# Image Segmentation
# in the Era of Deep Learning

Institute of Information Theory
and Automation of the AS CR

# Outline

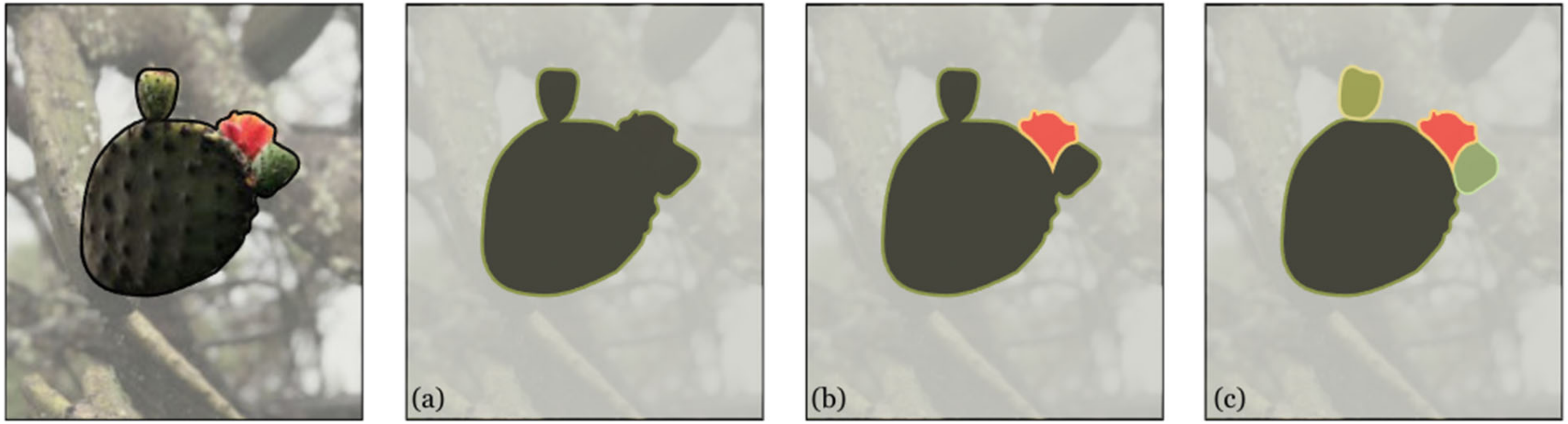
- Segmentation tasks
- Supervised training – datasets, loss
- Architectures
- Evaluations metrics

# Segmentation Tasks

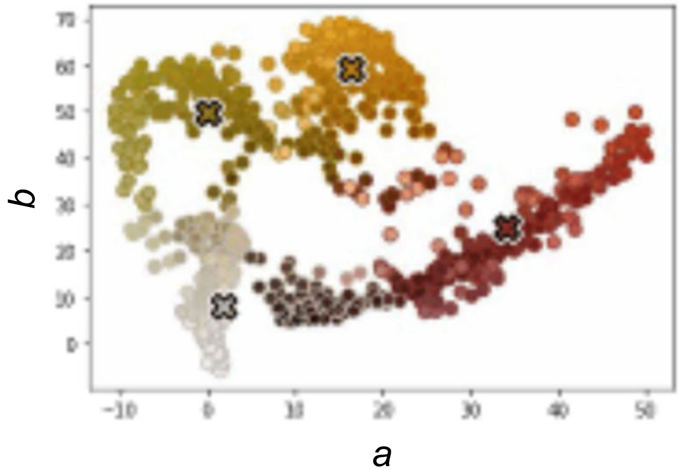
- Image classification
- Object detection
- Segmentation
  - Semantic - SS
  - Instance - IS
  - Panoptic (objects, stuff) - PS

Segmentation Survey, PAMI 24

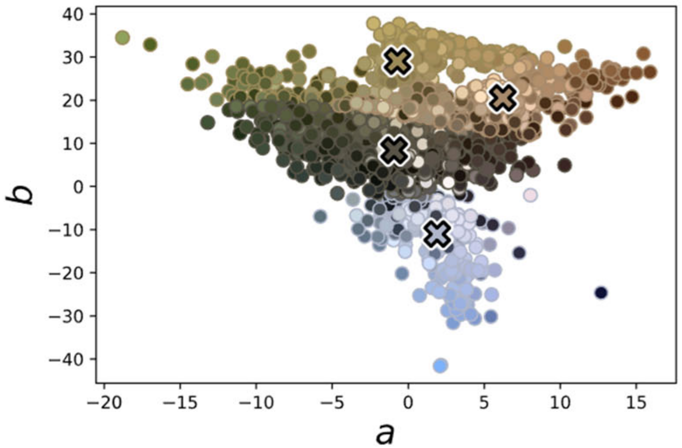
# What is segmentation?



# Clustering Problem





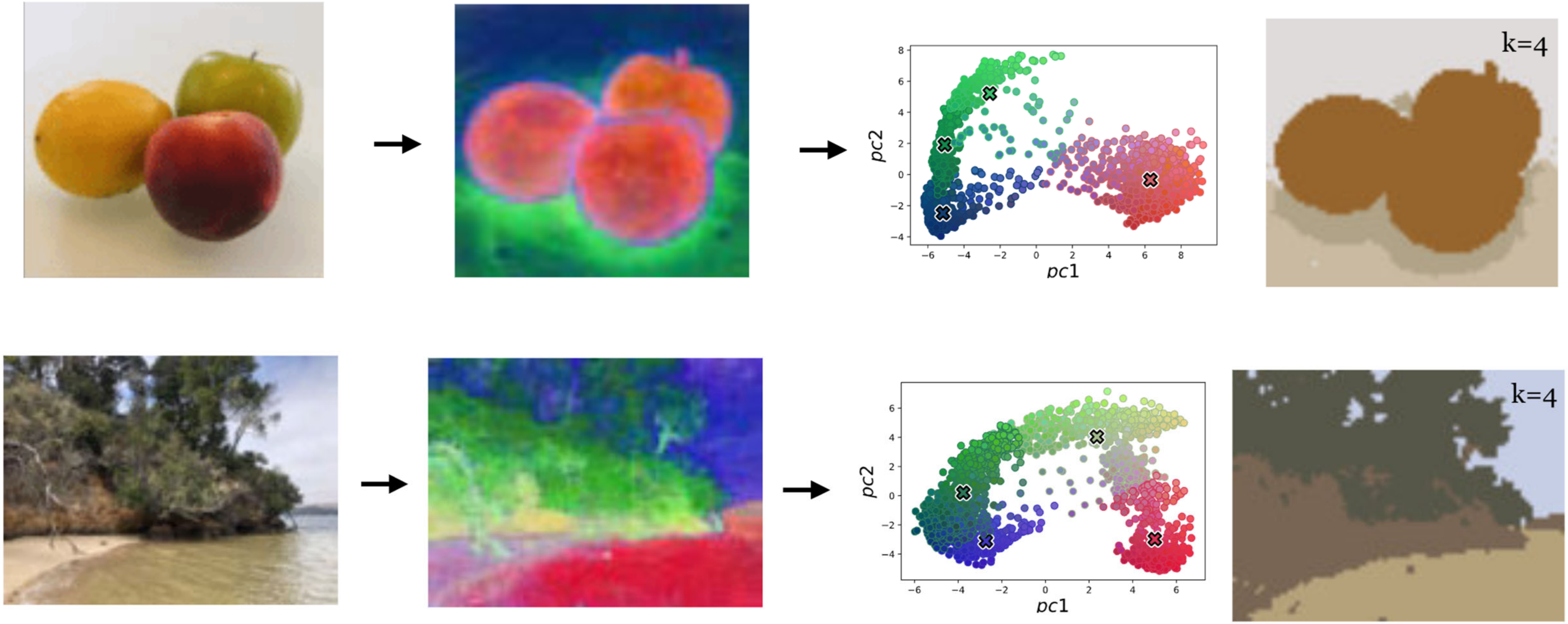




A. Torralba et al., Foundations of Computer Vision, MITPress

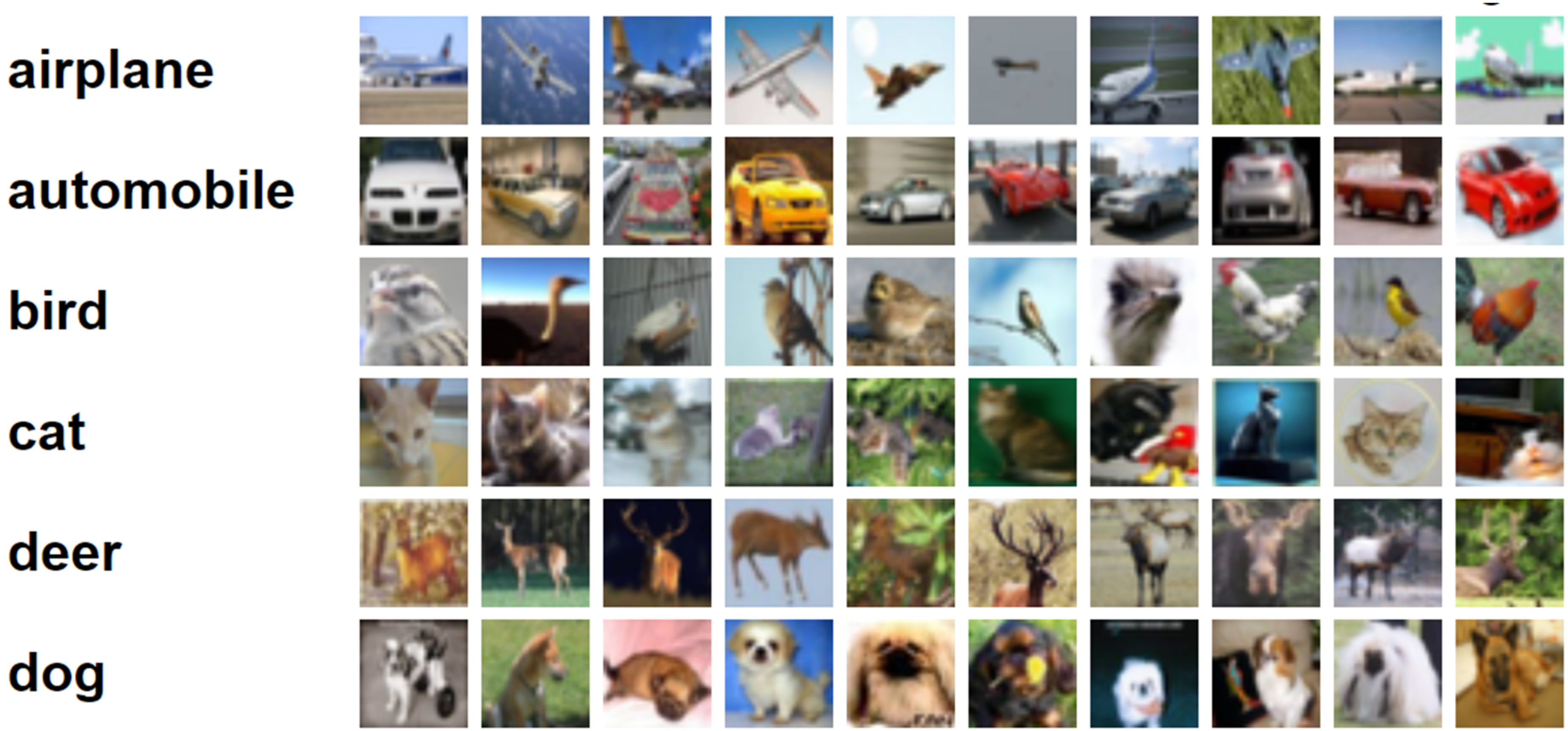
# Clustering Problem

| Input | Feature map (DINO) | Feature space (2 PCA components) | Segmentation (k-means clustering) |
|---|---|---|---|



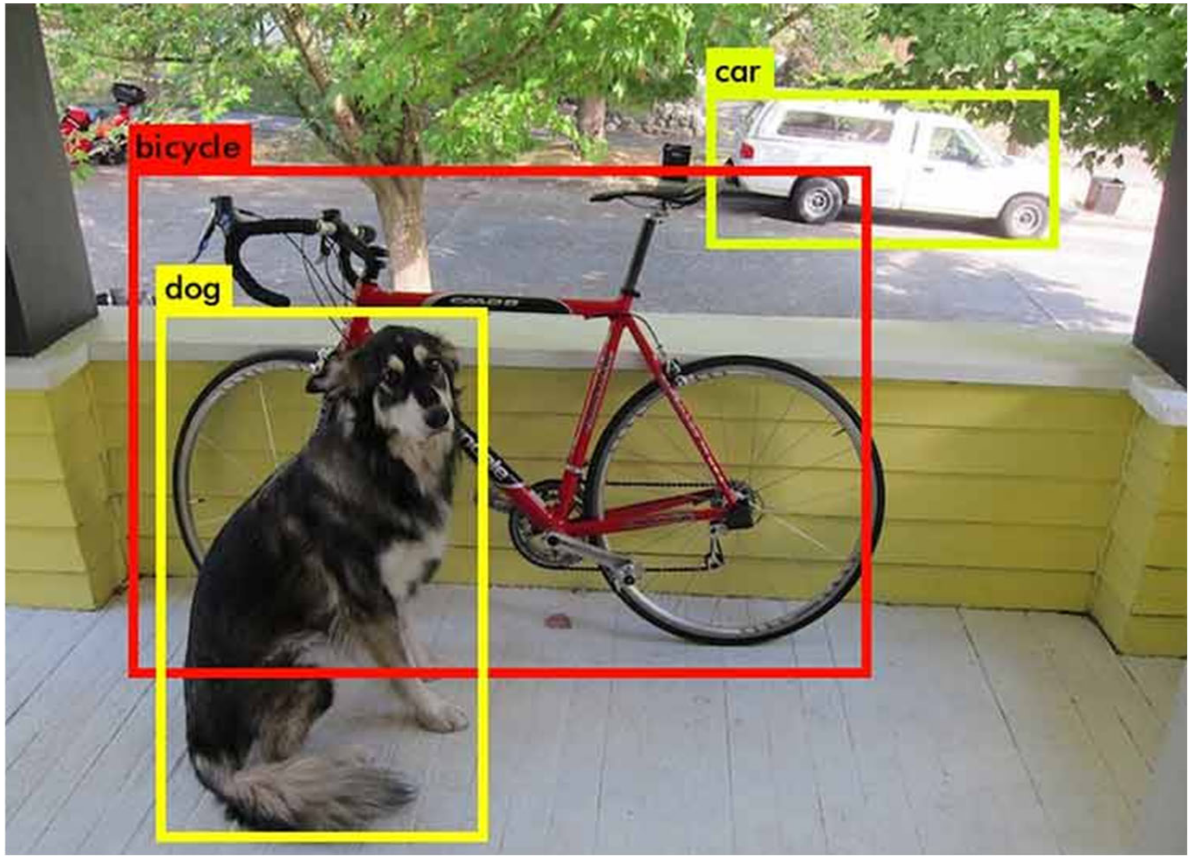
# Image Classification

- Image -> class label



# Object Detection

- Image -> { label, bounding_box (x,y,w,h) }



# Segmentation

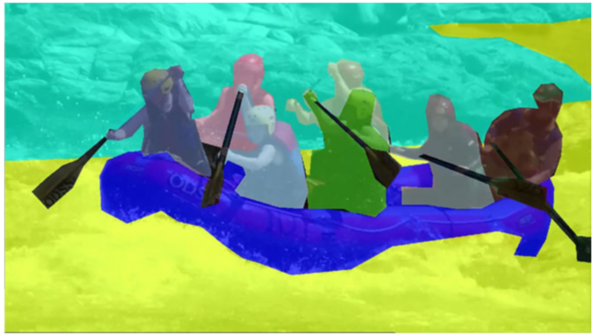
- Semantic

label and mask

- Instance (countable objects = things)

label and mask per instance

- Panoptic (semantic + instance)
- things (person) : label and mask per instance
- stuff (water): label and mask

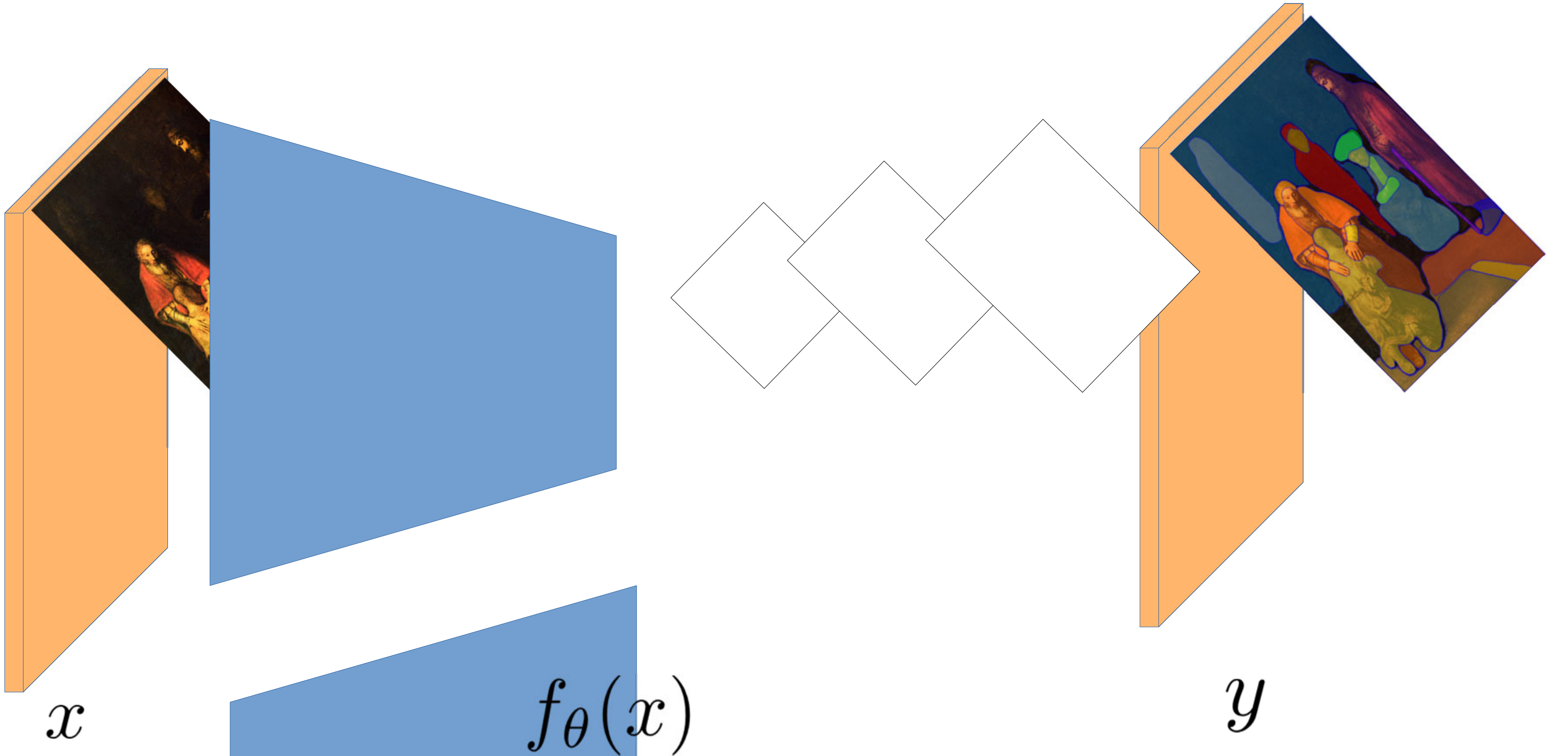


# Datasets for SS,IS,PS

- **Pascal Visual Object Classes** (Pascal VOC): many different object classes, bounding boxes and robust segmentation maps. (20 object categories, only for SS)
- **MS COCO**: 330k images and annotations for many tasks including image captioning (80 thing and 91 stuff categories)
- **Cityscapes**: data from urban environments mad up of 5k images with 20k annotations and 30 class labels.
- **ADE20k**: 20k annotated images of scene categories from the SUN and Places database.
- **Cell Tracking Challenge**: 2D and 3D time-lapse images (for SS and IS)

# Supervised Training

Annotation: {x,y} = {“input image”, “output segmentation”}




Training: $\min_{\theta} \sum_{m} \mathcal{L}(f_\theta(x^{(m)}), y^{(m)})$ Must be differentiable!

# Cross-Entropy Loss

- Training set: $(x^{(i)}, y^{(i)})$ i … image (pixel) index

$$y^{(i)} \in \mathbb{R}^C$$

- one-hot vector

$$y_c^{(i)} = \begin{cases} 1 & \text{if } x^{(i)} \text{ is from the c-th class} \\ 0 & \text{elsewhere} \end{cases}$$

$$y^{(i)} = [\underset{\text{1. class (car)}}{0}, \quad \underset{\text{2. class (bird)}}{\color{red}{1}}, \quad 0, \quad \ldots, \quad \underset{\text{C-th class}}{0}]$$

- Prediction: $\hat{y}^{(i)}$

- CE Loss:

$$L = -\sum_{i \in \mathbb{B}} \sum_{c=1}^{C} y_c^{(i)} \log(\hat{y}_c^{(i)})$$

- Binary CE: $C = 2$

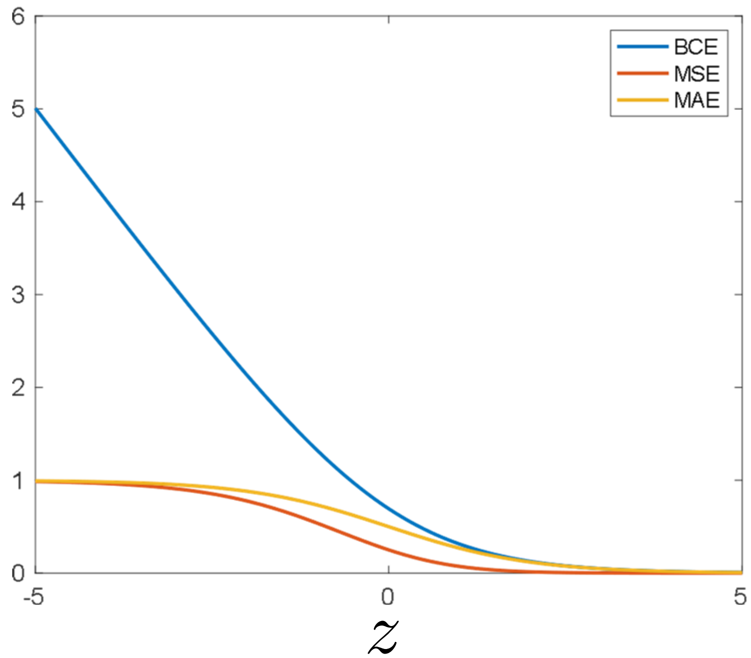
$$L = -\sum_{i} y_1^{(i)} \log(\hat{y}_1^{(i)}) + (1 - y_1^{(i)}) \log(1 - \hat{y}_1^{(i)})$$

# $L_p$ norm

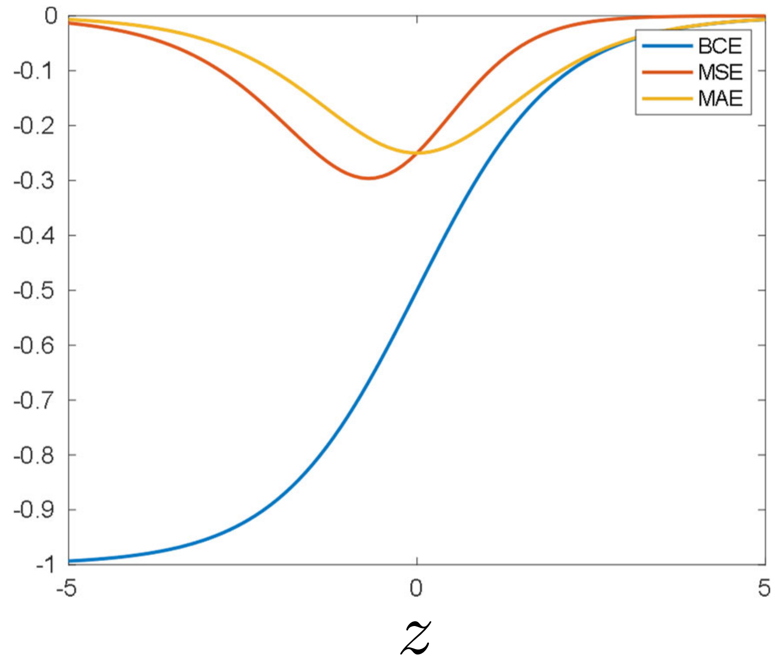
- GT: $y \in \mathbb{R}^C$
- Prediction: $\hat{y} = \sigma(z)$
- $L_2$: $\sum_{c=1}^{C} (y_c - \hat{y}_c)^2$

CE: $-\sum_{c=1}^{C} y_c \log(\hat{y}_c)$

- 

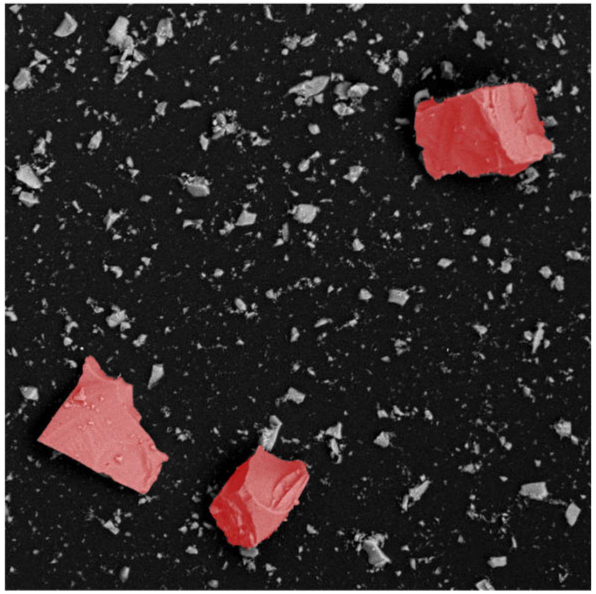



$$\sum_{c=1}^{C} |y_c - \hat{y}_c|$$



# Unbalanced classes

- CE: larger areas contribute to the error more



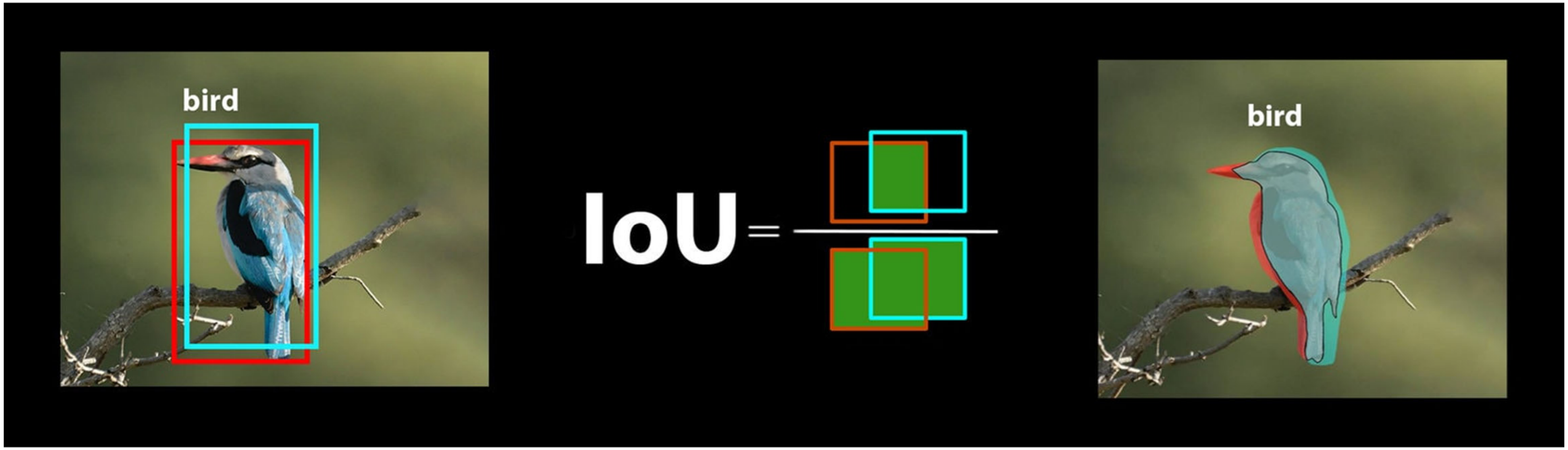
- weighted CE:

$$L = -\sum_{i \in \mathbb{B}} \sum_{c=1}^{C} w_c y_c^{(i)} \log(\hat{y}_c^{(i)})$$

- IoU

# IoU - Intersection over Union

- Jaccard index




$$\mathrm{IoU} \in \langle 0, 1 \rangle \qquad L = 1 - \mathrm{IoU}$$

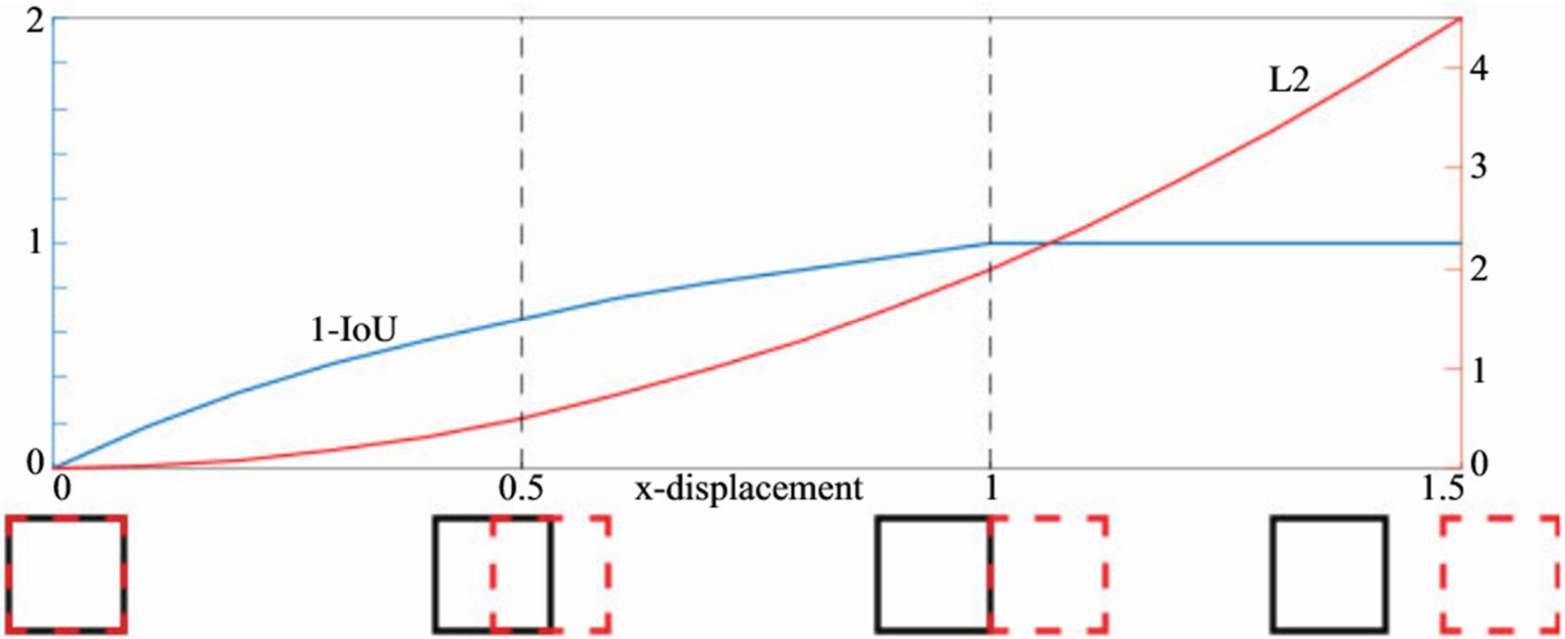
- For masks:

$$L = 1 - \sum_{c=1}^{C} \frac{\sum_i y_c^{(i)} \hat{y}_c^{(i)}}{\sum_i y_c^{(i)} + \hat{y}_c^{(i)} - y_c^{(i)} \hat{y}_c^{(i)}}$$

- Dice loss (3DV 2016)

# Bounding Box Loss

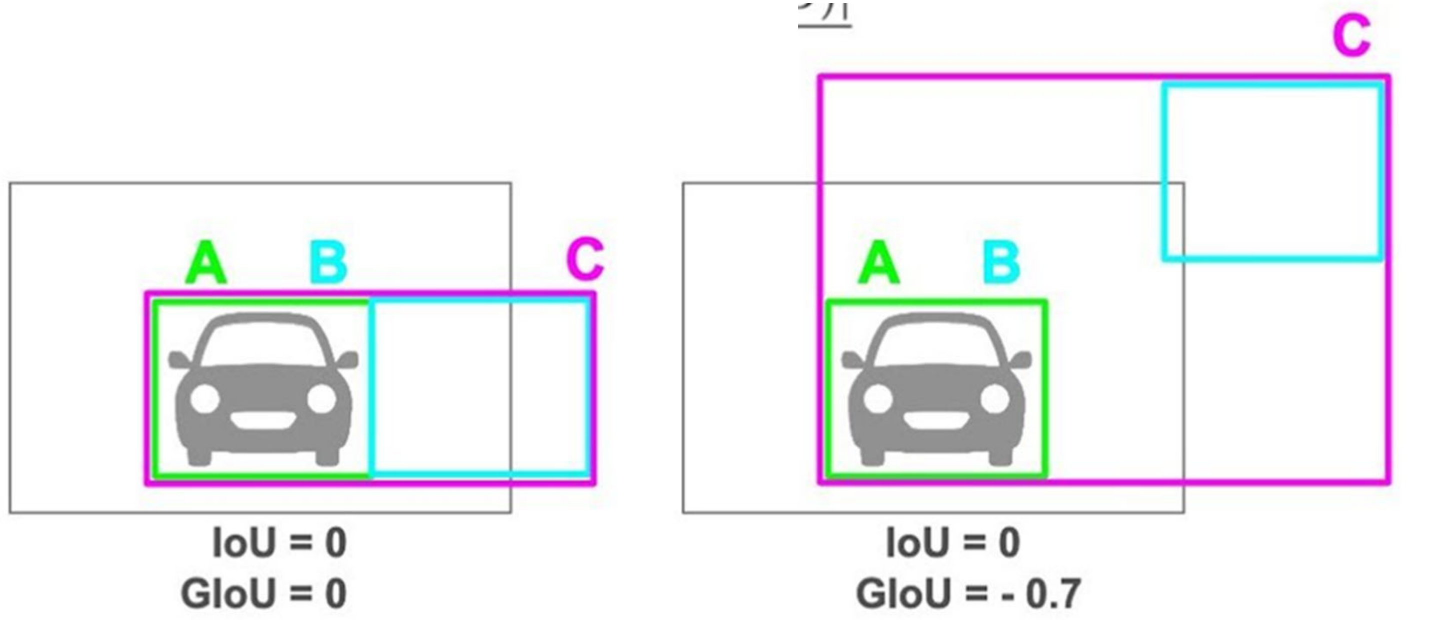
- Bounding box: [x,y,width,height]
- $L_p$ norm: same error irrespective of the area size
- IoU: no overlap has zero gradient




- Loss = $L_p$ + ( 1-IoU )

# Generalized IoU

$$\mathrm{GIoU} = \mathrm{IoU} - \frac{|C \setminus (A \cup B)|}{|C|}$$




$$\mathrm{GIoU} \in \langle -1, 1 \rangle$$

Defined only for bounding boxes.

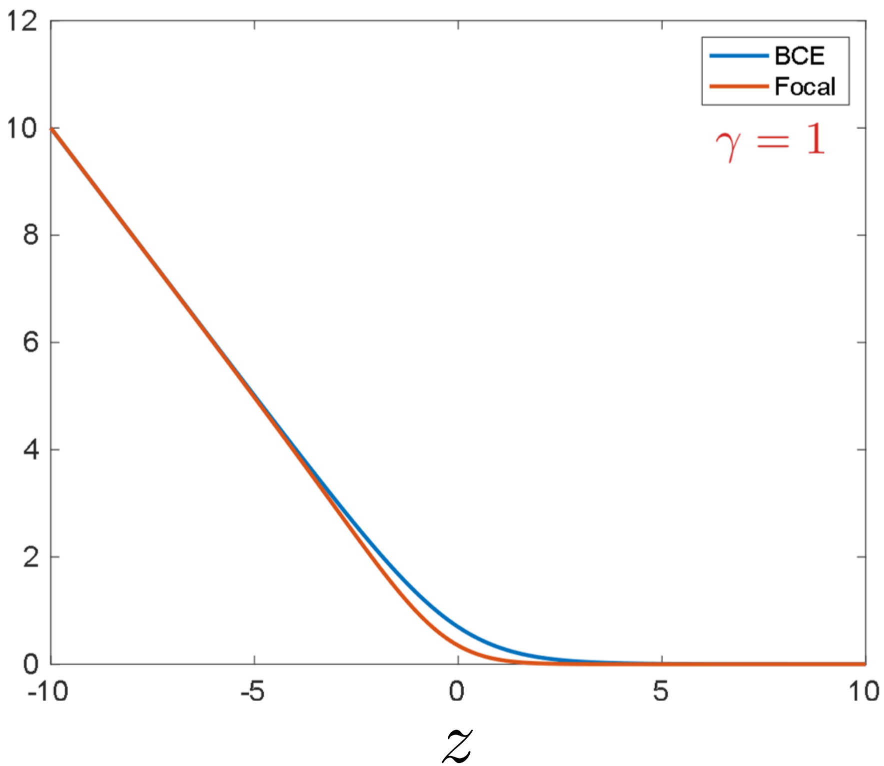
# Focal Loss

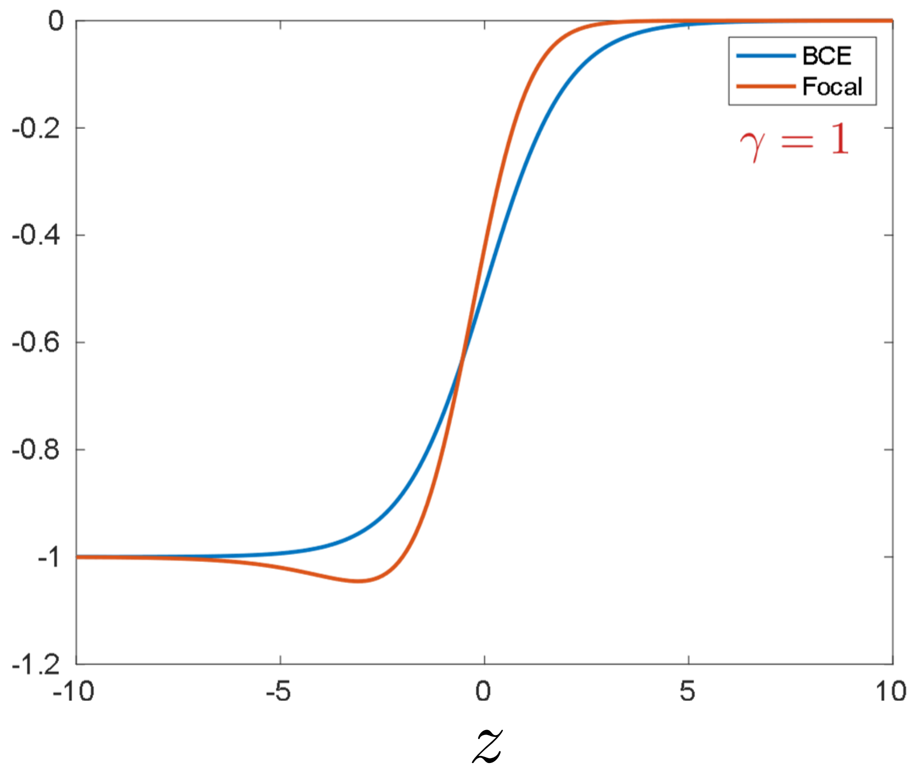
- CE:

$$-\sum_{c=1}^{C} y_c \log(\hat{y}_c)$$

Focal:

$$-\sum_{c=1}^{C} y_c \underline{(1-\hat{y}_c)^{\gamma}} \log(\hat{y}_c)$$



# Other Semantic Loss Functions

- Dice loss (3DV 2016)
- Focal loss (ICCV 2017)
- …
- A survey of semantic losses (arXiV 2023)

# Semantic Segmentation

- Classify every pixel
- Output: $y \in \mathbb{R}^{W \times H \times C}$

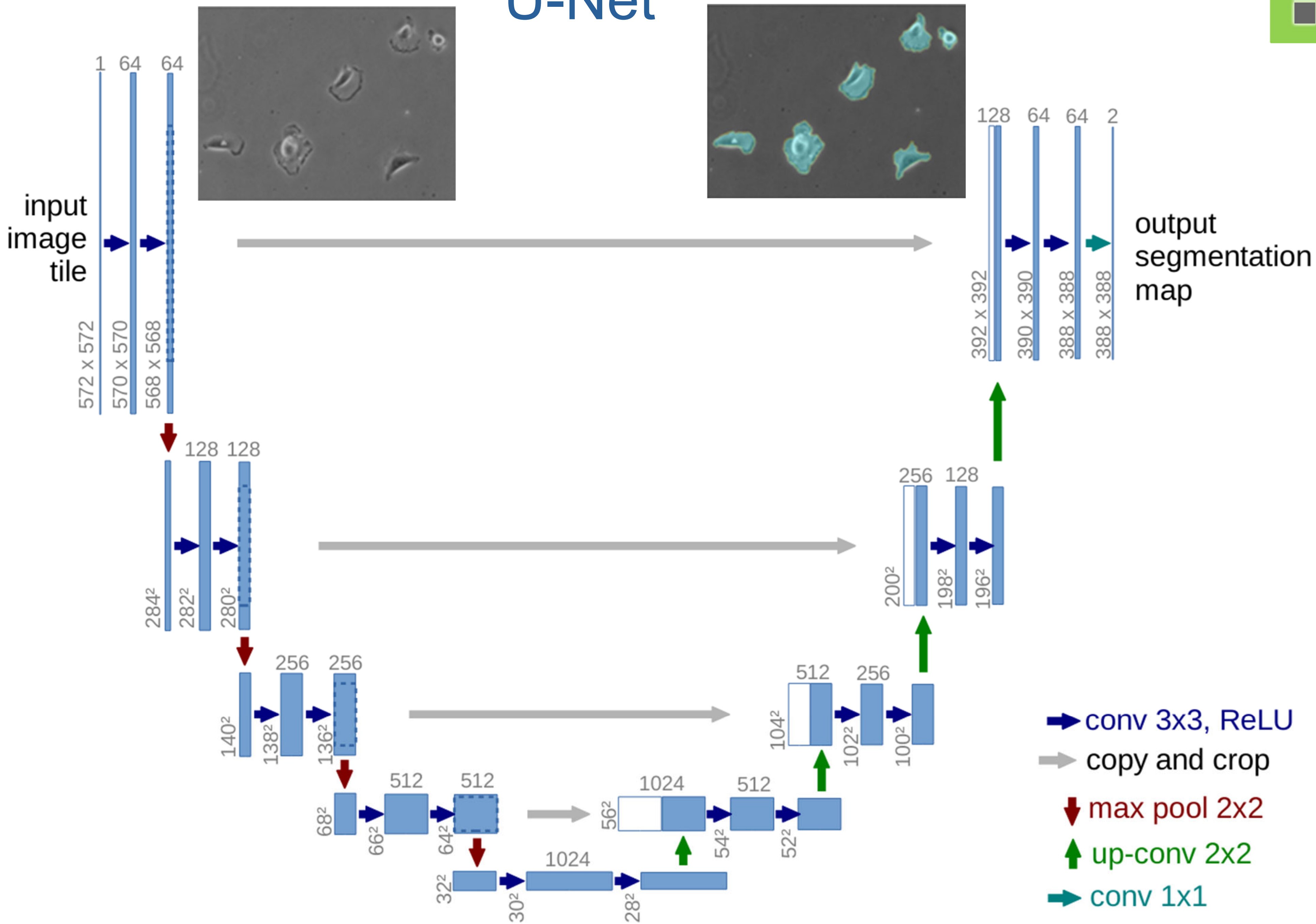
$$[p_1, p_2, \ldots, p_i, \ldots, p_{C-1}, p_C]$$

0 0 1 0 0

$H$

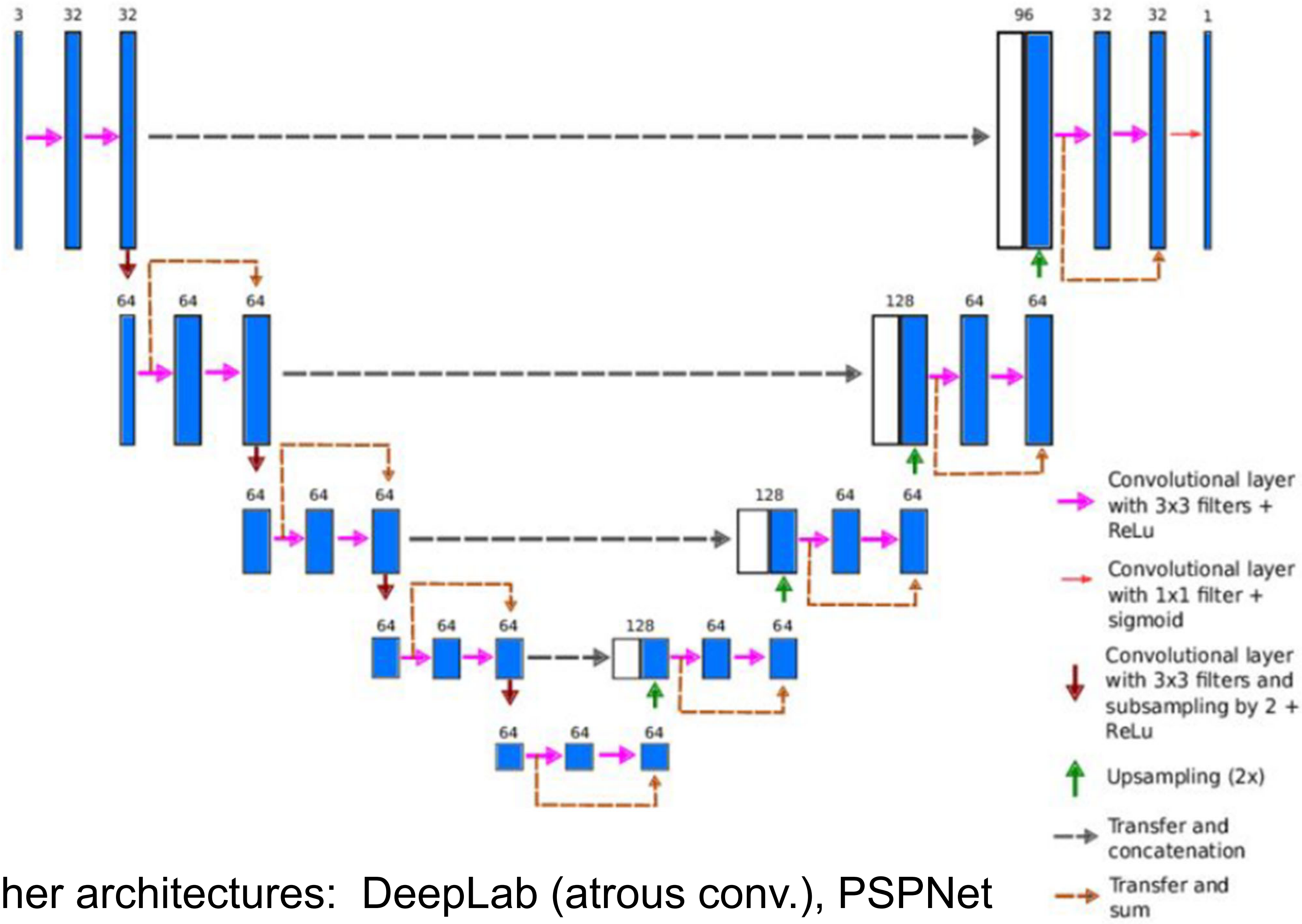
$W$

# U-Net




Ronneberger et al., U-Net, 2015

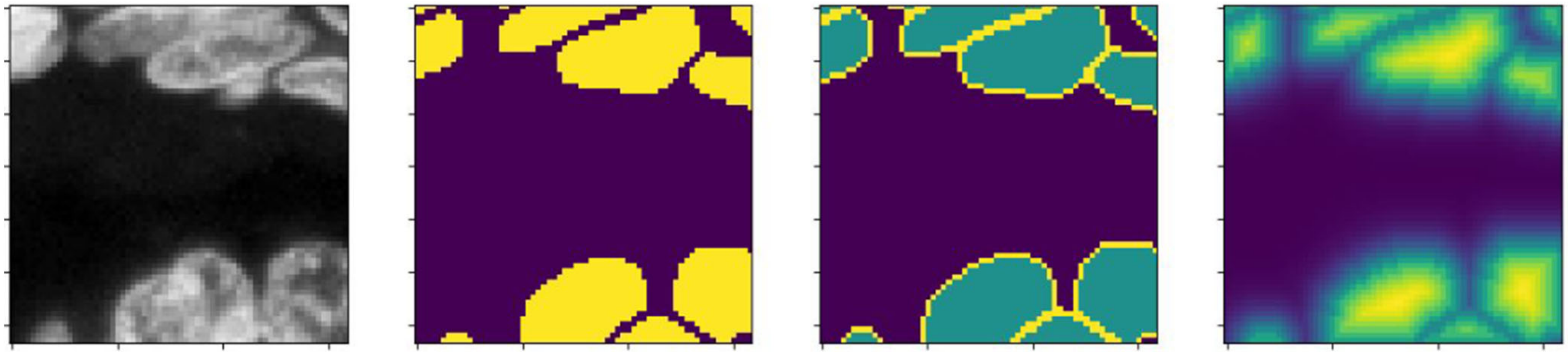
# Res U-Net




Other architectures: DeepLab (atrous conv.), PSPNet

# Bottom-Up IS

- Clustering in the (feature & spatial) domain
- Do not predict just object probability per pixel
- Predict distance from the border

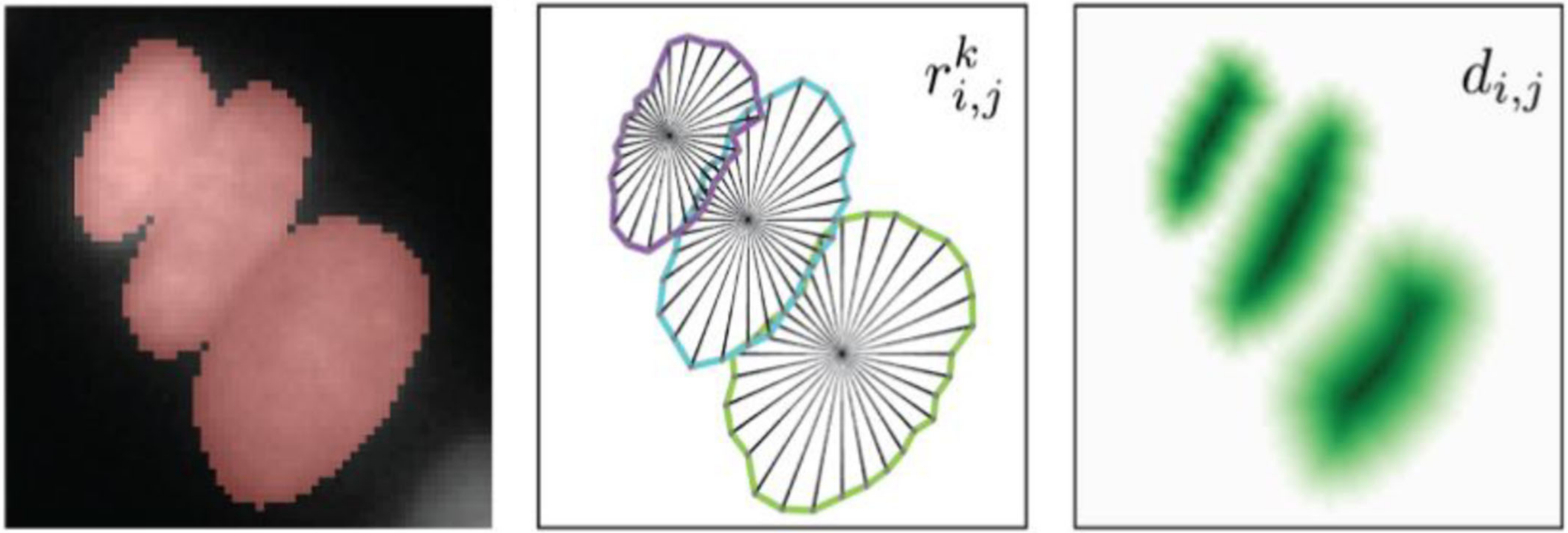


SDM, StarDist, Cellpose, Omnipose, DenoiSeg

SDM 2019

# StarDist

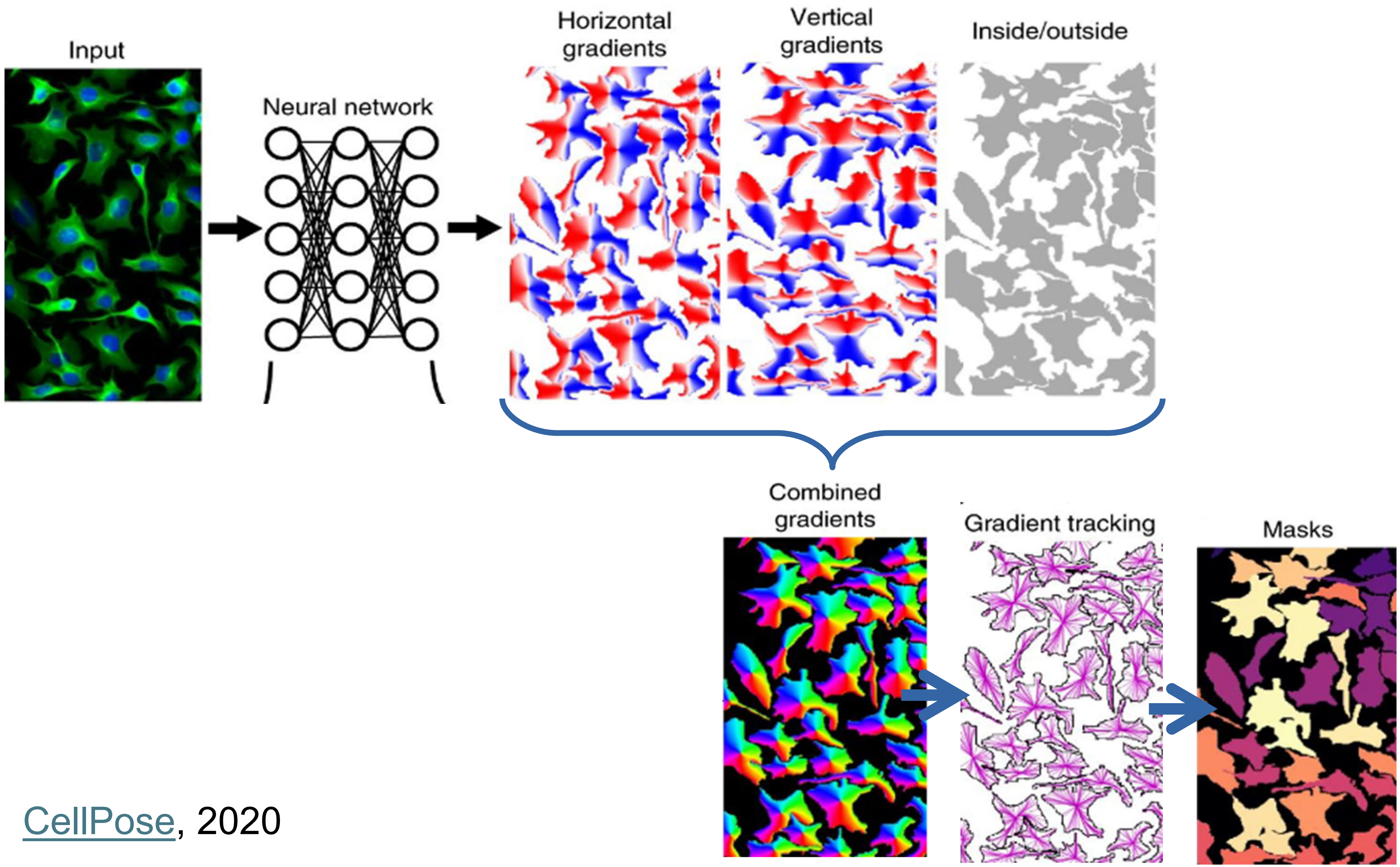
- Predict distance map and star-shape polygon




StarDist, 2018

# CellPose

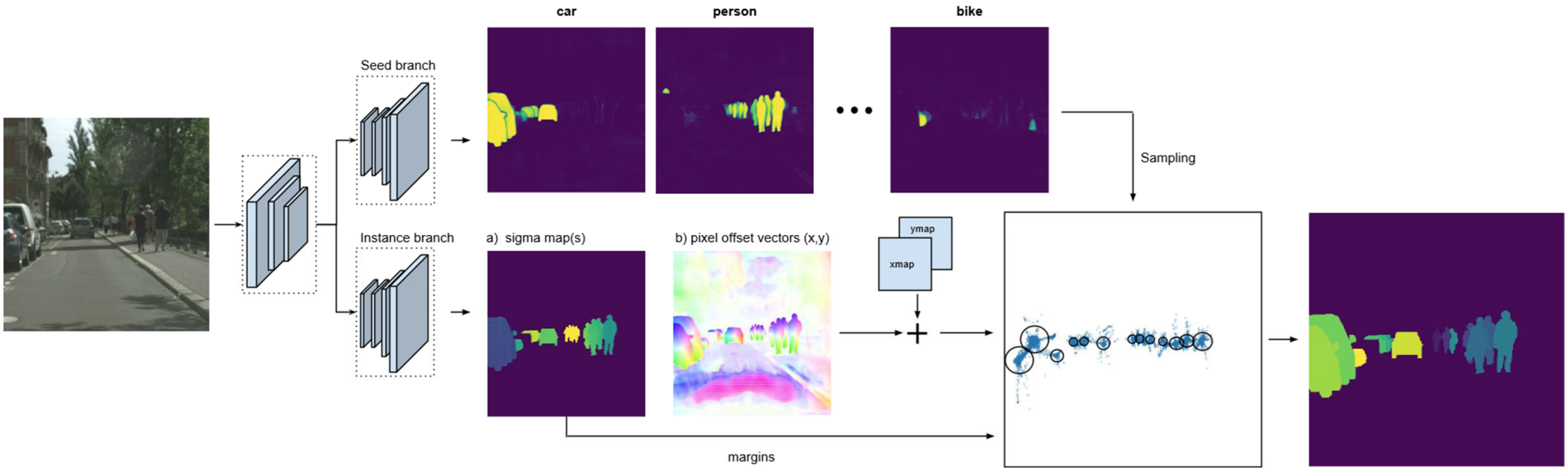
- Predict binary mask and gradient flow




[CellPose](), 2020

# Pixel offset

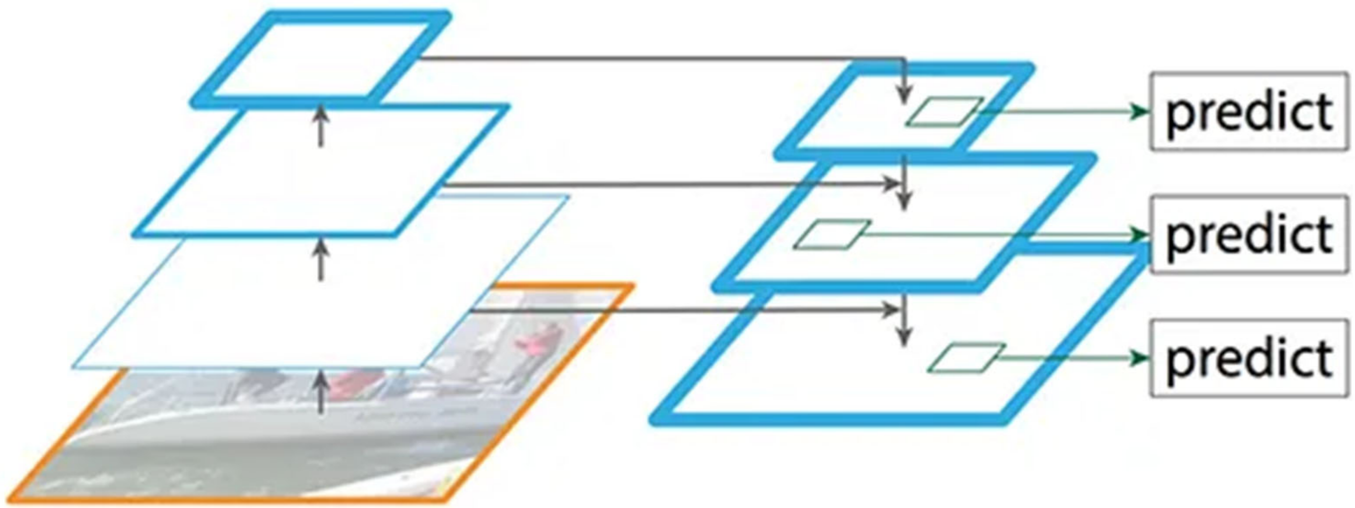
- Predict distance, pixel offset from the center, sigma maps (cluster size)




CVPR 2019

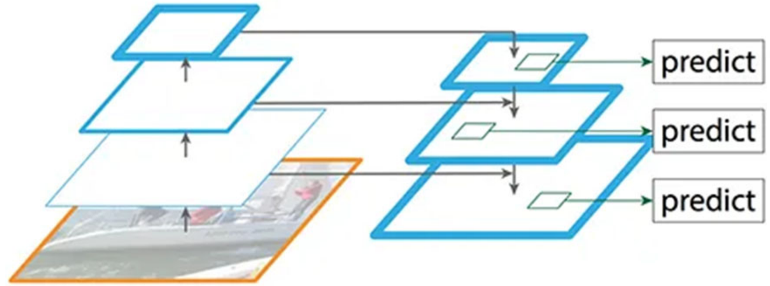
# Object Detection

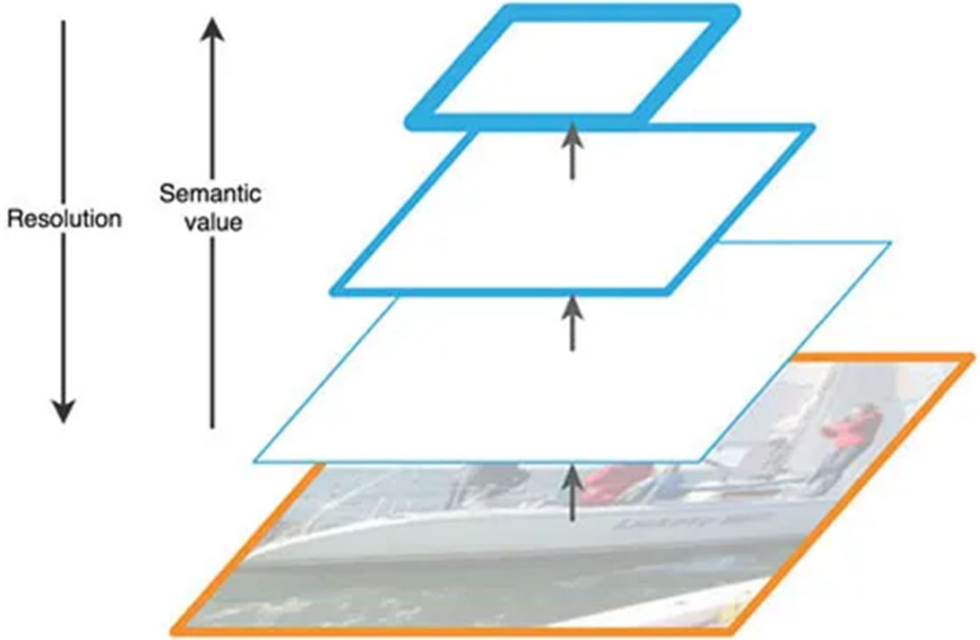
- Extract features – Feature Pyramid Network
- (Select regions – Region Proposal Network)
- Classify each region (objectness, class, bounding box)
- Non-maximum suppression



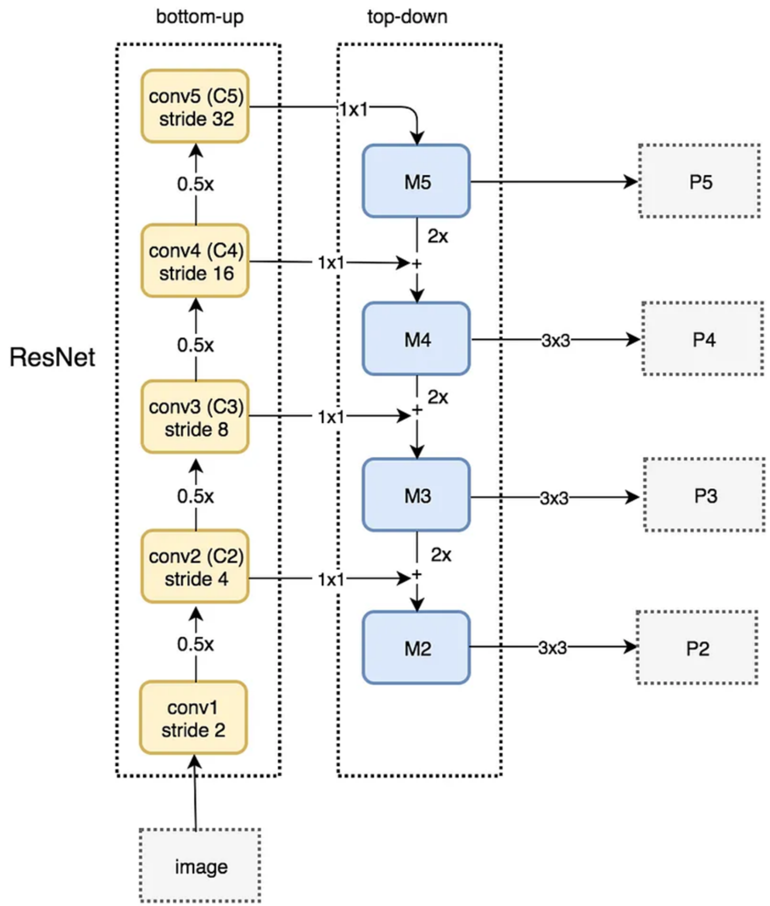
# Feature Pyramid Network

- Similar to U-Net
- Asymmetric design
- Lateral connections use 1x1conv with addition.

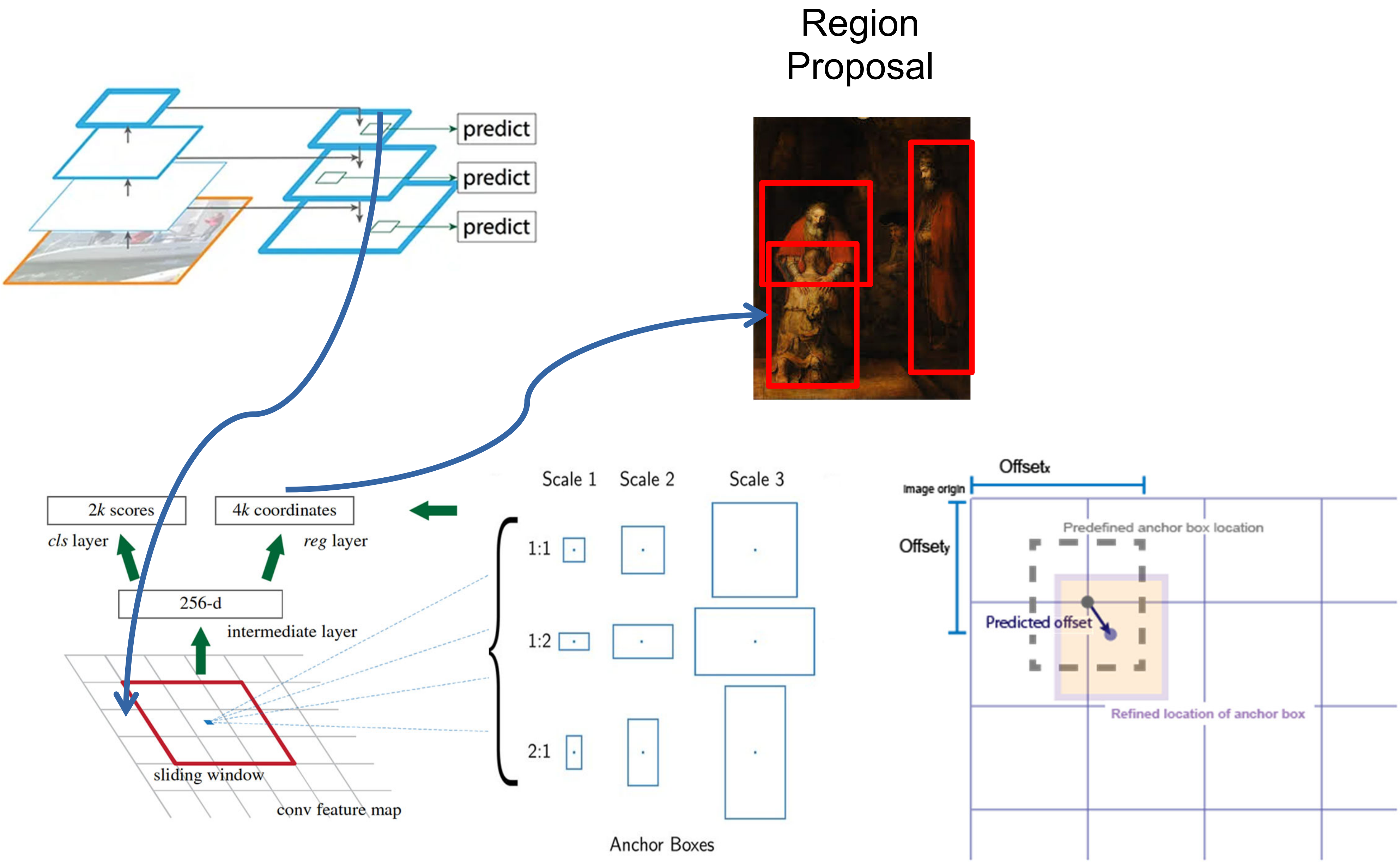


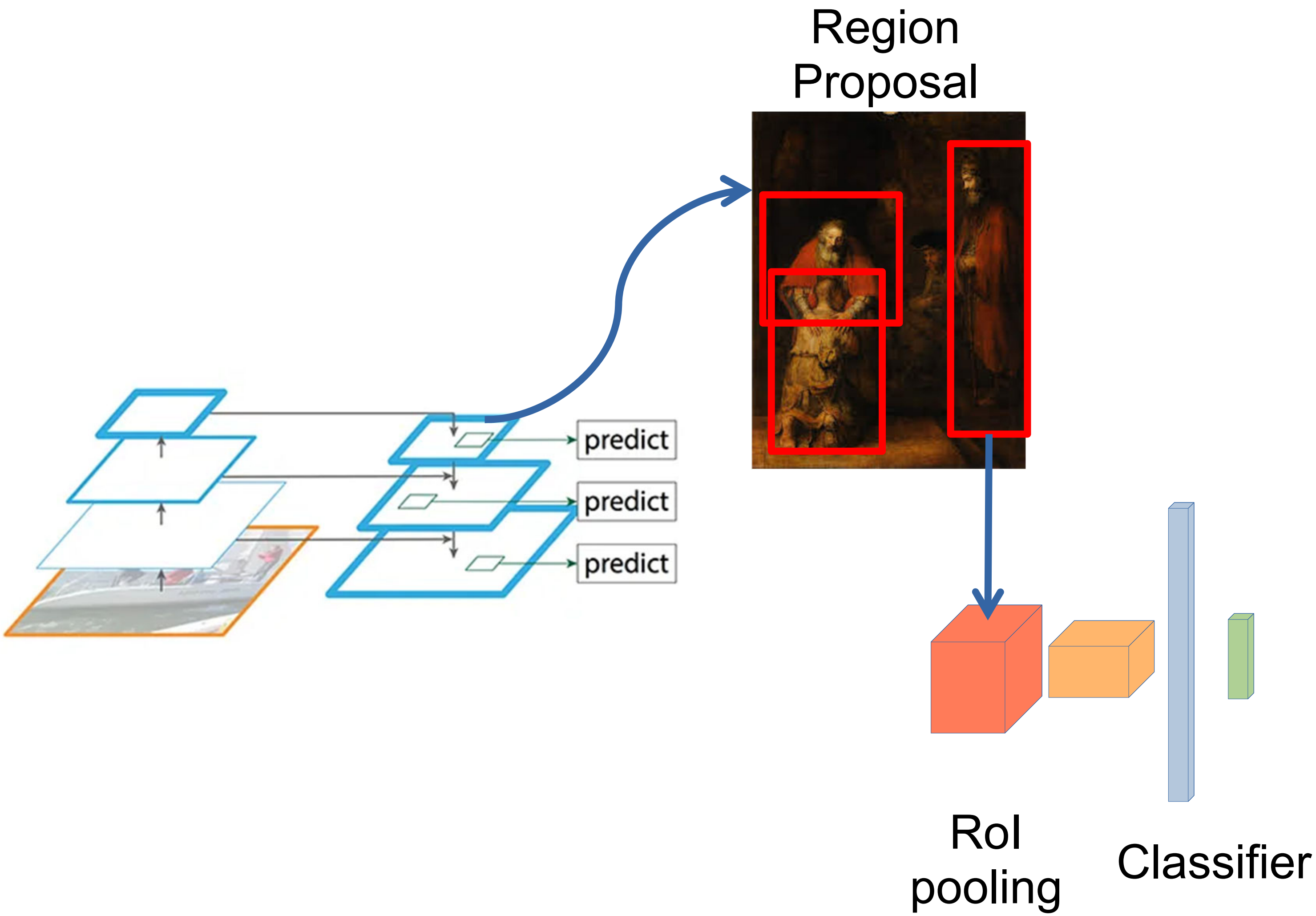

VGG , ResNet



# Region Proposal Network

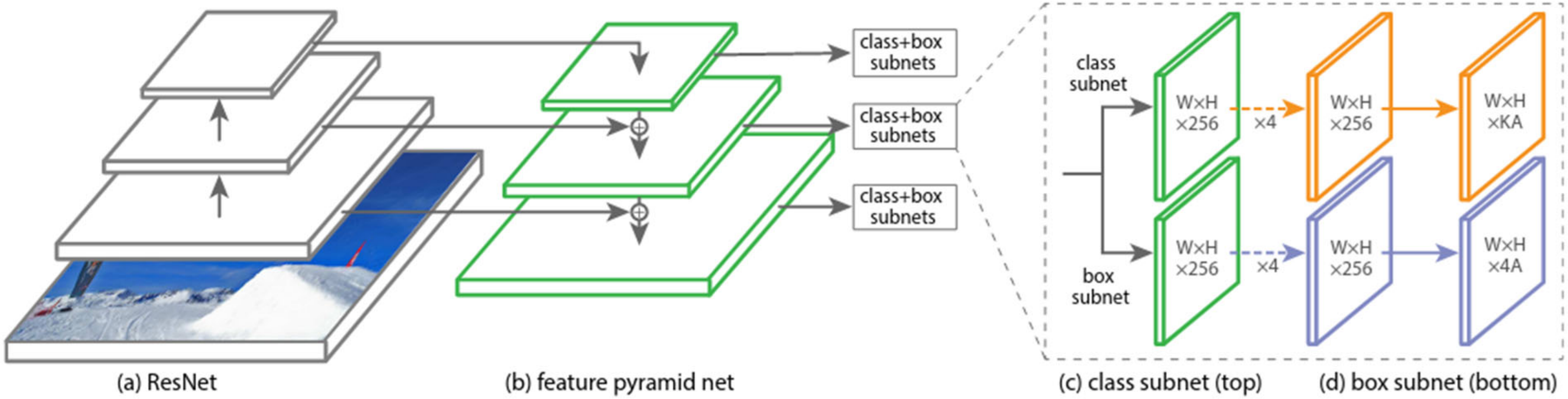


# Two-stage Detector



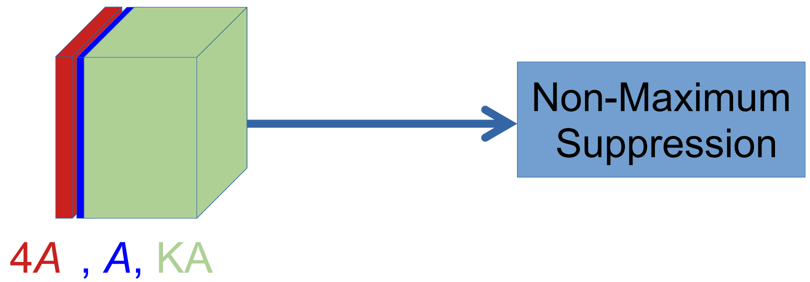

Girshick et al., R-CNN, 2015
Girshick, Fast R-CNN, 2015
Ren et al., Faster R-CNN, 2016

# One-stage Detector




One-stage vs. two-stage:

faster but less accurate.

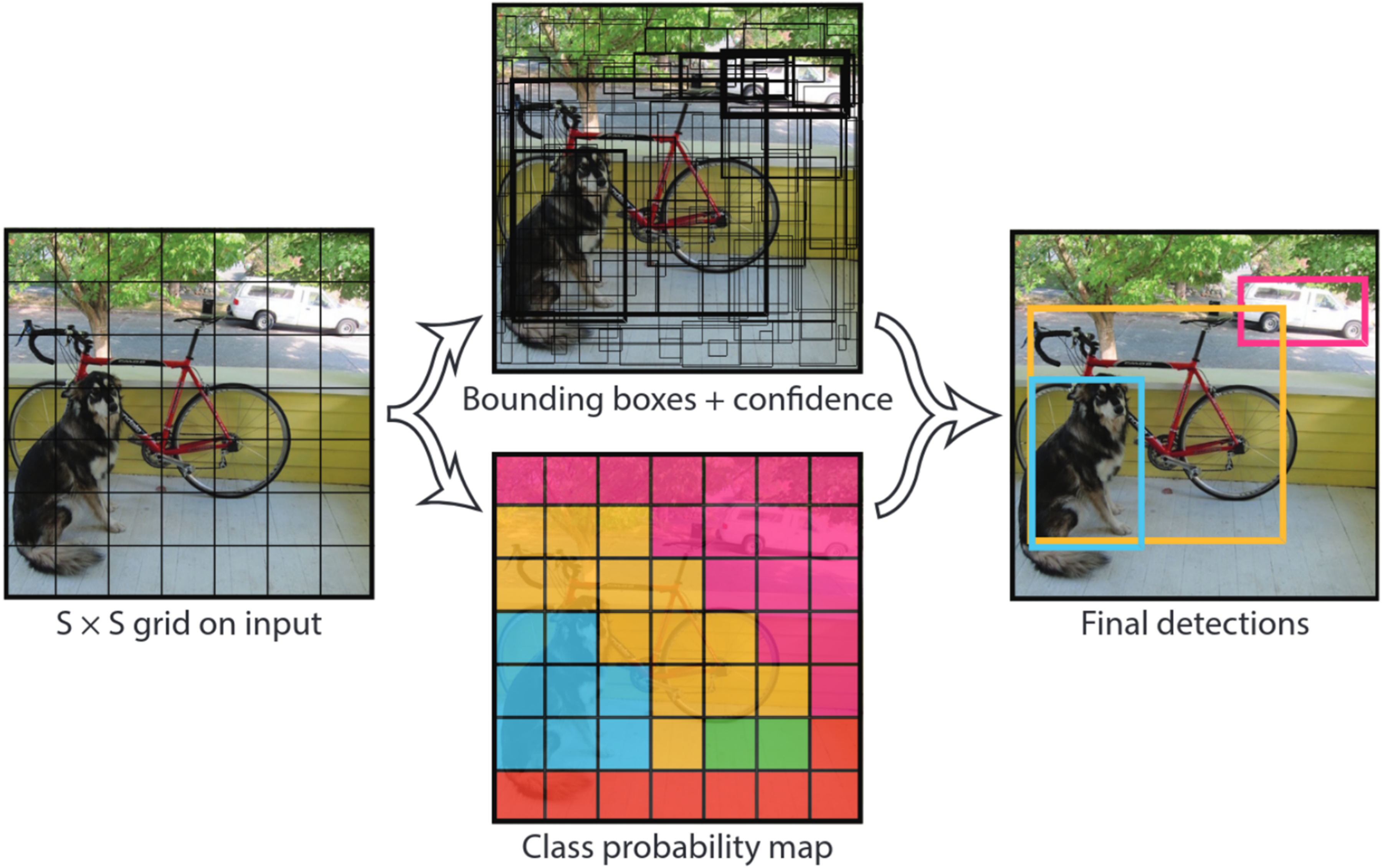



- *A* regions (x,y,w,h) for every location
- P(classes + background) for every location
- Objectness (prediction of IoU)

Redmon et al., YOLO, CVPR 2016
Liu et al., SSD, ECCV 2017
Lin et al., RetinaNet, 2018

# YOLO

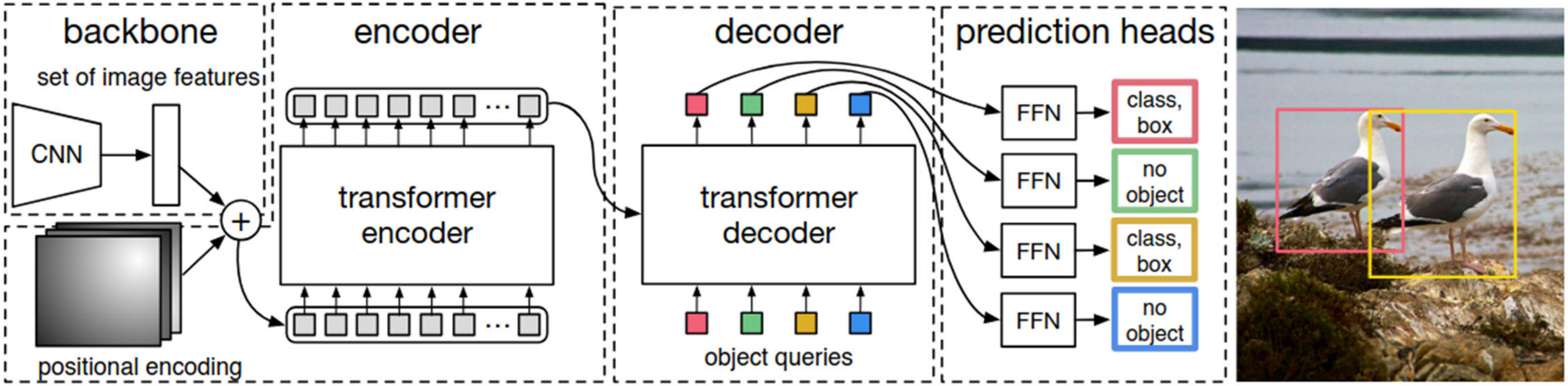
- You Only Live/Look Once



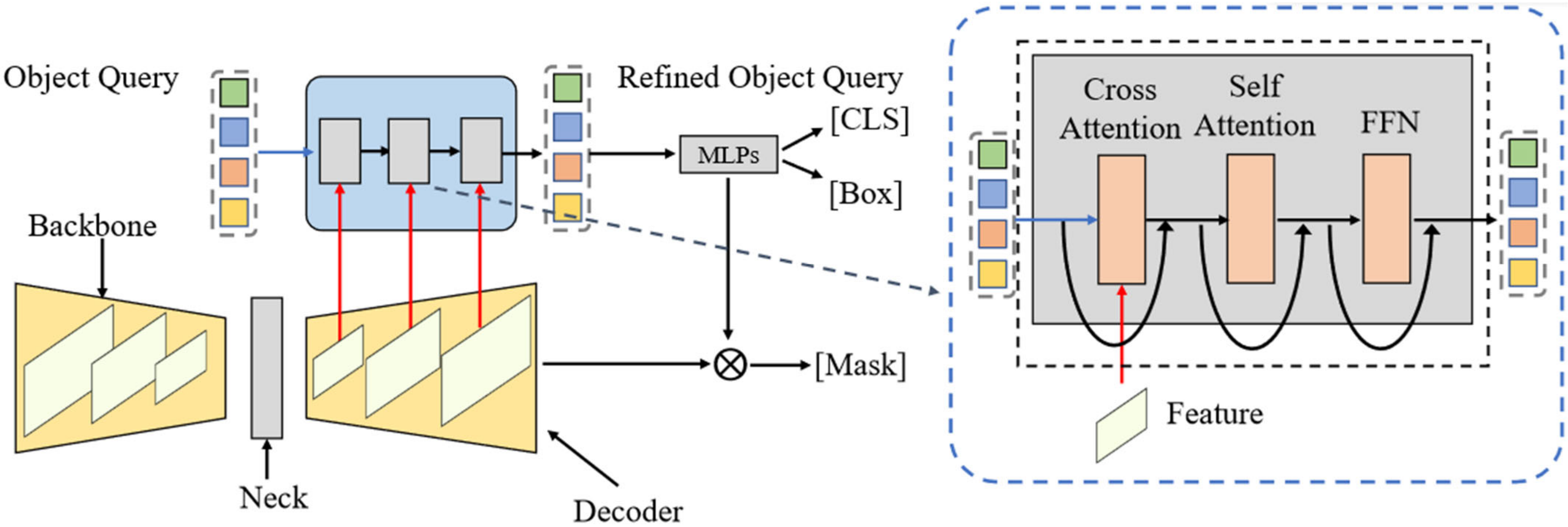

Redmon et al., [YOLO](), CVPR 2016

# Detection with Transformers

- [DETR, ECCV2020]



# Meta Architecture

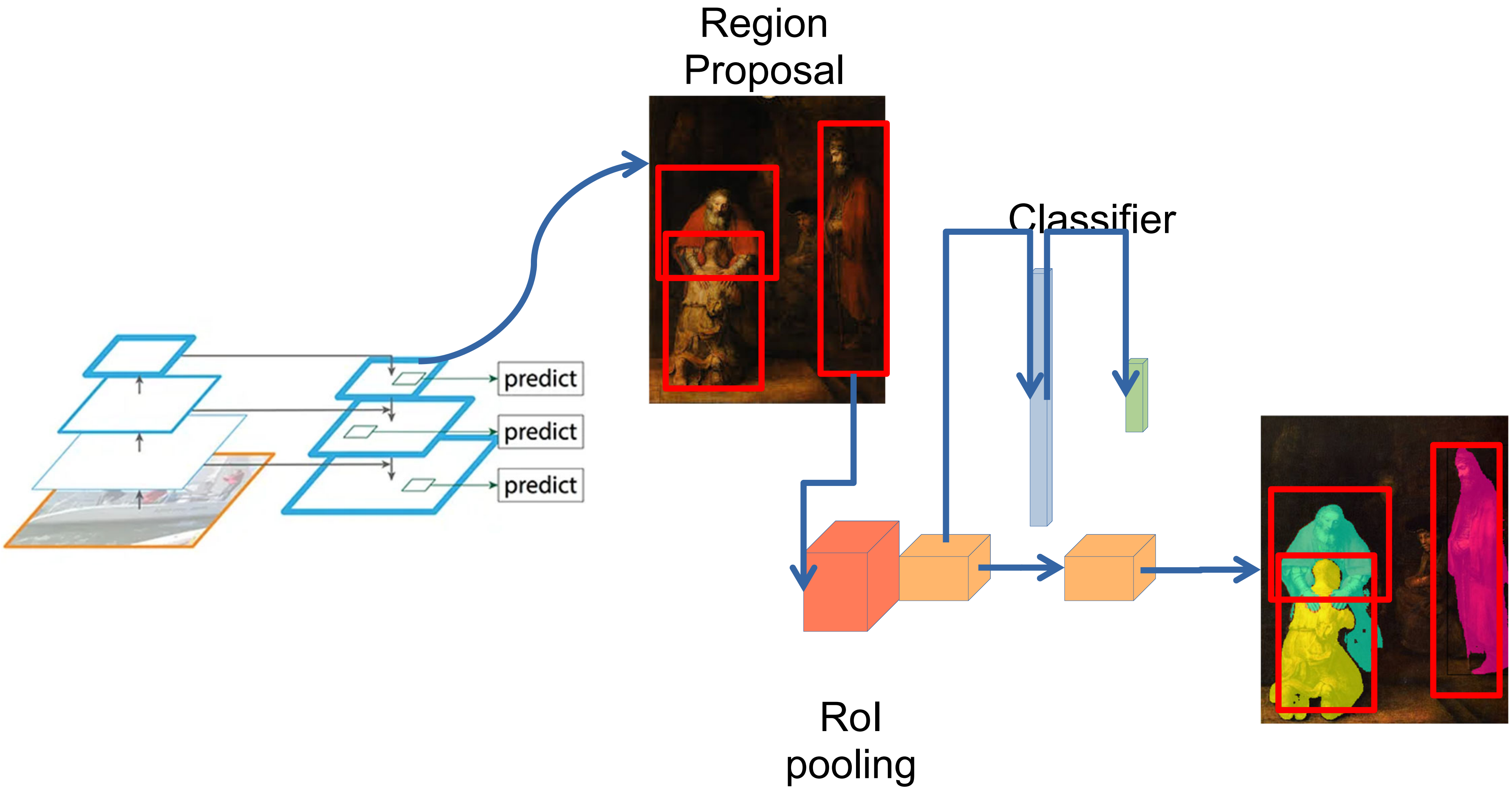



ResNet, ViT, Swin, ConvNeXt

Segmentation Survey, PAMI 24

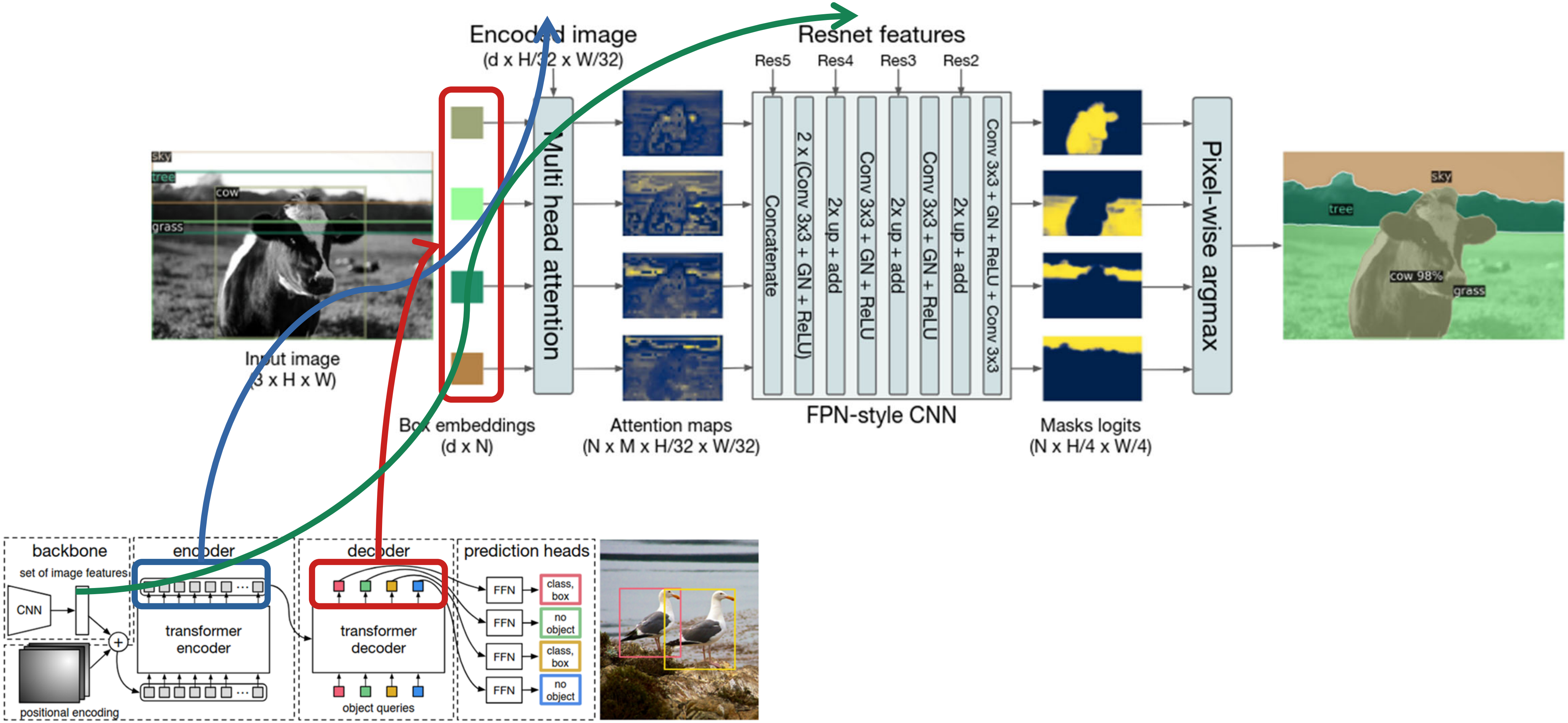
# Top-Down IS

- Mask R-CNN, 2018



# Top-Down IS

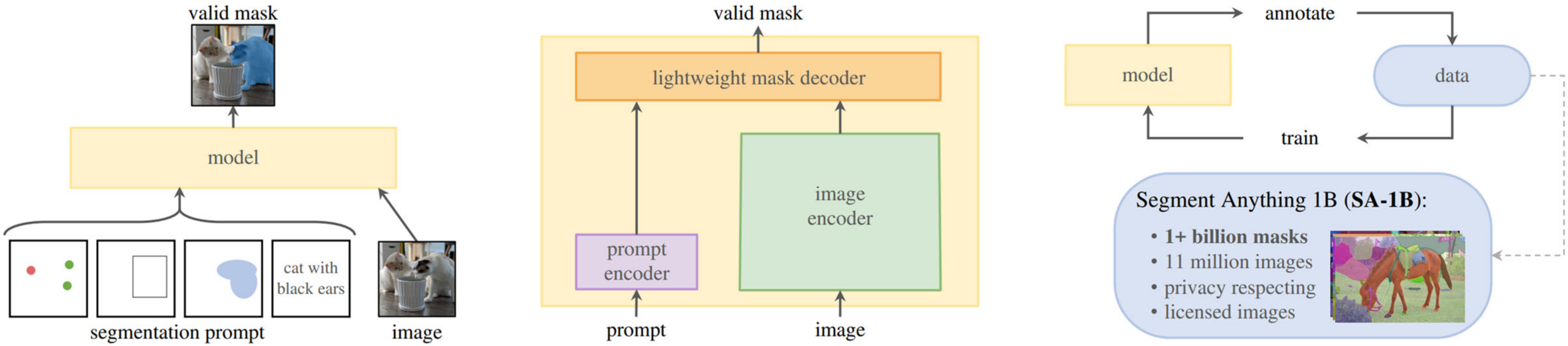
- DETR - Panoptic



# Panoptic Segmentation

- Universal segmentation
- OneFormer
- Masked-attention Mask Transformer for Universal Image Segmentation

# Segment Anything Model




(a) **Task**: promptable segmentation (b) **Model**: Segment Anything Model (**SAM**) (c) **Data**: data engine (top) & dataset (bottom)

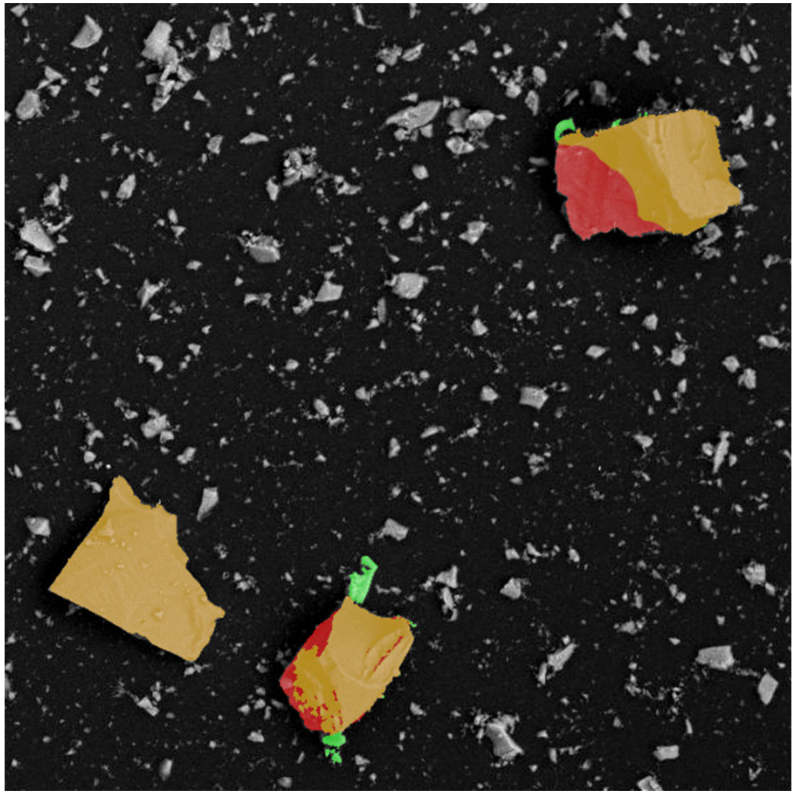
CVAT.AI (annotation platform)

[SAM 2023]

# Evaluation Metric

- Semantic: mIoU
- Instance: mAP
- Panoptic: Panoptic Quality

# Intersection over Union

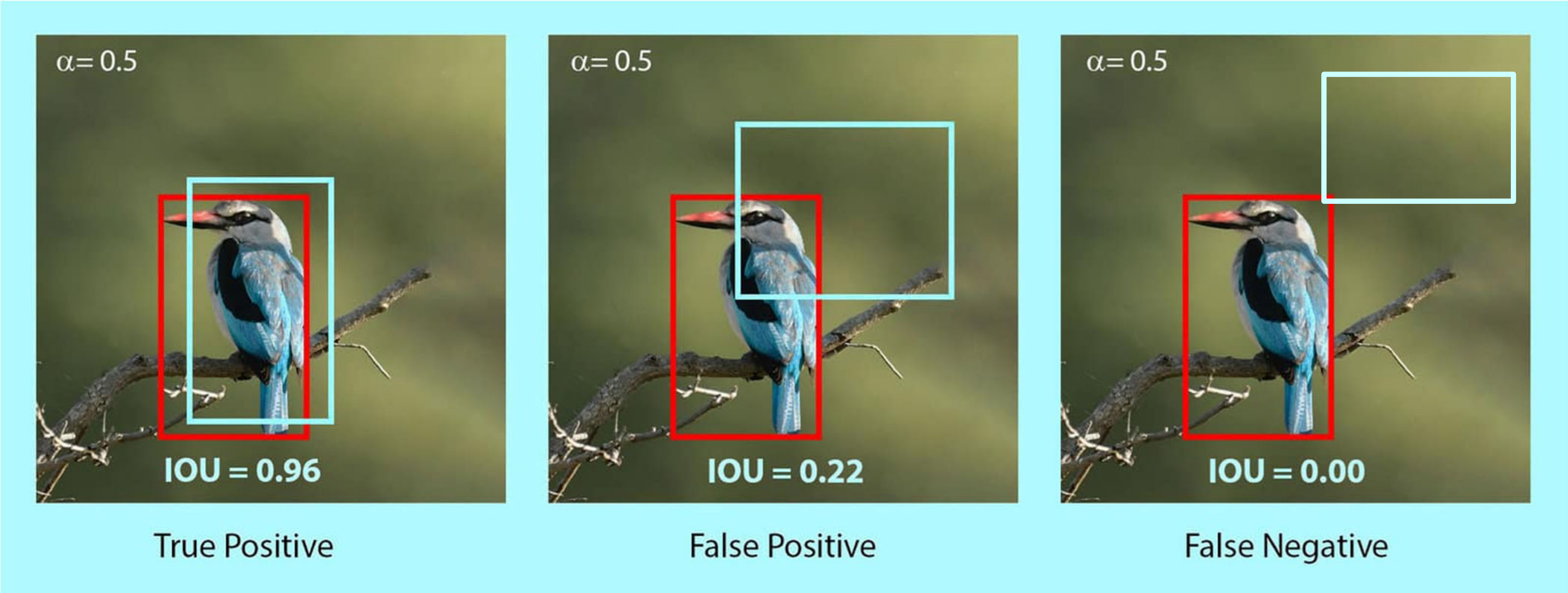


$$\mathrm{IoU} = \frac{|\mathrm{G} \cap \mathrm{P}|}{|\mathrm{G}| + |\mathrm{P}| - |\mathrm{G} \cap \mathrm{P}|}$$

- Mean IoU: $$\mathrm{mIoU} = \sum_{c} \mathrm{IoU}_c$$

# AP - Average Precision

- Threshold: $\mathrm{IoU} > \alpha$




$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \qquad \mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

$$\mathrm{F1} = 2\frac{\mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}$$

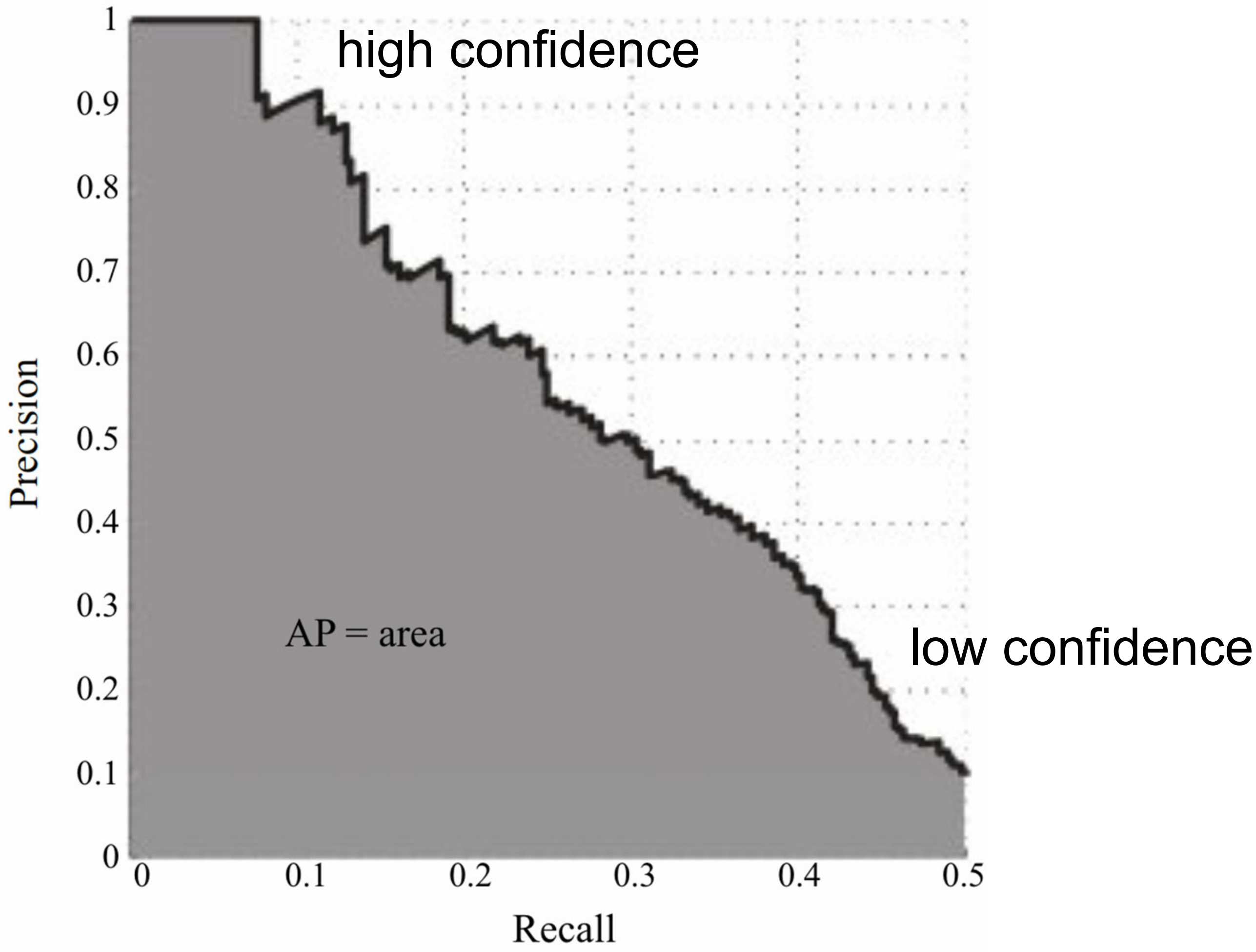
https://www.ridgerun.ai/post/mean-average-precision-map-and-other-object-detection-metrics

https://jss367.github.io/what-do-these-different-ap-values-mean.html

# mAP

- For the given threshold of IoU:

$$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$$




$$\mathrm{mAP} = \sum_{c=1}^{C} \mathrm{AP}_c$$

$$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$$

Thank You

# Panoptic Quality

- PQ per class:

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

$$\mathrm{PQ} = \sum_c \mathrm{PQ}_c$$