

Lecture 13: Spatial Localization and Image Segmentation

Administrivia

Expect TA feedback on **project proposal** by 9/30

Reminder, that **Homework 2** is due 9/29

Midterm

- Nov 16, in class
- Closed book
- Syllabus includes everything till the Nov. 9 lecture

Happy Dusshera / Vijaydashami (may you defeat the non-converging networks)

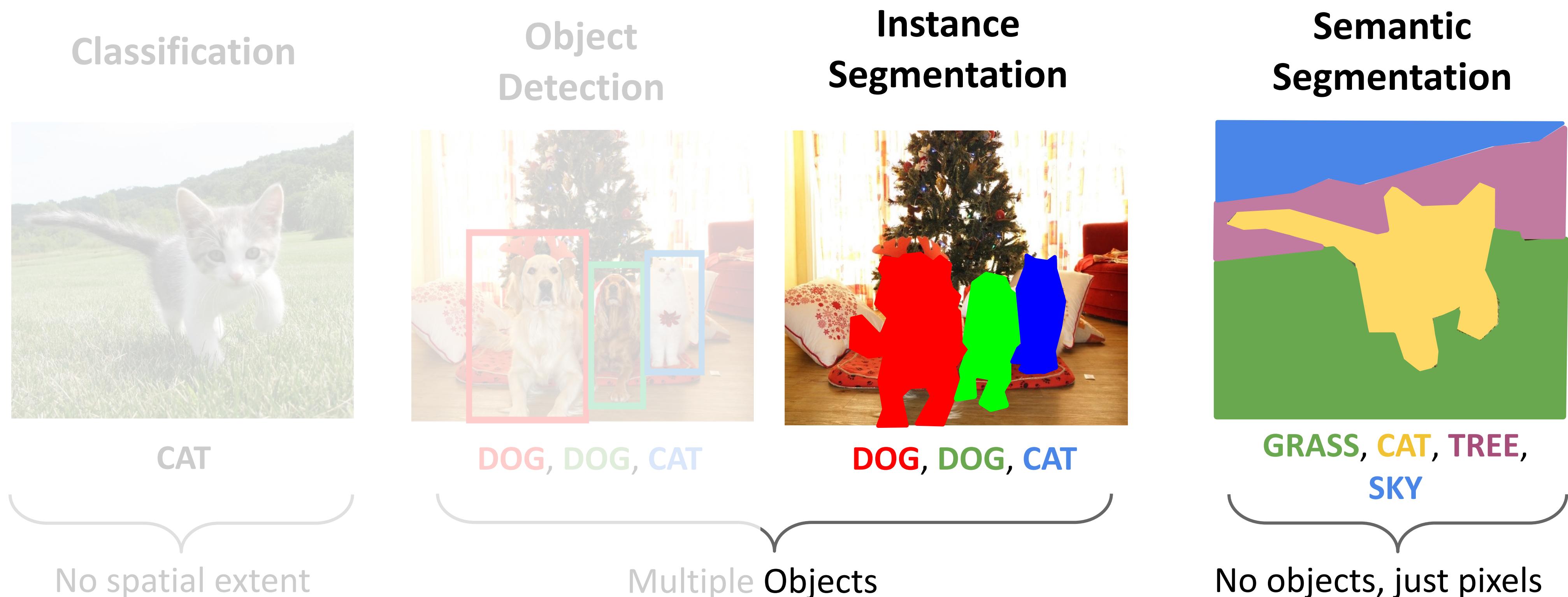
Project milestone (due 11/5)

Your project milestone report should be between **2 - 3 pages** using the [provided template](#). The following is a suggested structure for your report:

- Title, Author(s)
- Introduction: this section introduces your problem, and the overall plan for approaching your problem
- Problem statement: Describe your problem precisely specifying the dataset to be used, expected results and evaluation
- Technical Approach: Describe the methods you intend to apply to solve the given problem
- Intermediate/Preliminary Results: State and evaluate your results upto the milestone

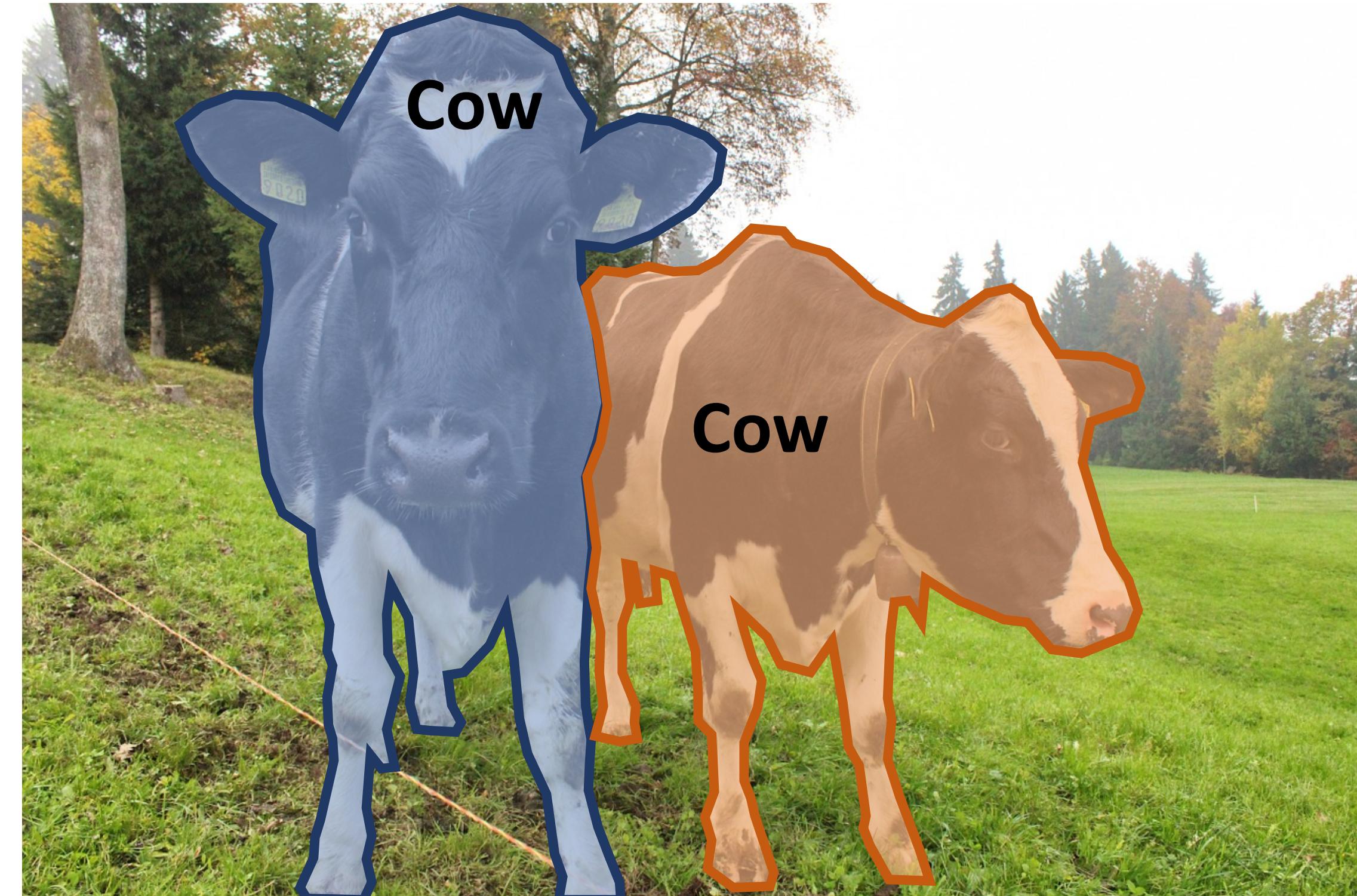
Submission: Please upload a PDF file to Gradescope. Please coordinate with your teammate and **submit only under ONE of your accounts**, and add your teammate on Gradescope.

Computer Vision Tasks



Instance segmentation

Instance Segmentation:
Detect all objects in the image, and identify the pixels that belong to each object

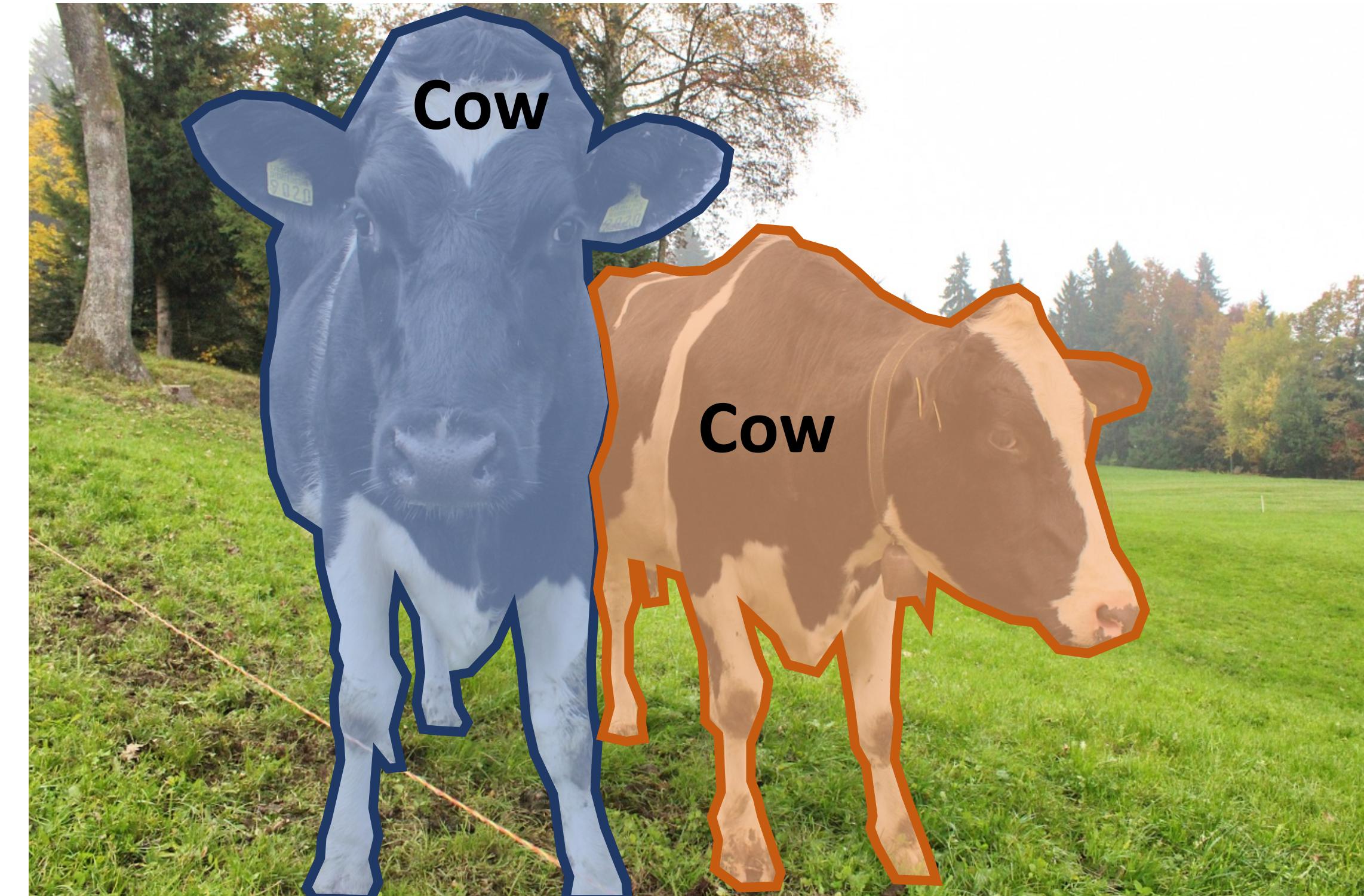


[This image is CC0 public domain](#)

Instance segmentation

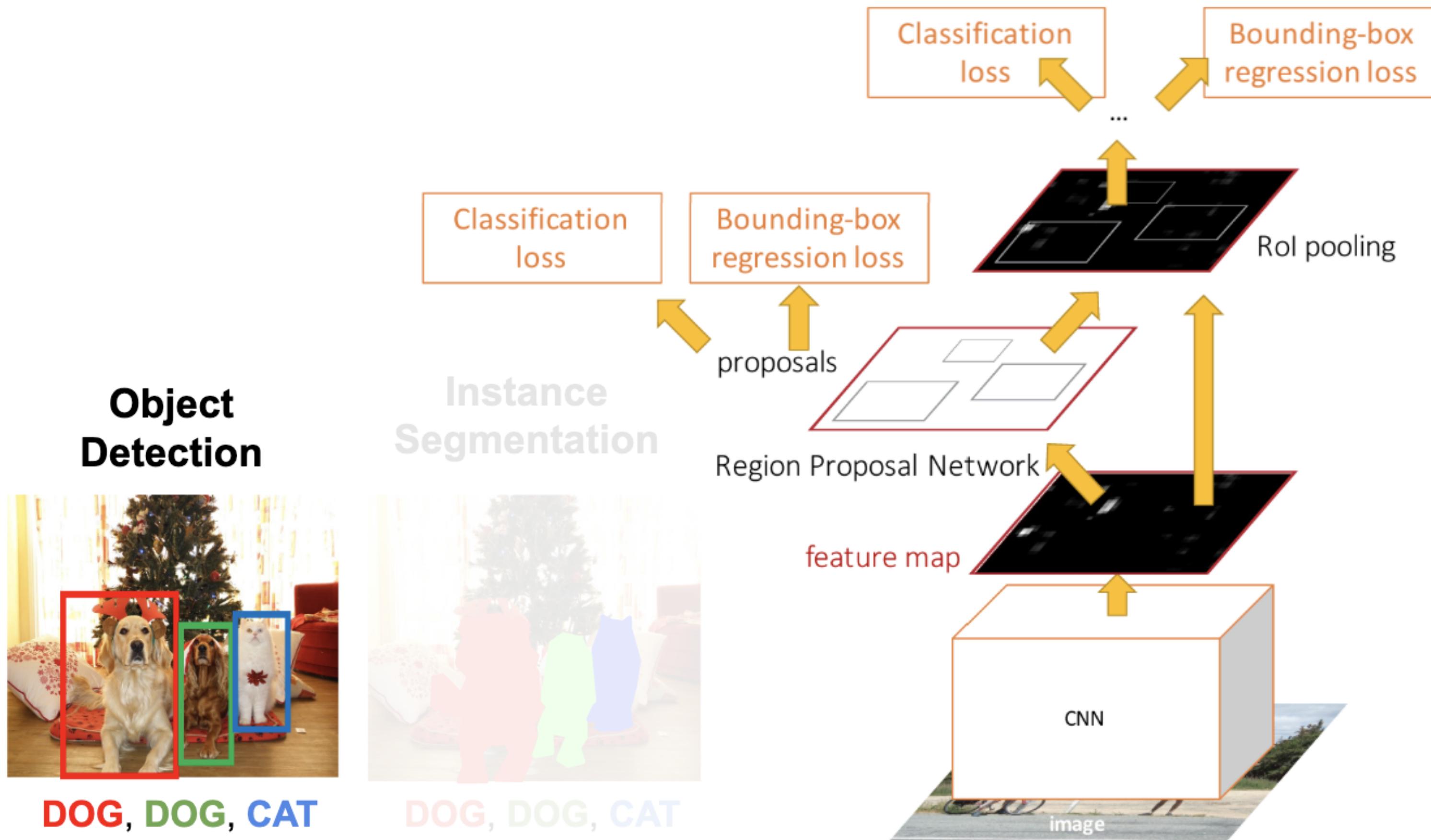
Instance Segmentation:
Detect all objects in the image, and identify the pixels that belong to each object

Approach: Perform object detection, then predict a segmentation mask for each object!



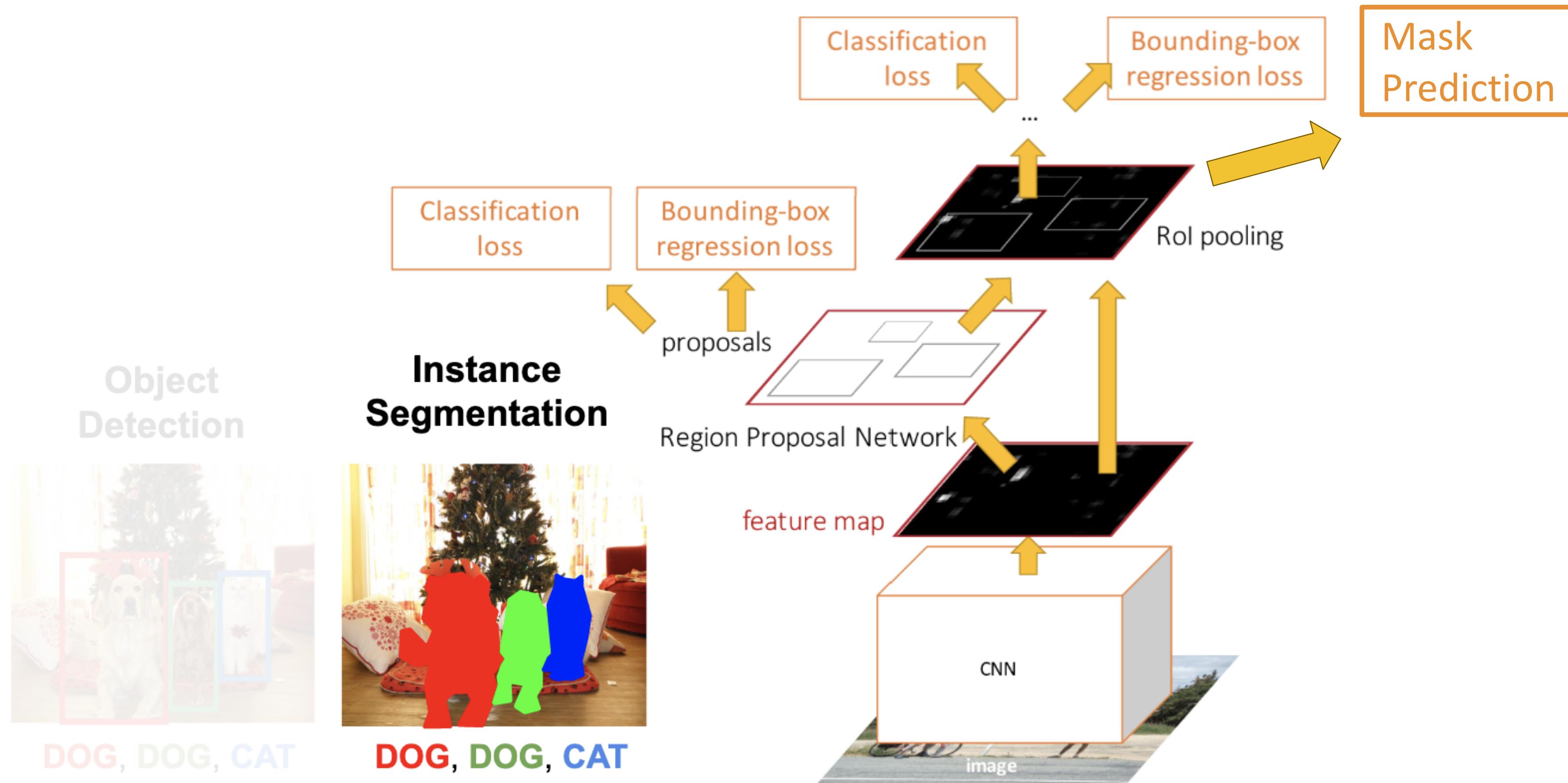
[This image](#) is CC0 public domain

Object Detection: Faster R-CNN



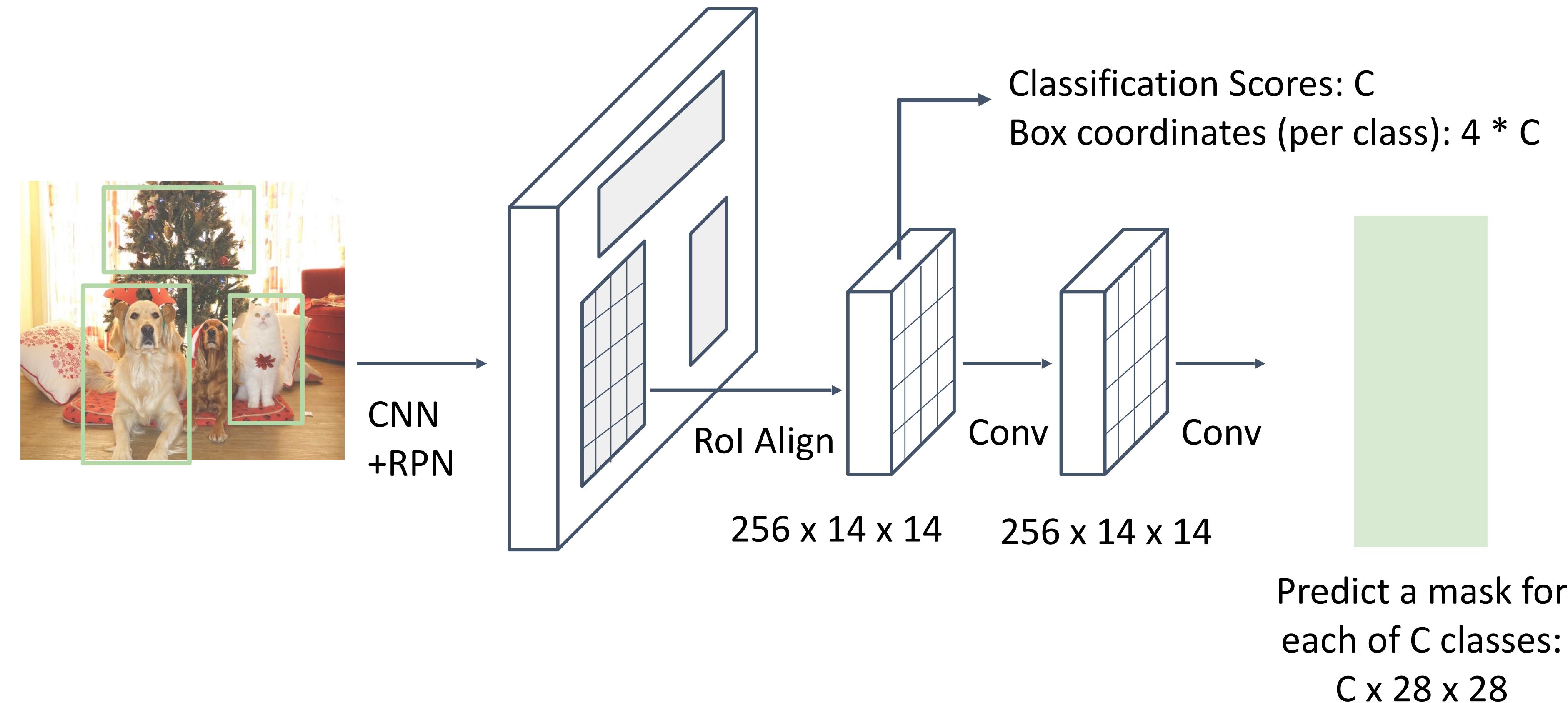
Ren et al, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Instance Segmentation: Mask R-CNN



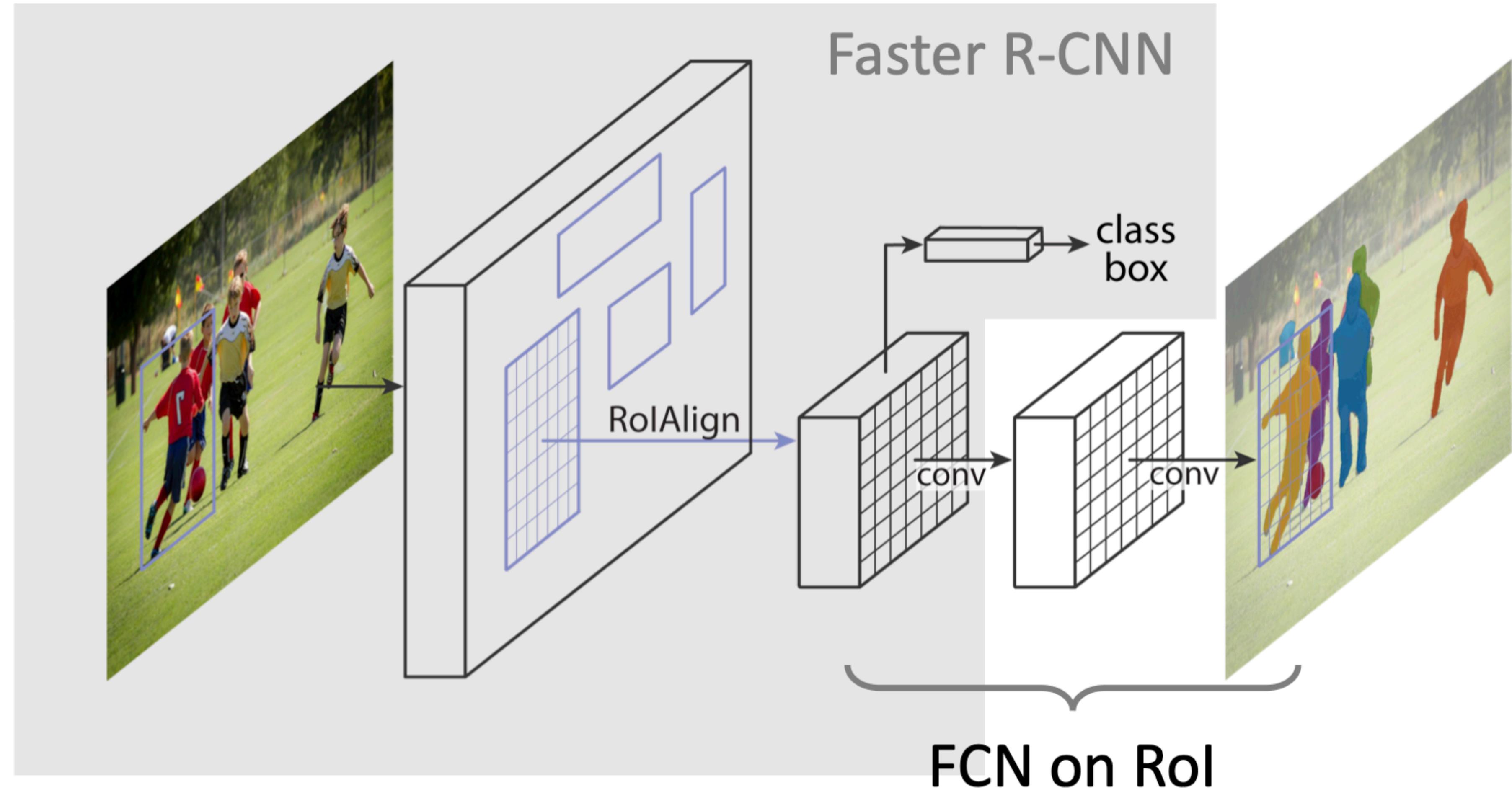
He et al, "Mask R-CNN", ICCV 2017

Mask R-CNN



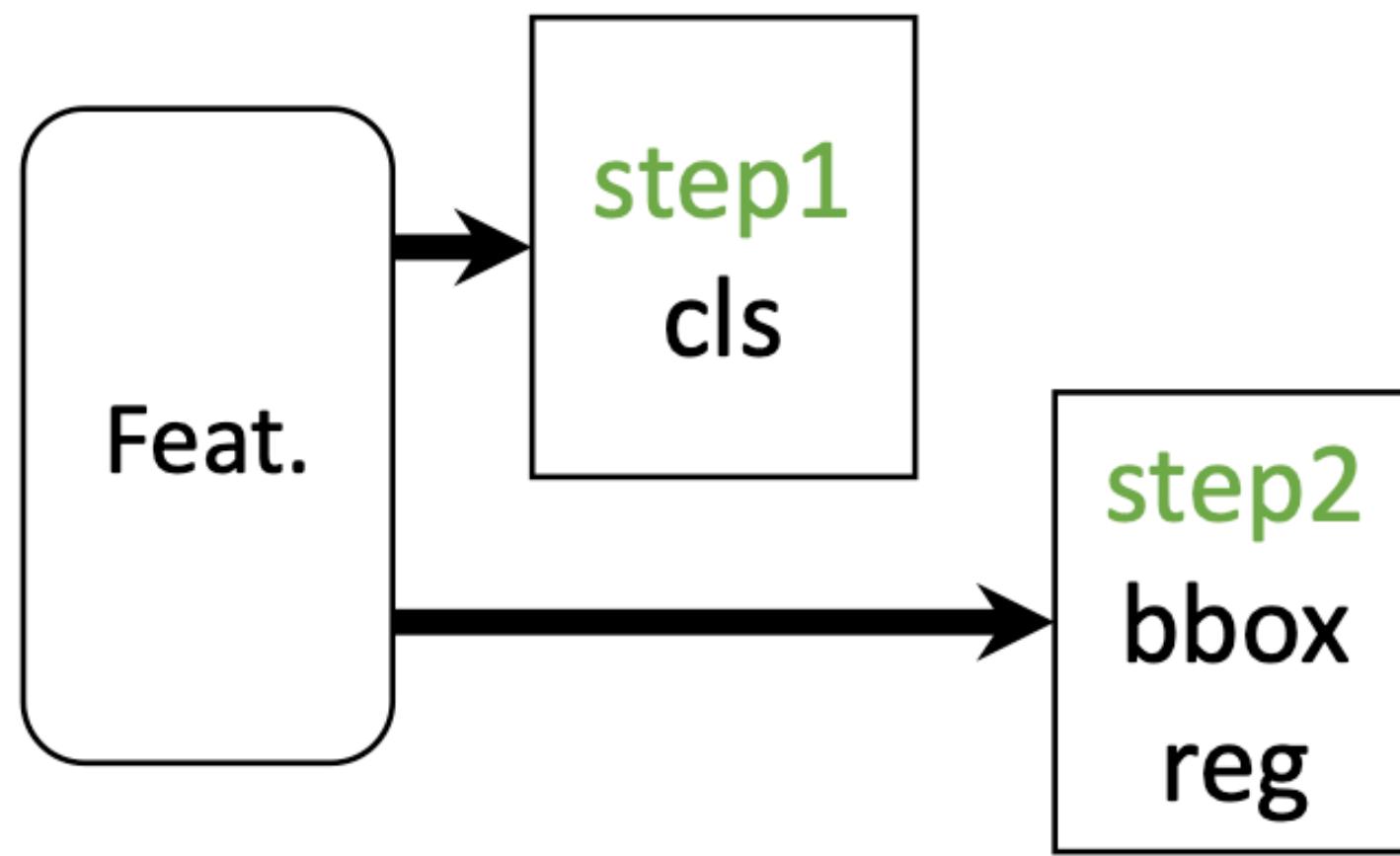
Mask R-CNN

- Mask R-CNN = **Faster R-CNN** with **FCN** on Rols

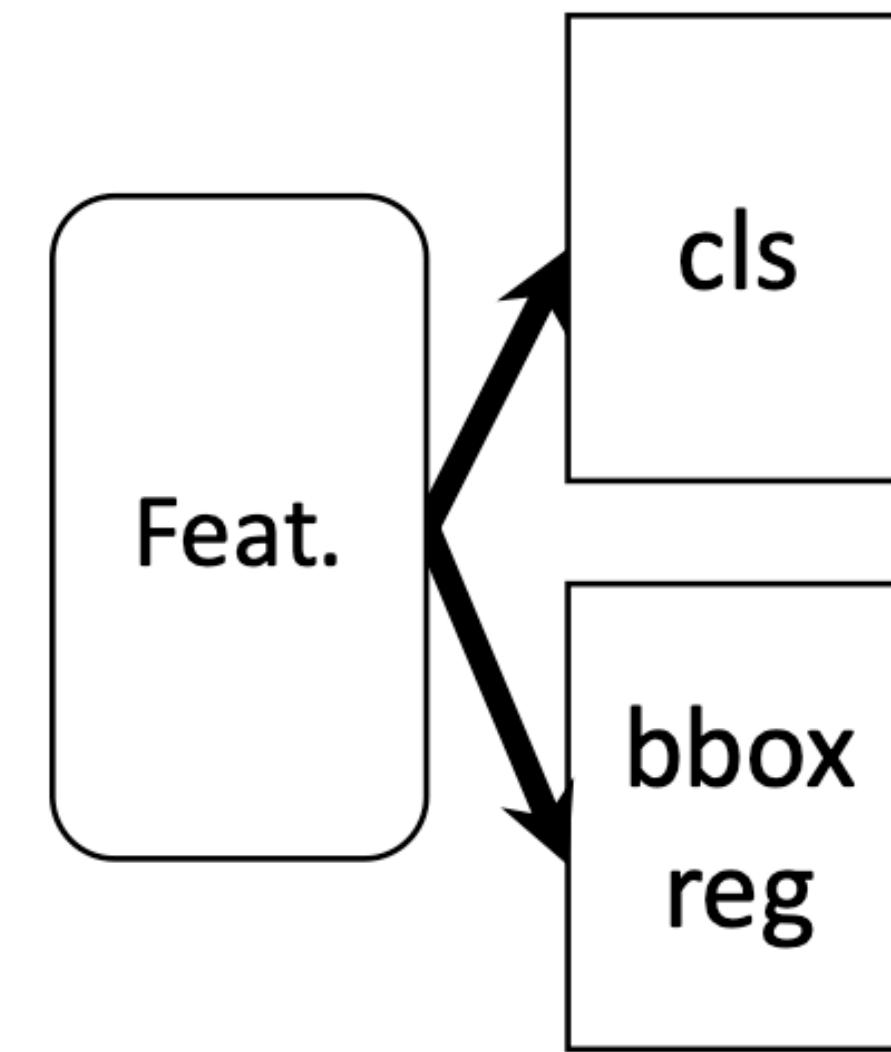


Parallel Heads

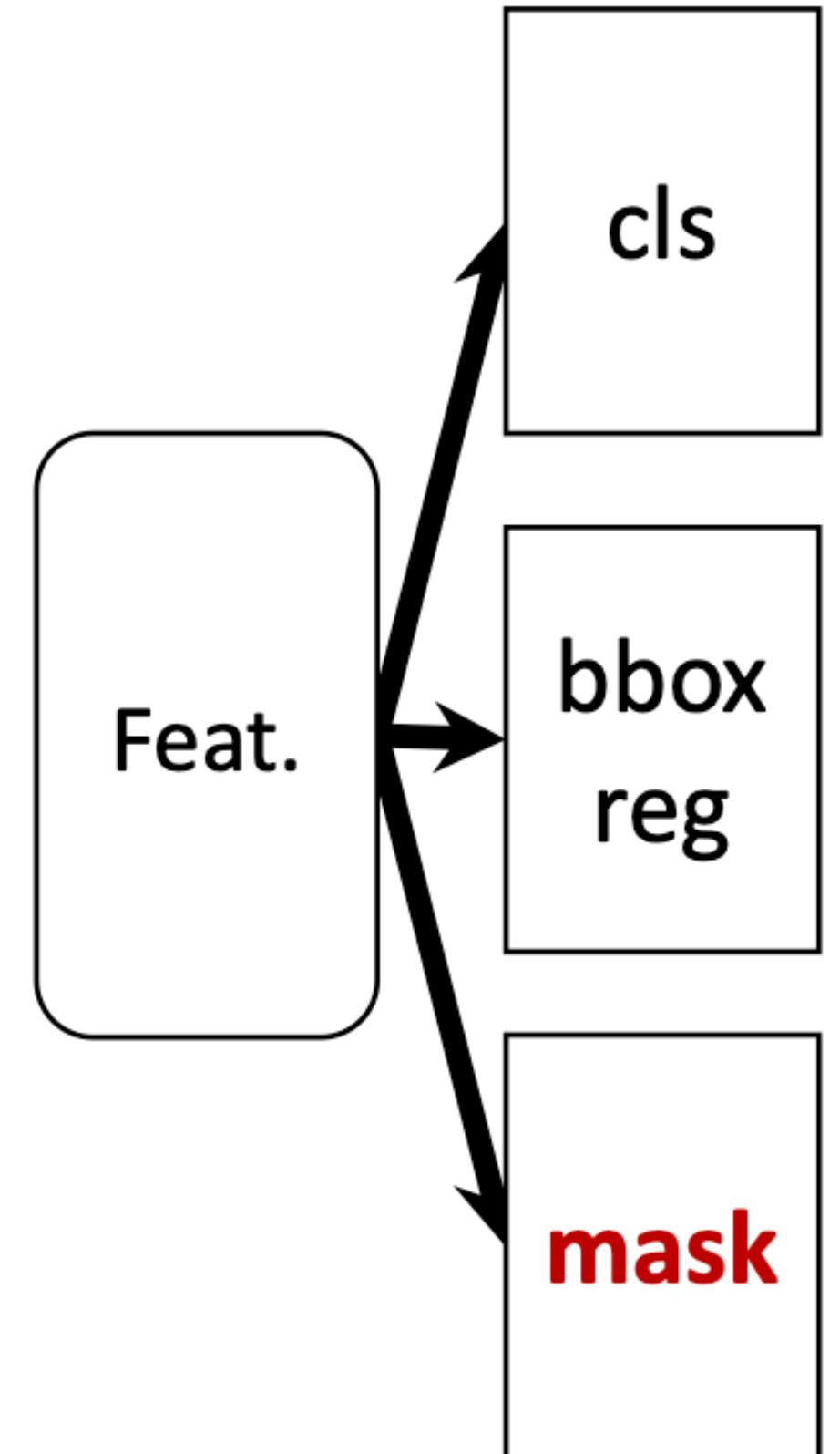
- Easy, fast to implement and train



(slow) R-CNN

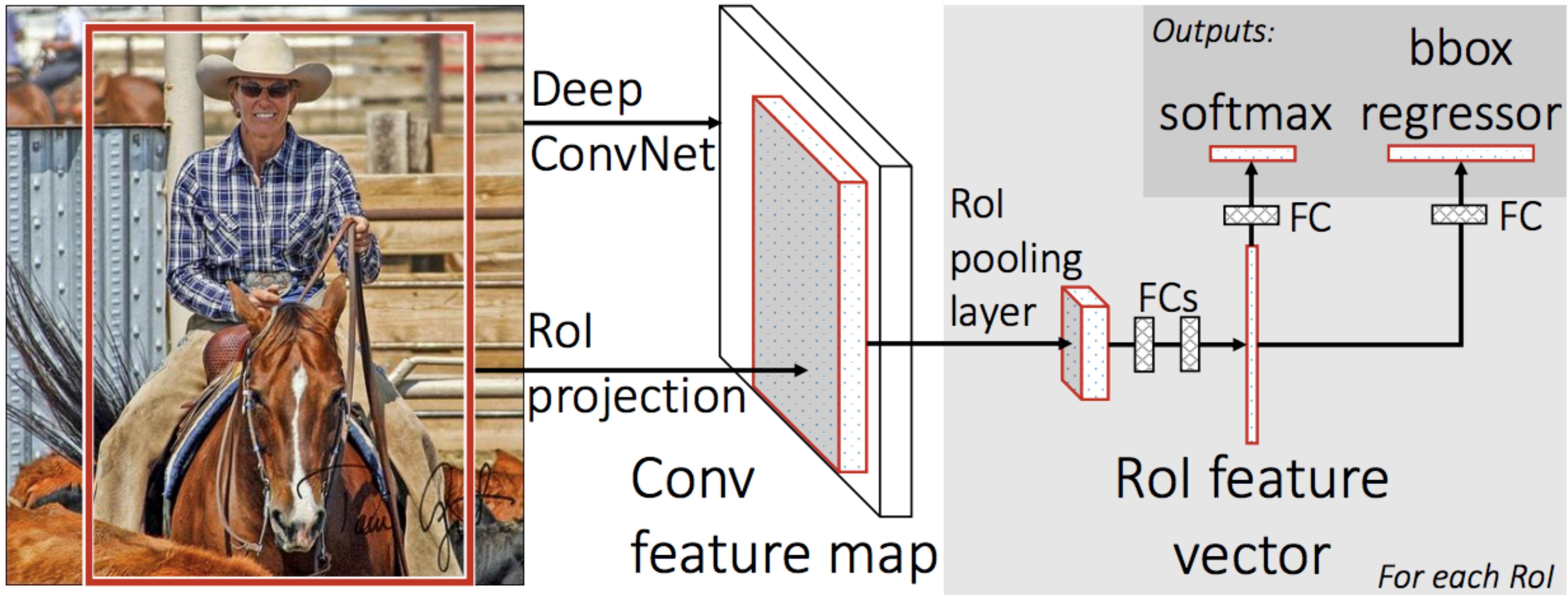


Fast/er R-CNN



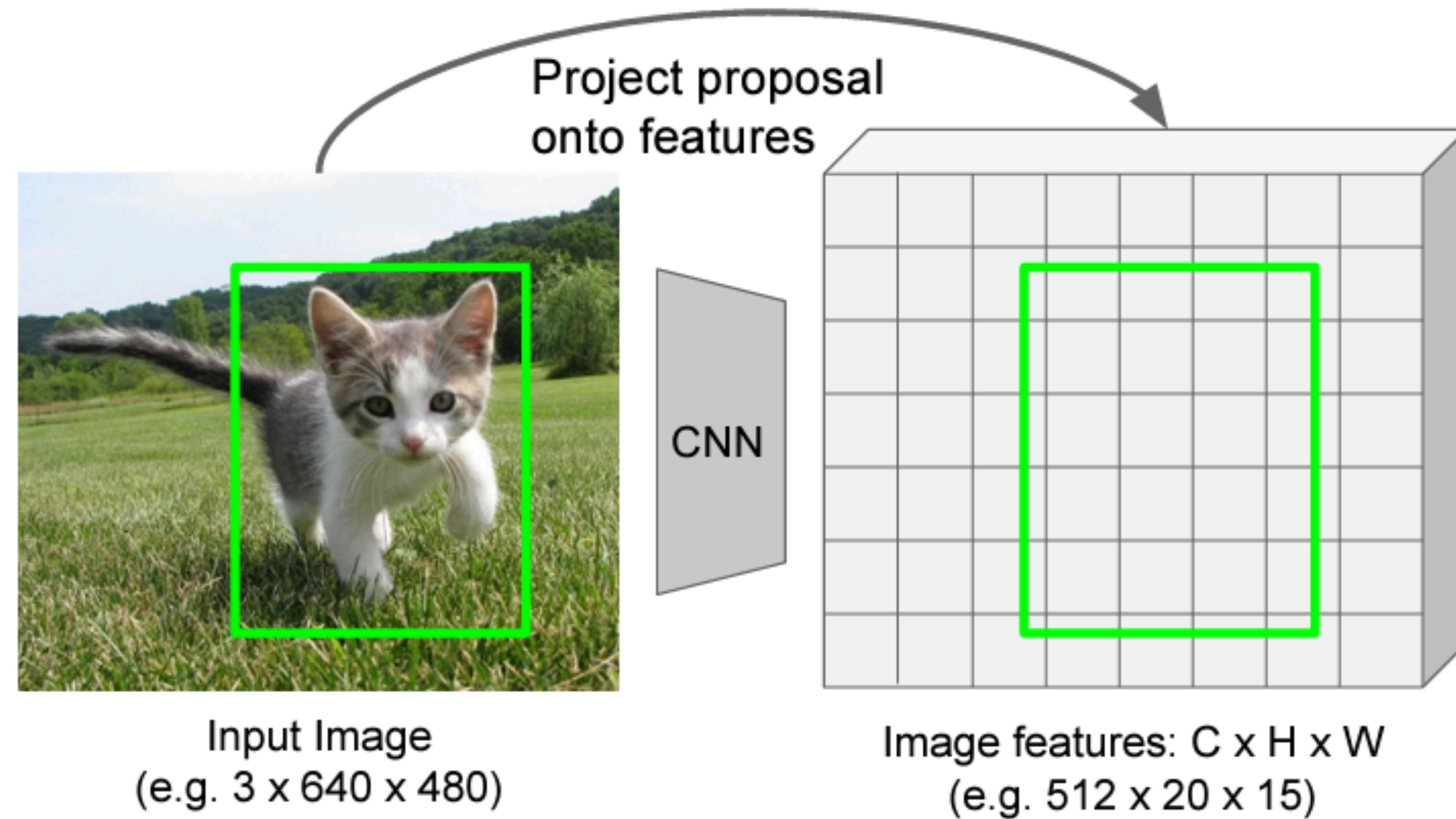
Mask R-CNN

RoIPool and RoIAvg



R. Girshick, [Fast R-CNN](#), ICCV 2015

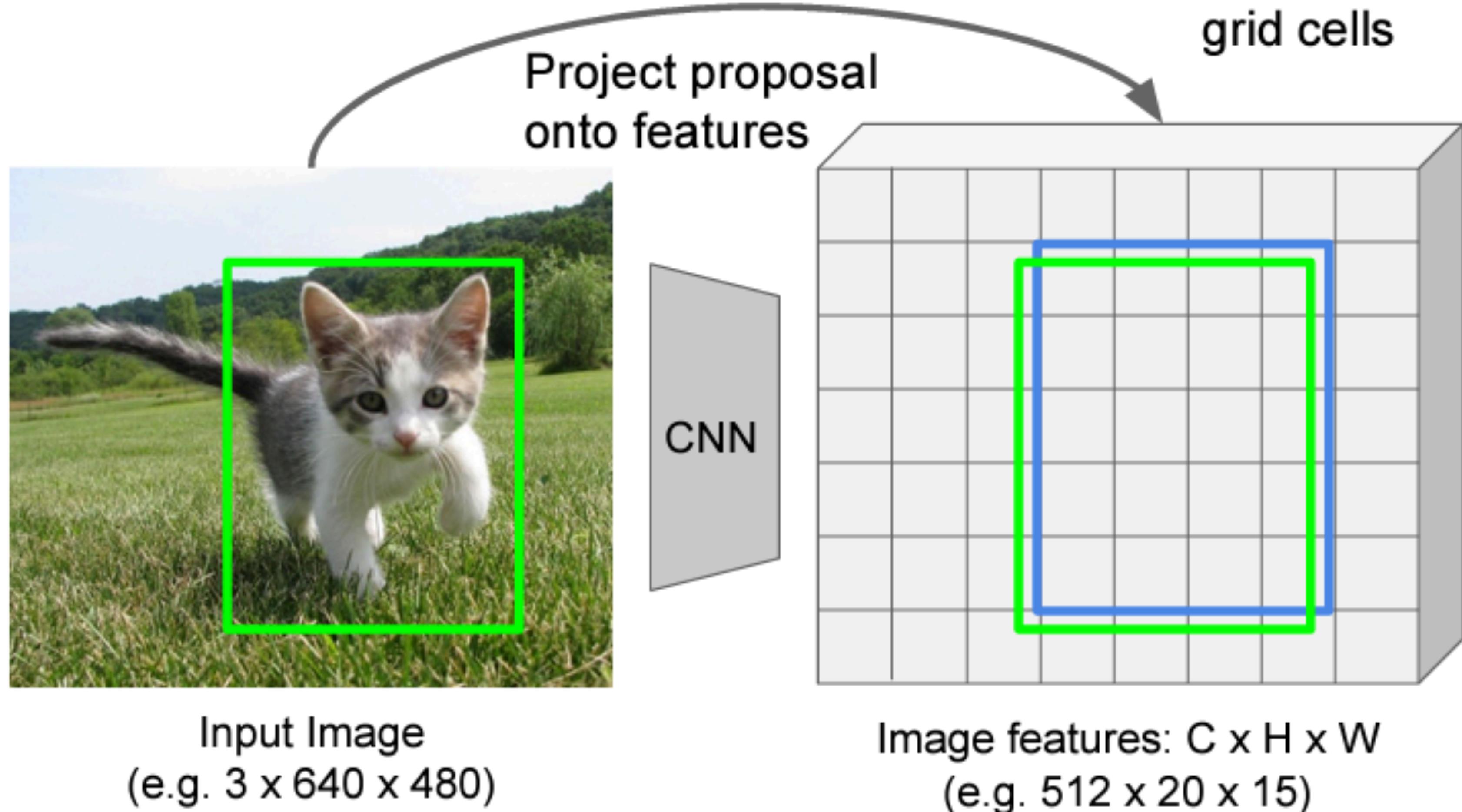
Cropping Features: RoI Pool



Girshick, "Fast R-CNN", ICCV 2015.

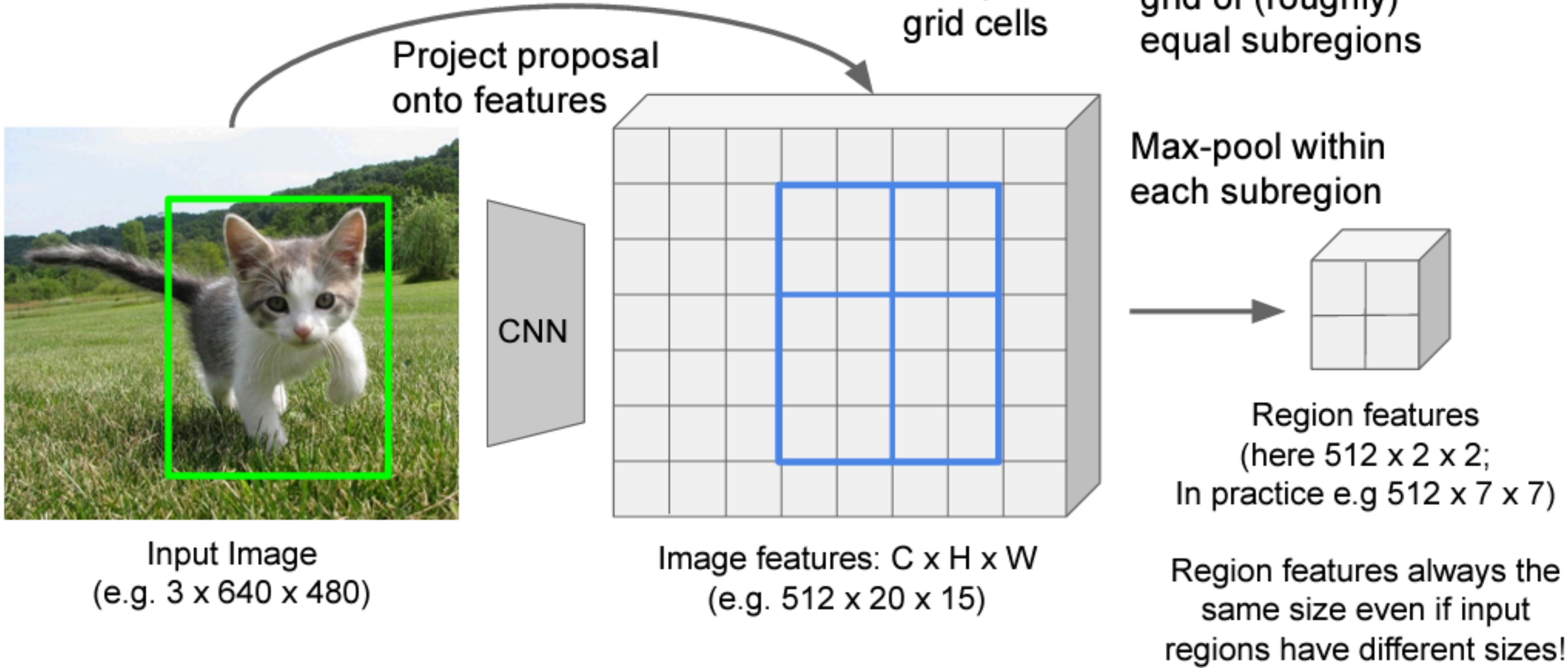
Girshick, "Fast R-CNN", ICCV 2015.

Cropping Features: RoI Pool



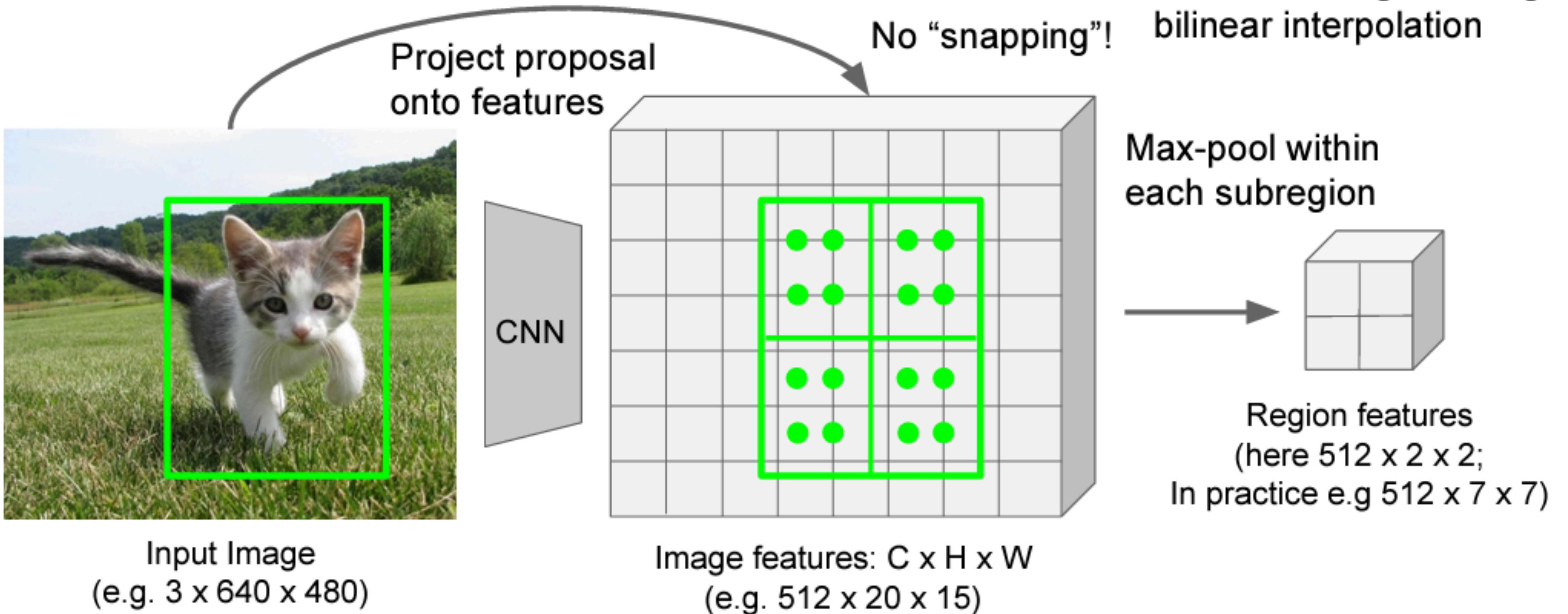
Girshick, "Fast R-CNN", ICCV 2015.

Cropping Features: RoI Pool



Girshick, "Fast R-CNN", ICCV 2015.

Cropping Features: RoI Align



He et al, "Mask R-CNN", ICCV 2017

Ablation: RoIPool vs RoIAlign

baseline: ResNet-50-Conv5 backbone, **stride=32**

	mask AP			box AP		
	AP	AP ₅₀	AP ₇₅	AP ^{bb}	AP ^{bb} ₅₀	AP ^{bb} ₇₅
<i>RoIPool</i>	23.6	46.5	21.6	28.2	52.7	26.9
<i>RoIAlign</i>	30.9	51.8	32.1	34.0	55.3	36.4
	+7.3	+ 5.3	+10.5	+5.8	+2.6	+9.5

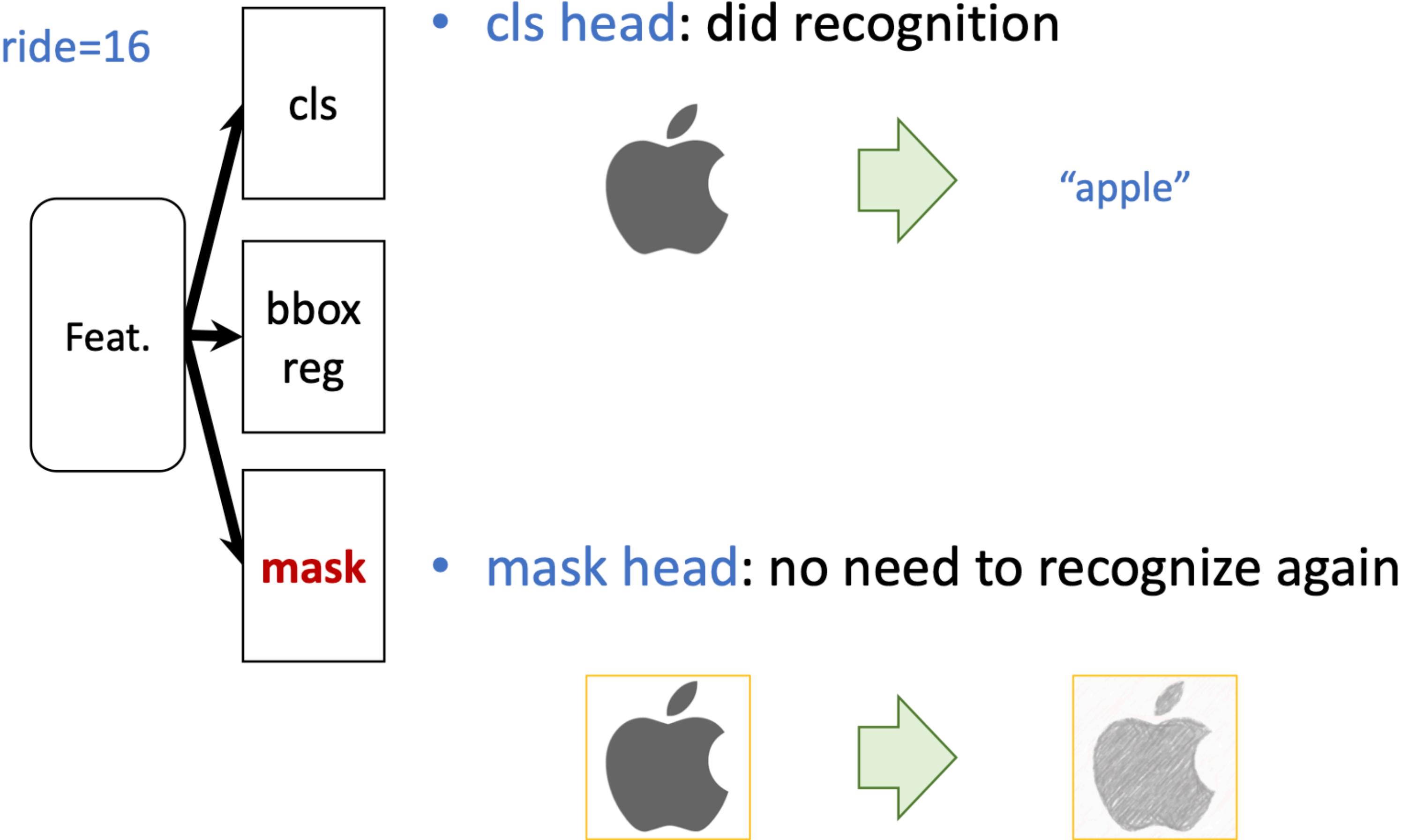


- huge gain at high IoU,
in case of big stride (32)

Ablation: Multinomial vs Binary Segmentation

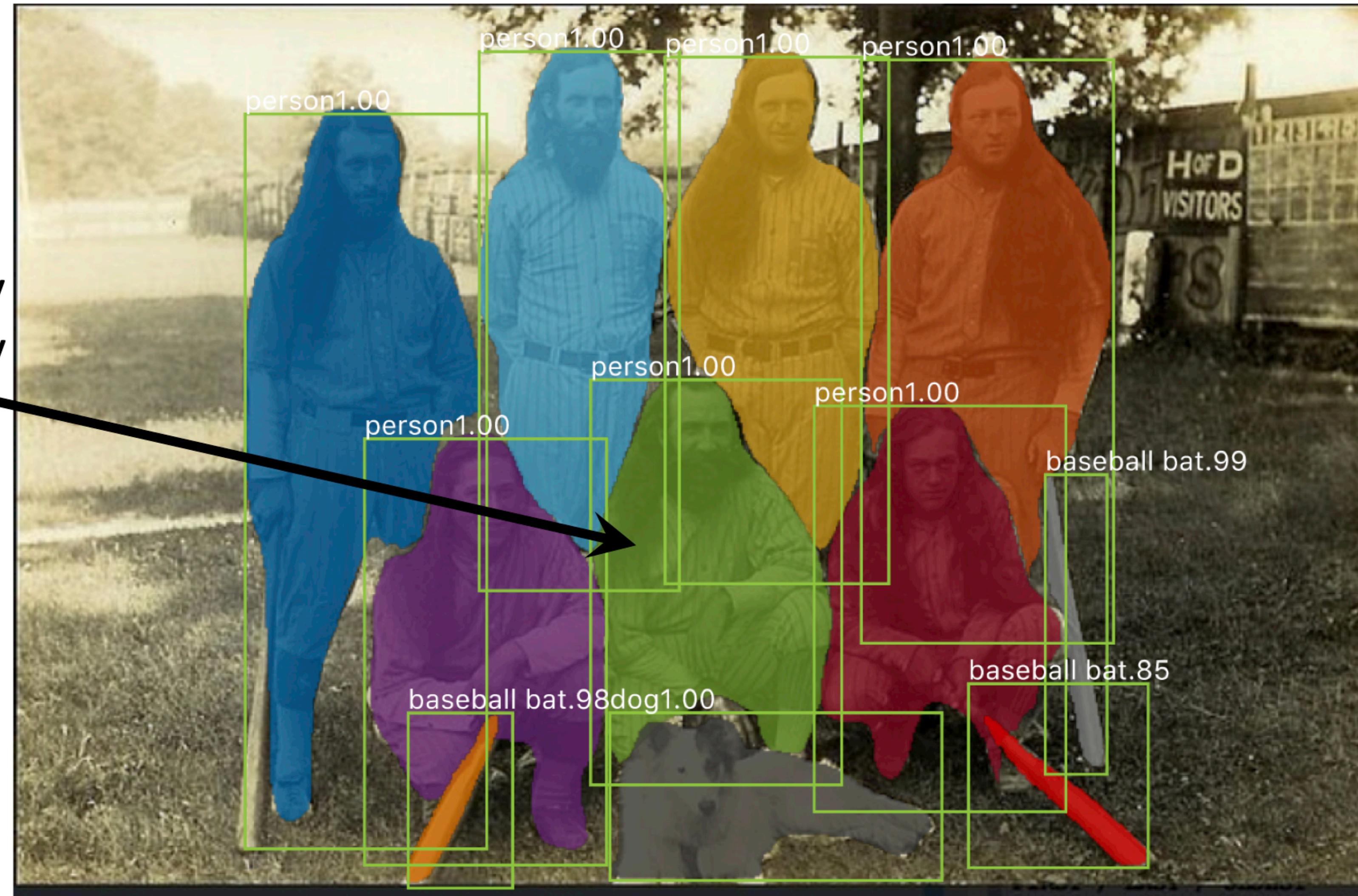
baseline: ResNet-50-Conv4 backbone, stride=16

	AP	AP ₅₀	AP ₇₅
<i>softmax</i>	24.8	44.1	25.1
sigmoid	30.3	51.2	31.5
	+5.5	+7.1	+6.4



Mask R-CNN: Very Good Results!

object
surrounded by
same-category
objects



Mask R-CNN results on COCO

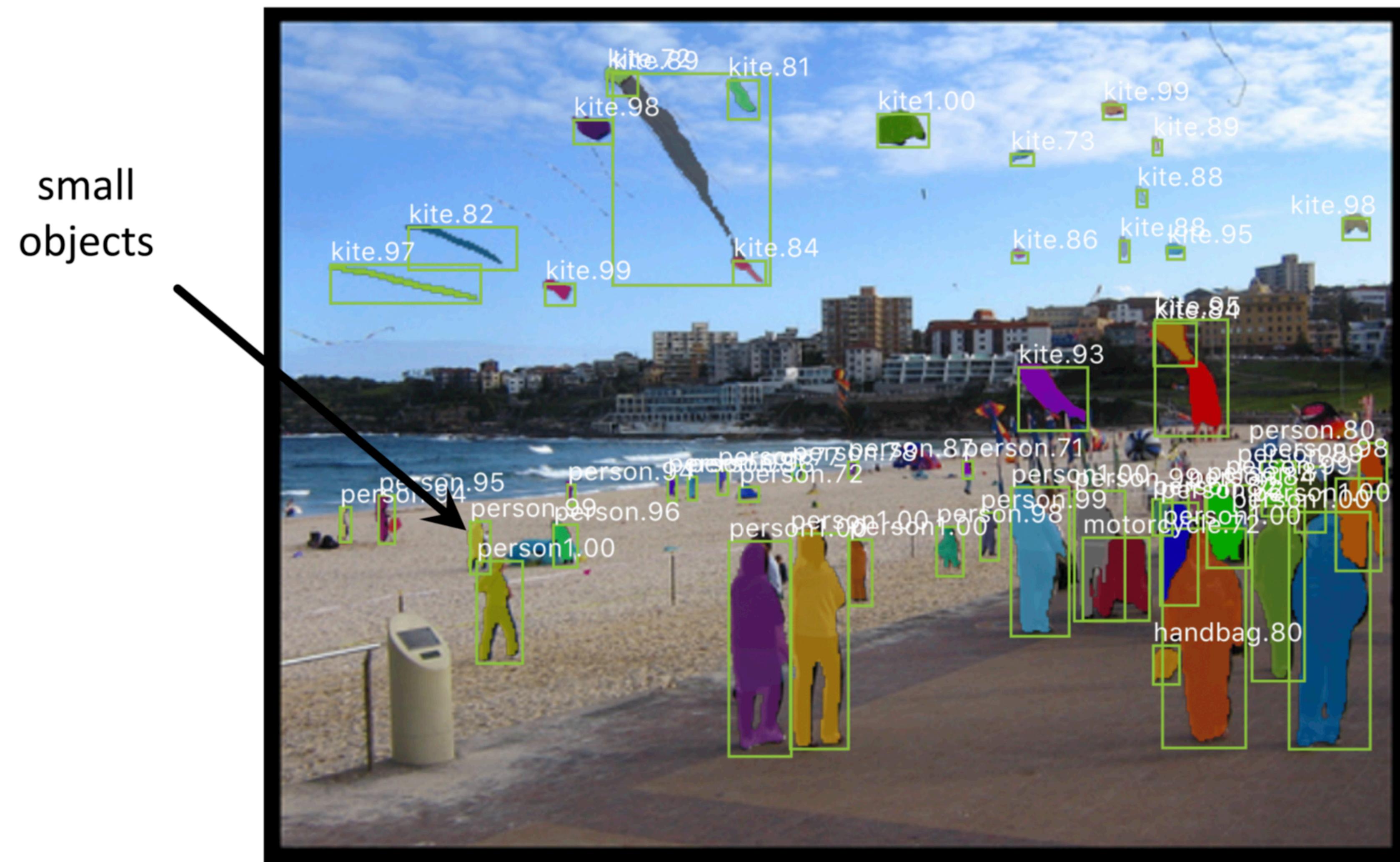
Mask R-CNN: Very Good Results!

disconnected
object



Mask R-CNN results on COCO

Mask R-CNN: Very Good Results!



Mask R-CNN results on COCO

Mask R-CNN: Failure Case

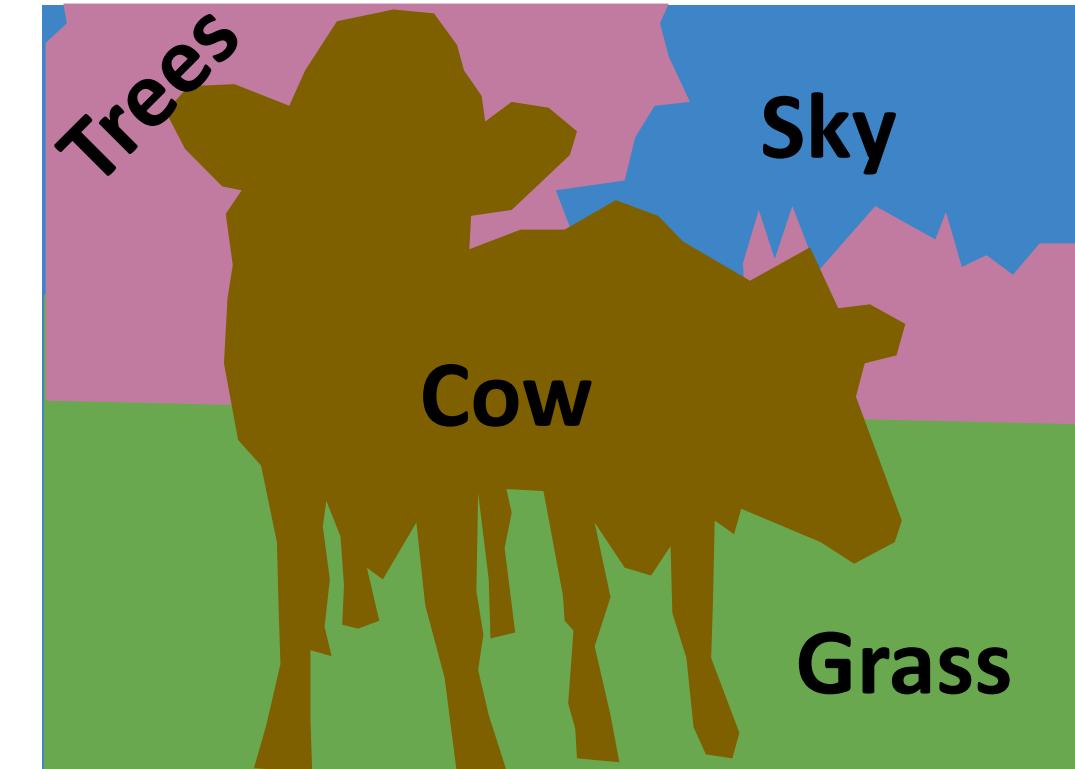
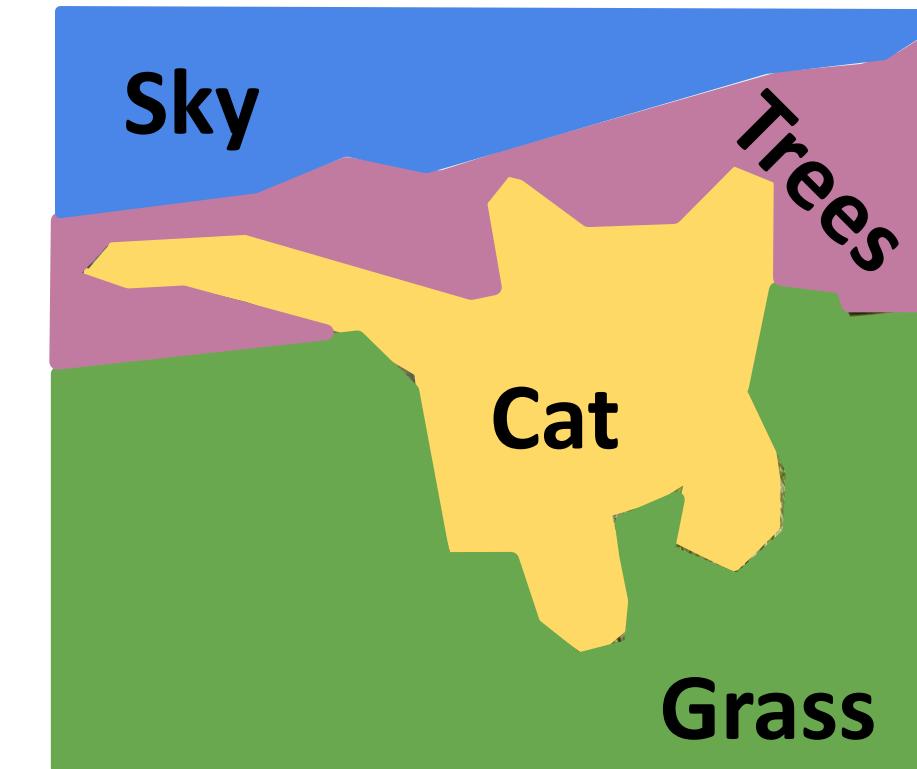


Mask R-CNN results on COCO

Semantic Segmentation

Label each pixel in the image with a category label

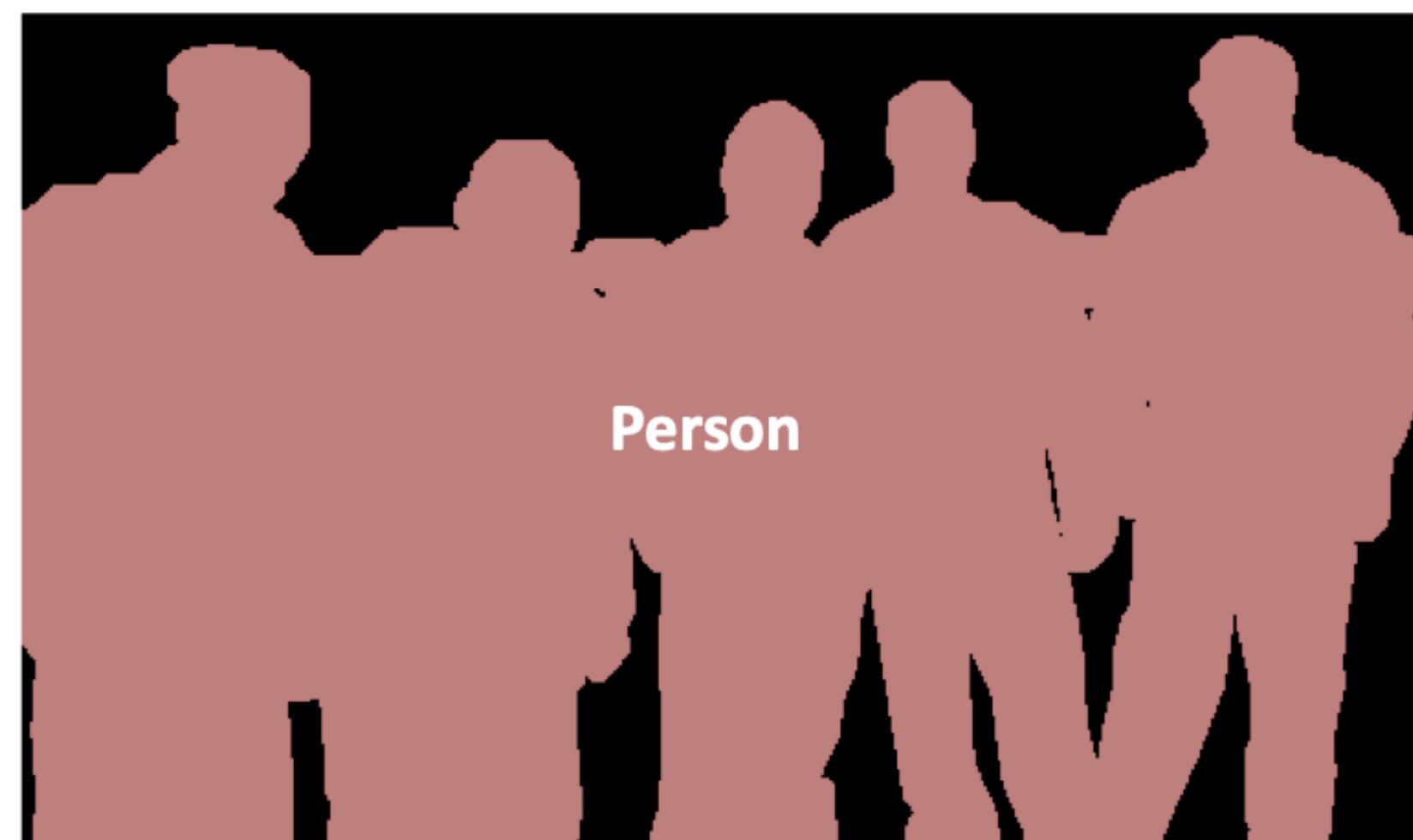
Don't differentiate instances, only care about pixels



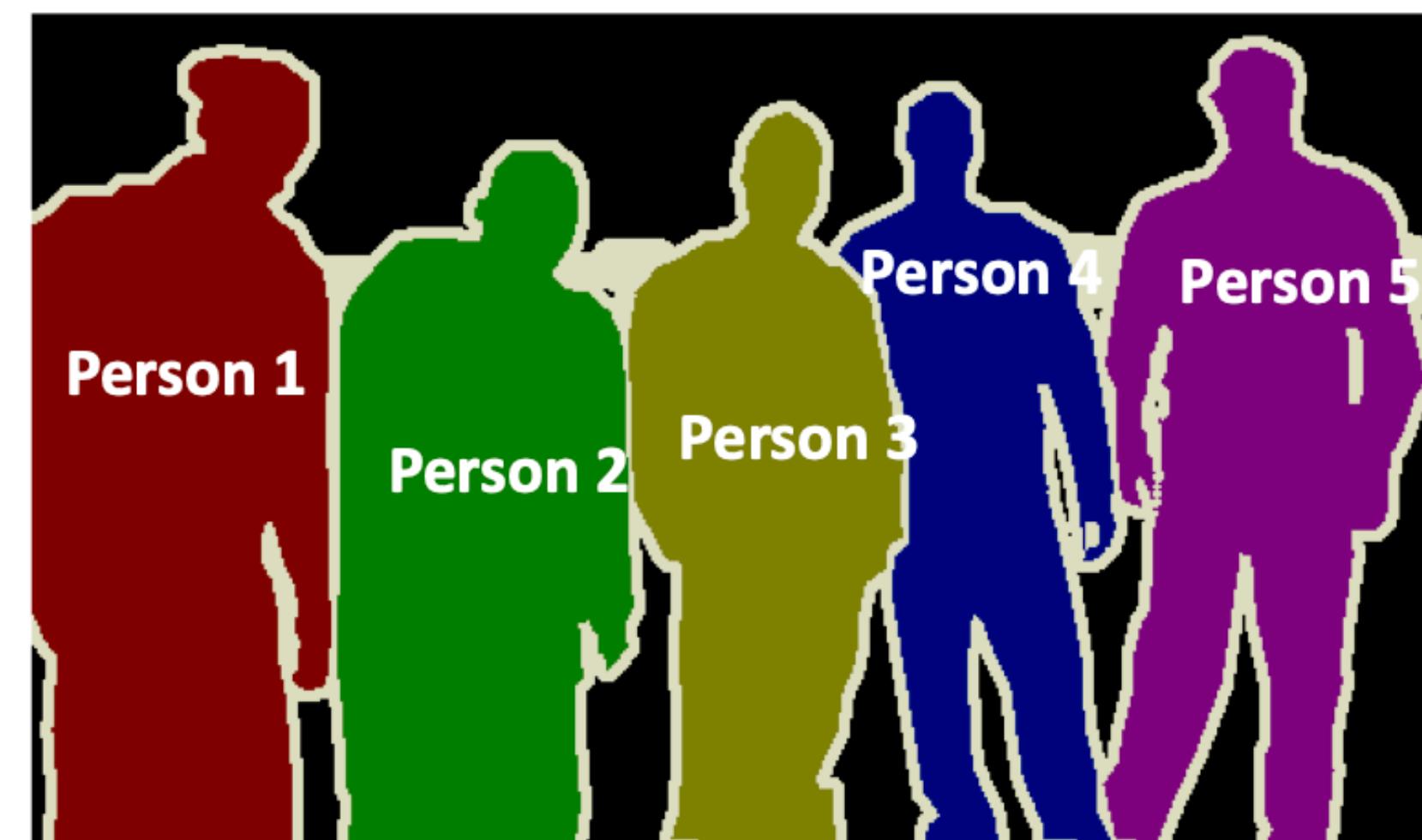
Semantic vs Instance Segmentation



Object Detection

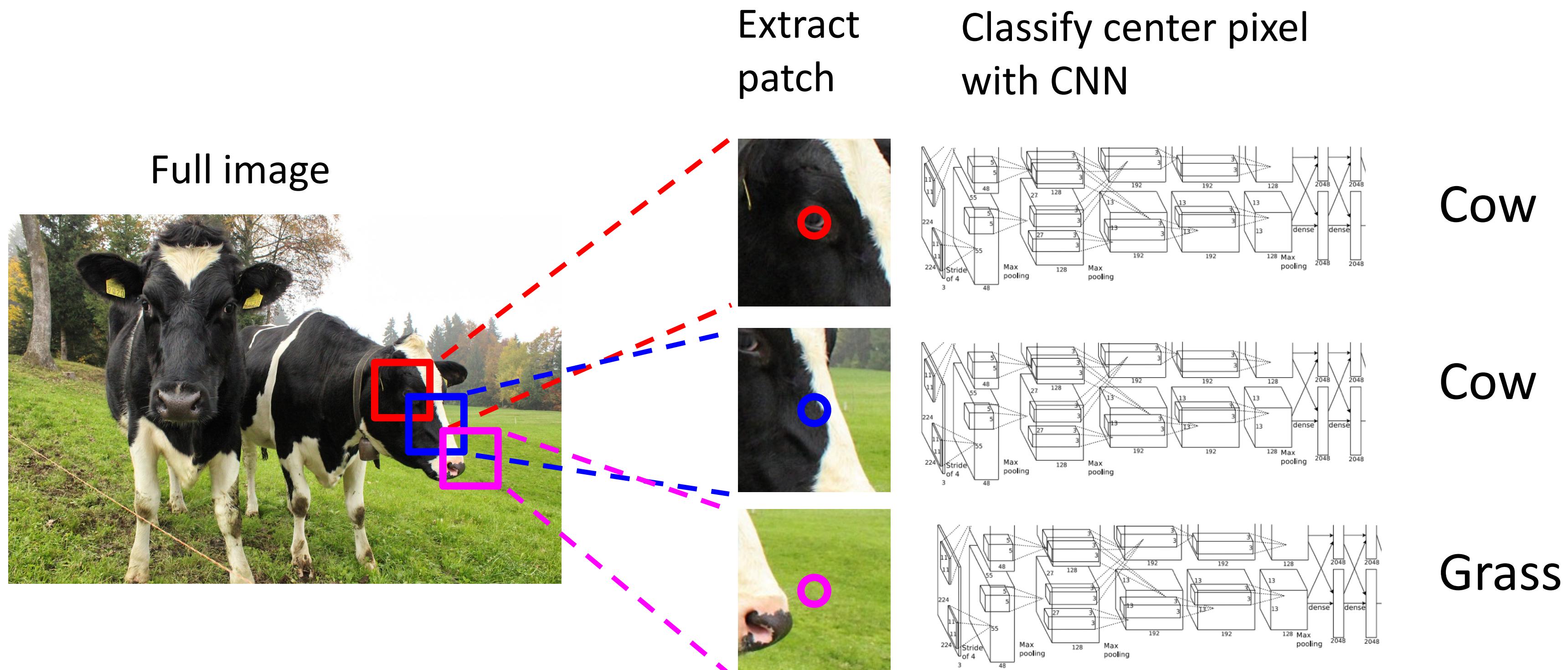


Semantic Segmentation



Instance Segmentation

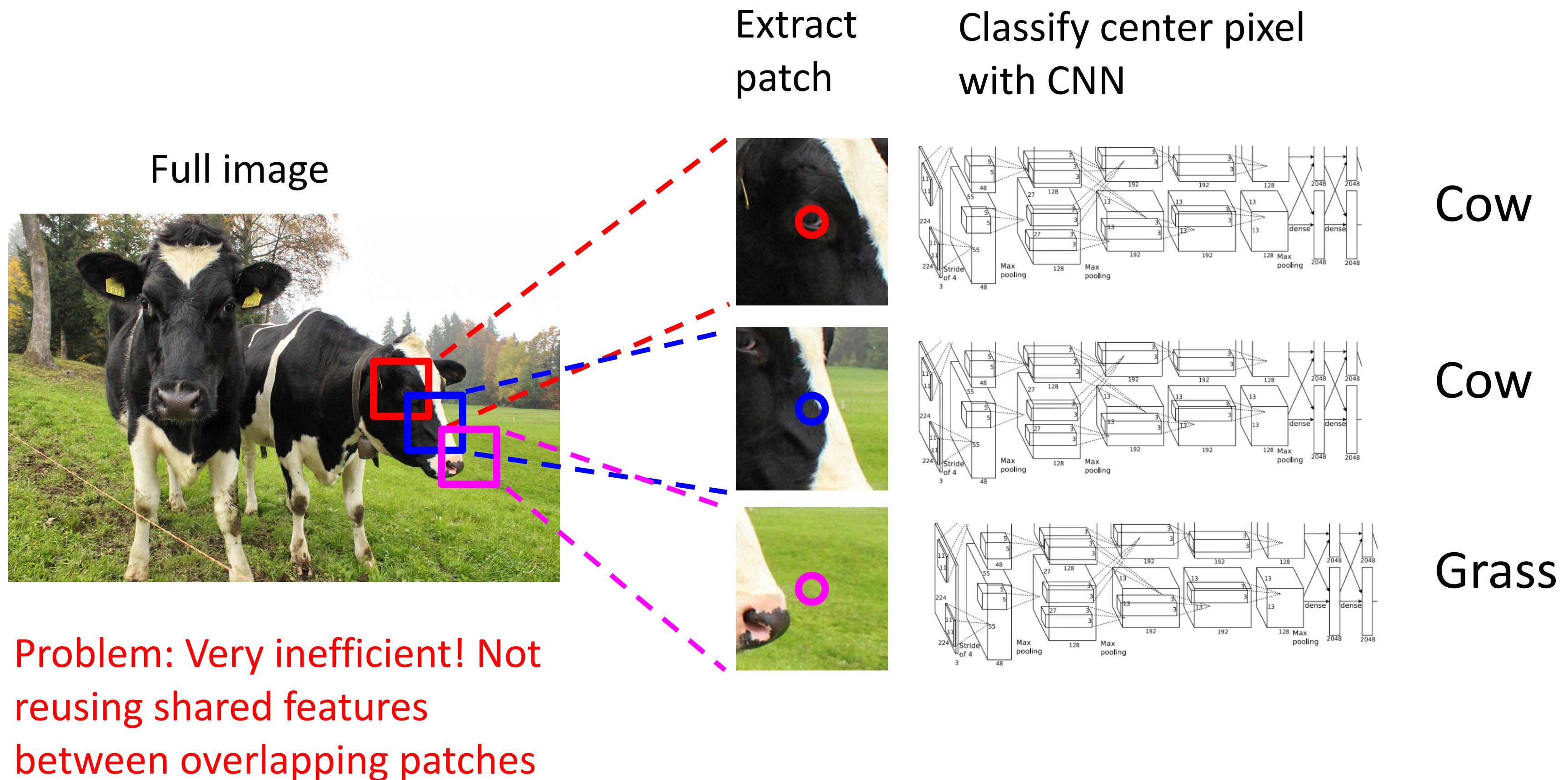
Segmentation: Sliding Window



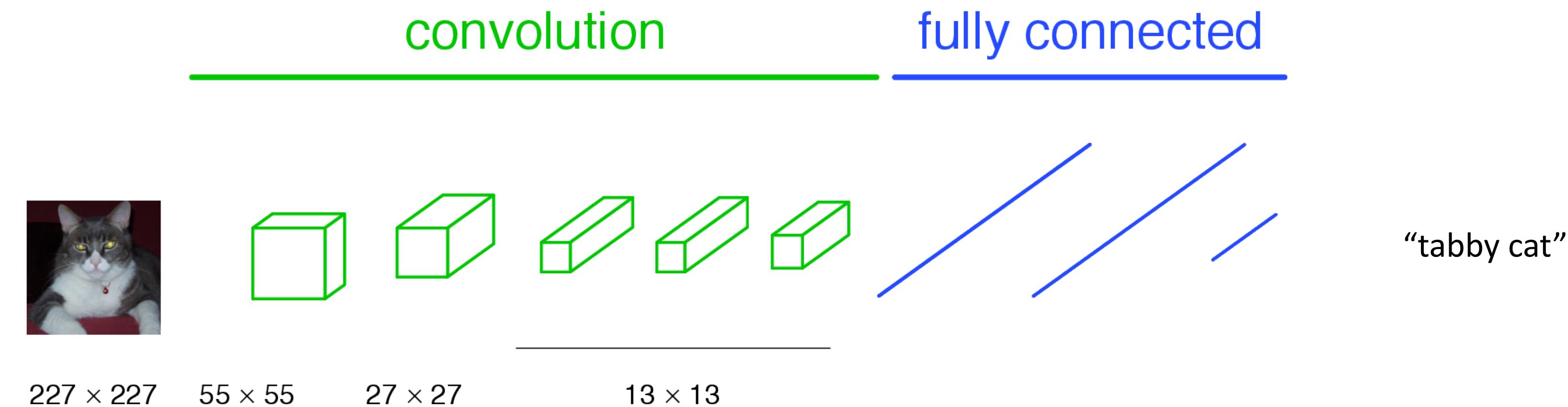
Farabet et al, "Learning Hierarchical Features for Scene Labeling," TPAMI 2013

Pinheiro and Collobert, "Recurrent Convolutional Neural Networks for Scene Labeling", ICML 2014

Segmentation: Sliding Window

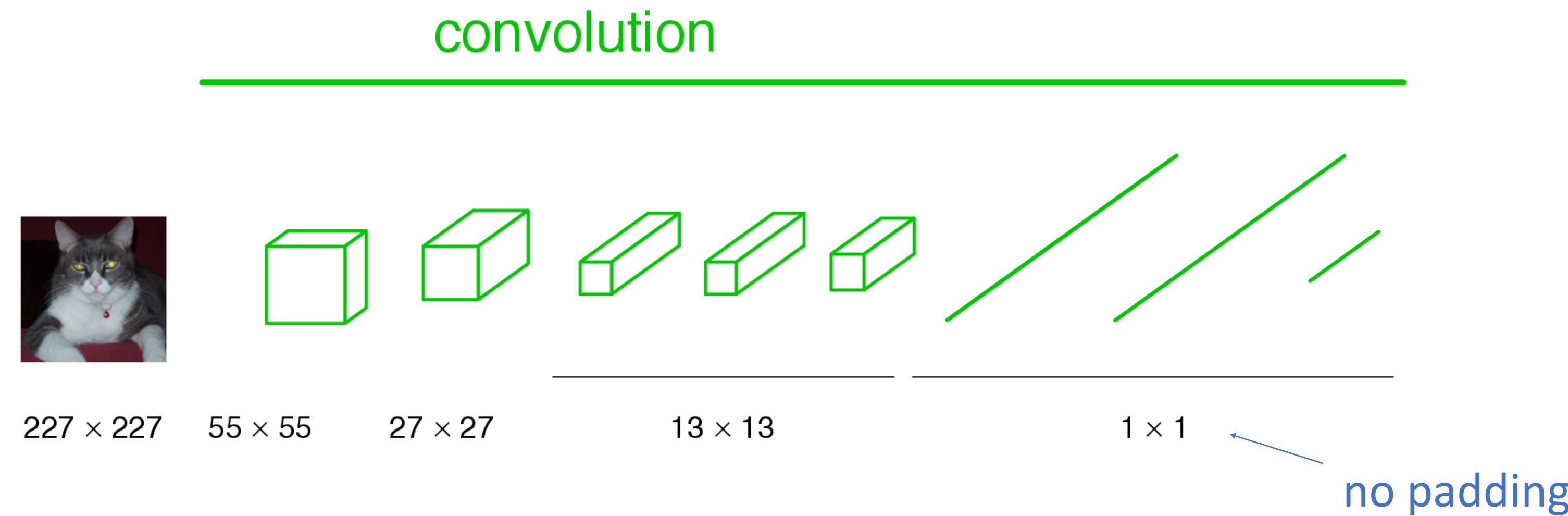


A Classification Network



Fully Convolutional Networks for Semantic Segmentation.
Jon Long, Evan Shelhamer, Trevor Darrell. CVPR 2015

Becoming Fully Convolutional

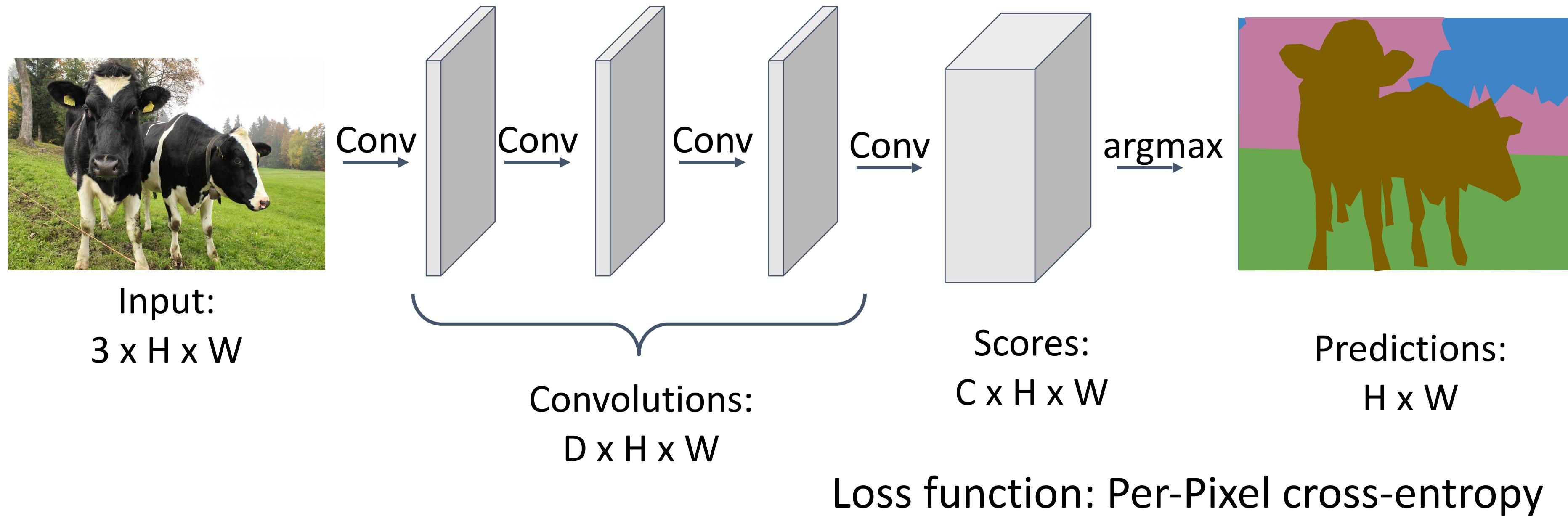


A **fully-connected layer** is equivalent to a **convolution layer**.

Note: “Fully Convolutional” and “Fully Connected” aren’t the same thing.
They’re almost opposites, in fact.

Fully Convolutional Network

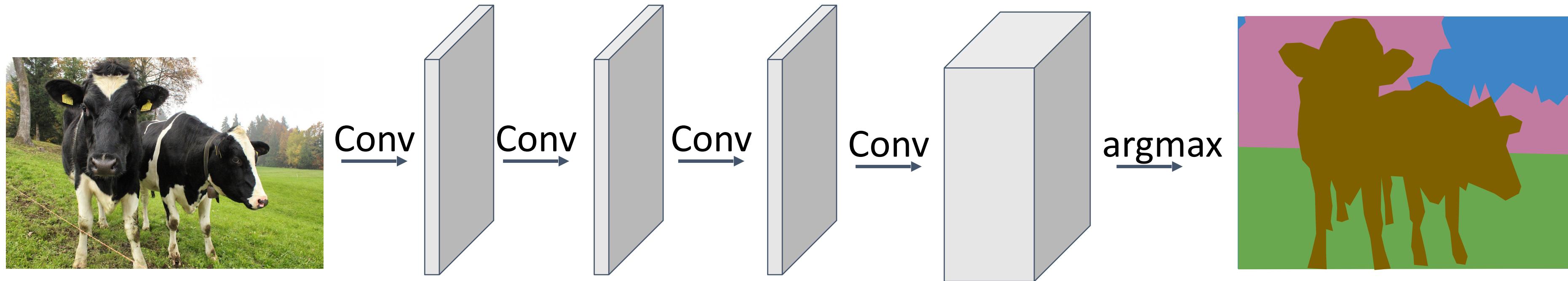
Design a network as a bunch of convolutional layers to make predictions for pixels all at once!



Long et al, “Fully convolutional networks for semantic segmentation”, CVPR 2015

Fully Convolutional Network

Design a network as a bunch of convolutional layers to make predictions for pixels all at once!

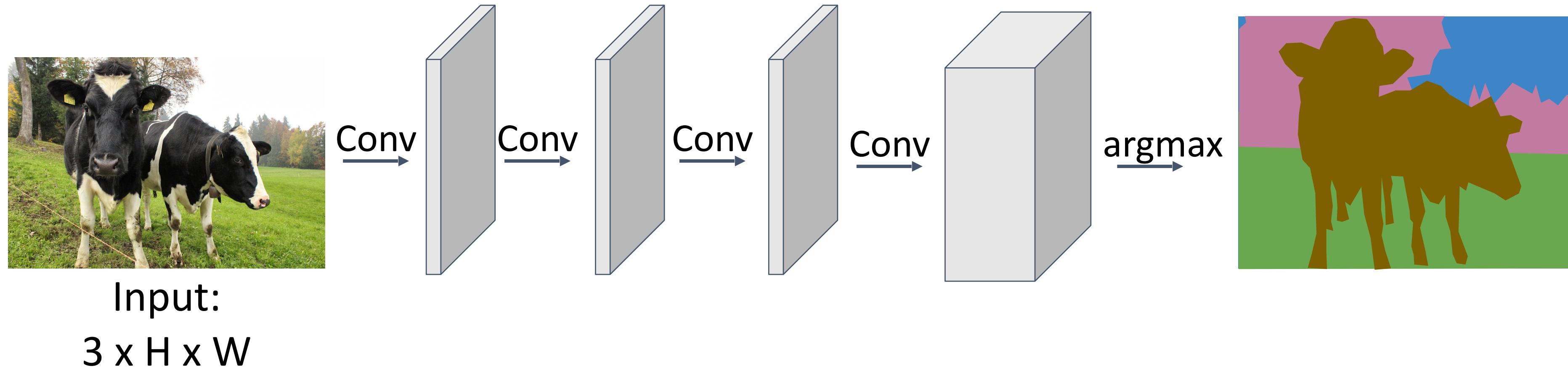


Input:
 $3 \times H \times W$

Problem #1: Effective receptive field size is linear in number of conv layers: With L 3×3 conv layers, receptive field is $1+2L$

Fully Convolutional Network

Design a network as a bunch of convolutional layers to make predictions for pixels all at once!

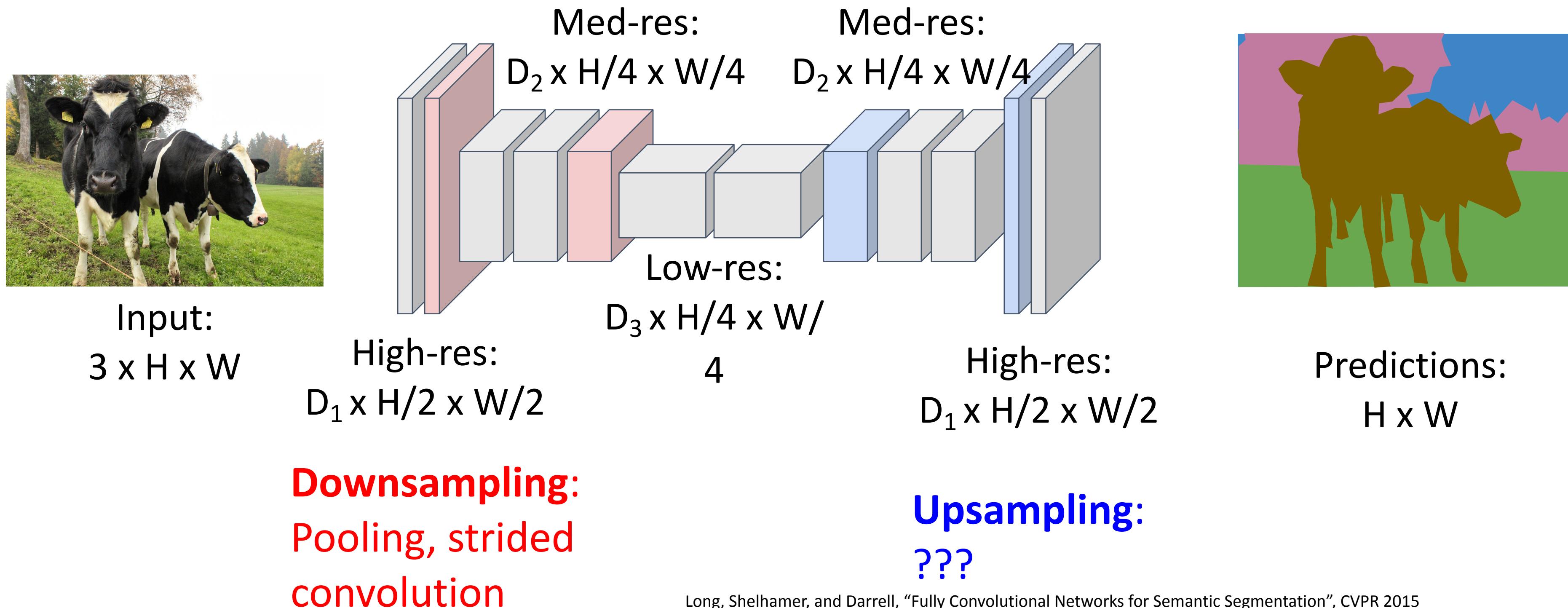


Problem #1: Effective receptive field size is linear in number of conv layers: With L 3×3 conv layers, receptive field is $1+2L$

Problem #2: Convolution on high res images is expensive!

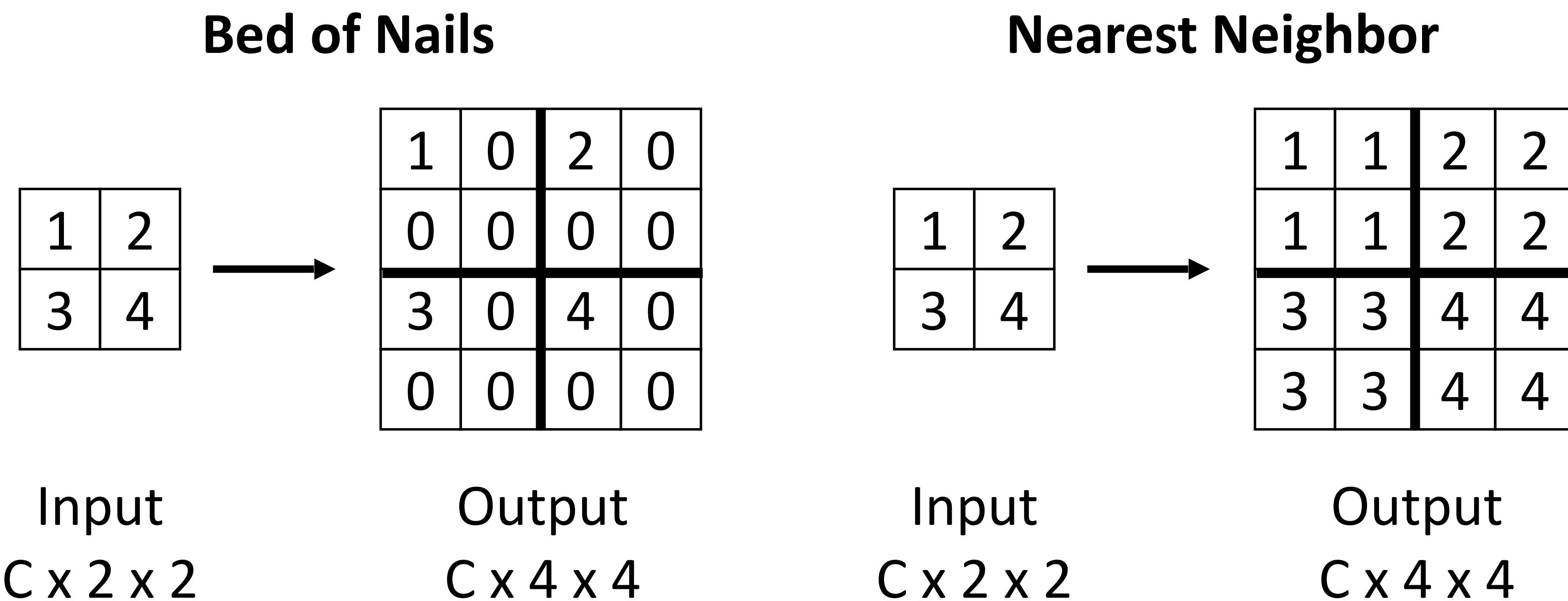
Fully Convolutional Network

Design network as a bunch of convolutional layers, with
downsampling and **upsampling** inside the network!

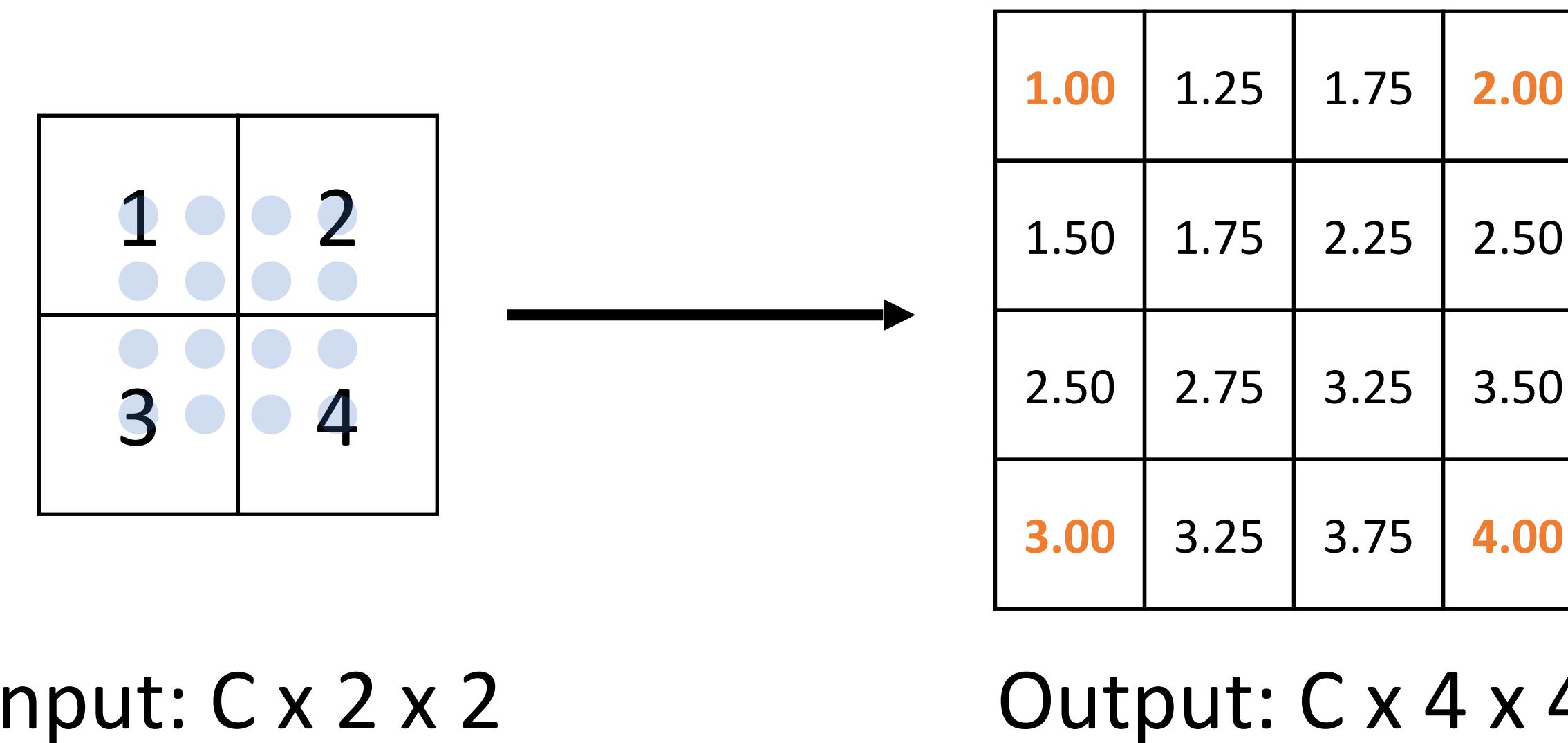


Long, Shelhamer, and Darrell, "Fully Convolutional Networks for Semantic Segmentation", CVPR 2015
Noh et al, "Learning Deconvolution Network for Semantic Segmentation", ICCV 2015

In-Network Upsampling: “Unpooling”



Upsampling: Bilinear Interpolation

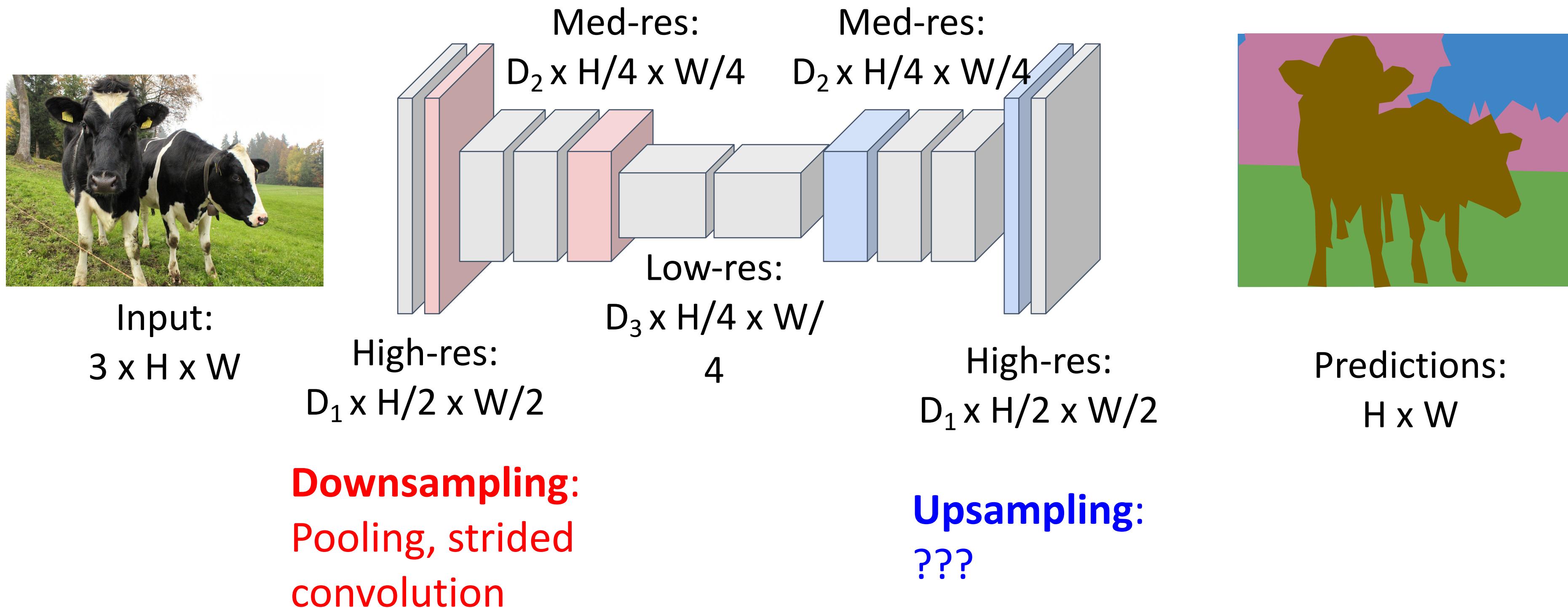


$$f_{x,y} = \sum_{i,j} f_{i,j} \max(0, 1 - |x - i|) \max(0, 1 - |y - j|) \quad i \in \{\lfloor x \rfloor - 1, \dots, \lceil x \rceil + 1\} \\ j \in \{\lfloor y \rfloor - 1, \dots, \lceil y \rceil + 1\}$$

Use two closest neighbors in x and y
to construct linear approximations

Fully Convolutional Network

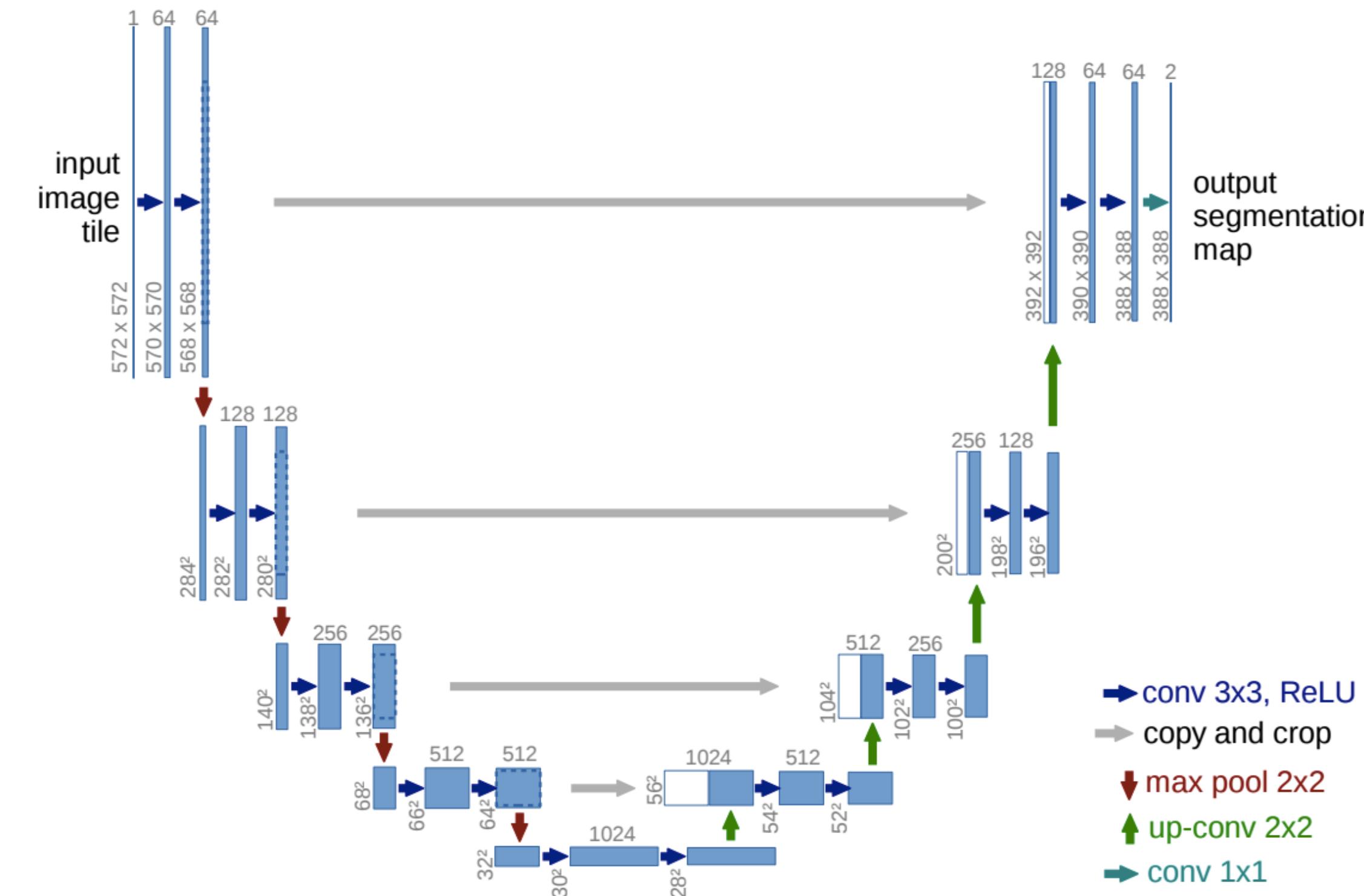
Design network as a bunch of convolutional layers, with
downsampling and **upsampling** inside the network!



U-Net

O. Ronneberger, P. Fischer, T. Brox, [U-Net: Convolutional Networks for Biomedical Image Segmentation](#), MICCAI 2015

- Like FCN, fuse upsampled higher-level feature maps with higher-res, lower-level feature maps
- Unlike FCN, fuse by concatenation, predict at the end



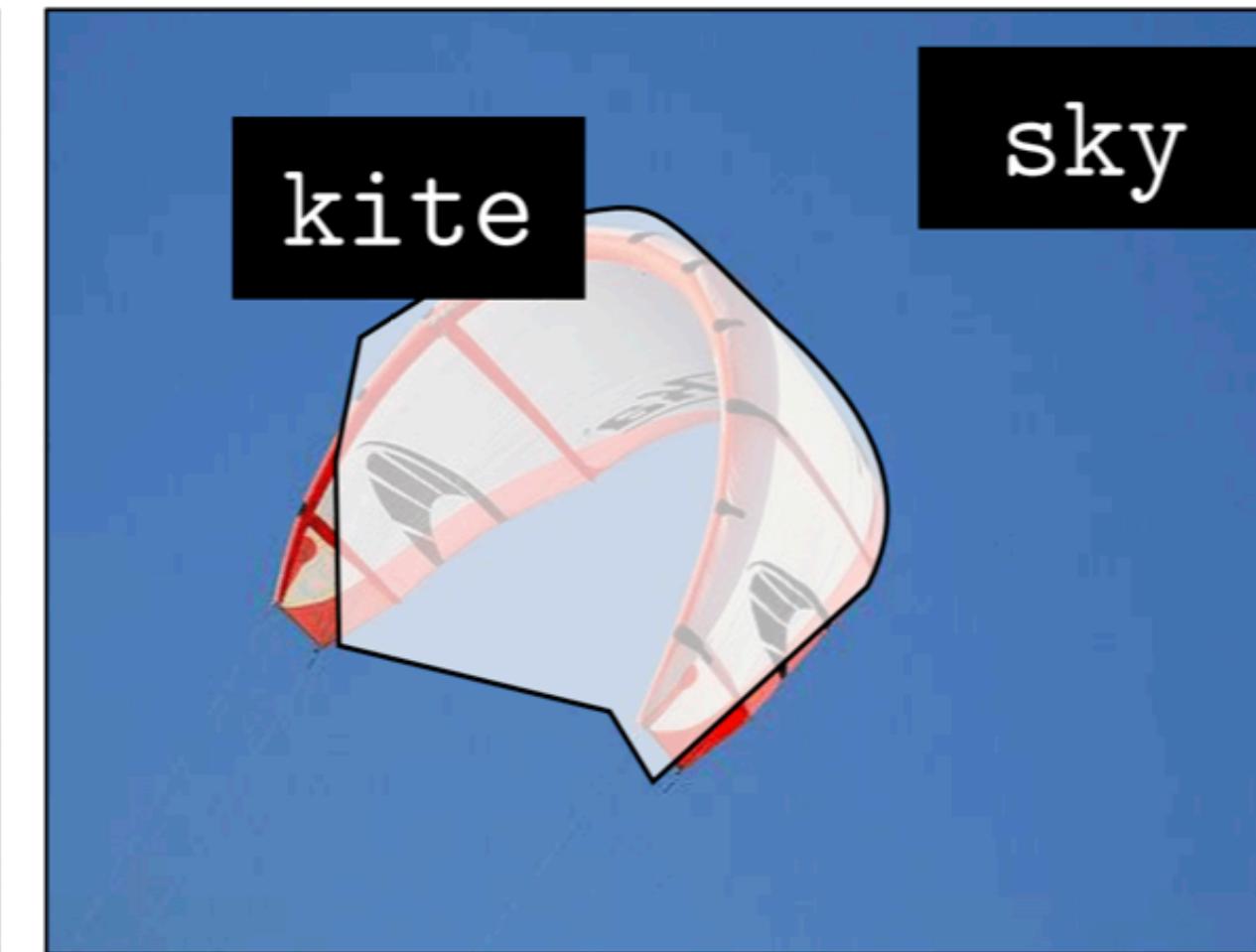


road
sidewalk
building
wall
fence
pole
traffic light
traffic sign
vegetation
terrain
sky
person
rider
car
truck
bus
train
motorcycle
bicycle

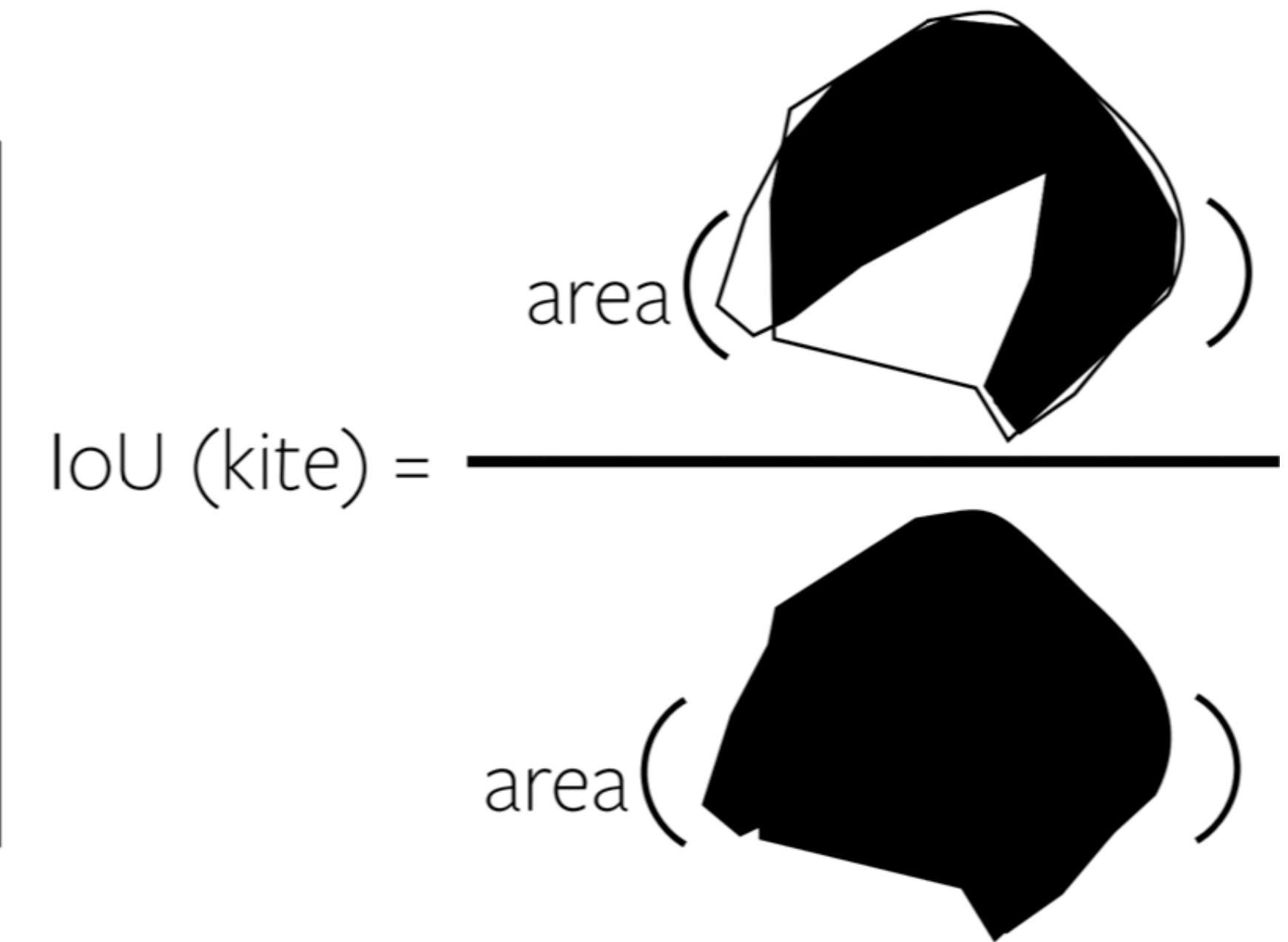
Evaluation of Semantic Segmentation



ground truth

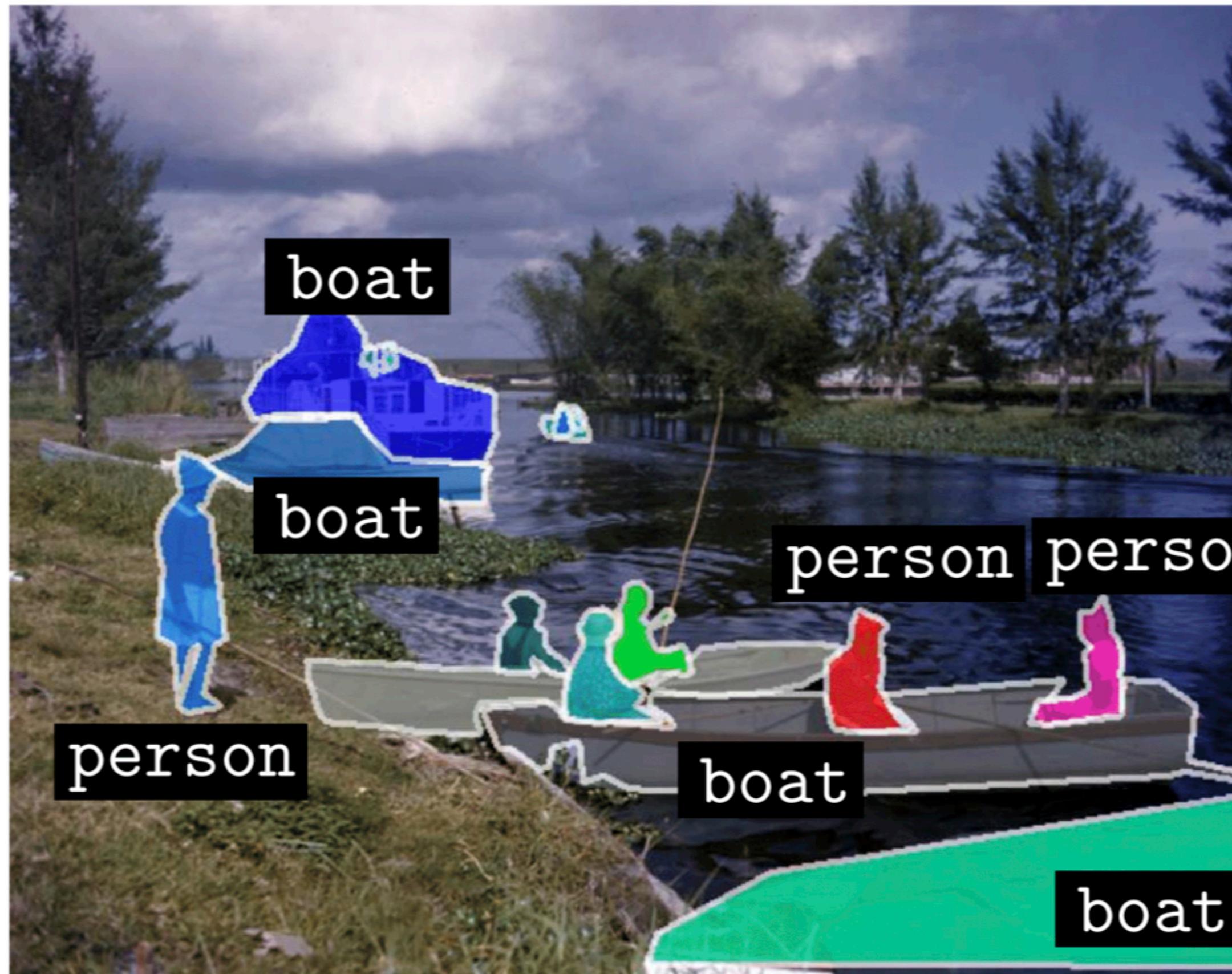


prediction



mIoU (mean IoU) per class

Instance and Semantic Segmentation



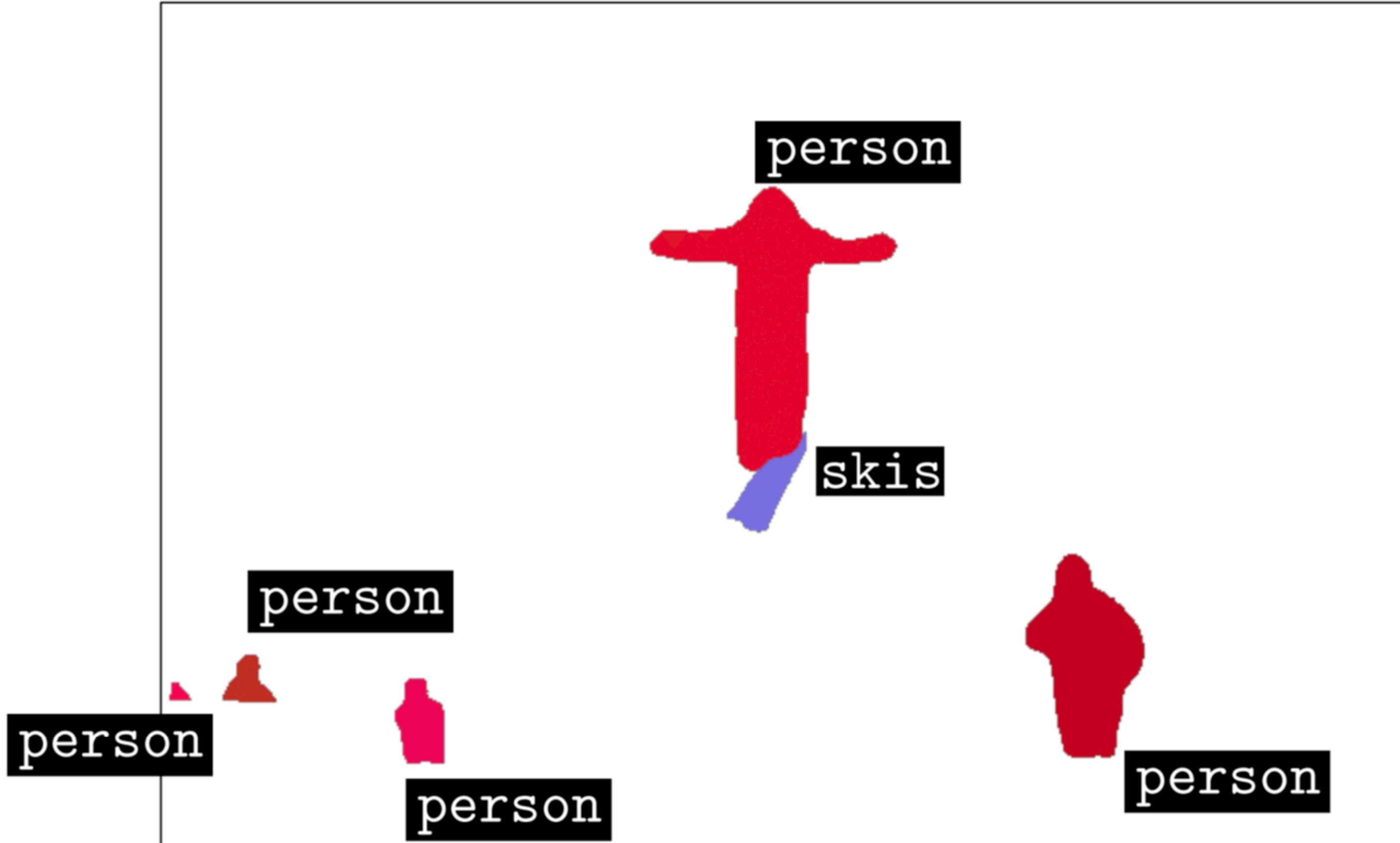
instance segmentation

real-world application likely requires both modalities



semantic segmentation

What do instance segmentation models see?



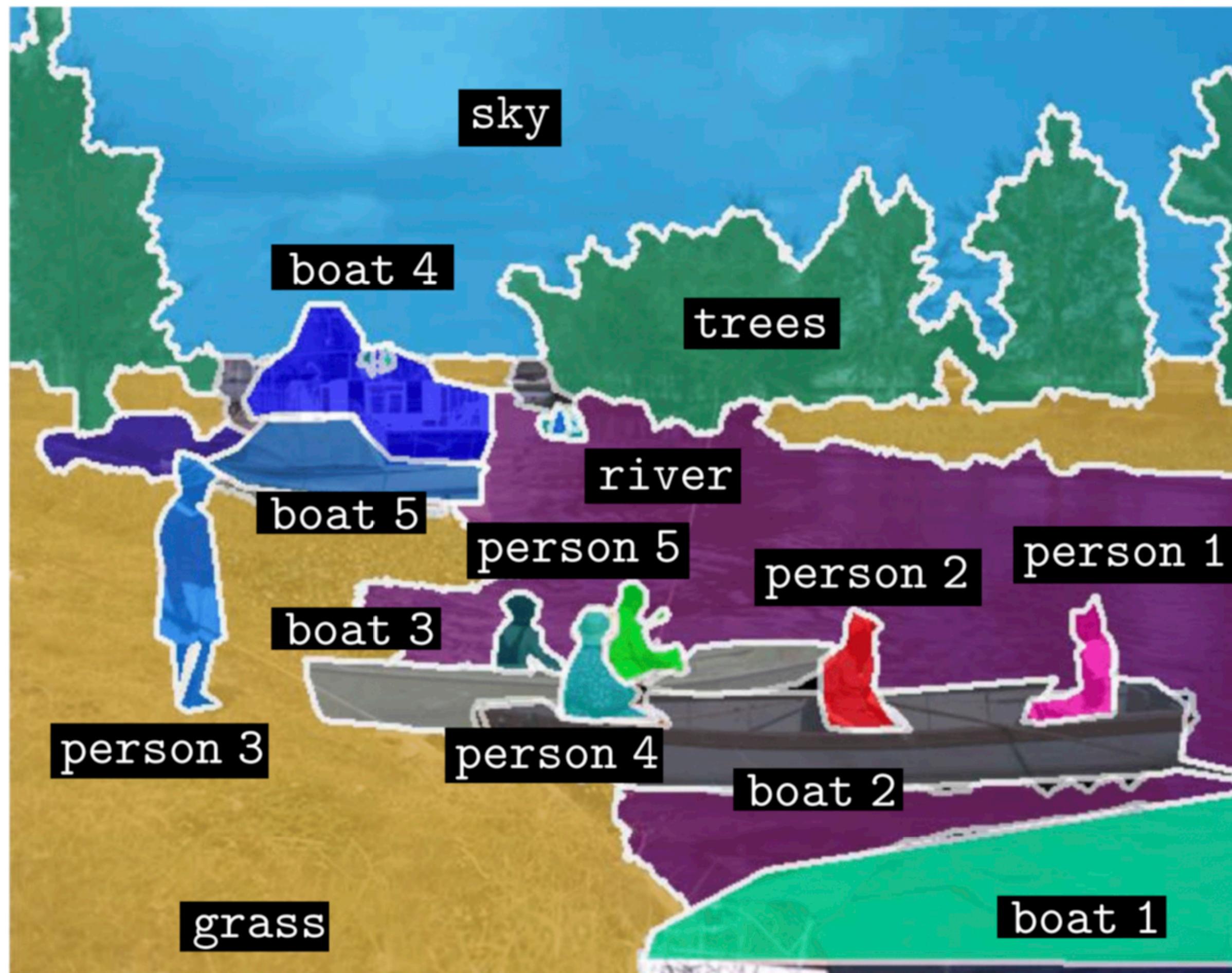
no understanding of the general scene layout

What do semantic segmentation models see?



Does not differentiate
different instances

Panoptic Segmentation: Unified Segmentation



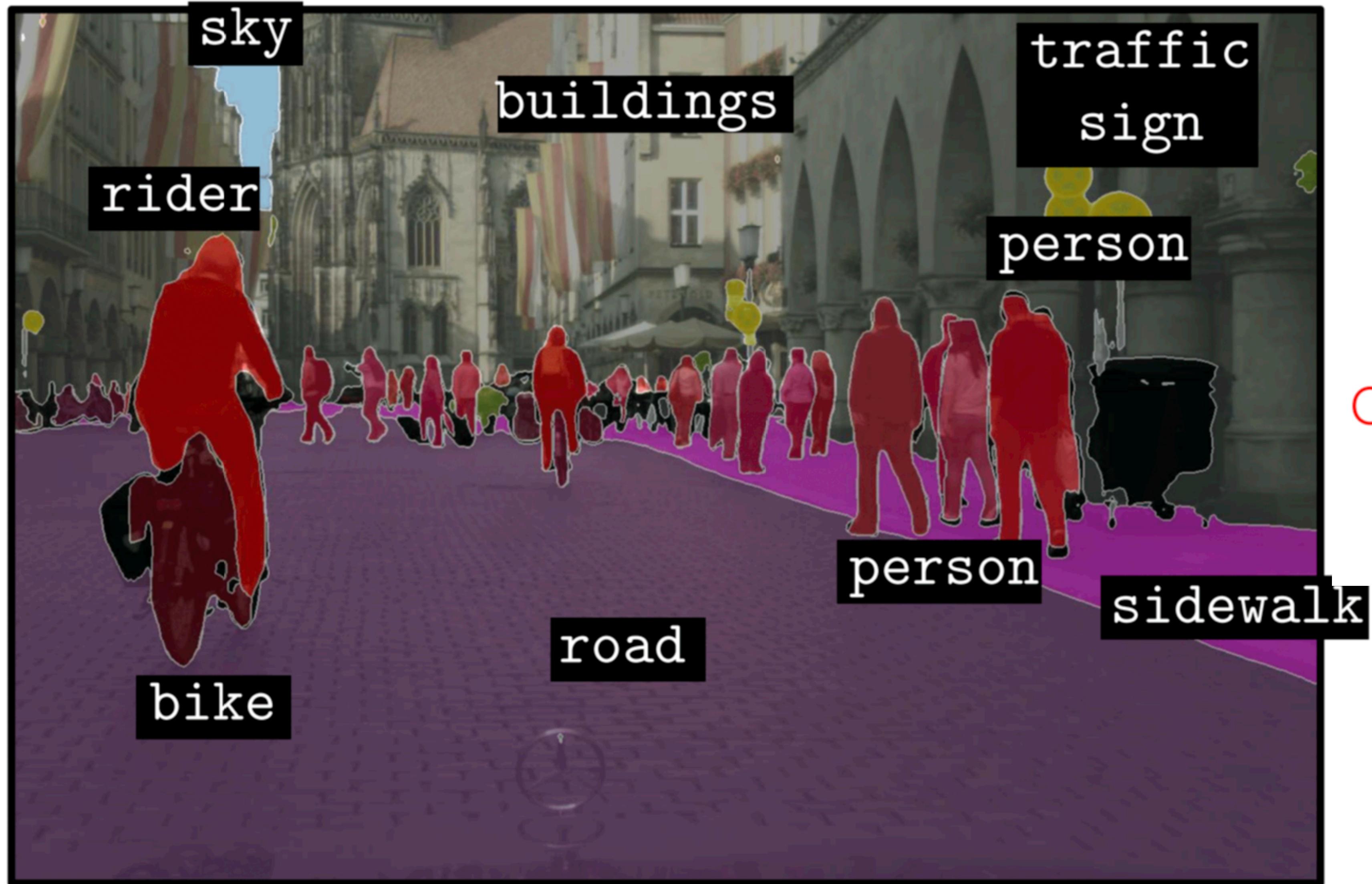
single task that combines semantic and instance segmentation

things: categories with instance-level annotation (person, boat)

stuff: categories without the notion of instances (sky, road)

Panoptic: see everything at once

Panoptic Segmentation



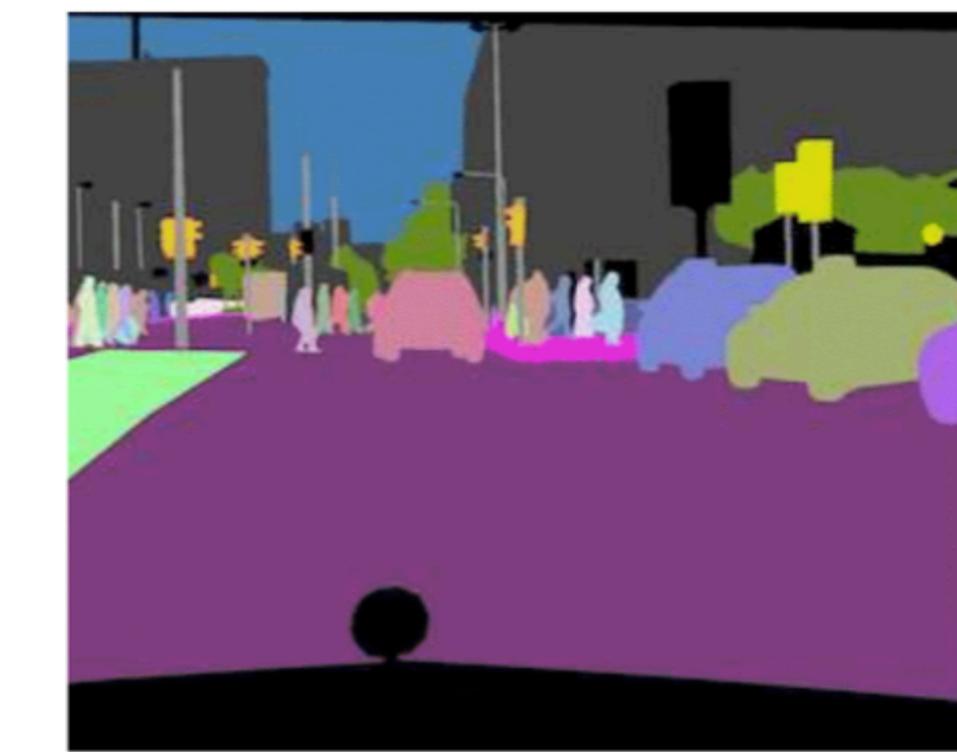
Available Panoptic Segmentation Datasets



CO (2014) + COCO-stuff (2017)
COCO-panoptic challenges:
ECCV`18, ICCV`19



Mapillary Vistas (2017)
Vistas-panoptic challenges:
ECCV`18, ICCV`19

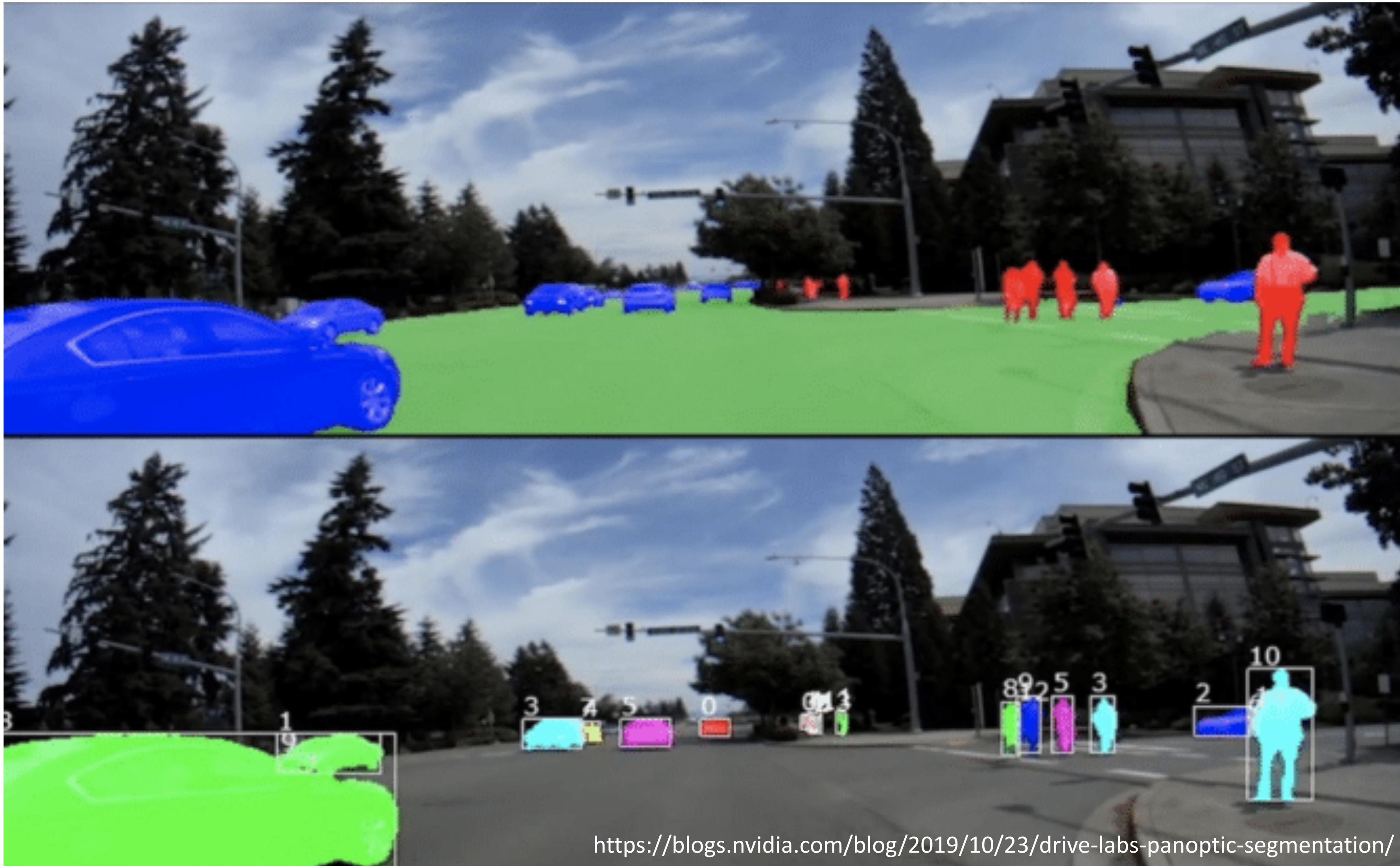


Cityscapes (2015)
panoptic test set
leaderboard (2019)

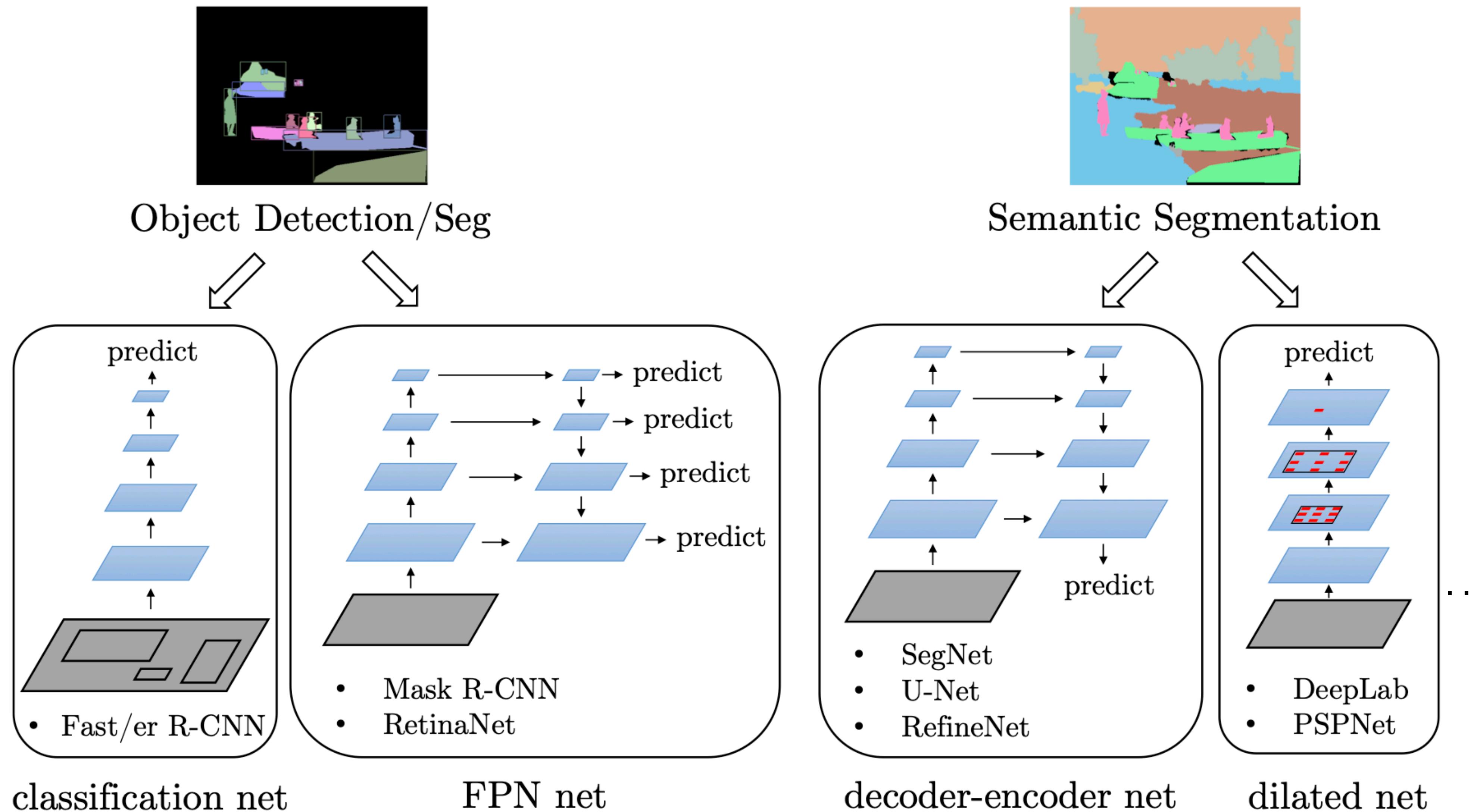


ADE2ok (2016)
>22k images, 150 categories

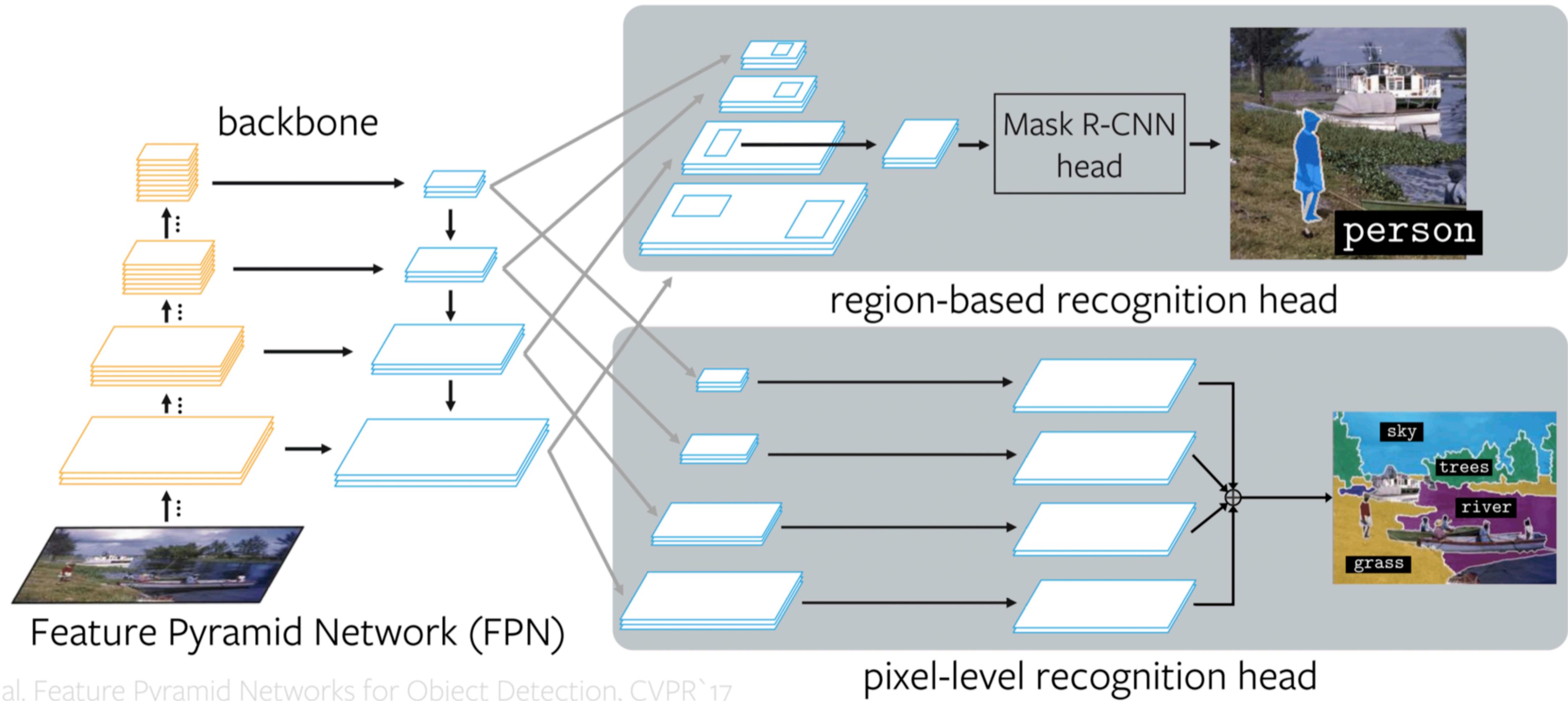
Panoptic Segmentation for Autonomous Driving



Deep Networks for Segmentation Tasks



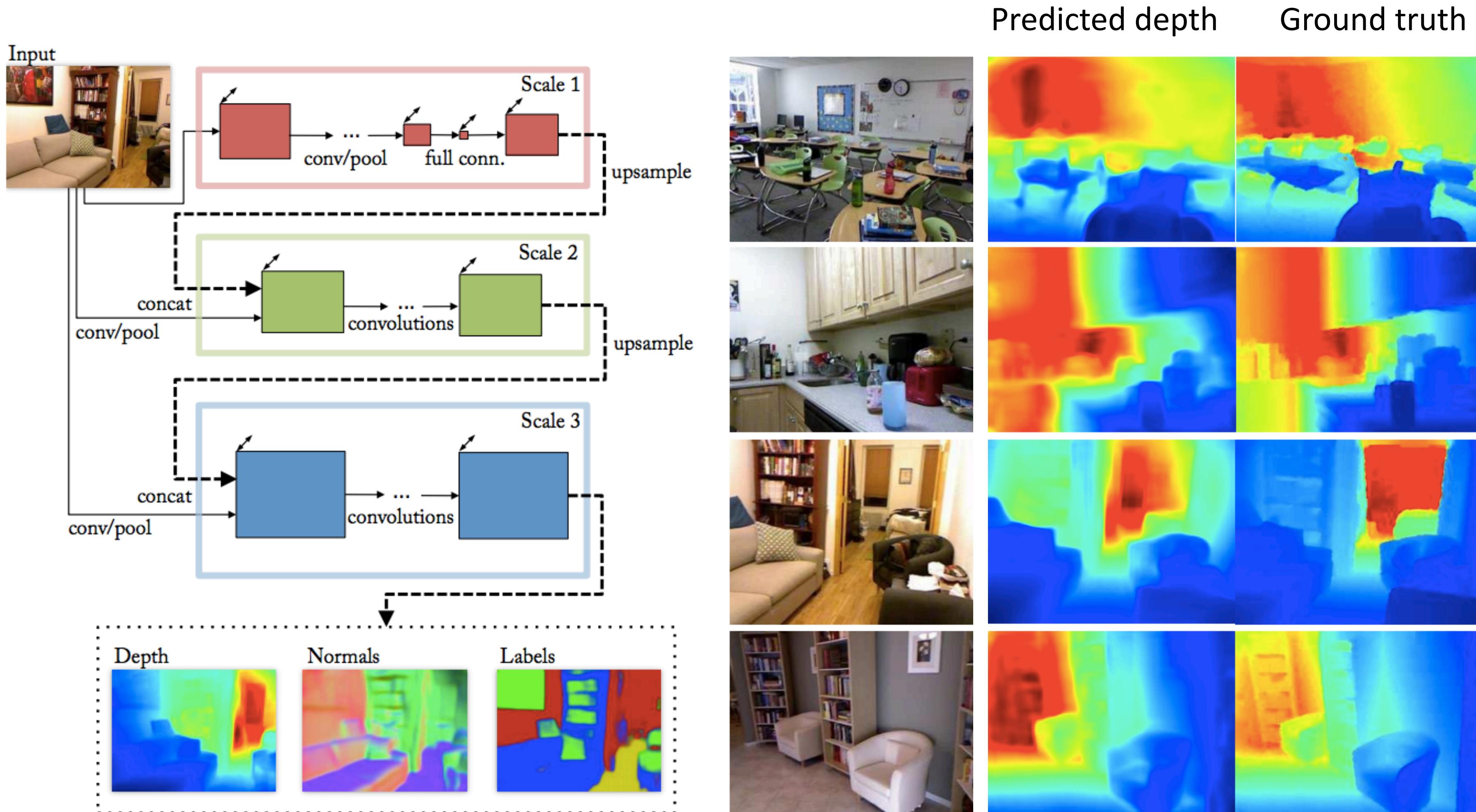
Panoptic FPN



et al. Feature Pyramid Networks for Object Detection, CVPR'17

Figure Credit: Alexander Kirillov

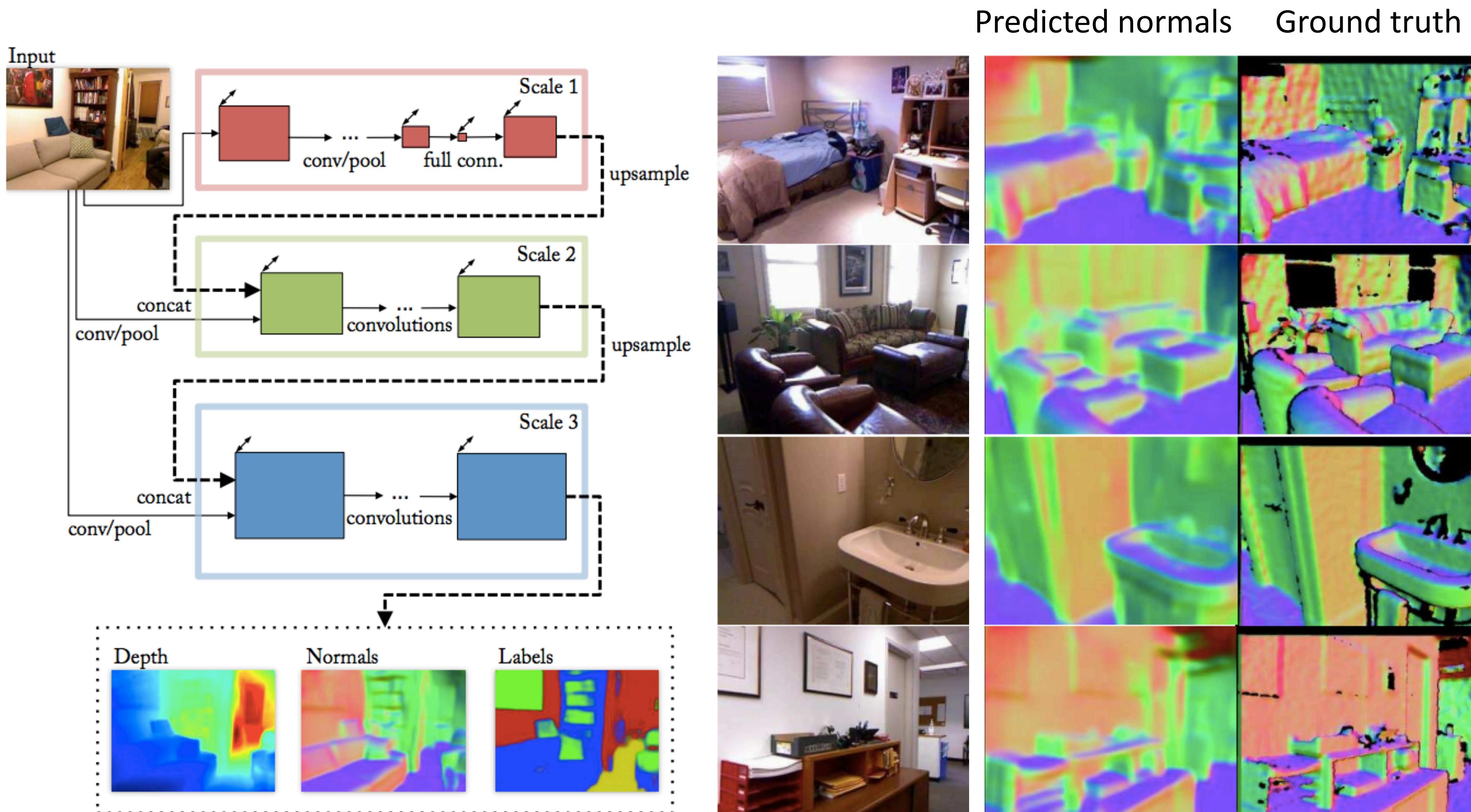
Dense Prediction: Depth and normal estimation



D. Eigen and R. Fergus, [Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-Scale Convolutional Architecture](#), ICCV 2015

Slide credit: S. Lazebnik

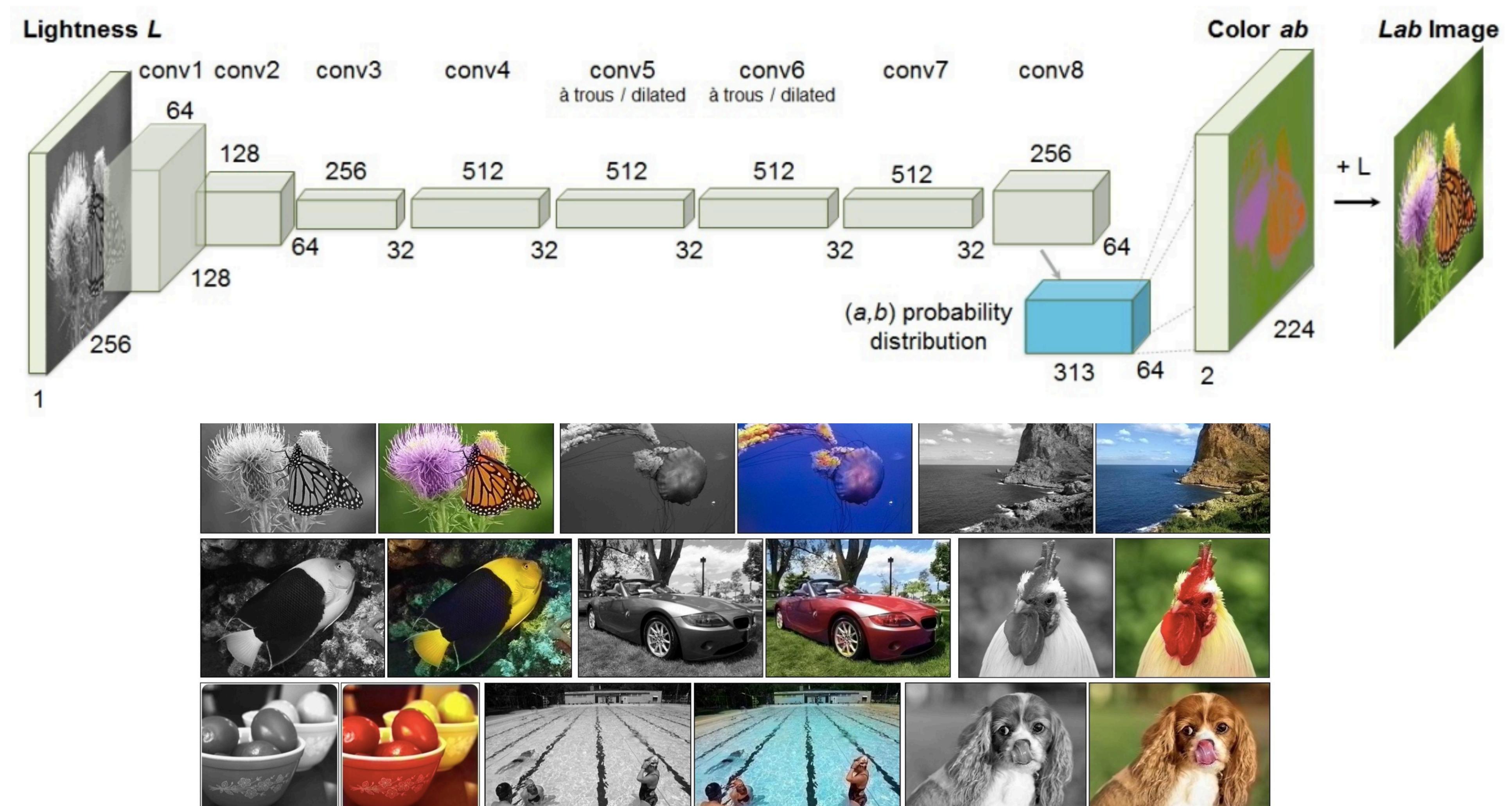
Dense Prediction: Depth and normal estimation



D. Eigen and R. Fergus, [Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-Scale Convolutional Architecture](#), ICCV 2015

Slide credit: S. Lazebnik

Dense Prediction: Colorization



Slide credit: S. Lazebnik

R. Zhang, P. Isola, and A. Efros, [Colorful Image Colorization](#), ECCV 2016