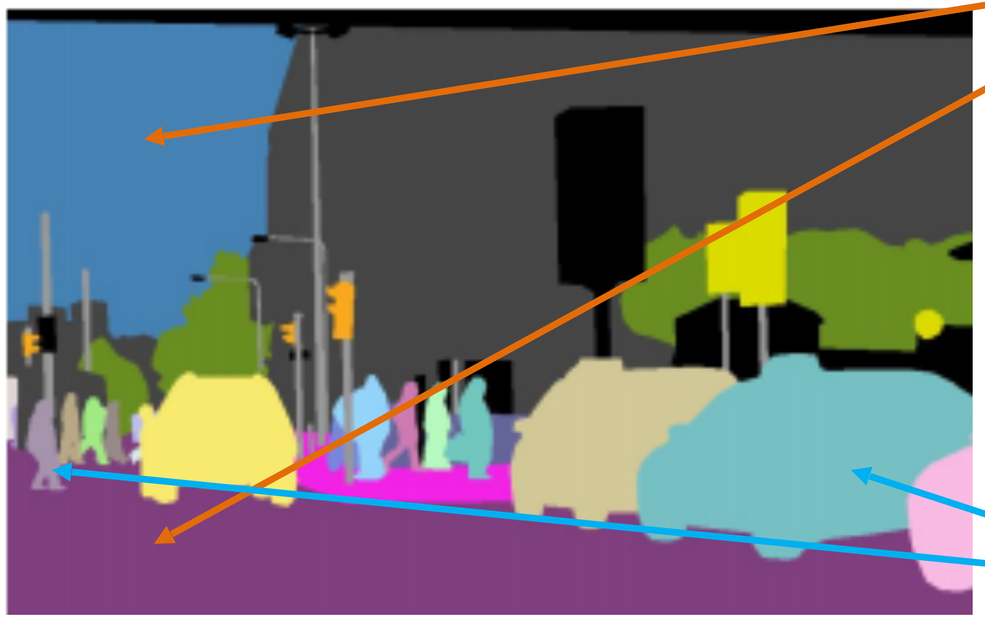
(e.g. Mask R-CNN)

=

Panoptic segmentation

(e.g. Panoptic FPN)

Panoptic segmentation



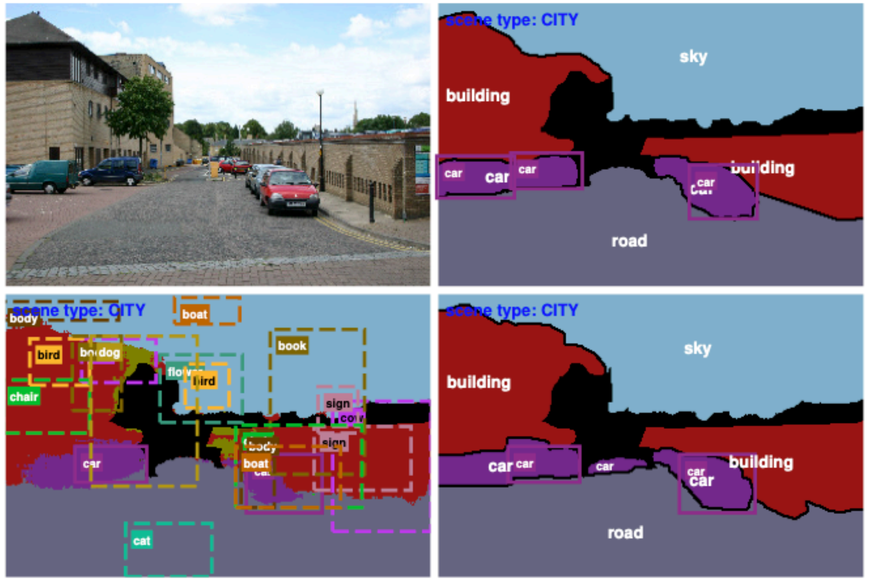
It gives labels to uncountable objects called "stuff" (sky, road, etc), similar to FCN-like networks.

It differentiates between pixels coming from different instances of the same class (countable objects) called "things" (cars, pedestrians, etc).

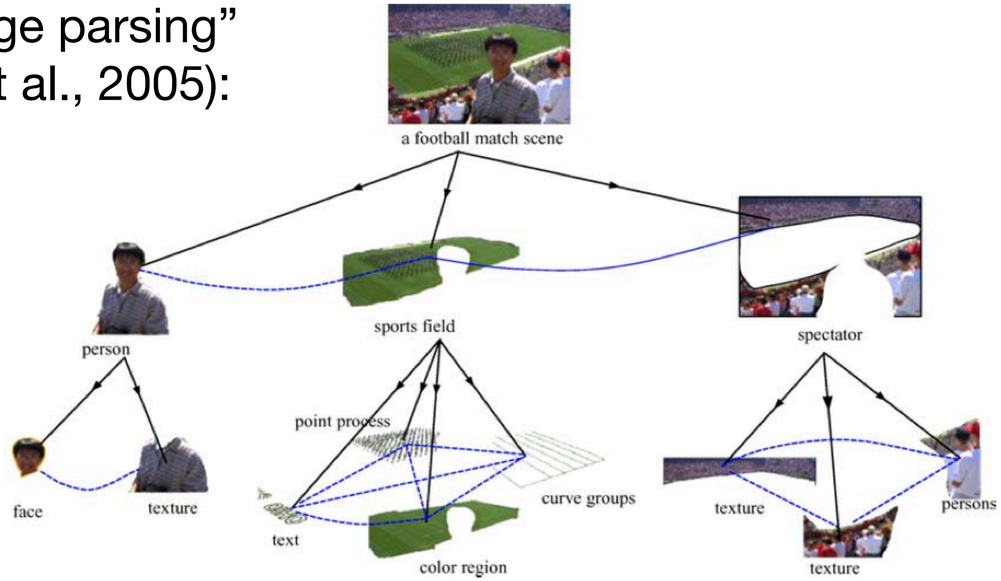
Back in the day

- The task is not new...

"Holistic scene understanding"
(Yao et al., 2012):



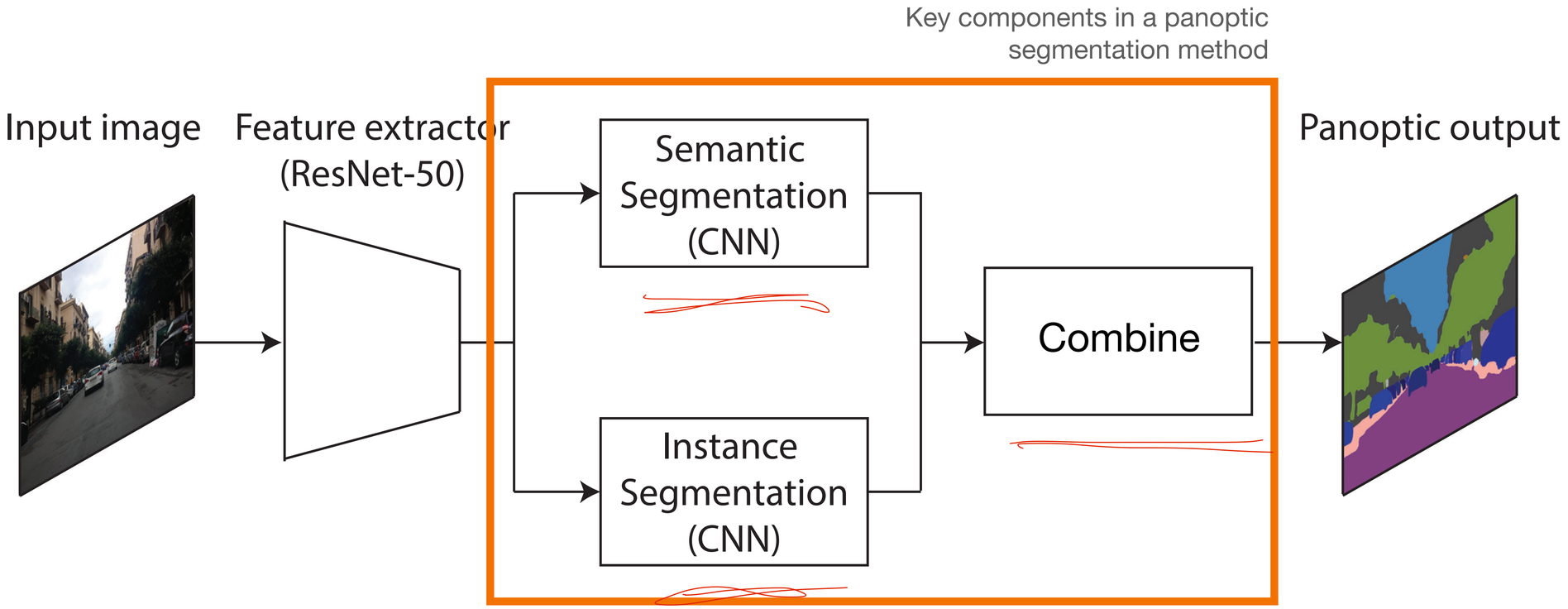
"Image parsing"
(Tu et al., 2005):



... but deep learning makes it feasible.

Overview

- Typical architecture:



Adapted from [de Geus et al., 2018].

Panoptic segmentation

Challenges:

- Can we harmonise architectures for predicting “stuff” and “things”?
  - semantic and instance segmentation pipelines are yet very different.
- Can we improve computational efficiency via parameter sharing?

Two broad categories:

- Top-down: typically two-stage proposal-based.
- Bottom-up: learn suitable feature representation for grouping pixels.

Overview

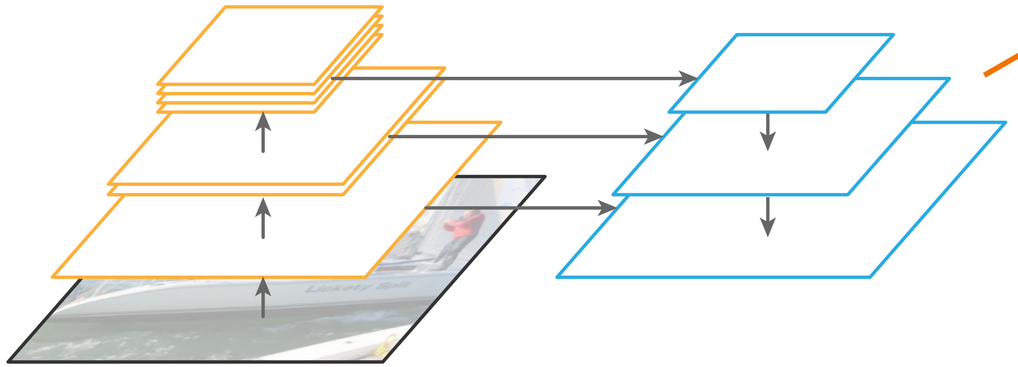
- Kirillov et al., “Panoptic Feature Pyramid Networks”, CVPR 2019.
- Xiong et al., “UPSNet: A Unified Panoptic Segmentation Network”, CVPR 2019.
- Cheng et al., “Panoptic-DeepLab: A Simple, Strong, and Fast Baseline for Bottom-Up Panoptic Segmentation”, CVPR 2020.
- Li et al., “Fully Convolutional Networks for Panoptic Segmentation”, CVPR 2021.

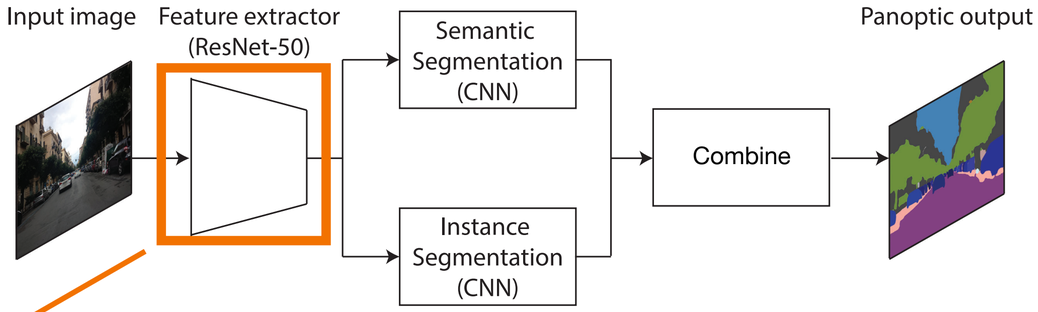
Overview

- **Kirillov et al., “Panoptic Feature Pyramid Networks”, CVPR 2019.**
- Xiong et al., “UPSNet: A Unified Panoptic Segmentation Network”, CVPR 2019.
- Cheng et al., “Panoptic-DeepLab: A Simple, Strong, and Fast Baseline for Bottom-Up Panoptic Segmentation”, CVPR 2020.
- Li et al., “Fully Convolutional Networks for Panoptic Segmentation”, CVPR 2021.

Panoptic FPN

- Feature pyramid backbone:

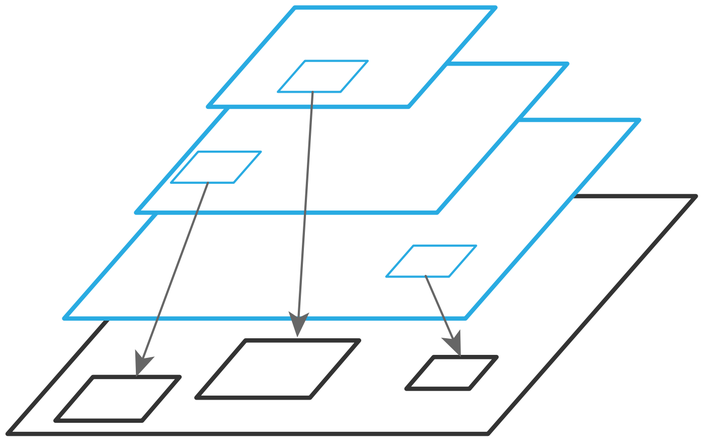


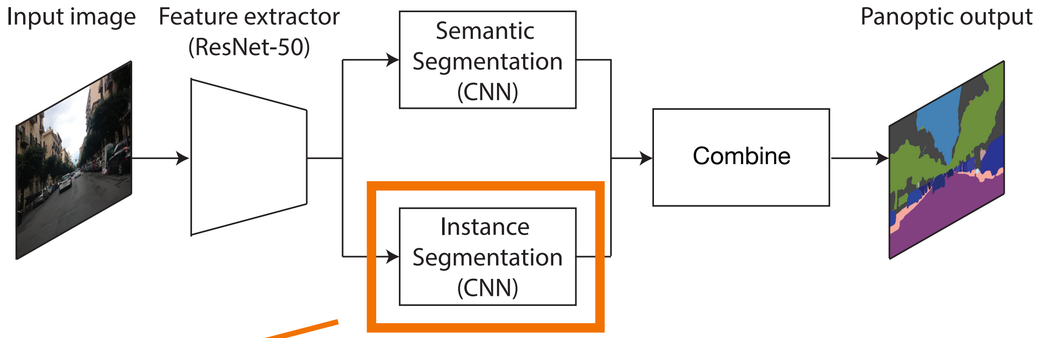


Kirillov et al., “Panoptic Feature Pyramid Networks”. CVPR 2019

Panoptic FPN

- Mask R-CNN for instance segmentation

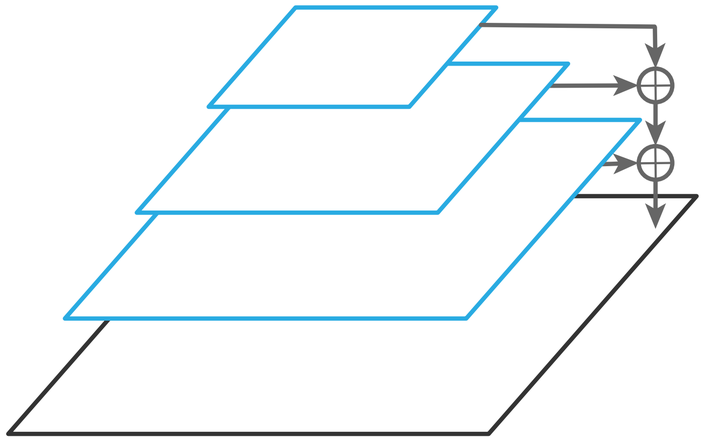


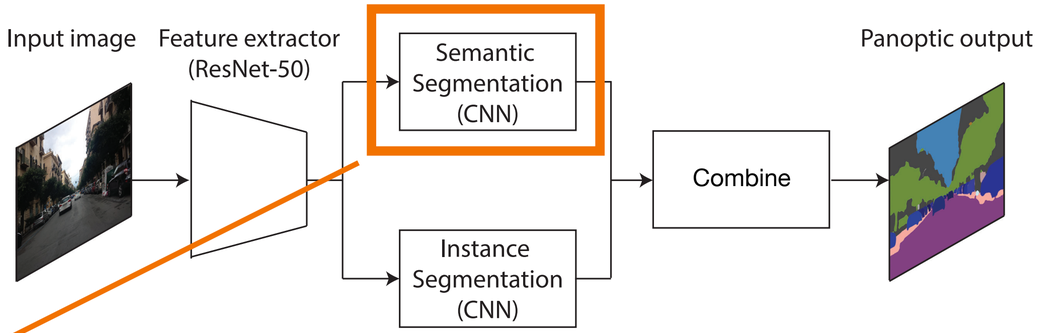


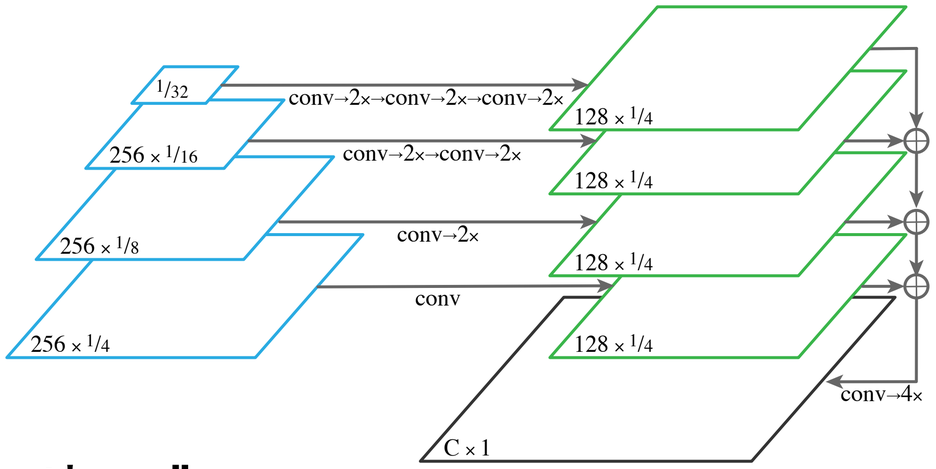
Kirillov et al., “Panoptic Feature Pyramid Networks”. CVPR 2019

Panoptic FPN

- Semantic segmentation decoder:





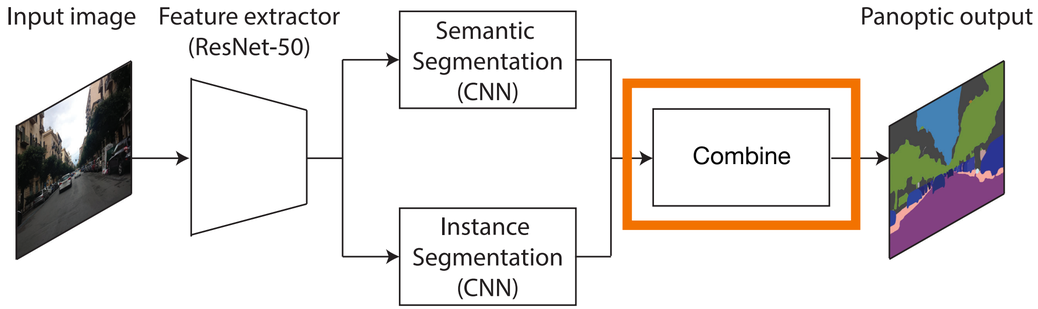


- Replace things classes with 1 class “other”

Kirillov et al., “Panoptic Feature Pyramid Networks”. CVPR 2019

Panoptic FPN

- Merge things and stuff:
  - NMS on instances.
  - Resolve stuff-things conflicts in favour of things. (Why?)
  - Remove any stuff regions labelled “other” or with a small area.



Kirillov et al., “Panoptic Feature Pyramid Networks”. CVPR 2019

Panoptic FPN

- Loss function:

Trade-off hyperparameter

$$L = L_c + L_b + L_m + \lambda_s L_s$$

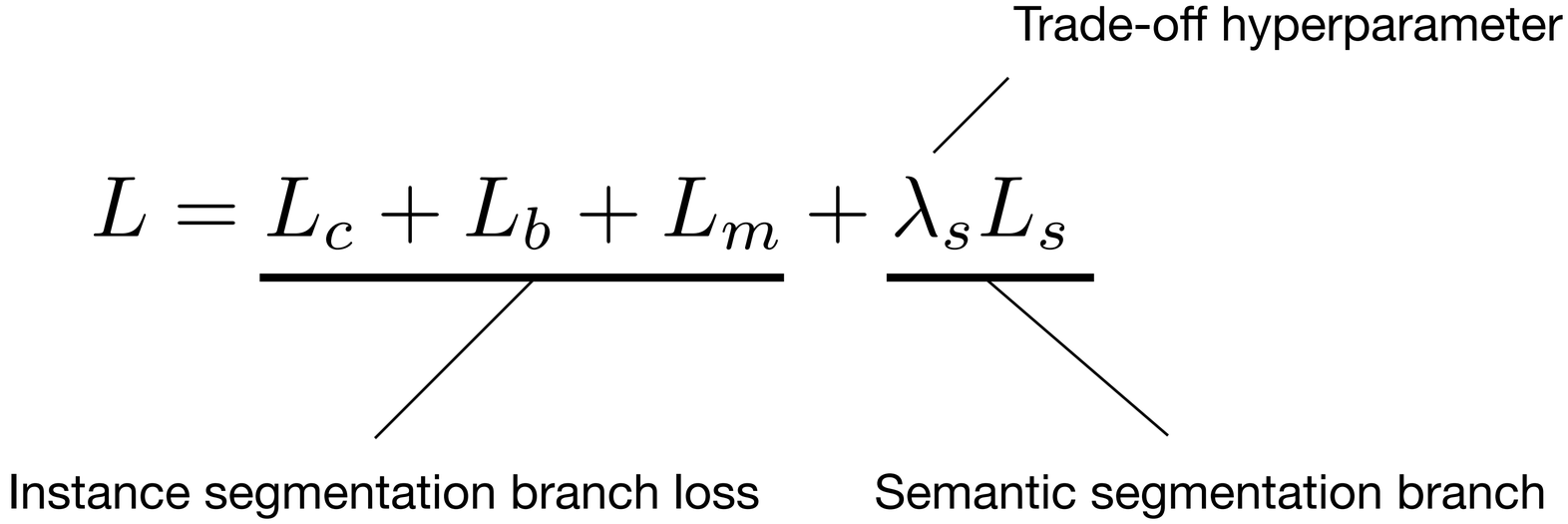
Instance segmentation branch loss

Semantic segmentation branch

Kirillov et al., “Panoptic Feature Pyramid Networks”. CVPR 2019

Panoptic FPN

- Loss function:



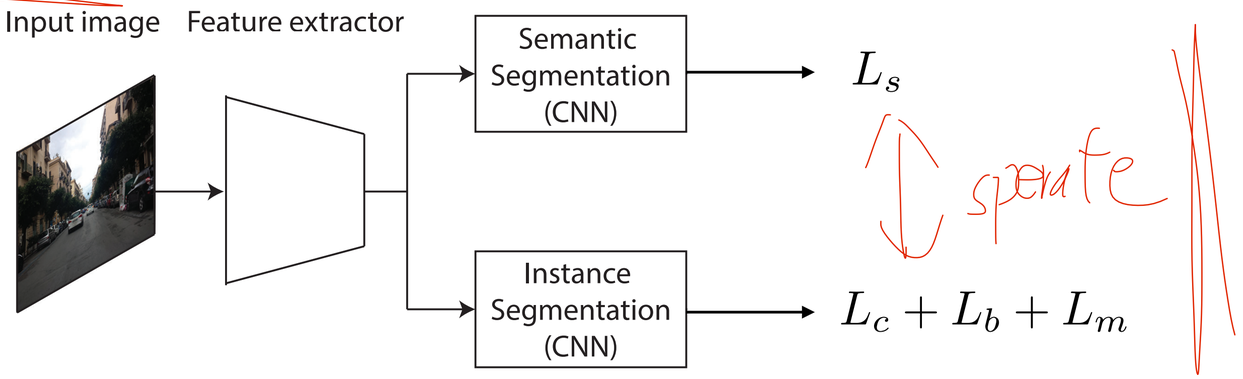
- Remark: Training with multiple loss terms (“multi-task learning”) can be challenging, as different loss terms may “compete” for desired feature representation.

Kirillov et al., “Panoptic Feature Pyramid Networks”. CVPR 2019

Kirillov et al., “Panoptic Feature Pyramid Networks”. CVPR 2019

Panoptic FPN: Summary

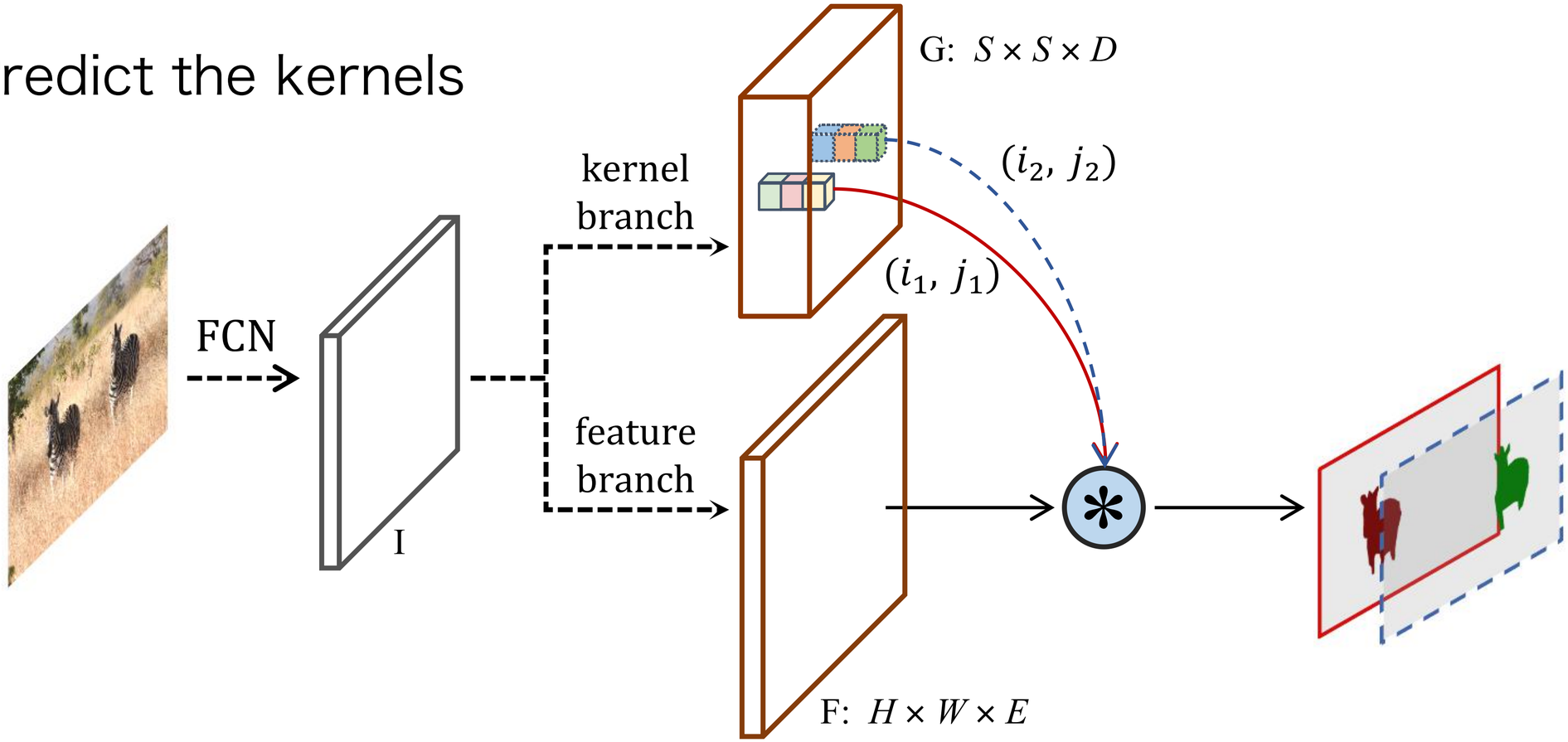
- Simple heuristics for merging things and stuff
- The instance and semantic segmentation branches are treated independently
  - i.e. semantic segmentation branch receives no gradient from instance supervision and vice-versa.



Even simpler?

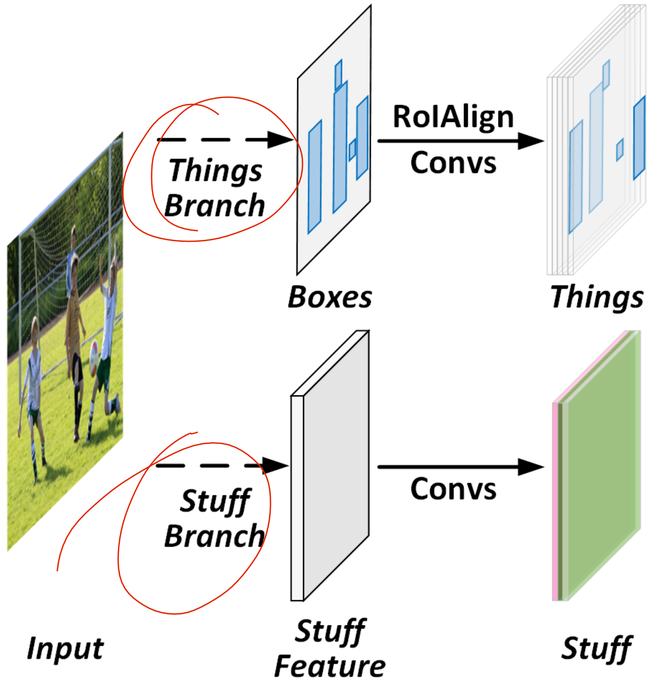
Recall SOLOv2:

Idea: Predict the kernels

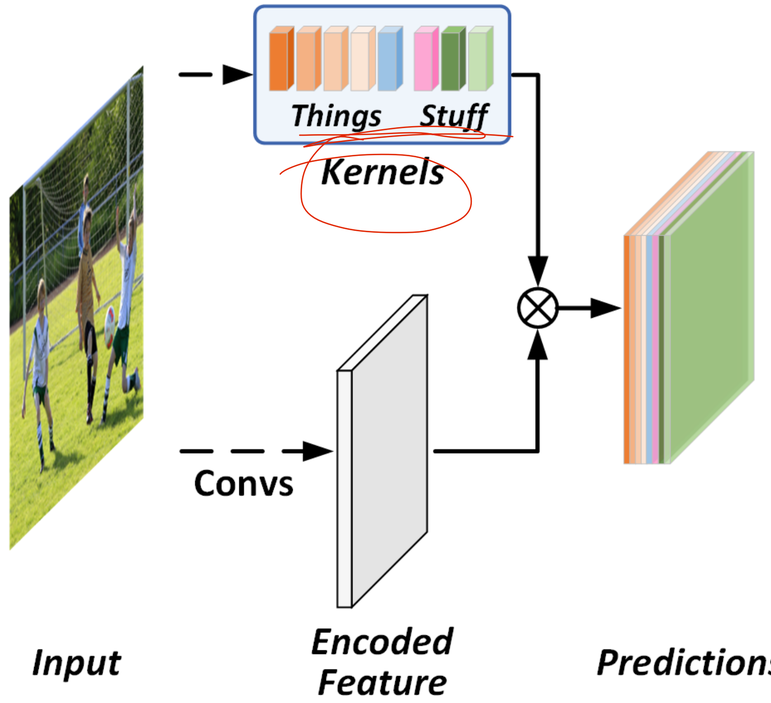


Panoptic FCN

Panoptic FPN

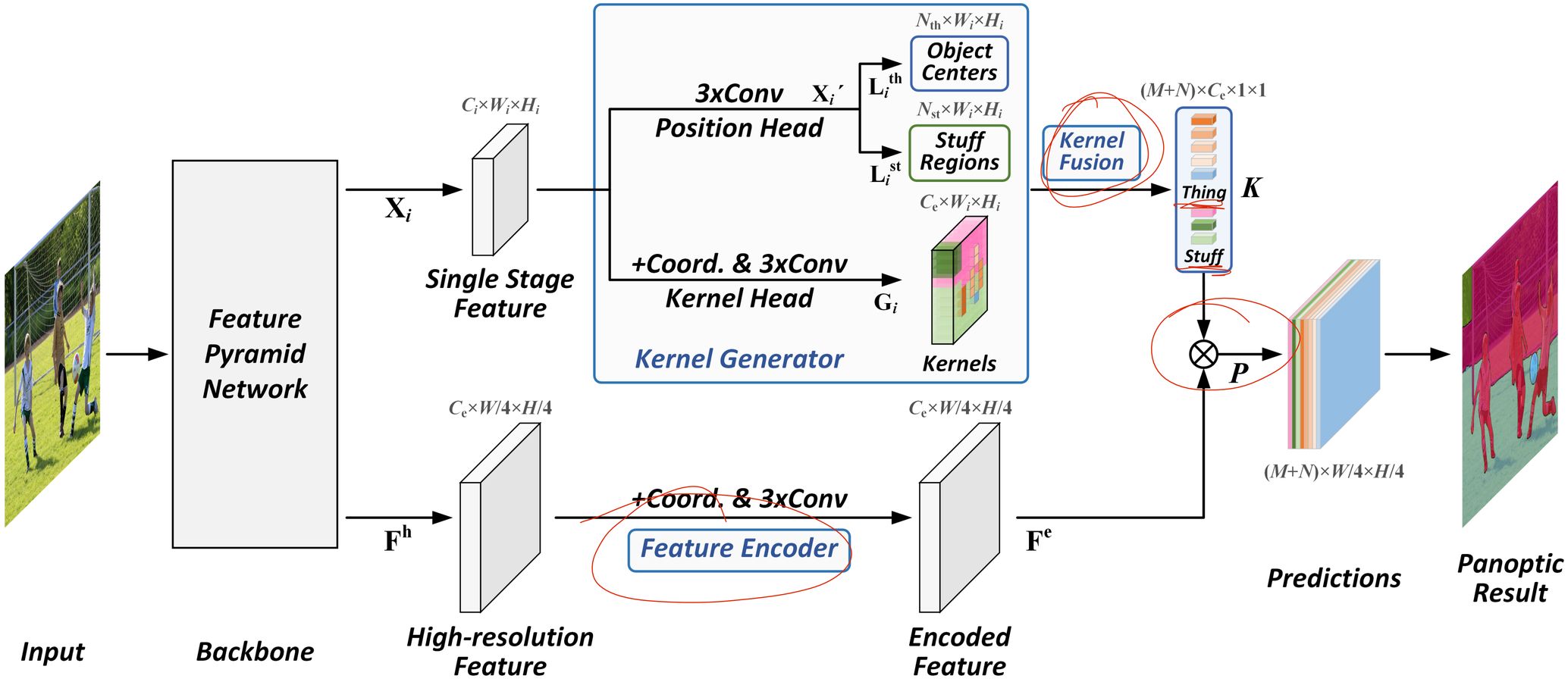


Panoptic FCN (Fully Convolutional Network)



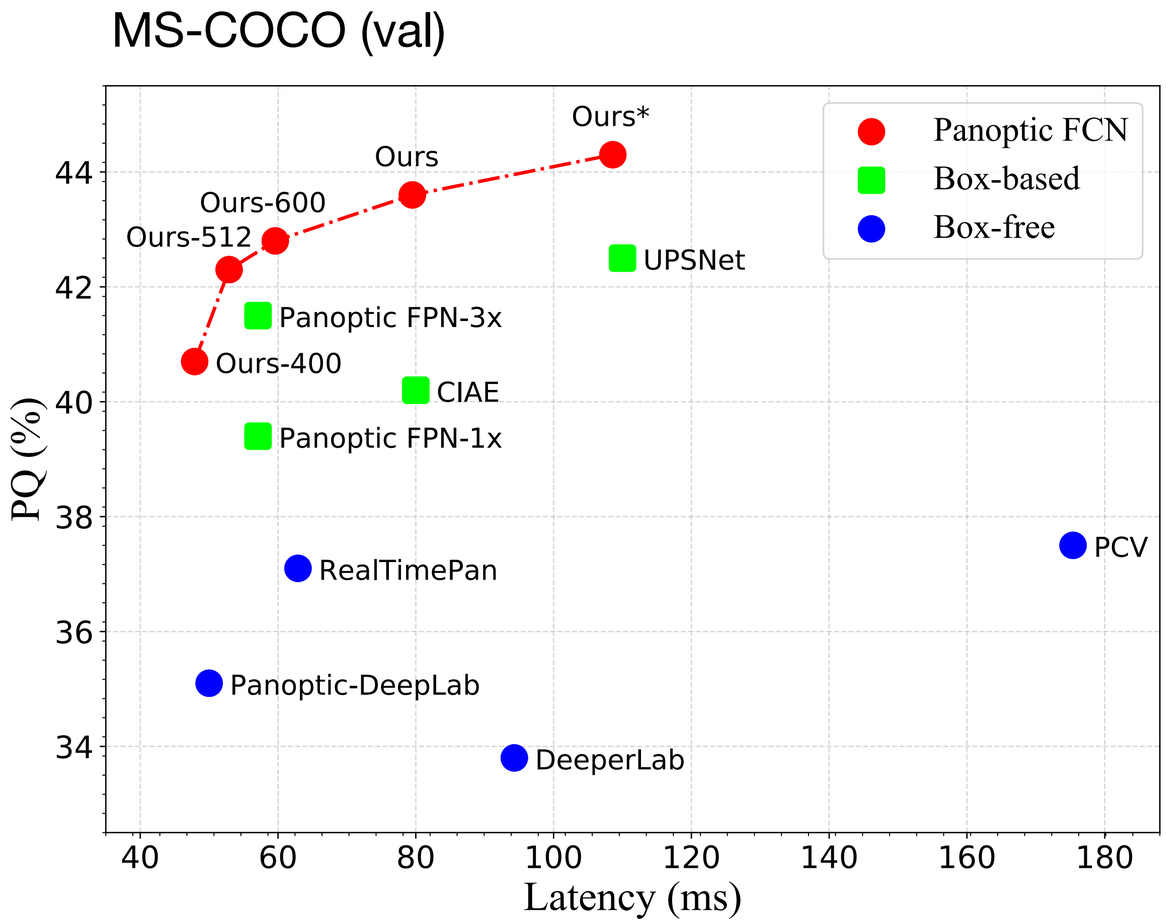
Li et al., “Fully Convolutional Networks for Panoptic Segmentation”, CVPR 2021.

Panoptic FCN



Li et al., “Fully Convolutional Networks for Panoptic Segmentation”, CVPR 2021.

Panoptic FCN



- Improved efficiency and accuracy
- Simpler architecture

Li et al., “Fully Convolutional Networks for Panoptic Segmentation”, CVPR 2021.

Panoptic FCN: Qualitative examples

Further reading

With CNNs:

- Xiong et al., "UPSnet: A unified panoptic segmentation network." CVPR 2019.
- Wang et al., “Axial-DeepLab: Stand-alone axial-attention for panoptic segmentation.” ECCV 2020.
- Wang et al., “MaX-DeepLab: End-to-end panoptic segmentation with mask transformers.” CVPR 2021.

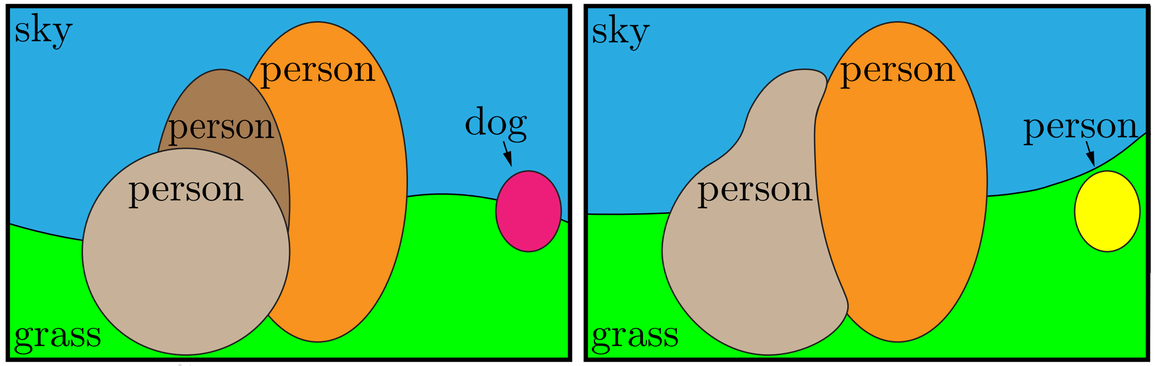
With Transformer networks (later in the course):

- Cheng et al., “Masked-attention Mask Transformer for Universal Image Segmentation.” CVPR 2022.

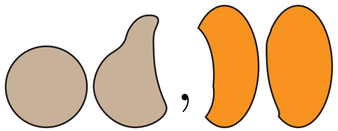
Evaluating panoptic segmentation

Panoptic quality (PQ)

- Example:



Ground truth Prediction

Person — TP: {  }; FN: {  }; FP: {  }

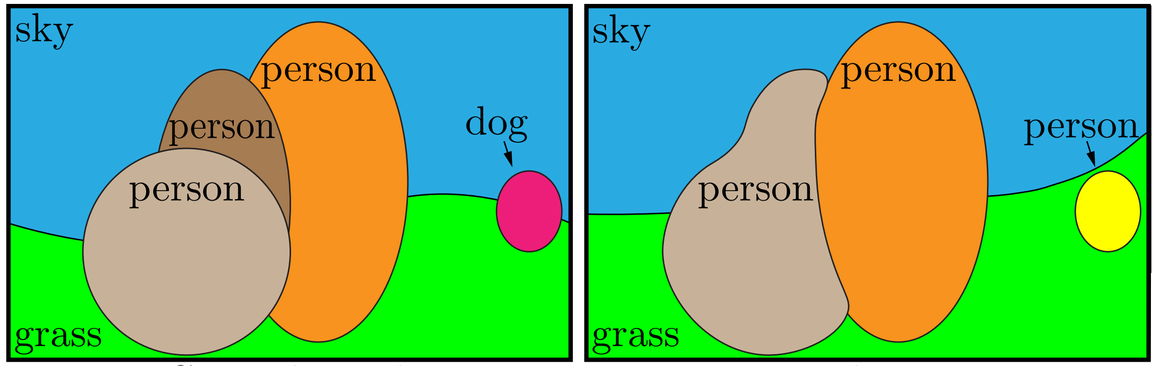
TP = True positive, FN = False negative, FP = false positive

- Wait, but don't we need to define an IoU threshold?

Kirillov et al., "Panoptic Segmentation". CVPR 2019.

Panoptic quality (PQ)

- To compute PQ we specify that a prediction and a ground truth match only if their IoU is greater than 0.5.
- This match, if found, is **unique**.
- Unique matching theorem:
  - A ground-truth segment has an IoU greater than 0.5 with at most **one** prediction.



Ground truth Prediction

Person — TP: { , }; FN: { }; FP: { }

proof hint

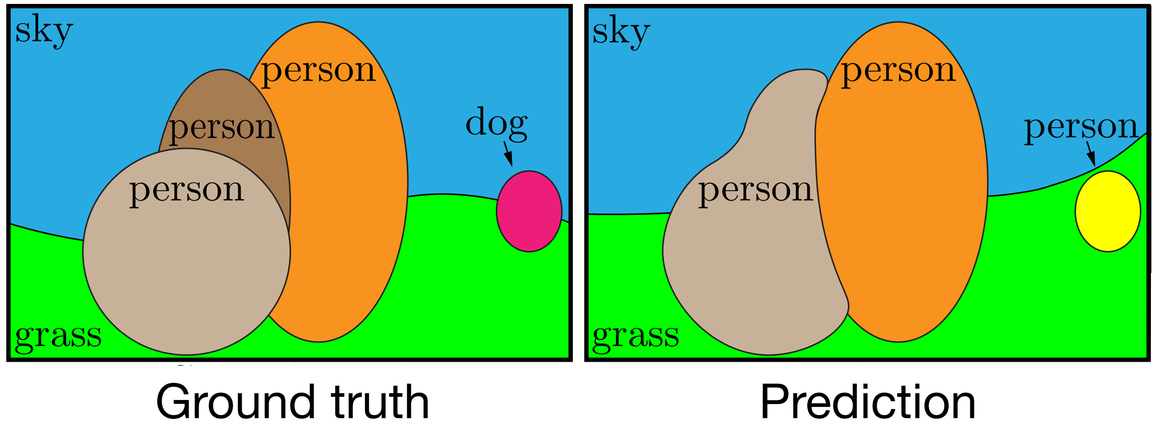
- With p_1 and p_2 being disjoint:

$$\mathrm{IoU}(p_1, g) + \mathrm{IoU}(p_2, g) \leq 1$$

Kirillov et al., “Panoptic Segmentation”. CVPR 2019.

Panoptic quality (PQ)

1. Establish matches between the ground-truth and predictions;
2. Count TPs, FPs and FNs;
3. Compute PQ for each class:



Ground truth Prediction

Person — TP: { , }; FN: { }; FP: { }

$$PQ = \frac{\sum_{(p,g)\in \mathrm{TP}} \mathrm{IoU}(p,g)}{|\ \mathrm{TP}\ |} \frac{|\ \mathrm{TP}\ |}{|\ \mathrm{TP}\ | + \frac{1}{2}|\ \mathrm{FP}\ | + \frac{1}{2}|\ \mathrm{FN}\ |}$$

...and then average.

Kirillov et al., “Panoptic Segmentation”. CVPR 2019.

Panoptic quality (PQ)

$$PQ = \frac{\sum_{(p,g)\in \mathrm{TP}} \mathrm{IoU}(p,g)}{|\ \mathrm{TP}\ |} \frac{|\ \mathrm{TP}\ |}{|\ \mathrm{TP}\ | + \frac{1}{2}|\ \mathrm{FP}\ | + \frac{1}{2}|\ \mathrm{FN}\ |}$$

Kirillov et al., “Panoptic Segmentation”. CVPR 2019.

Panoptic quality (PQ)

$$PQ = \underbrace{\frac{\sum_{(p,g)\in \mathrm{TP}} \mathrm{IoU}(p,g)}{|\ \mathrm{TP}\ |}}_{\mathrm{SQ}} \frac{|\ \mathrm{TP}\ |}{|\ \mathrm{TP}\ | + \frac{1}{2}|\ \mathrm{FP}\ | + \frac{1}{2}|\ \mathrm{FN}\ |}$$

- SQ = “Segmentation Quality”:
  - Average mask IoU for true positives;
  - Measures pixel-level accuracy of predicted masks.

Kirillov et al., “Panoptic Segmentation”. CVPR 2019.

Panoptic quality (PQ)

$$PQ = \frac{\sum_{(p,g)\in \mathrm{TP}} \mathrm{IoU}(p,g)}{|\ \mathrm{TP}\ |} \boxed{\frac{|\ \mathrm{TP}\ |}{|\ \mathrm{TP}\ | + \frac{1}{2}|\ \mathrm{FP}\ | + \frac{1}{2}|\ \mathrm{FN}\ |}}$$

RQ

- RQ = “Recognition Quality”:
  - Object-level accuracy.
  - Does it look familiar? (QUIZ)

Kirillov et al., “Panoptic Segmentation”. CVPR 2019.

Panoptic quality (PQ)

$$PQ = \frac{\sum_{(p,g)\in \mathrm{TP}} \mathrm{IoU}(p,g)}{|\ \mathrm{TP}\ |} \boxed{\frac{|\ \mathrm{TP}\ |}{|\ \mathrm{TP}\ | + \frac{1}{2}|\ \mathrm{FP}\ | + \frac{1}{2}|\ \mathrm{FN}\ |}}$$

RQ

- RQ = “Recognition Quality”:
  - Object-level accuracy.
  - Does it look familiar? This is F-score (F_1)
  - F-score is the harmonic mean of precision and recall.

Kirillov et al., “Panoptic Segmentation”. CVPR 2019.

Panoptic quality (PQ)

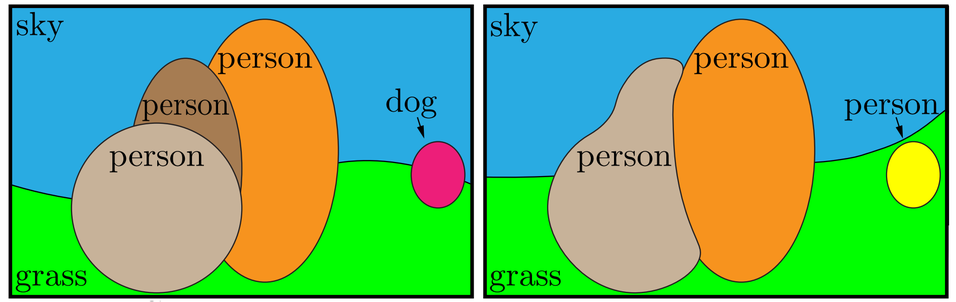
$$PQ = \frac{\sum_{(p,g)\in \mathrm{TP}} \mathrm{IoU}(p,g)}{|\ \mathrm{TP}\ |} \frac{|\ \mathrm{TP}\ |}{|\ \mathrm{TP}\ | + \frac{1}{2}|\ \mathrm{FP}\ | + \frac{1}{2}|\ \mathrm{FN}\ |}$$

- Observation 1: $PQ, RQ, SQ \in [0, 1]$

Kirillov et al., “Panoptic Segmentation”. CVPR 2019.

Panoptic quality (PQ)

$$PQ = \frac{\sum_{(p,g)\in \mathrm{TP}} \mathrm{IoU}(p,g)}{|\ \mathrm{TP}\ |} \frac{|\ \mathrm{TP}\ |}{|\ \mathrm{TP}\ | + \frac{1}{2}|\ \mathrm{FP}\ | + \frac{1}{2}|\ \mathrm{FN}\ |}$$



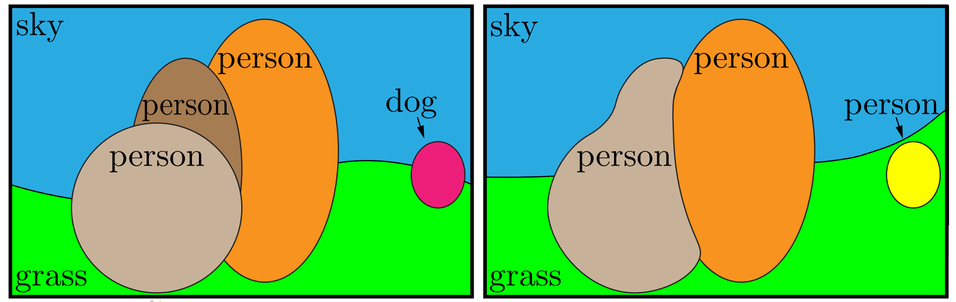
Ground truth Prediction

- Observation 2: What effect does missing one object have on PQ (e.g. “dog” above)?
  - Increment FN for that class (e.g. “dog”) AND FP for another class (e.g. “person”).

Kirillov et al., “Panoptic Segmentation”. CVPR 2019.

Panoptic quality (PQ)

$$PQ = \frac{\sum_{(p,g)\in \mathrm{TP}} \mathrm{IoU}(p,g)}{|\ \mathrm{TP}\ |} \frac{|\ \mathrm{TP}\ |}{|\ \mathrm{TP}\ | + \frac{1}{2}|\ \mathrm{FP}\ | + \frac{1}{2}|\ \mathrm{FN}\ |}$$



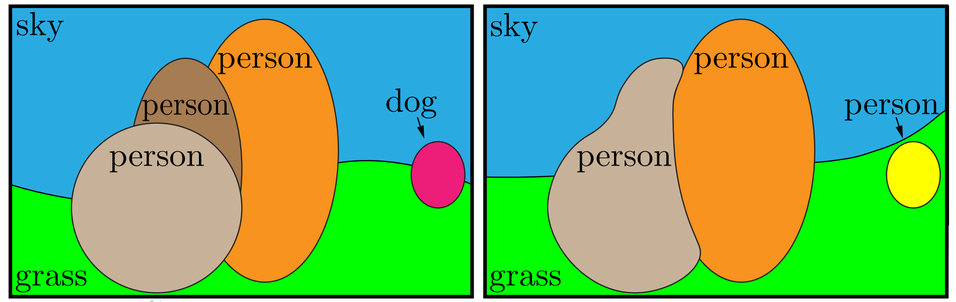
Ground truth Prediction

- Observation 2: What effect does missing one object have on PQ (e.g. “dog” above)?
  - Increment FN for that class (e.g. “dog”) AND FP for another class (e.g. “person”).
  - This reduces PQ for **two** classes.

Kirillov et al., “Panoptic Segmentation”. CVPR 2019.

Panoptic quality (PQ)

$$PQ = \frac{\sum_{(p,g)\in \mathrm{TP}} \mathrm{IoU}(p,g)}{|\ \mathrm{TP}\ |} \frac{|\ \mathrm{TP}\ |}{|\ \mathrm{TP}\ | + \frac{1}{2}|\ \mathrm{FP}\ | + \frac{1}{2}|\ \mathrm{FN}\ |}$$



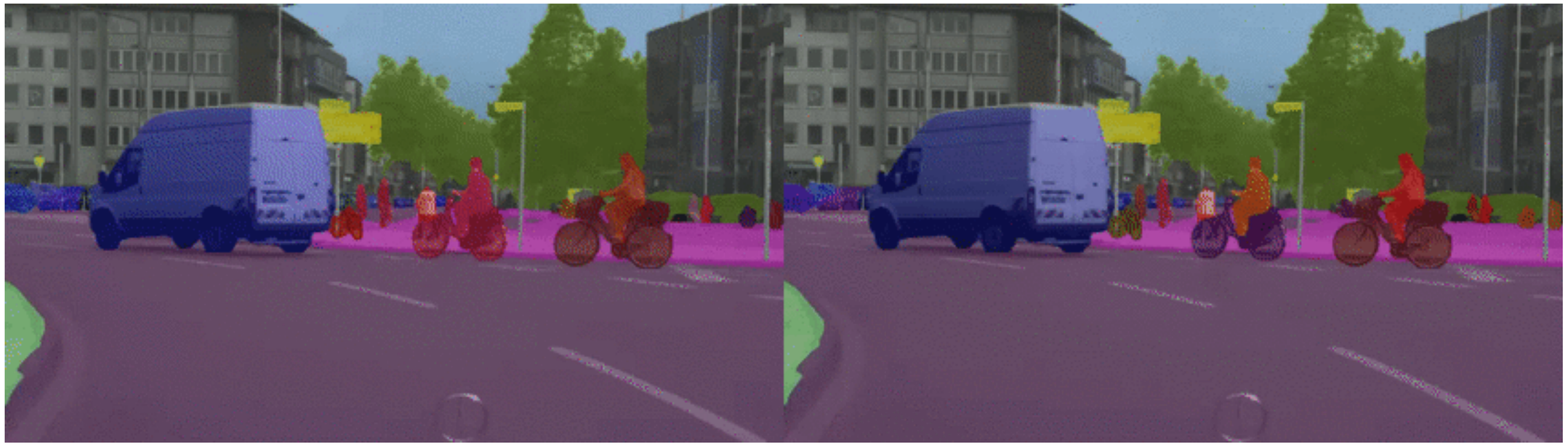
Ground truth Prediction

- Observation 2: What effect does missing one object have on PQ (e.g. “dog” above)?
  - Increment FN for that class (e.g. “dog”) AND FP for another class (e.g. “person”).
  - This reduces PQ of **two** classes.
  - Idea: Predict as “unknown” class instead. FP count will not affect another class.

Kirillov et al., “Panoptic Segmentation”. CVPR 2019.

Current research

Video panoptic segmentation (Kim et al., 2020):



Next lecture: Video Object Segmentation

See also:

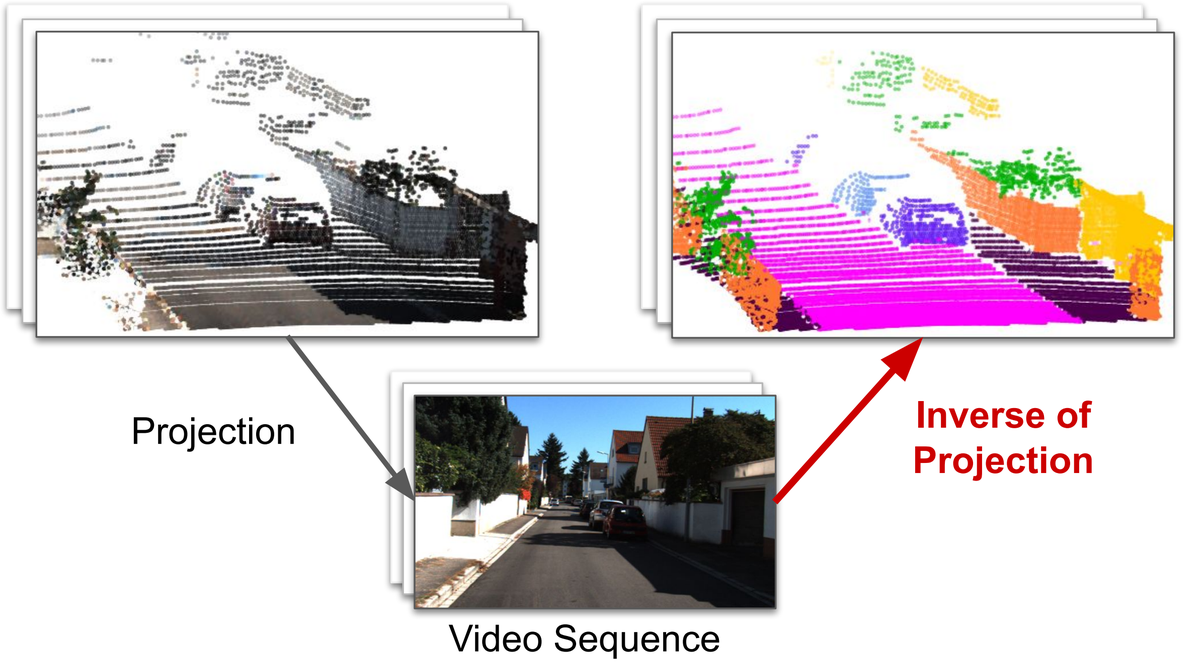
Weber et al., “STEP: Segmenting and Tracking Every Pixel” (2021).

Current research

Extending to other modalities (e.g. depth prediction):

Input: RGB video sequence

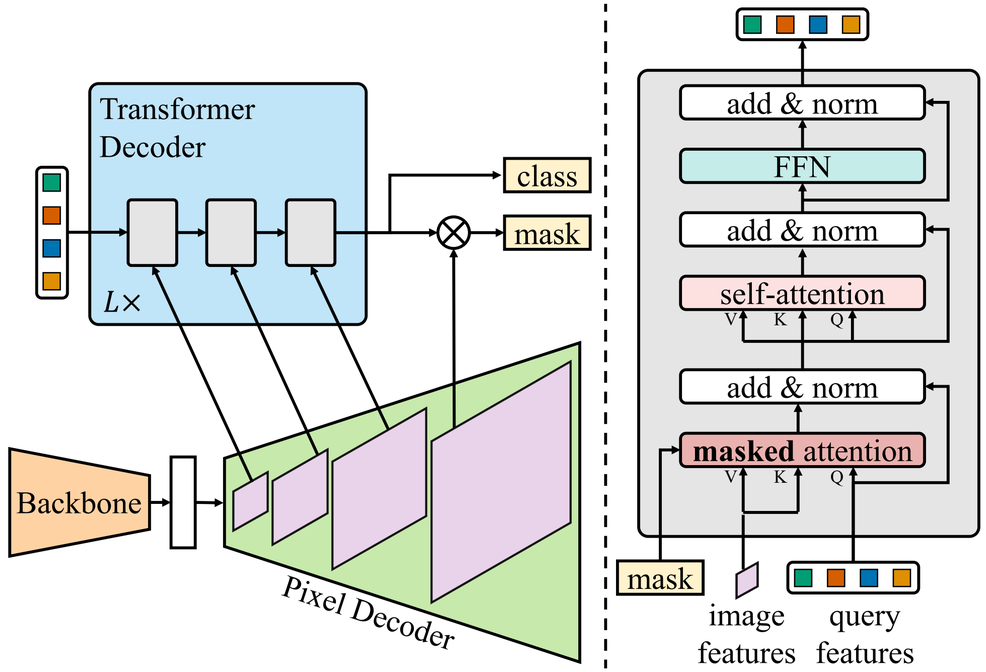
Output: semantic labels (panoptic) + depth

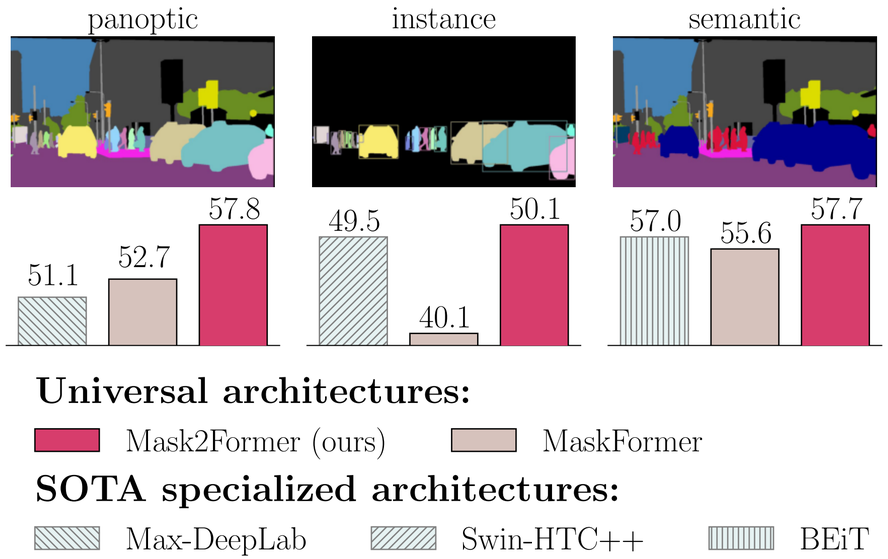


Qiao et al., “ViP-DeepLab: Learning Visual Perception with Depth-aware Video Panoptic Segmentation”, CVPR 2021.

Current research

Improved architecture with Transformers (Mask2Former):





Upcoming: Transformers

Cheng et al., “Masked-attention Mask Transformer for Universal Image Segmentation”, CVPR 2022.

Panoptic segmentation: Summary

- A natural semantic representation:
  - Discriminating pixels in terms of “stuff” and “things”.
- Panoptic annotation is expensive.
  - Even more so if we move to videos.
- Models can be more complex,
  - but there is an effort to unify the architectures (e.g. Panoptic FCN, Mask2Former)