Deep Learning for Computer Vision: Object Detection & Segmentation

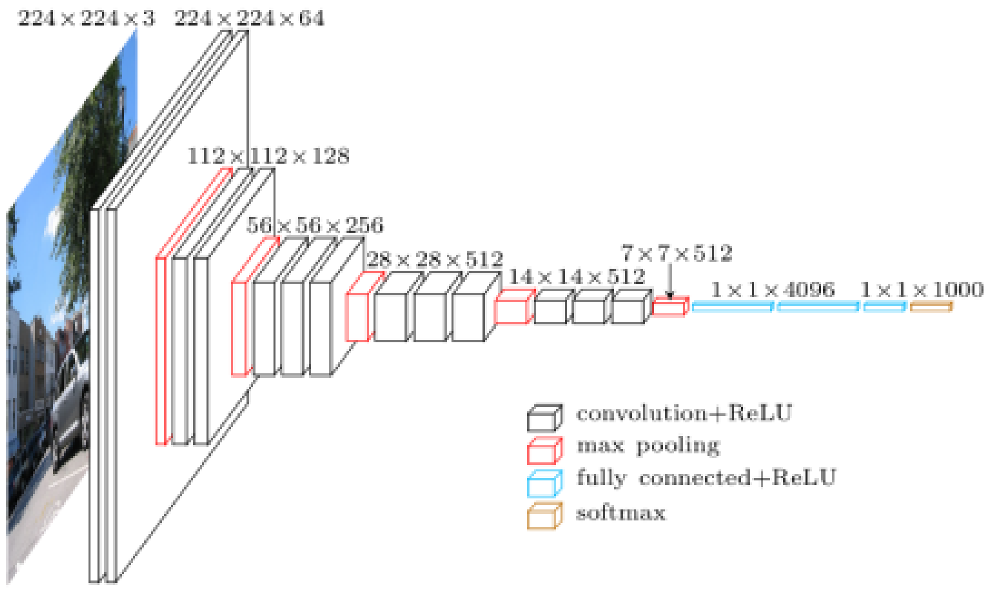
Shai Bagon

WAIC

Recap: Deep Learning

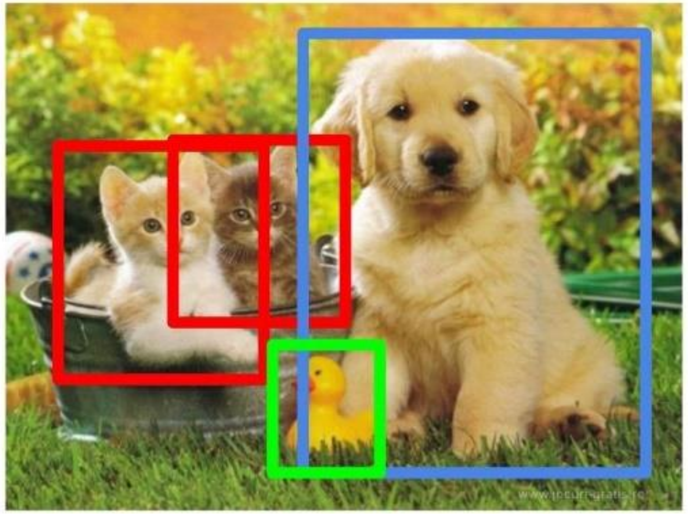
Key ingredients

- **Data:** labeled examples $\{(\boldsymbol{x}_i, y_i)\}_{i=1\ldots N}$
- **Model:** a “deep net” $\hat{y} = f(\boldsymbol{x}; \boldsymbol{\theta})$
- **Criterion:** how to “match” model and data $\mathcal{L}(\hat{y}, y)$
- **Optimization:** Stochastic Gradient Descent (SGD)



Additional Tasks

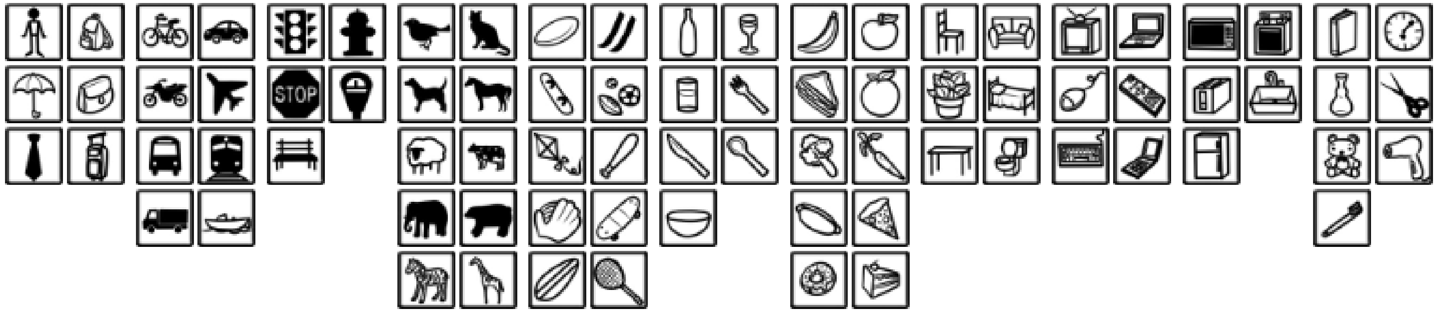
- Object detection
- Semantic segmentation
- Instance segmentation



Additional Tasks: Training Data

MS COCO

- 200K labeled images
- 1.5**M** instances
- 80 object categories



Object Detection

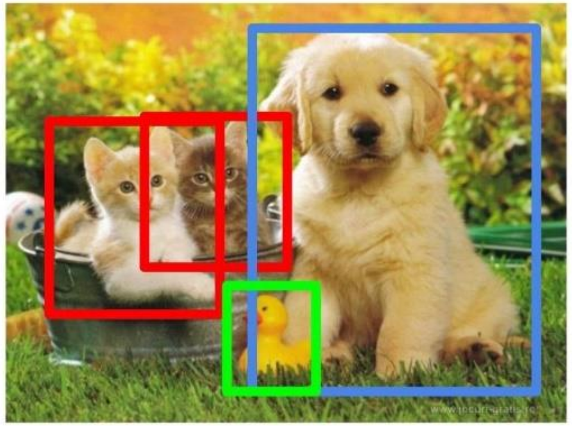
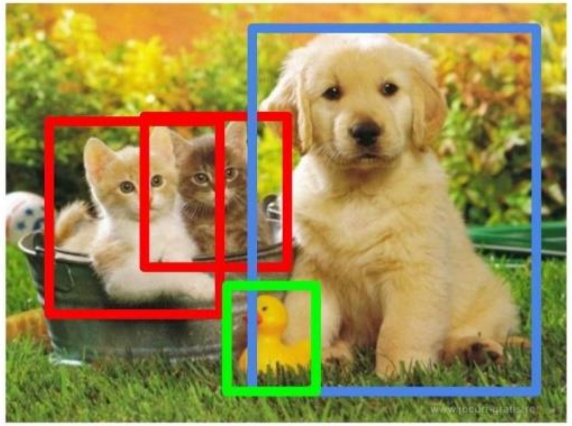


Image credit: medium

Object Detection - Challenges

- Multiple types of outputs
- Varying number of objects



Localization

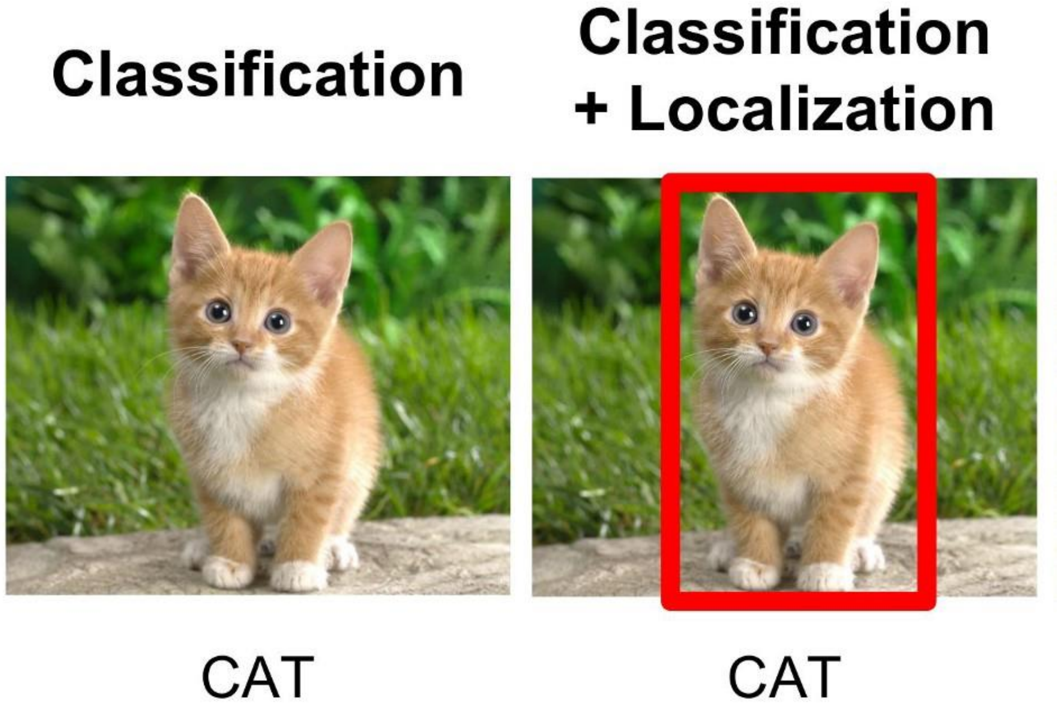
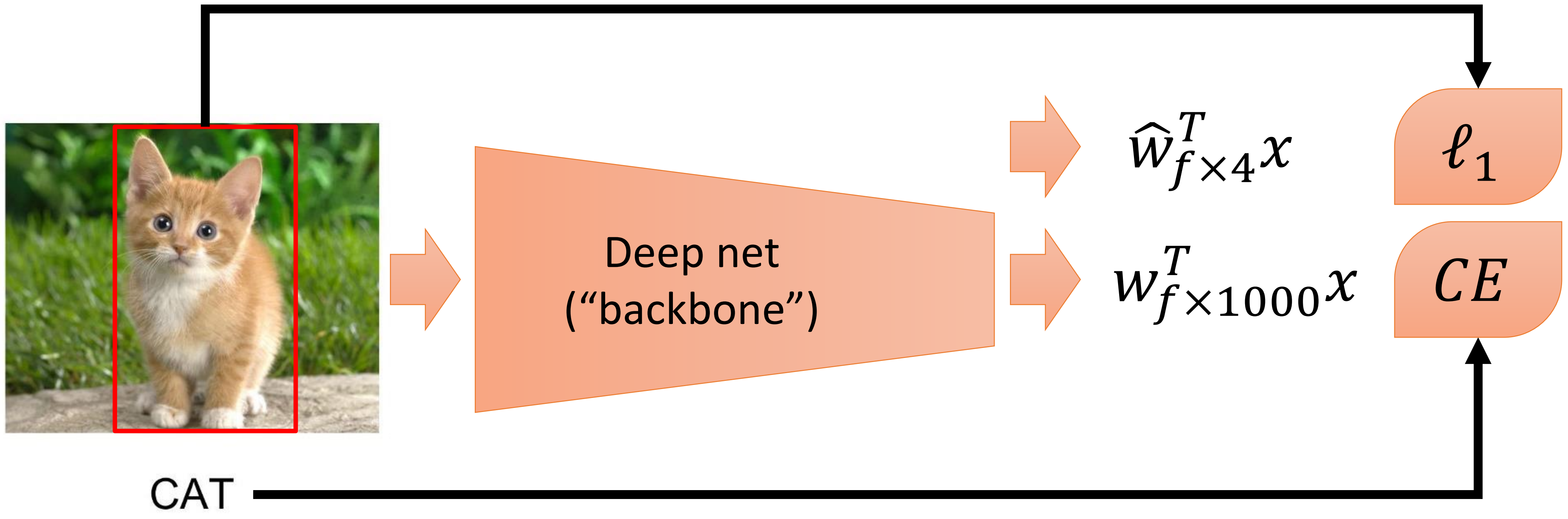
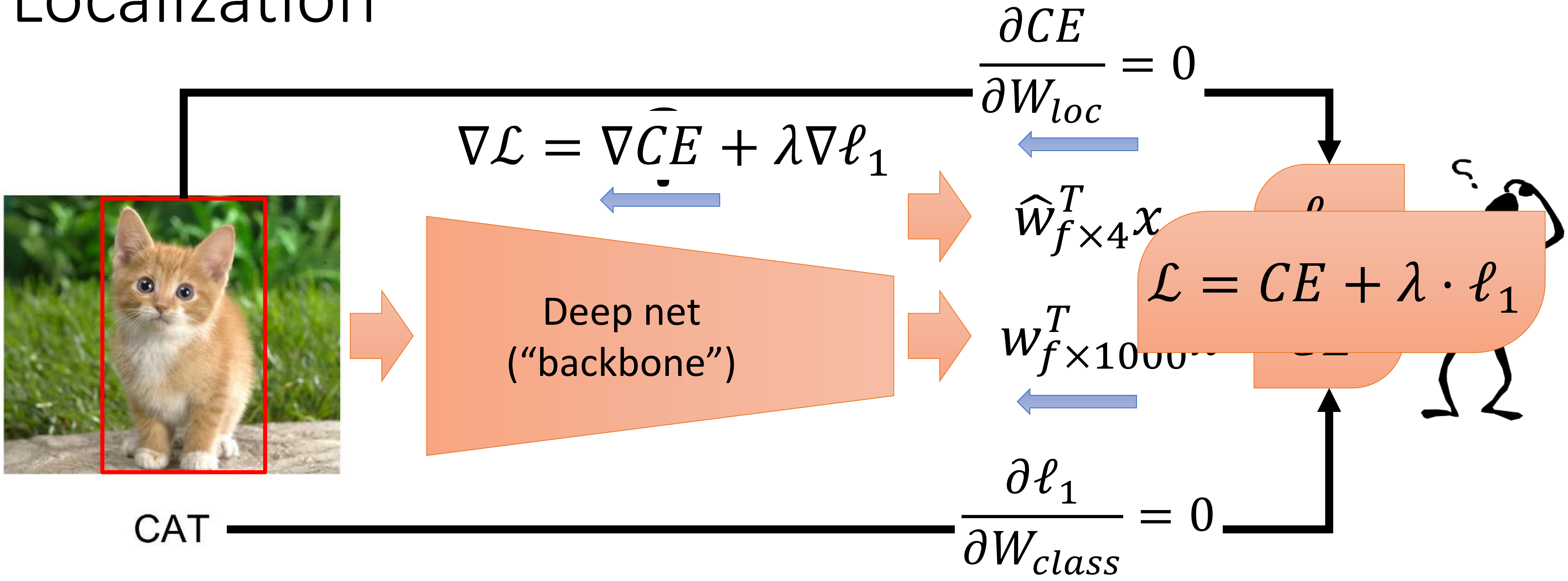


Image credit: medium

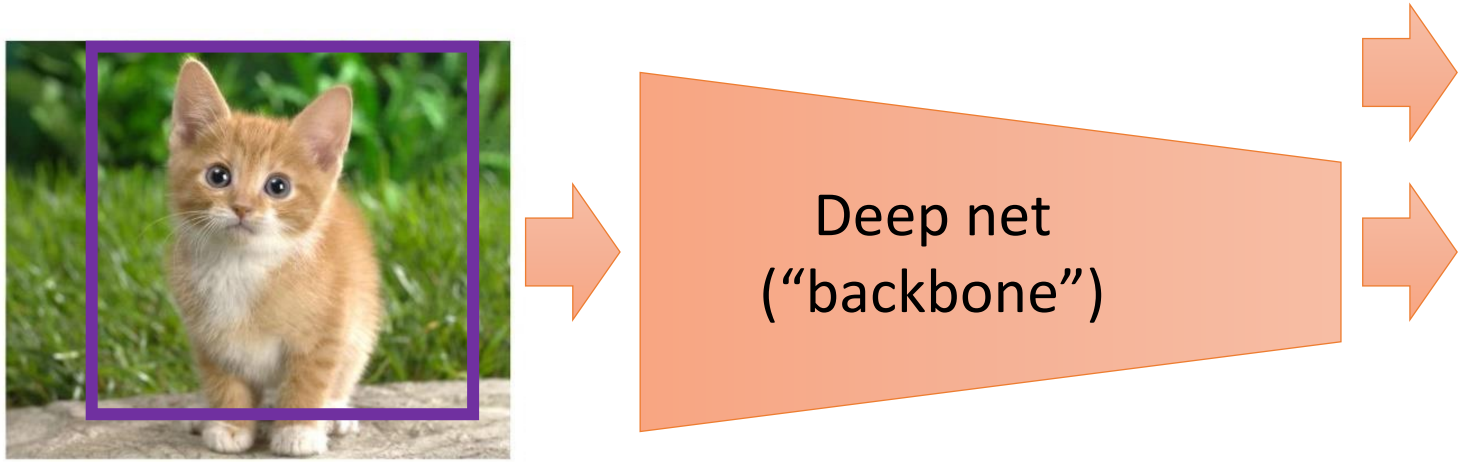
Localization



Localization



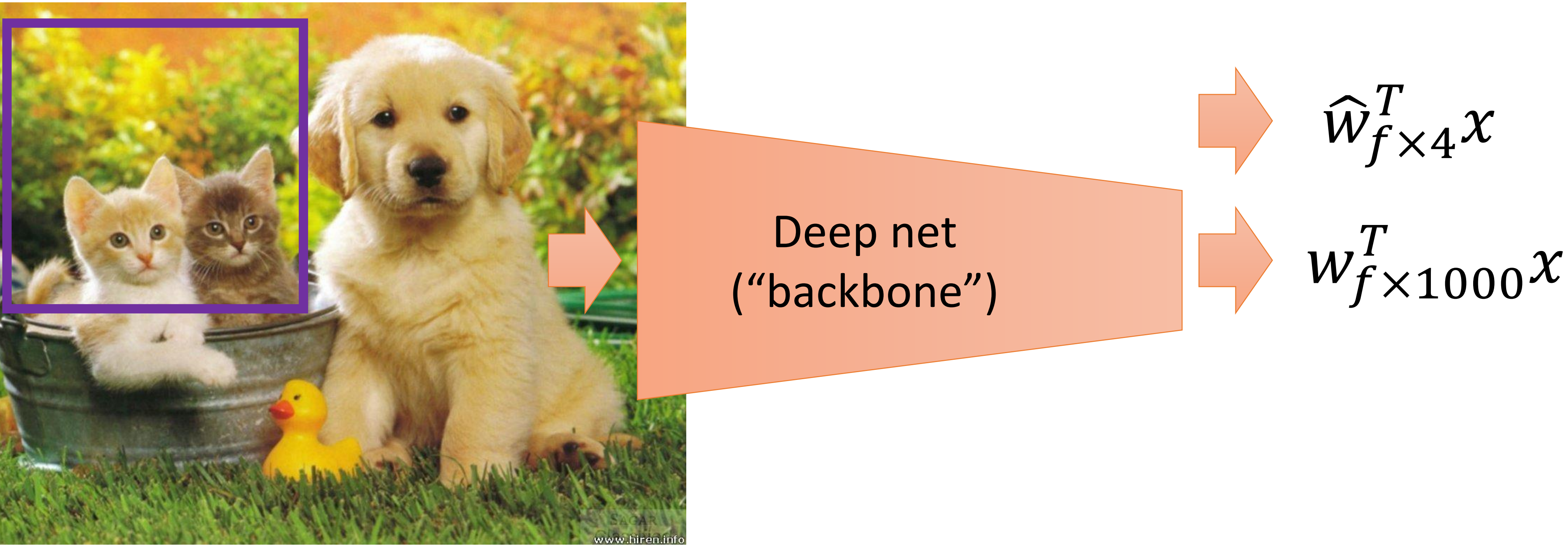
From Localization to Detection (v0)



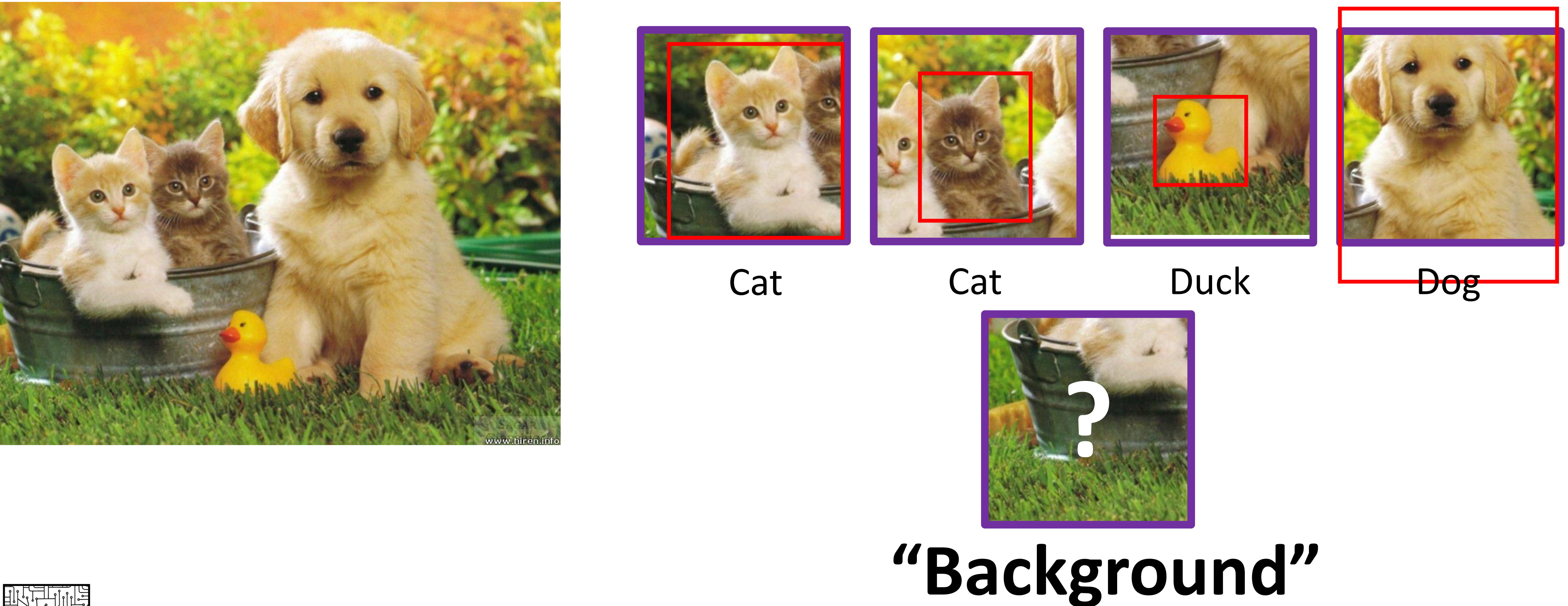
$$\hat{w}_{f\times 4}^{T}x$$

$$w_{f\times 1000}^{T}x$$

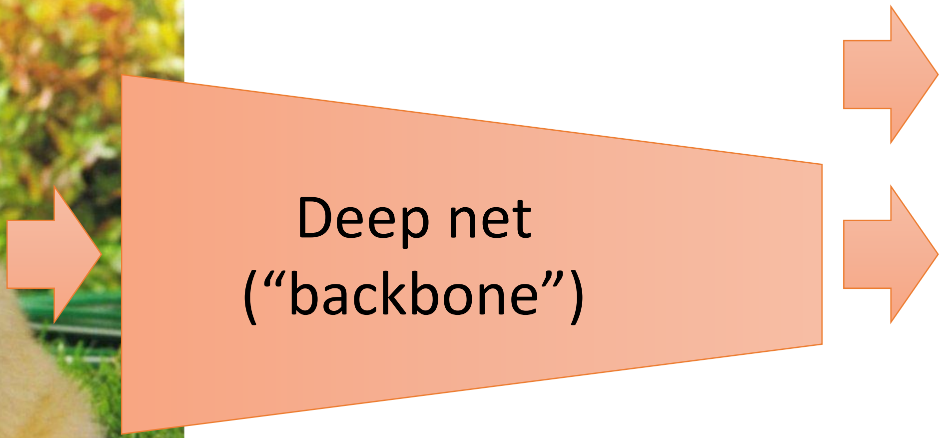
From Localization to Detection (v0)



From Localization to Detection (v0)



From Localization to Detection (v0)



$$\hat{w}_{f\times 4}^{T}x$$

$$w_{f\times 1000}^{T}x$$

Challenges:

- Multiple types of outputs
- Number of objects

From Localization to Detection (v0)

Deep net
("backbone")

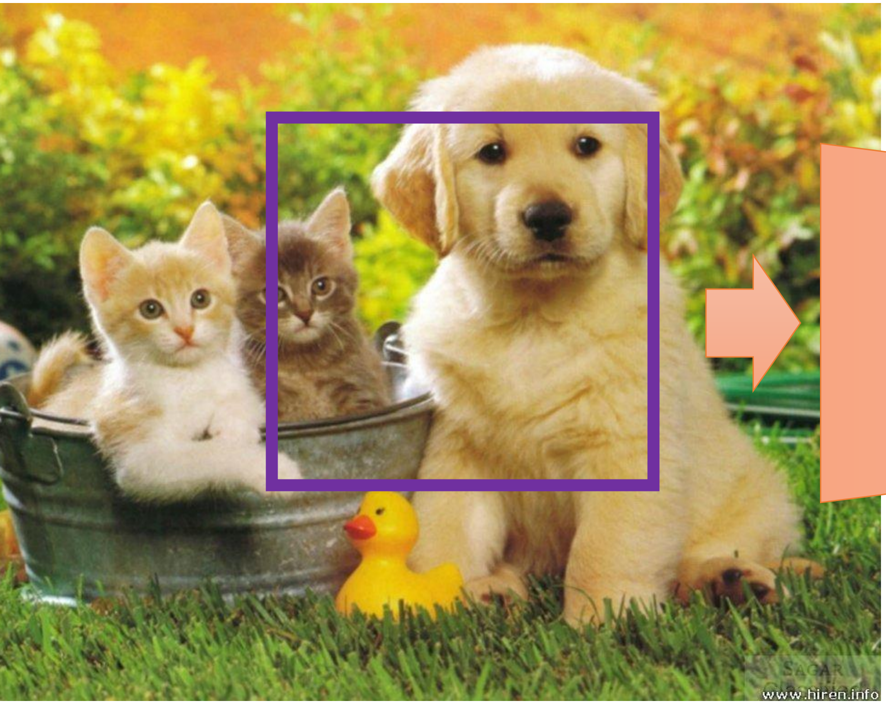
$\hat{w}_{f\times 4}^{T}x$

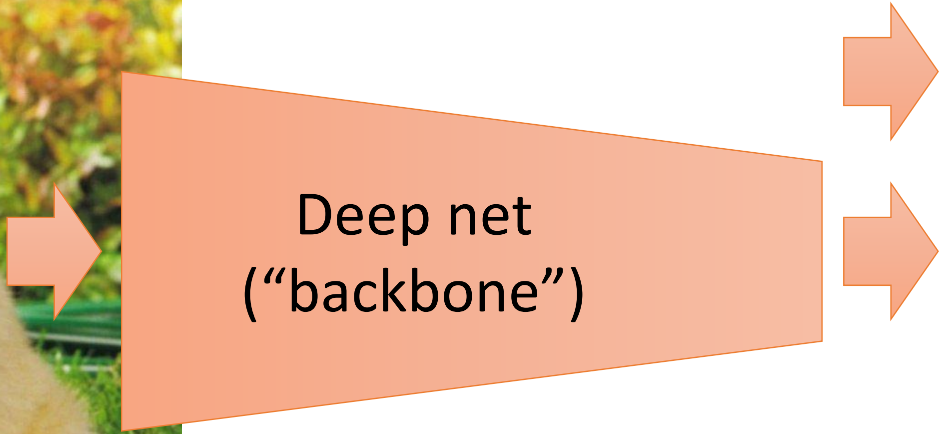
$w_{f\times 1000}^{T}x$

How many "sliding windows" are there?

There can easily be O(1M) windows!

From Localization to Detection (v1)





$$\hat{w}^{loc}_{f\times1\times1\times4} * x$$

$$w^{class}_{f\times1\times1\times C} * x$$

Object Detection

Single Shot:
SSD, YOLO…

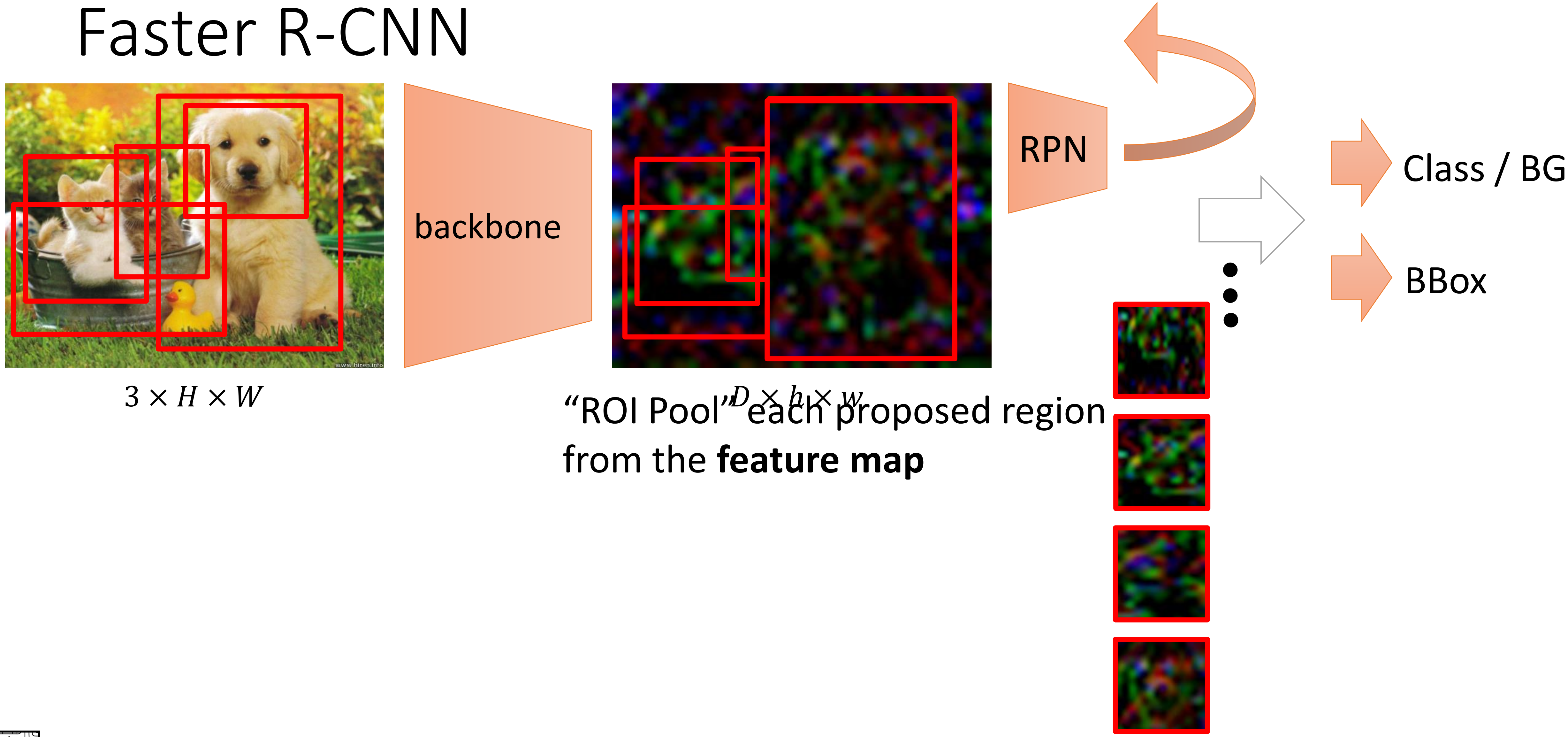
Fast
High false rate

Two Shots:
R-CNN…

Slower
More accurate

Faster R-CNN

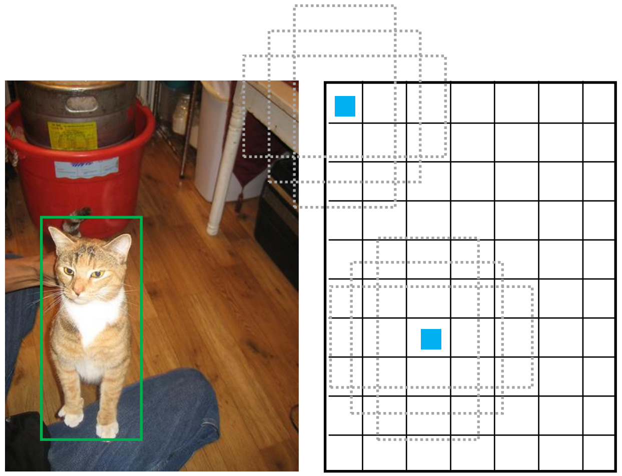
RPN: Region **P**roposal **N**etwork



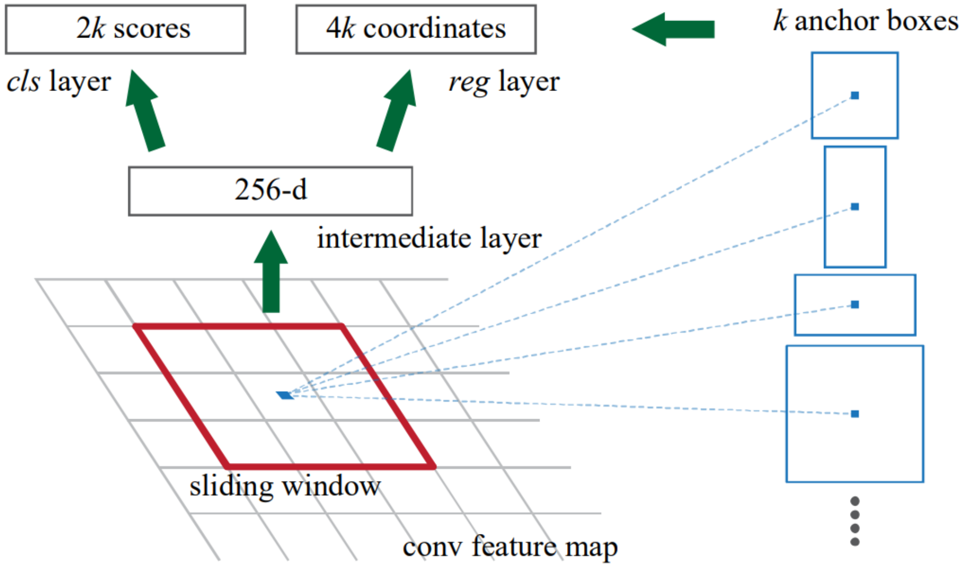
"ROI Pool" each proposed region from the feature map

Ren, He, Girshick, and Sun "[Faster r-cnn: Towards real-time object detection with region proposal networks]" (NIPS 2015)

RPN: Region Proposal Network

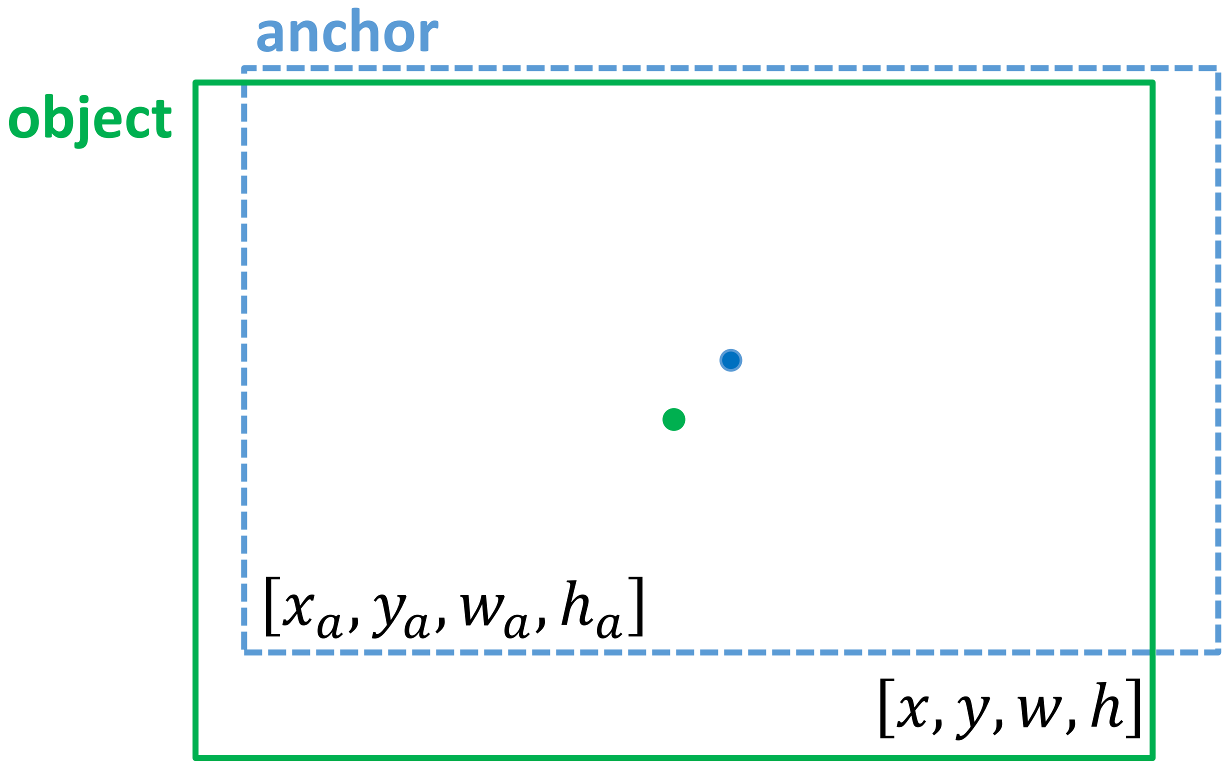


feature map



RPN: Region Proposal Network

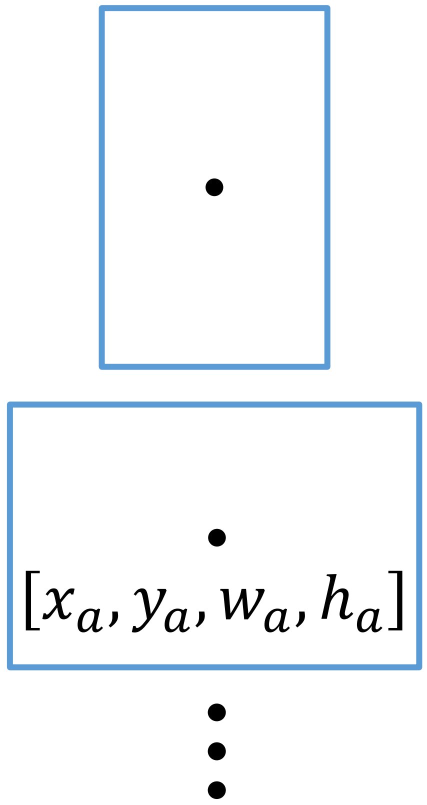
Predicting Bounding Box coordinates from anchors:



RPN: Region Proposal Network

Predicting a bounding box "corrections" $\boldsymbol{t} = [t_x, t_y, t_w, t_h]$:

$$t_x = \frac{x - x_a}{w_a}, \qquad t_y = \frac{y - y_a}{h_a}, \qquad t_w = \log\left(\frac{w}{w_a}\right), \qquad t_h = \log\left(\frac{h}{h_a}\right)$$



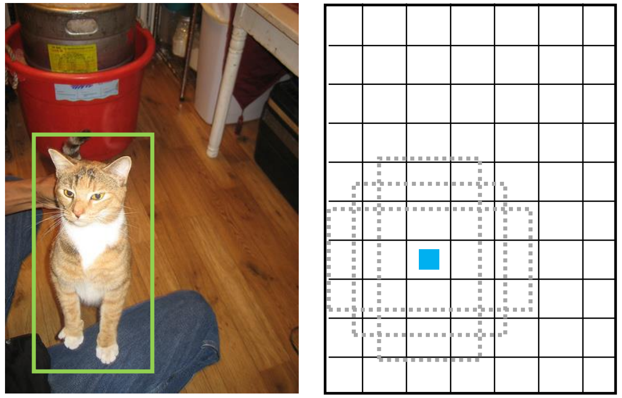
Recovering the actual BBox from $\boldsymbol{t} = [t_x, t_y, t_w, t_h]$:

$$\mathrm{x} = \mathrm{w_a} \cdot t_x + x_a, \qquad y = h_a \cdot t_y + y_a,$$

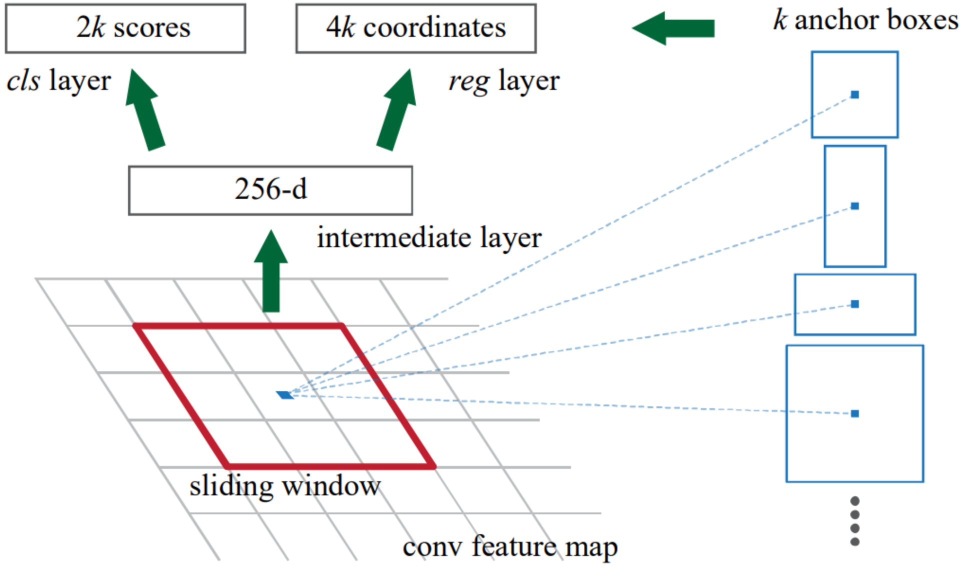
$$w = e^{t_w} \cdot w_a, \qquad h = e^{t_h} \cdot h_a$$

The learned "corrections" $\boldsymbol{t}$ are relative to anchor position and scale.

Object Detection



Training

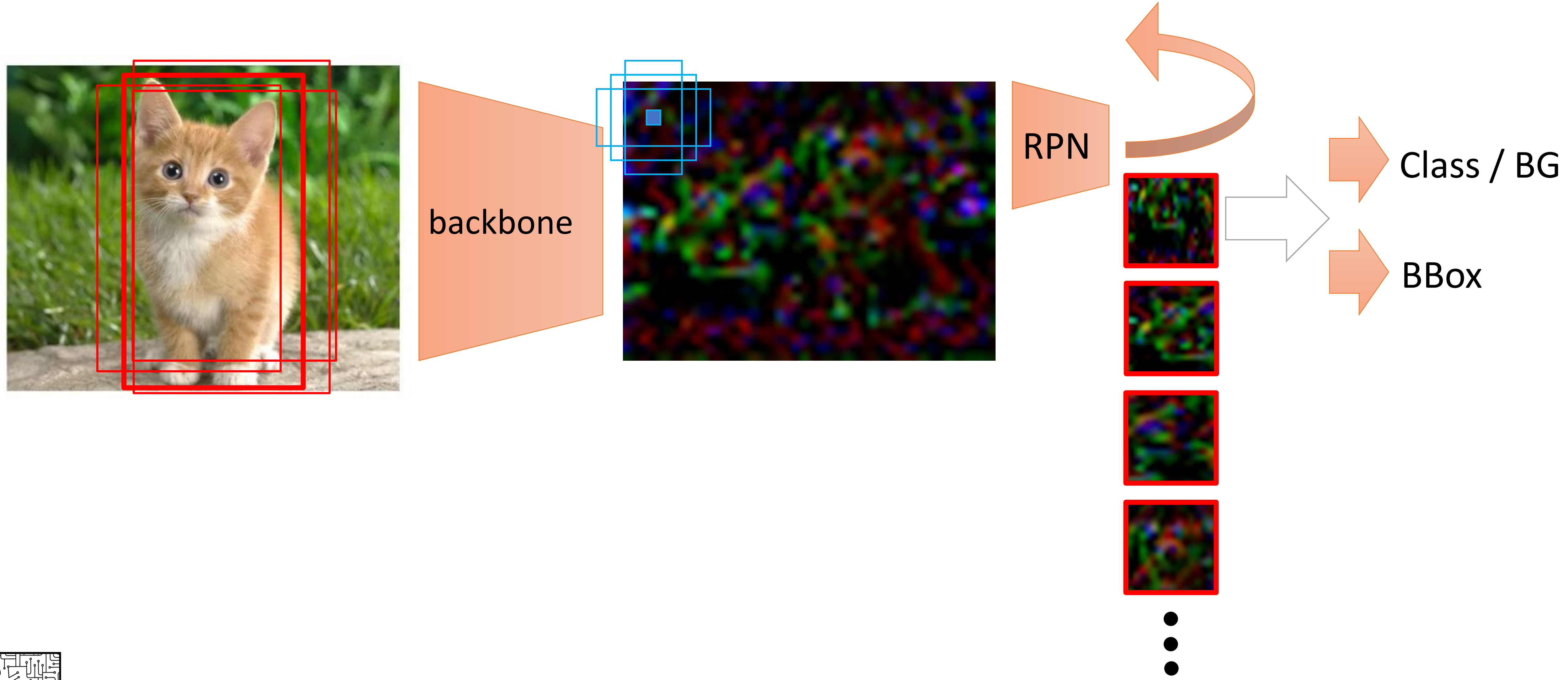


object

anchor

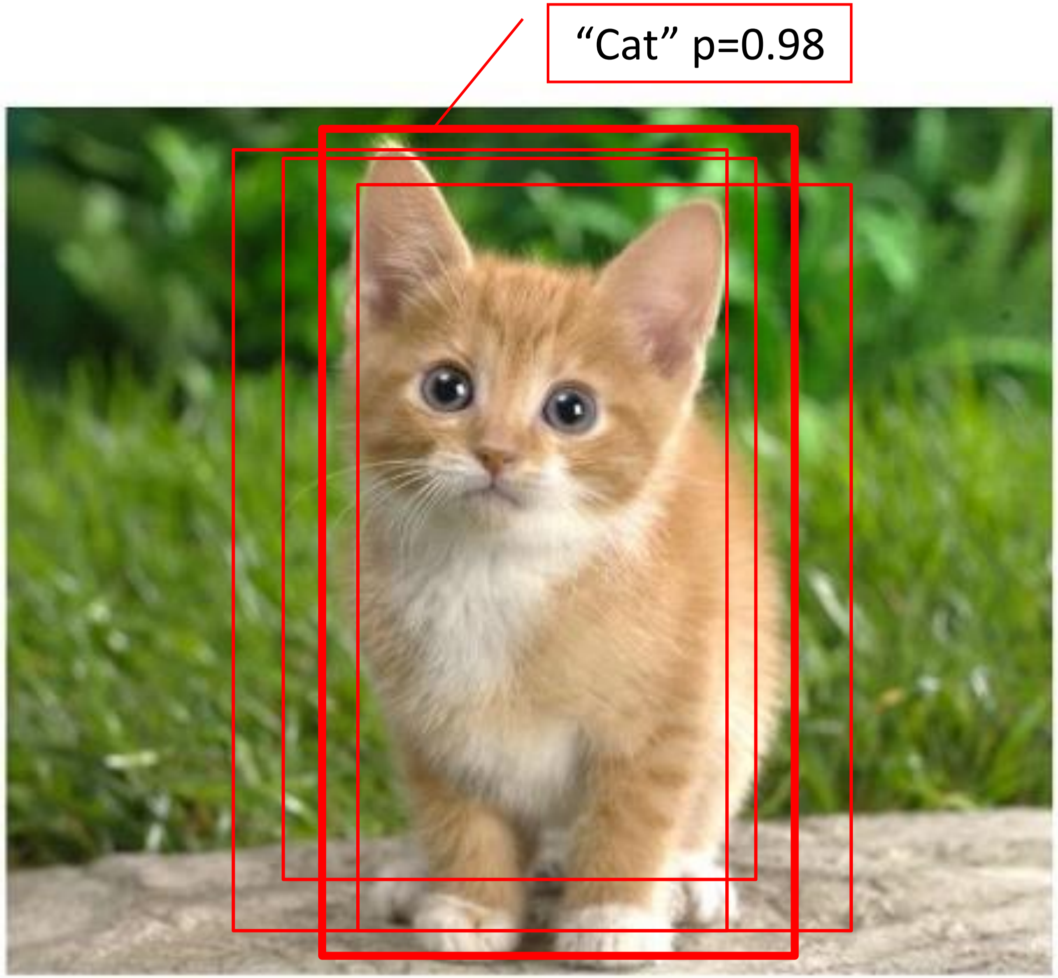
IoU = ―――――――

Inference

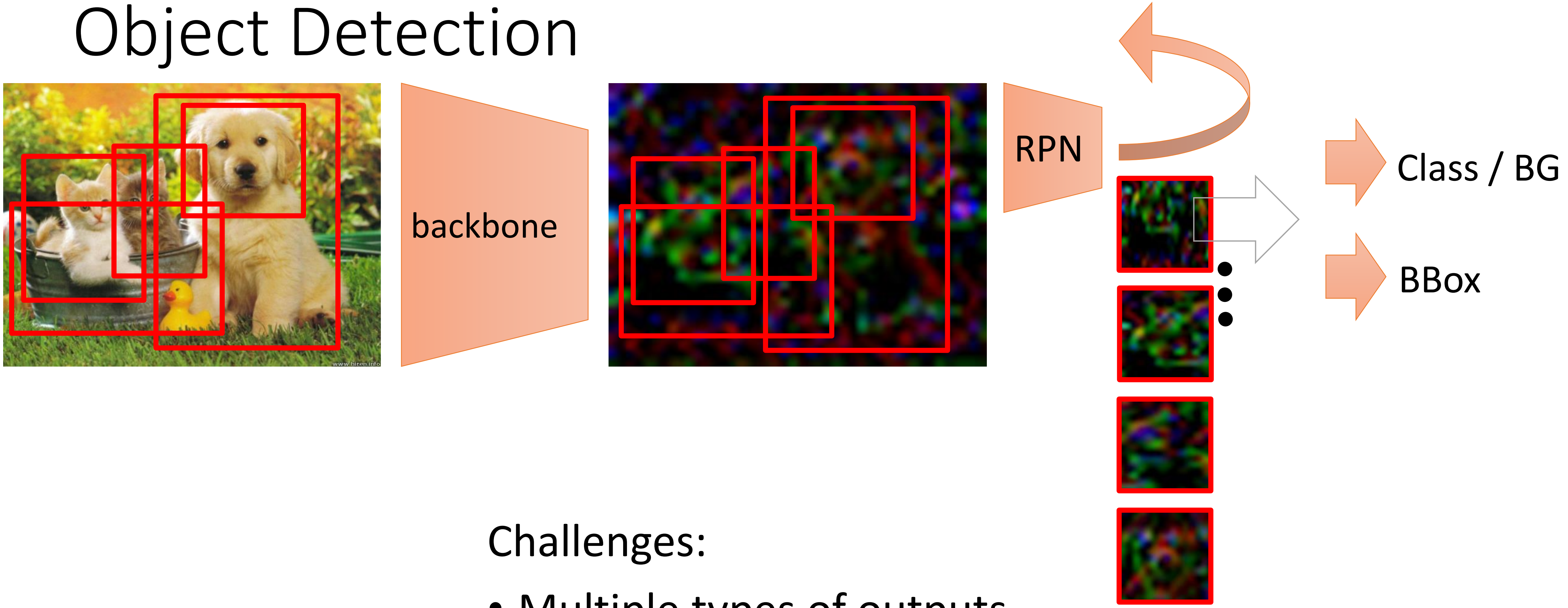


Inference: Non Maximal Suppression (NMS)

- Sort detection by score (per class)
- Take most confident
- Remove all overlapping
- Repeat



Object Detection



Challenges:

- Multiple types of outputs
- Number of objects

Object Detection: Pitfalls

- Imbalance
- Multiscale

Imbalance

Number of “negative” anchors ~O(10K)
Number of “positive” anchors ~O(10)

What happens to CE loss in this case?

$$CE(p_i, y_i) = -y_i \log(p_i) - (1 - y_i)\log(1 - p_i)$$

$$= -9 \cdot \log(0.5) - 9 \cdot \log(0.5)$$

$$=$$

Prediction p_i \ True y_i	BG	Obj
BG	9981	9
Obj	9	1

Unconfident:
p=0.5

Confident:
p=0.99

Imbalance

Number of “negative” anchors ~O(10K)
Number of “positive” anchors ~O(10)

What happens to CE loss in this case?

$$\nabla CE(p_i, y_i) = p_i - y_i$$

$$= 9 \cdot 0.5 + 9 \cdot (1 - 0.5)$$

Prediction p_i \ True y_i	BG	Obj
BG	9981	9
Obj	9	1

Unconfident: *p=0.5*

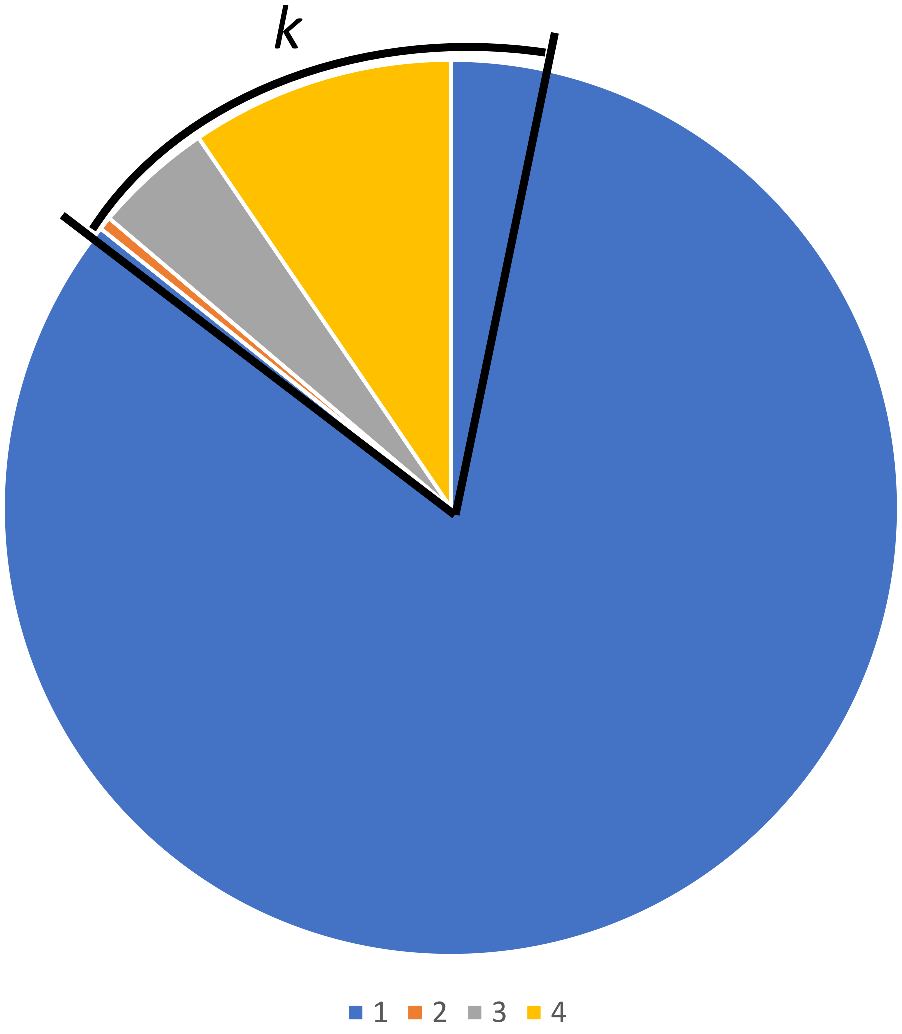
Confident: *p=0.99*

Imbalance – Hard Negative Mining

Compute loss for all *N* anchors

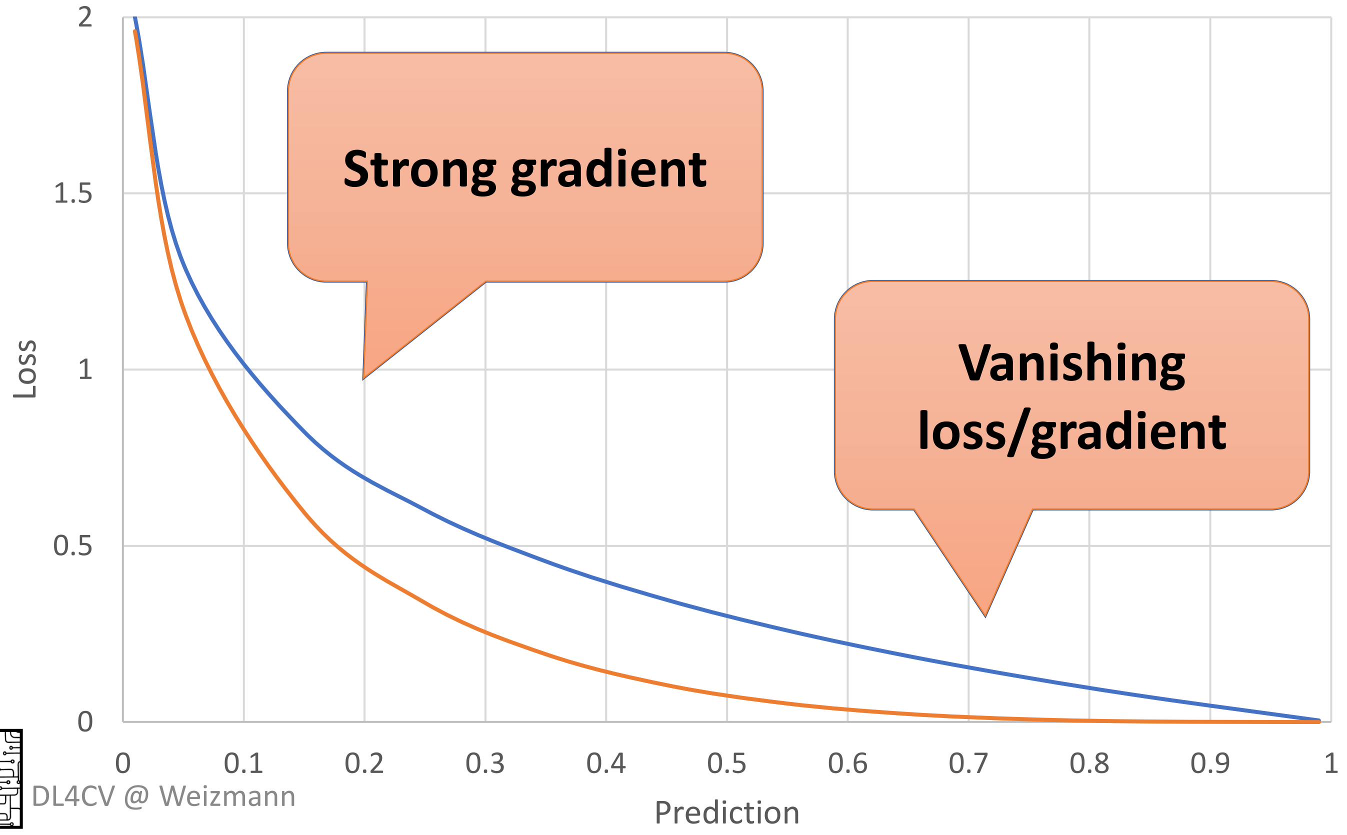
Select top *k* “hard” examples

Compute gradient for hard *k* ***only***



Imbalance – Focal Loss

Lin,Goyal, Girshick, He, and Dollár
Focal loss for dense object detection (PAMI 2018)



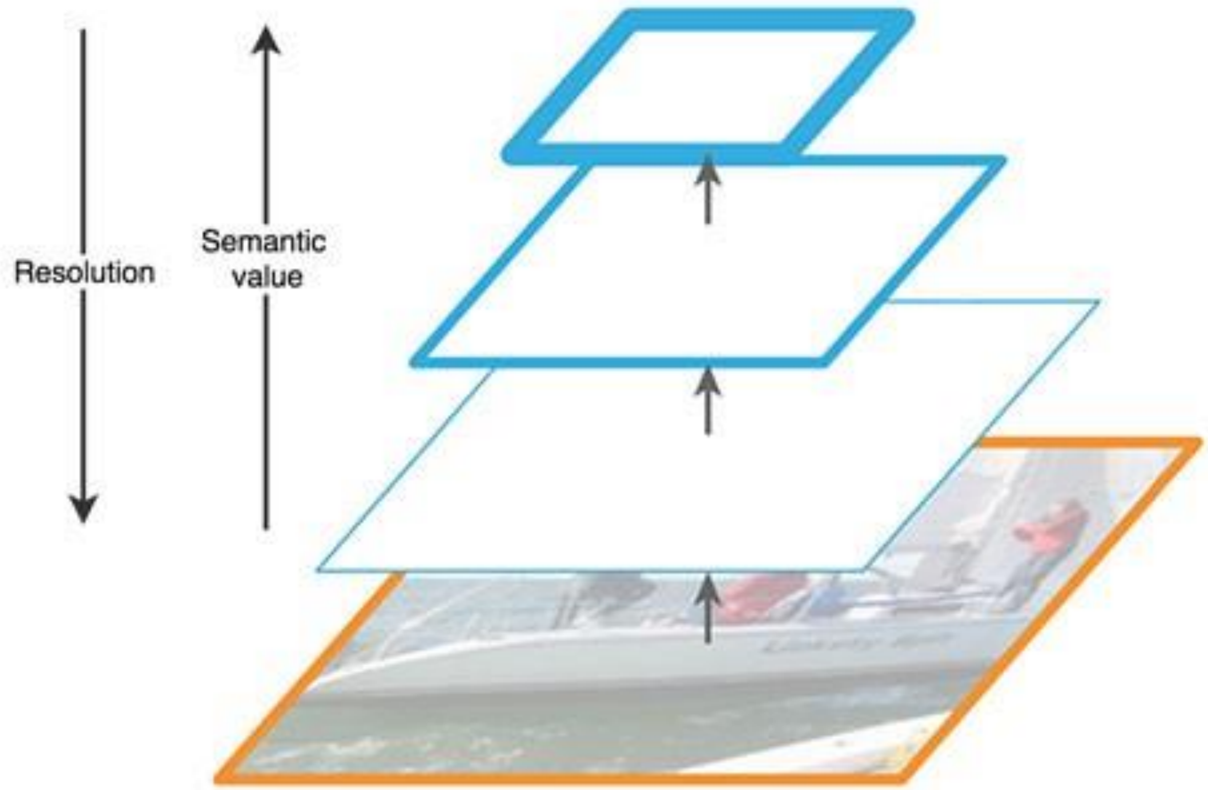
$CE = -y_i \log p_i$
$FL = -y_i (\mathbf{1} - \boldsymbol{p_i})^{\gamma} \log p_i$

Object Detection: Pitfalls

- Imbalance
- Multiscale

Resolution vs Semantic Value

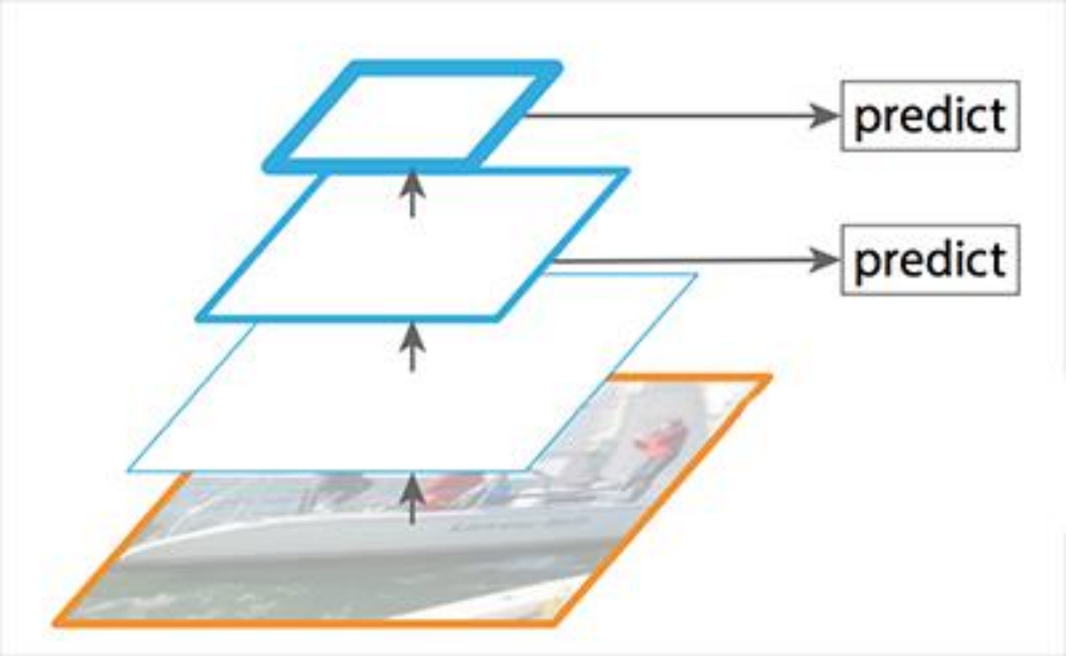
How to handle multiscale predictions?



Tsung-Yi, Dollár, Girshick, He, Hariharan and Belongie. [Feature Pyramid Networks for Object Detection] (CVPR 2017)

Resolution vs Semantic Value

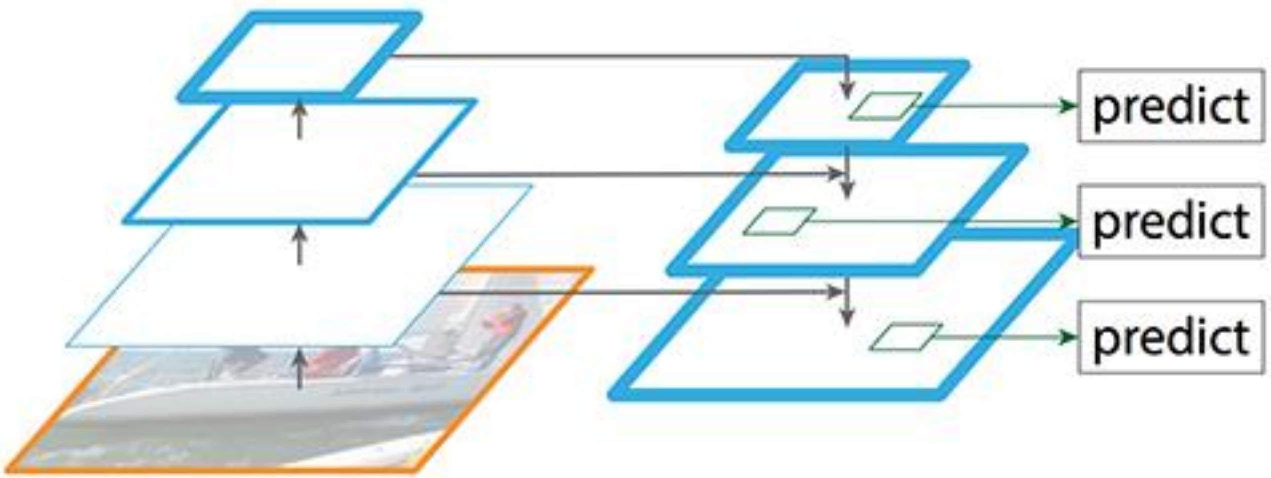
How to handle multiscale predictions?



Tsung-Yi, Dollár, Girshick, He, Hariharan and Belongie. [Feature Pyramid Networks for Object Detection] (CVPR 2017)

Feature Pyramid Network (FPN)

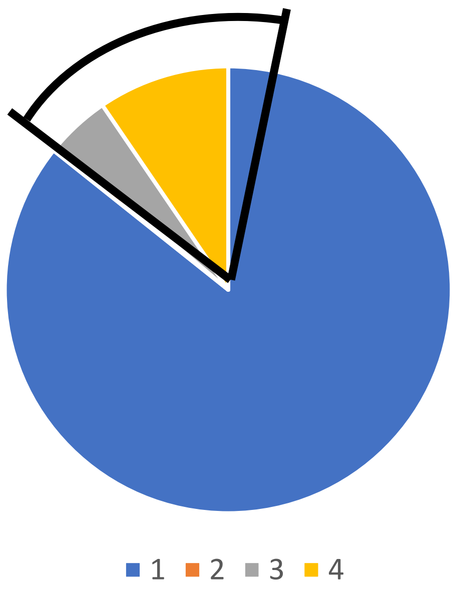
How to handle multiscale predictions?

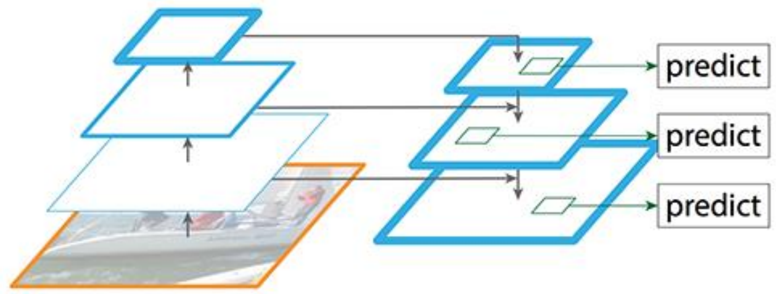


Tsung-Yi, Dollár, Girshick, He, Hariharan and Belongie. Feature Pyramid Networks for Object Detection (CVPR 2017)

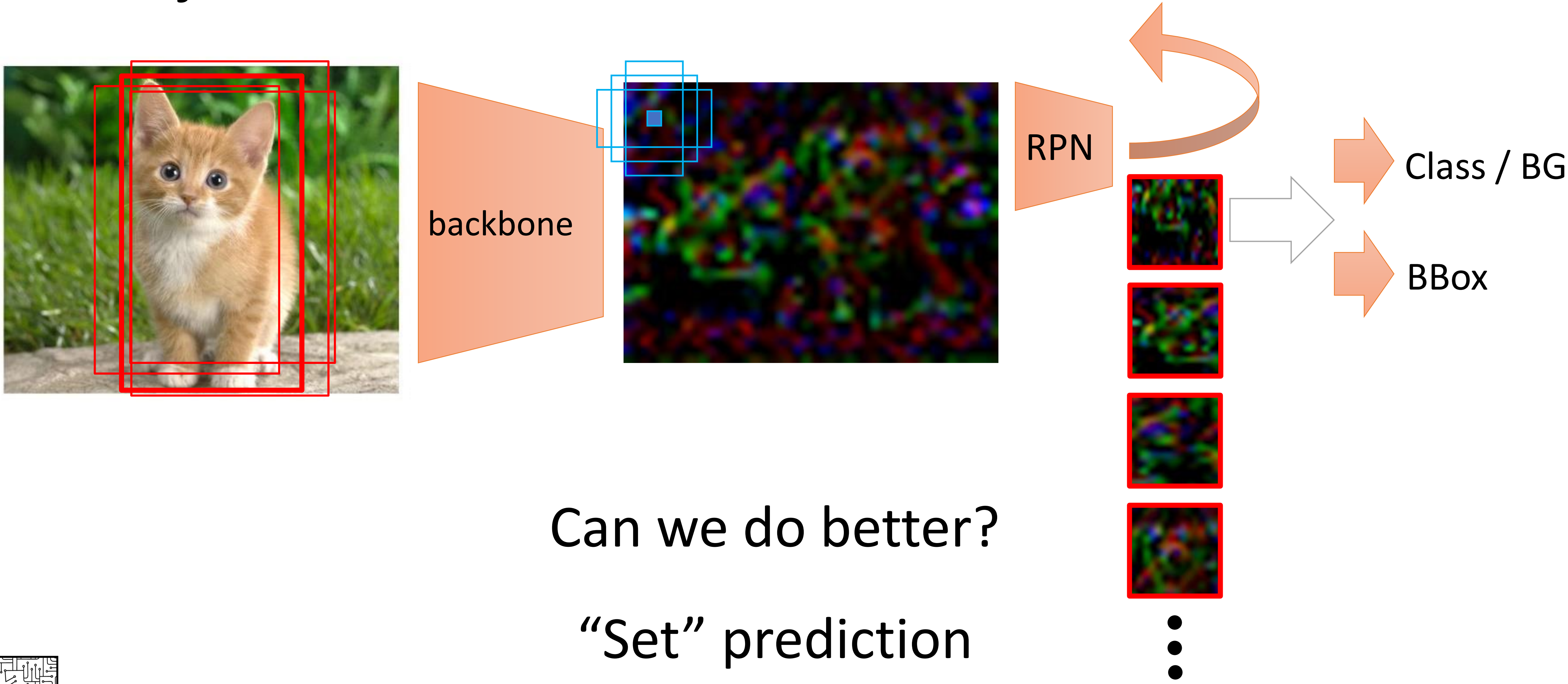
Object Detection

- Imbalance data
- Multiscale





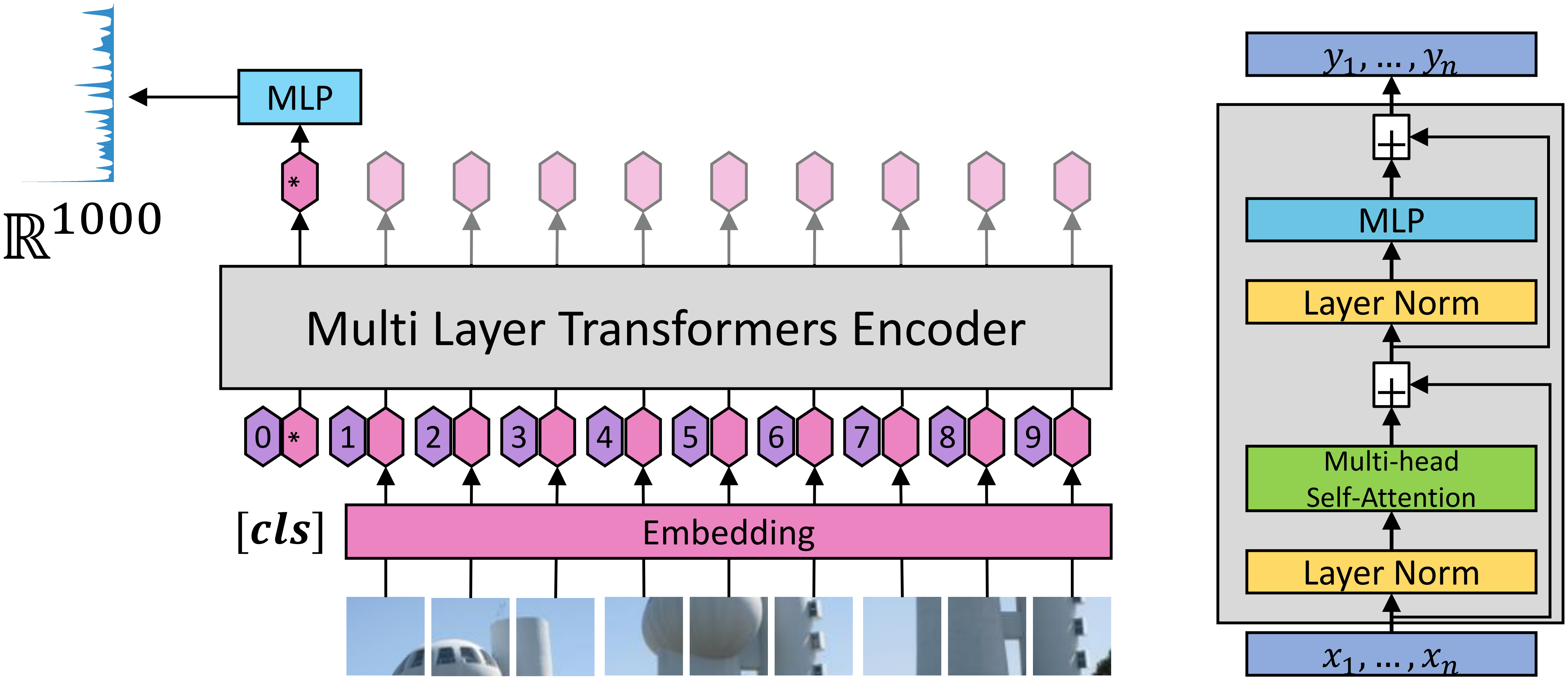
Object Detection



Can we do better?

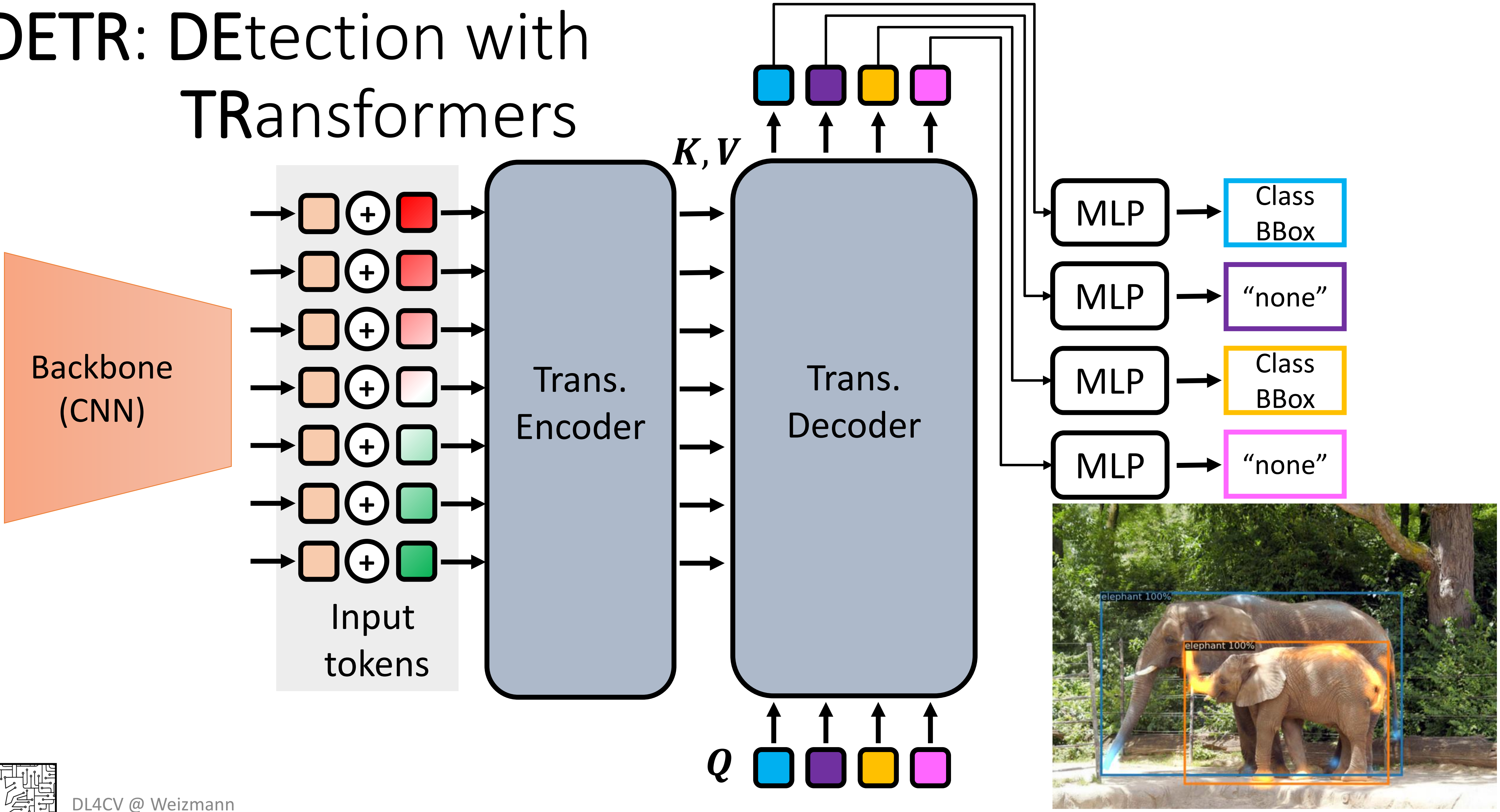
“Set” prediction

Vision Transformers (ViT)



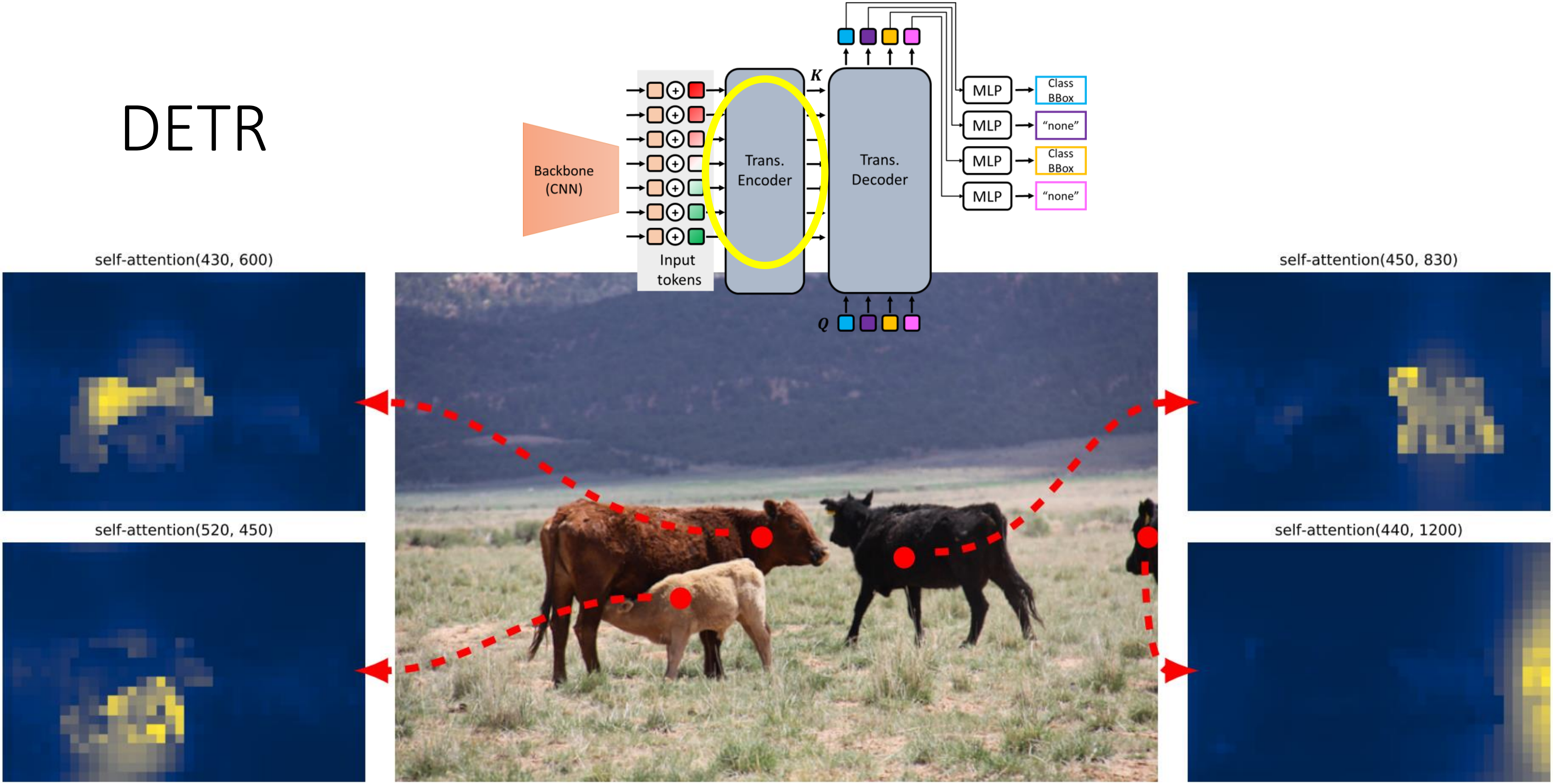
Dosovitskiy A., Beyer L., Kolesnikov A., Weissenborn D., Zhai X., Unterthiner T., Dehghani M., Minderer M., Heigold G., Gelly S., Uszkoreit J. and Houlsby N. "**An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale**" (ICLR 2021)

DETR: DEtection with TRansformers



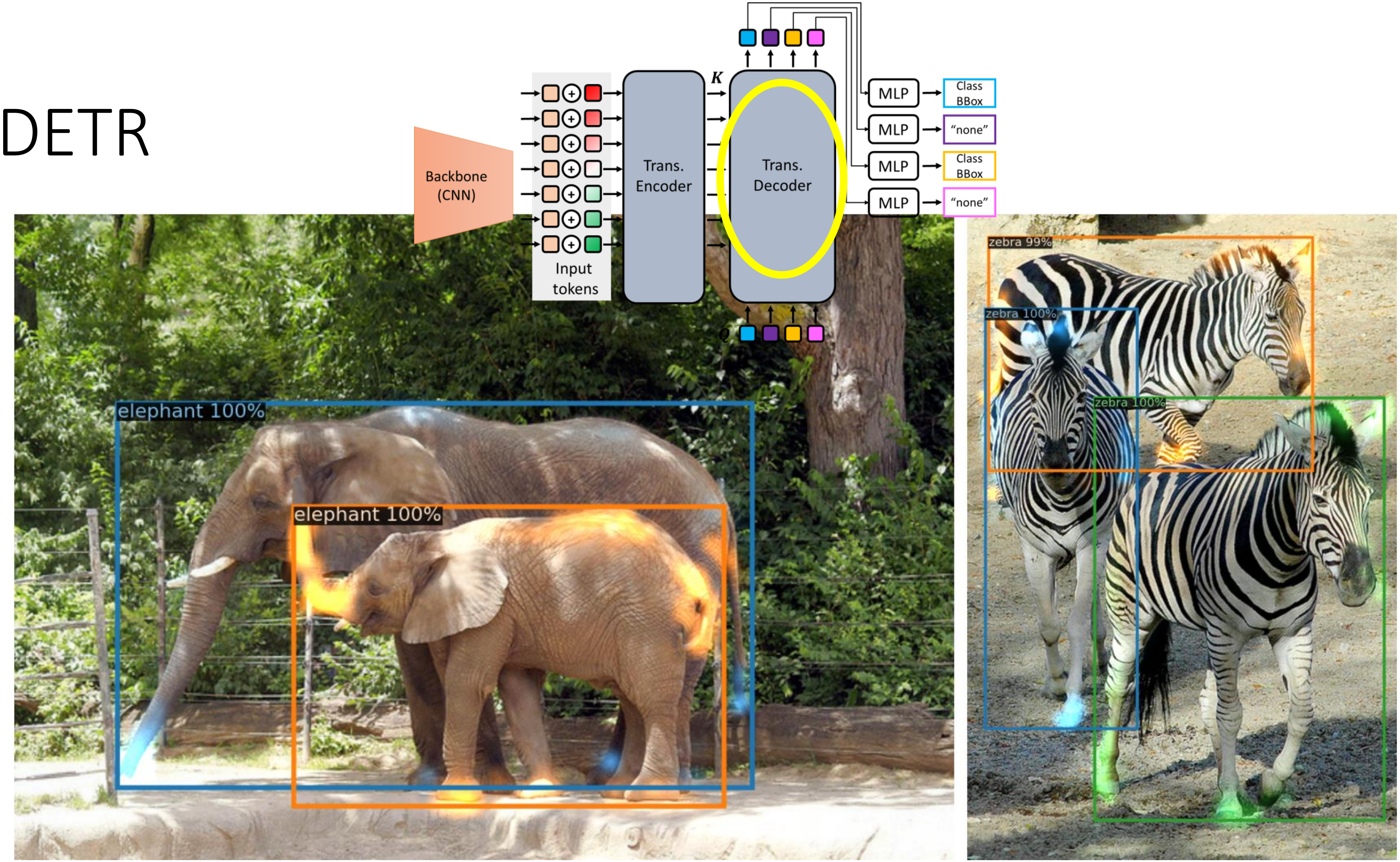
Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. [DETR: End-to-End Object Detection with Transformers]() (ECCV 2020)

DETR



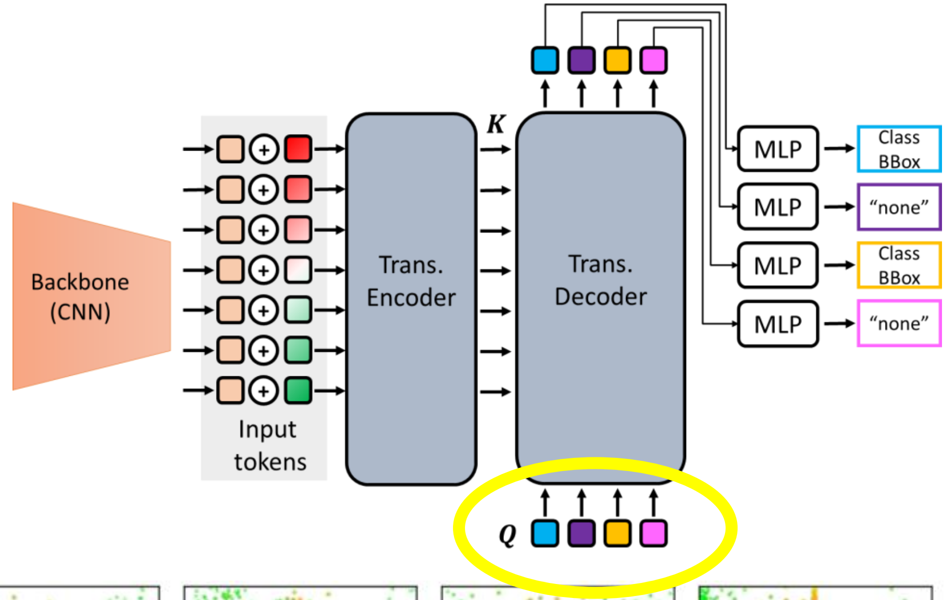
Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. [DETR: End-to-End Object Detection with Transformers] (ECCV 2020)

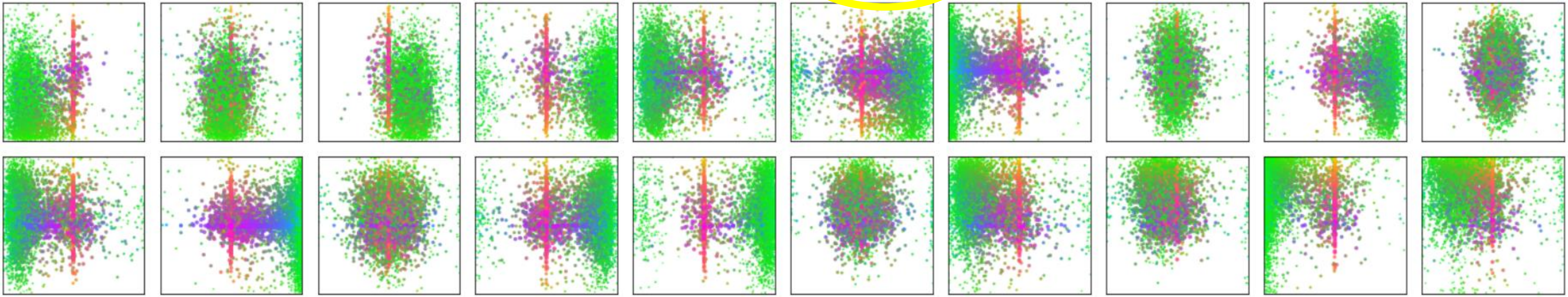
DETR

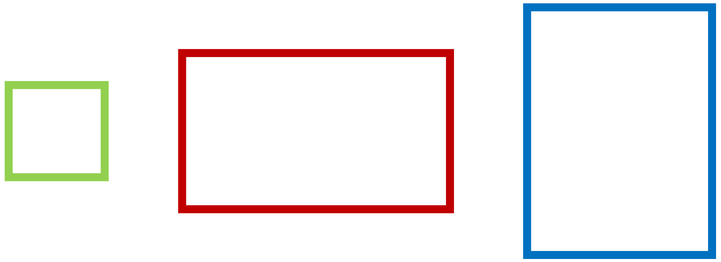


Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. DETR: End-to-End Object Detection with Transformers (ECCV 2020)

DETR

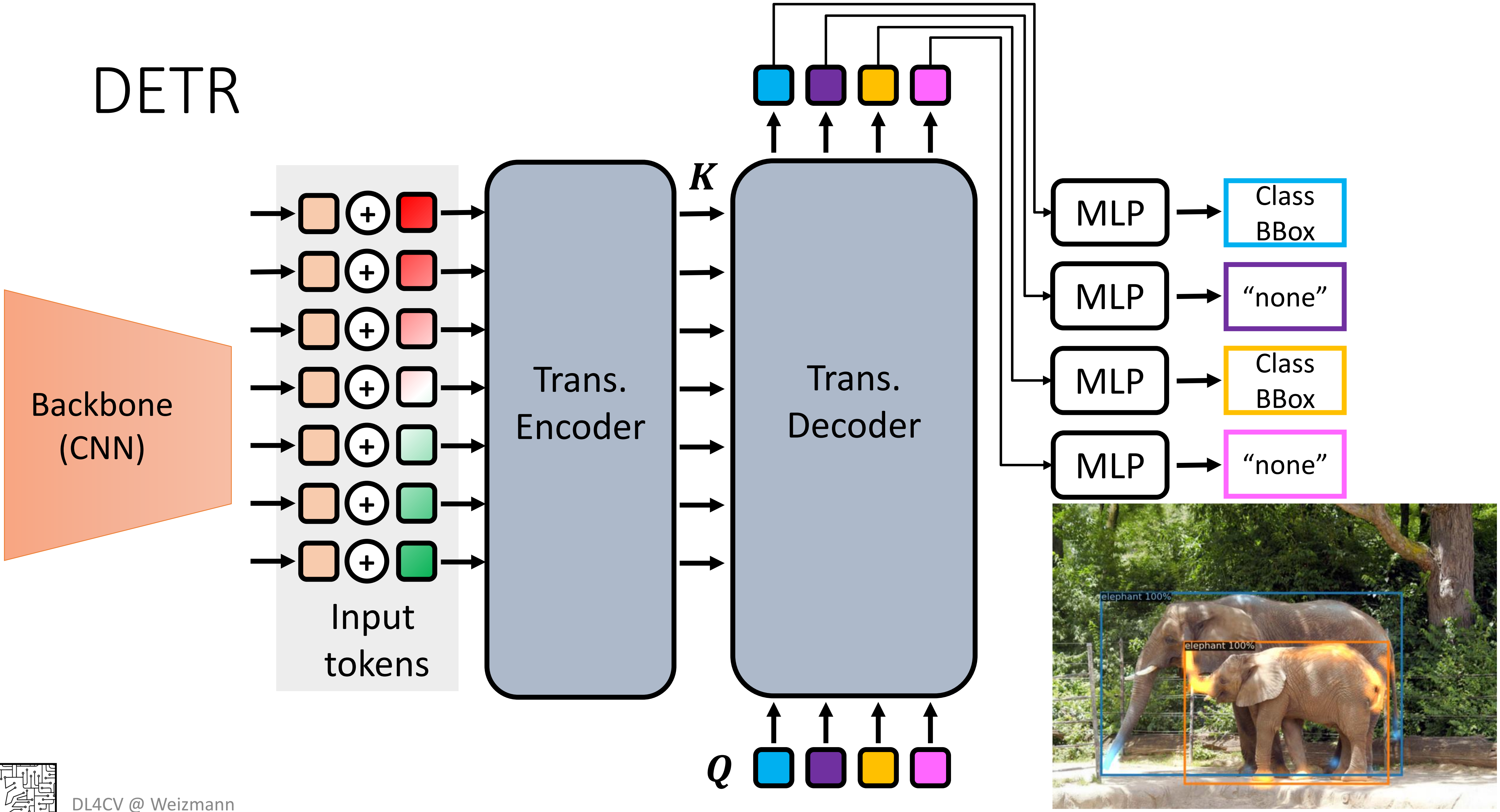






Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. [DETR: End-to-End Object Detection with Transformers](#) (ECCV 2020)

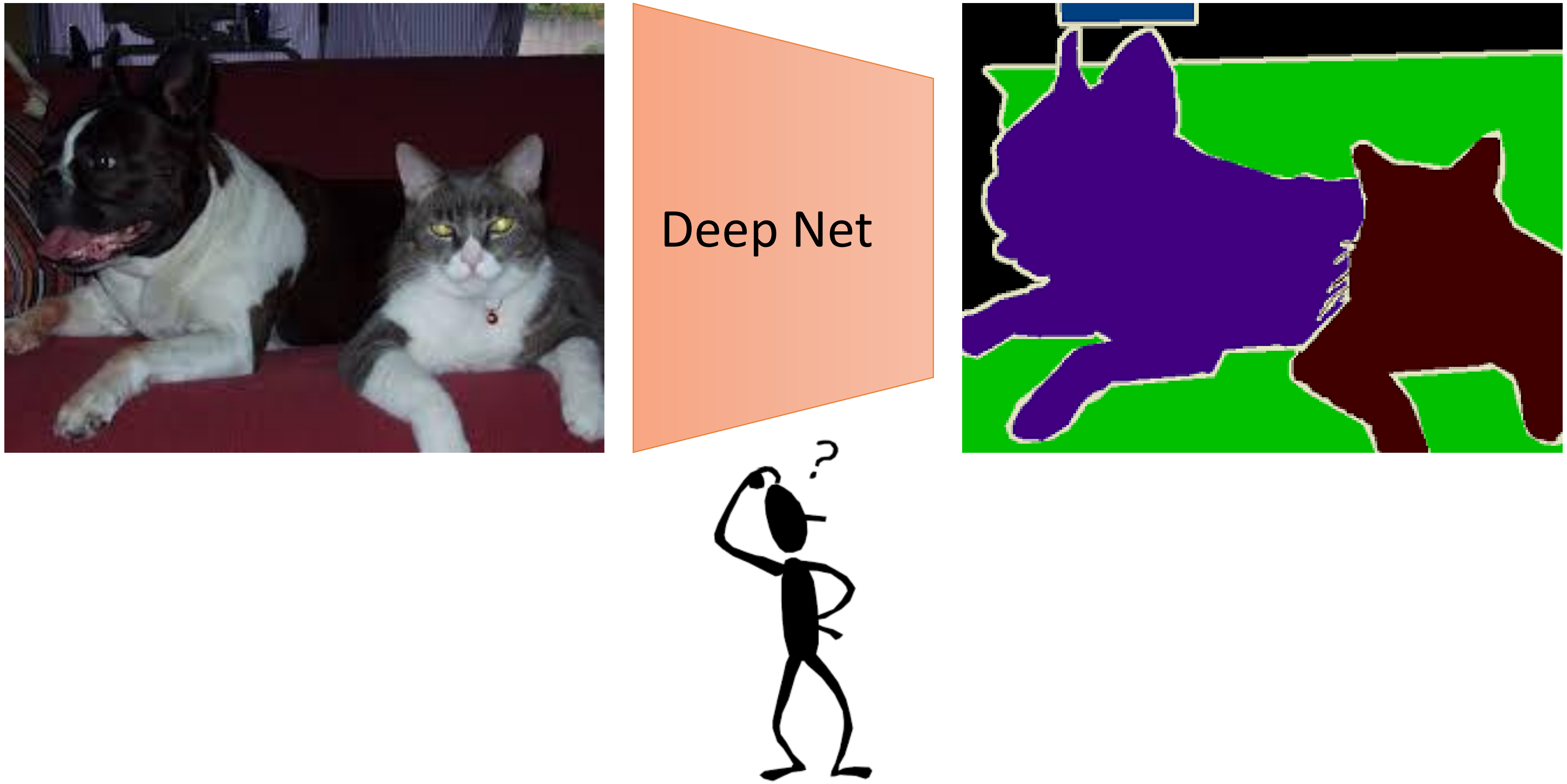
DETR



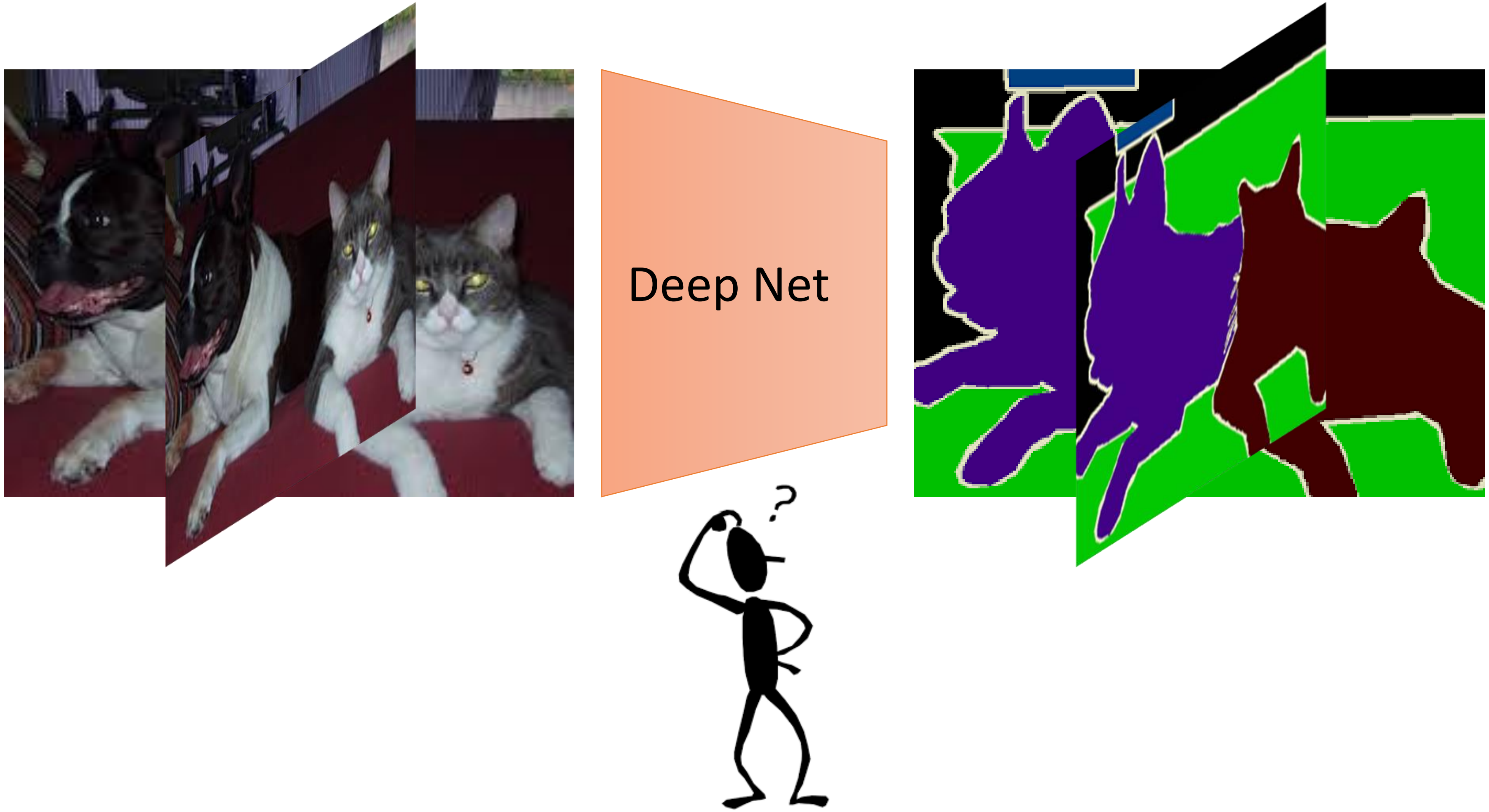
Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. [DETR: End-to-End Object Detection with Transformers]() (ECCV 2020)

Semantic Segmentation

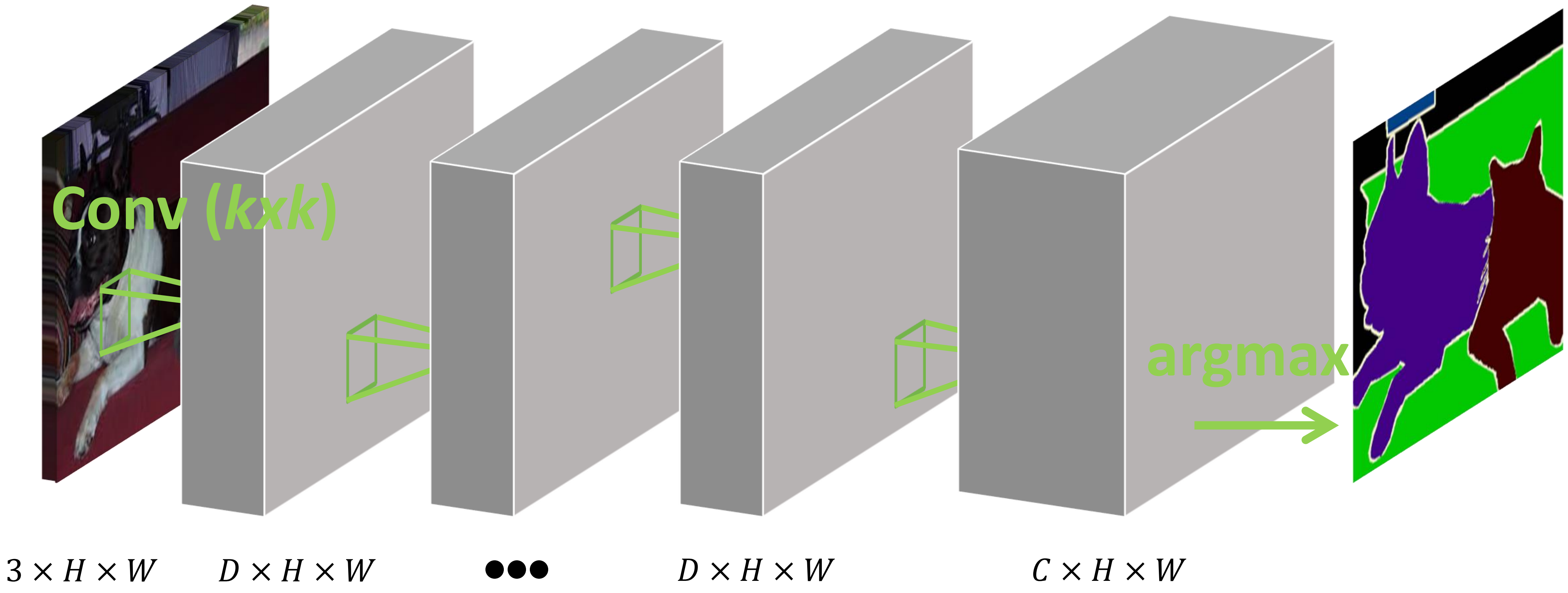
Semantic Segmentation



Semantic Segmentation

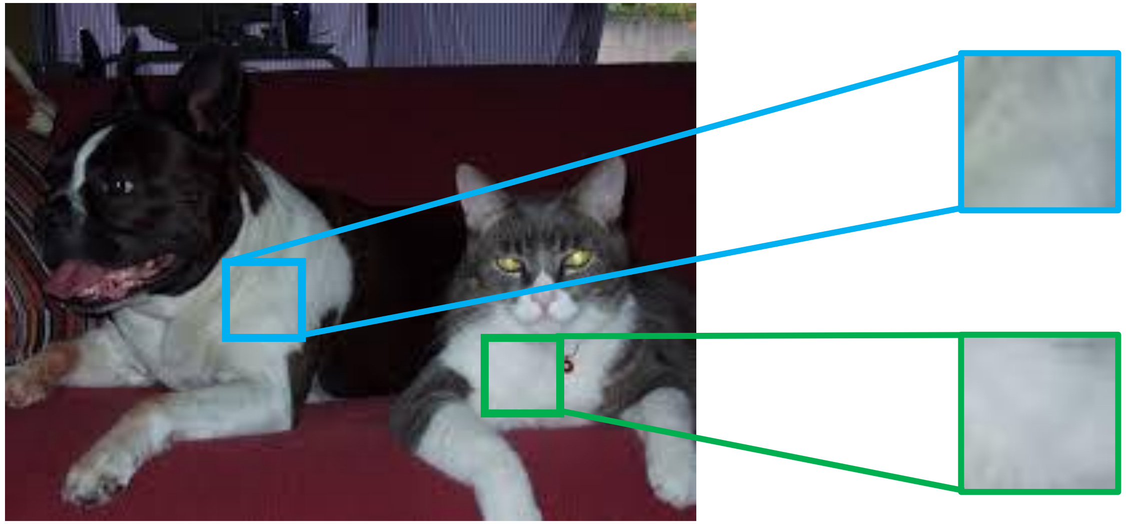


Semantic Segmentation (v0)



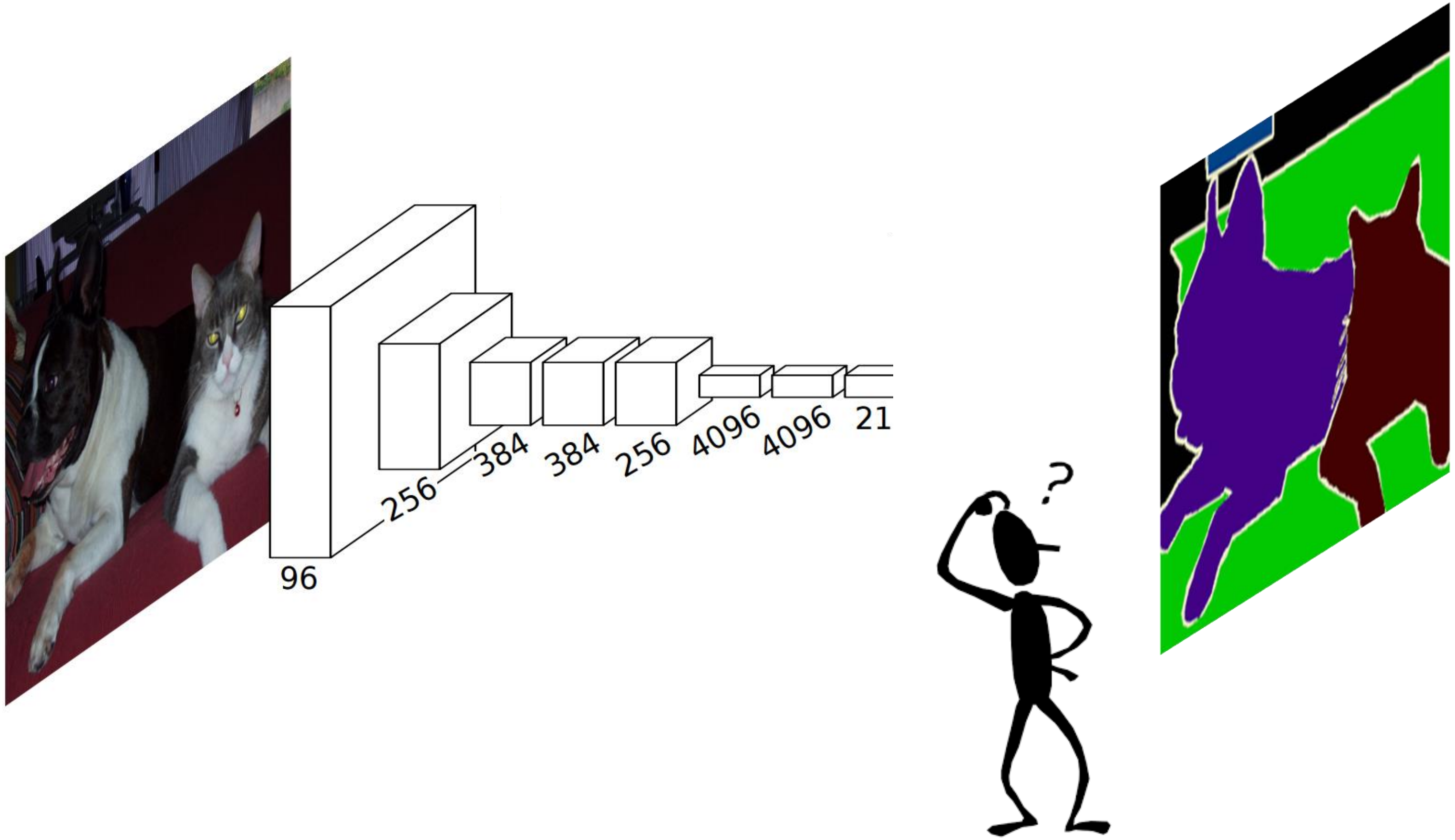
Semantic Segmentation (v0)

Challenge: Context (RF) vs Resolution



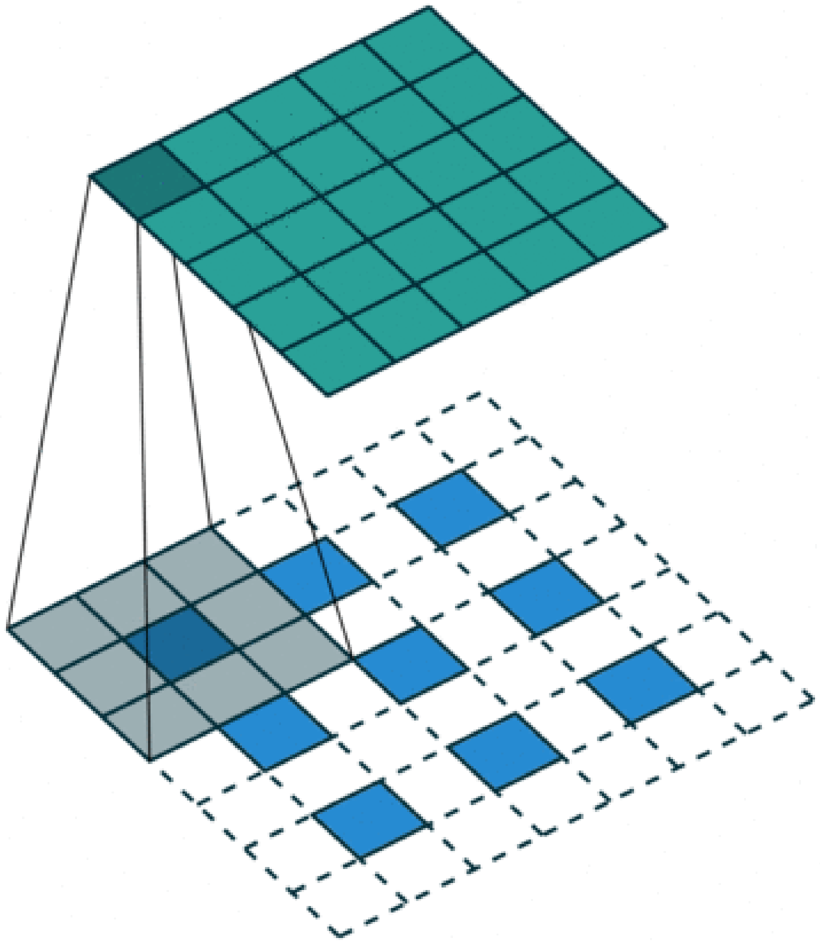
Semantic Segmentation - FCN

Replace FC layers with conv – “sliding window” classification



Long, Shelhamer and Darrell [Fully convolutional networks for semantic segmentation](#) (CVPR 2015)

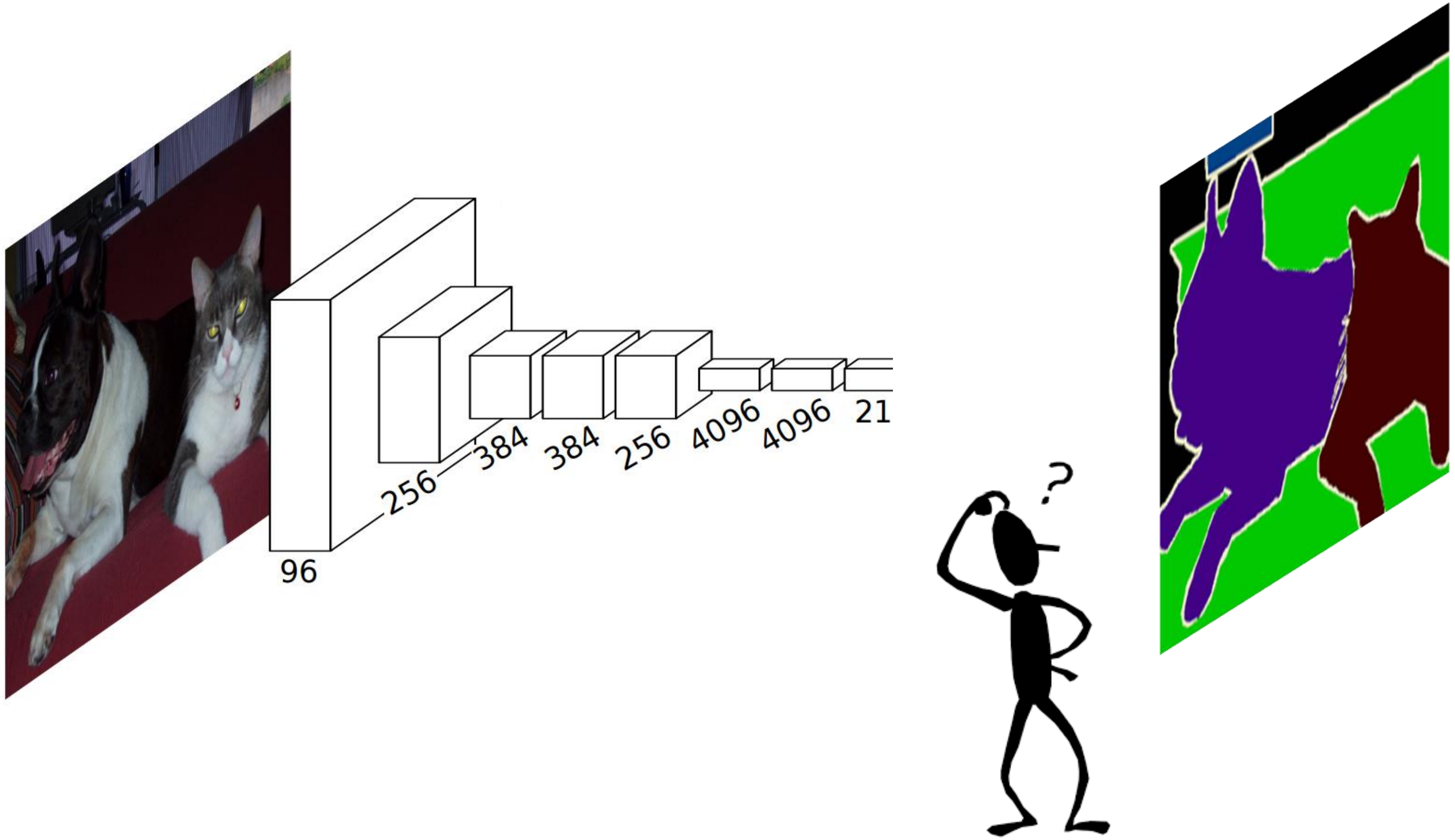
Transposed Convolution



In depth: [here]

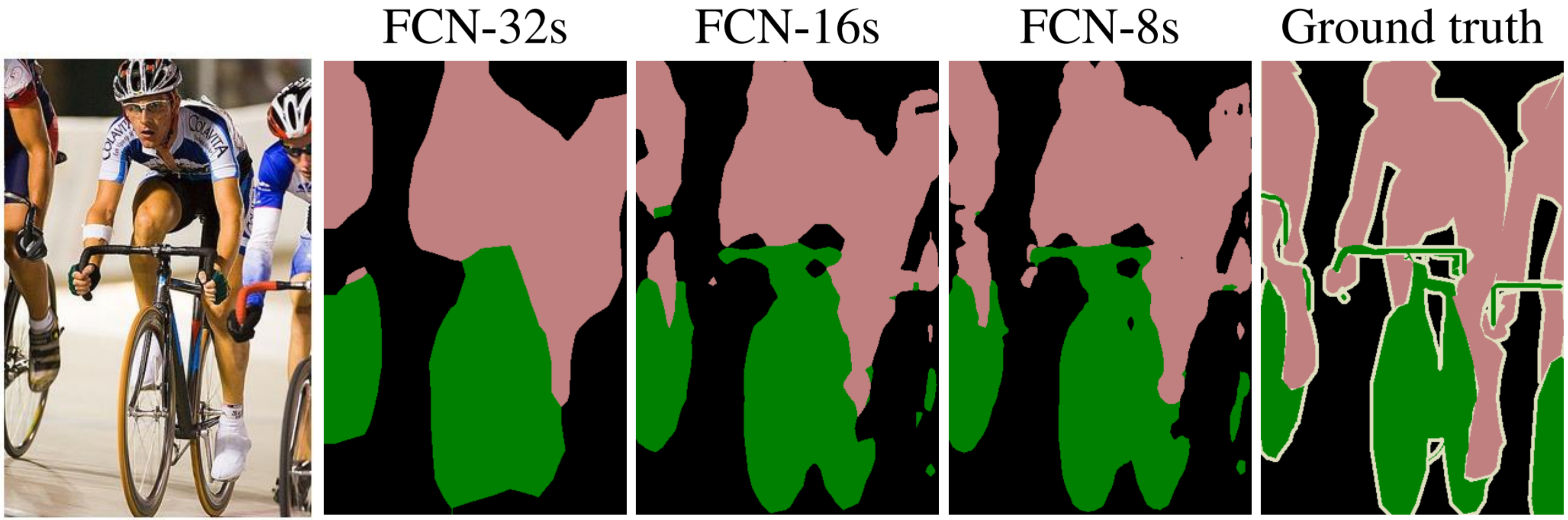
Semantic Segmentation - FCN

Replace FC layers with conv – “sliding window” classification



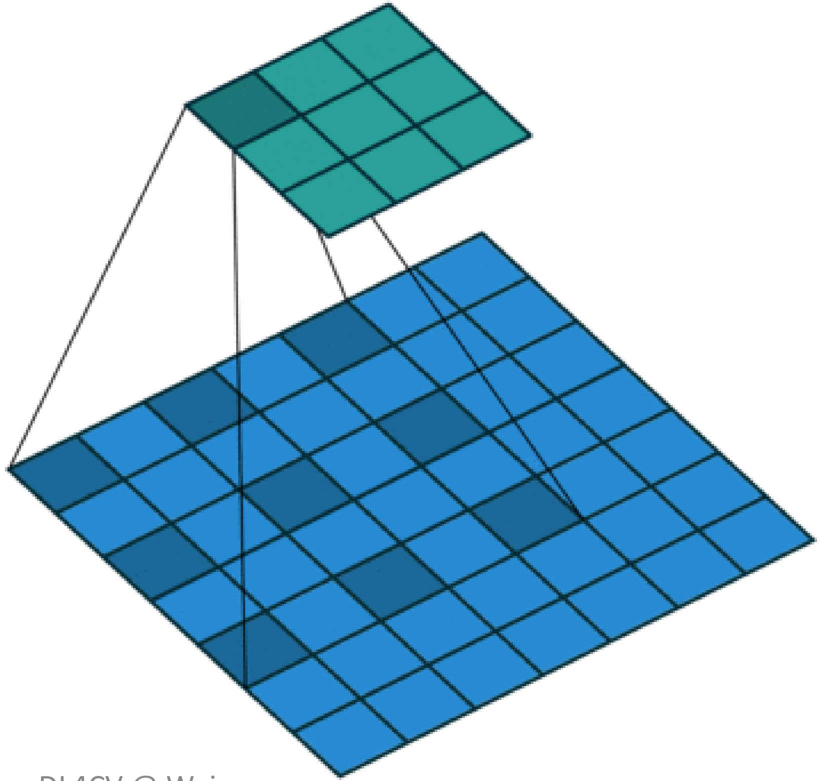
Long, Shelhamer and Darrell [Fully convolutional networks for semantic segmentation]() (CVPR 2015)

Semantic Segmentation - FCN

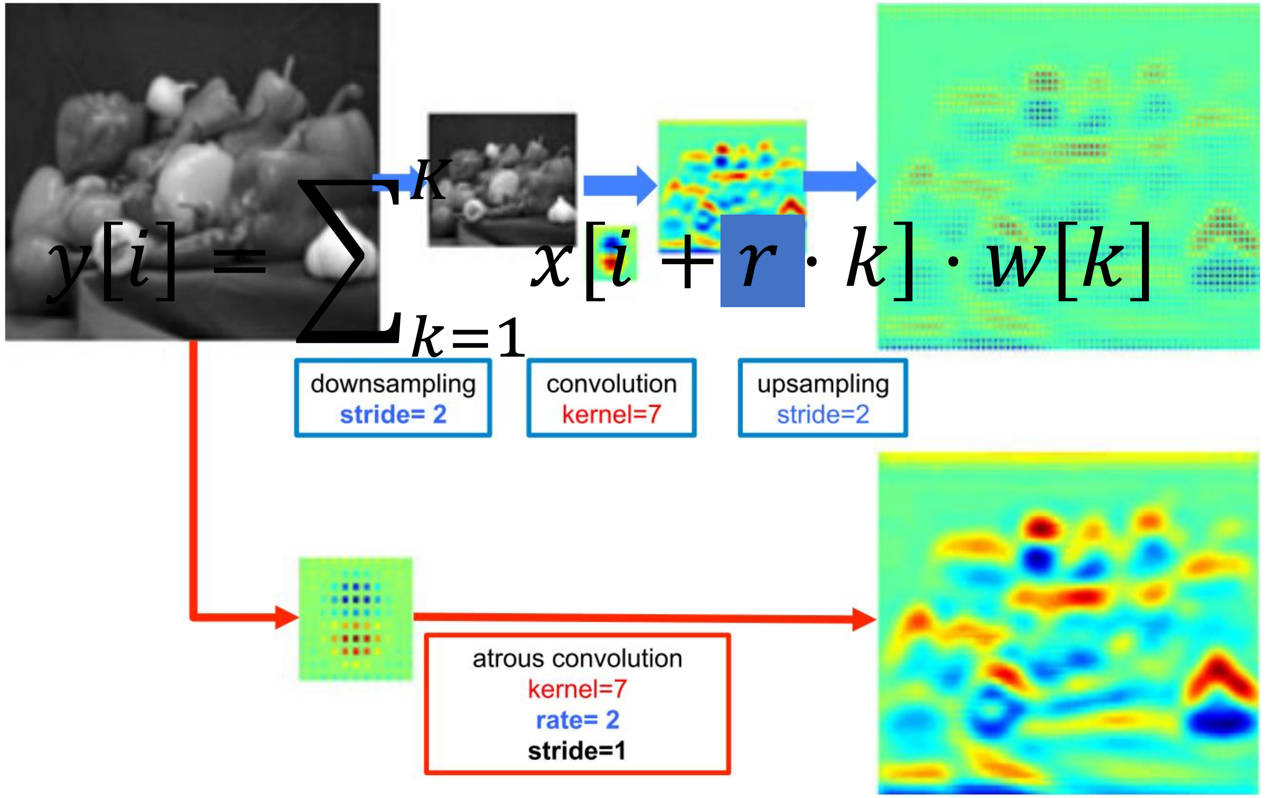


DeepLab: Atrous Convolution

Chen, Papandreou, Kokkinos, Murphy and Yuille
Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs (*PAMI 2018)*

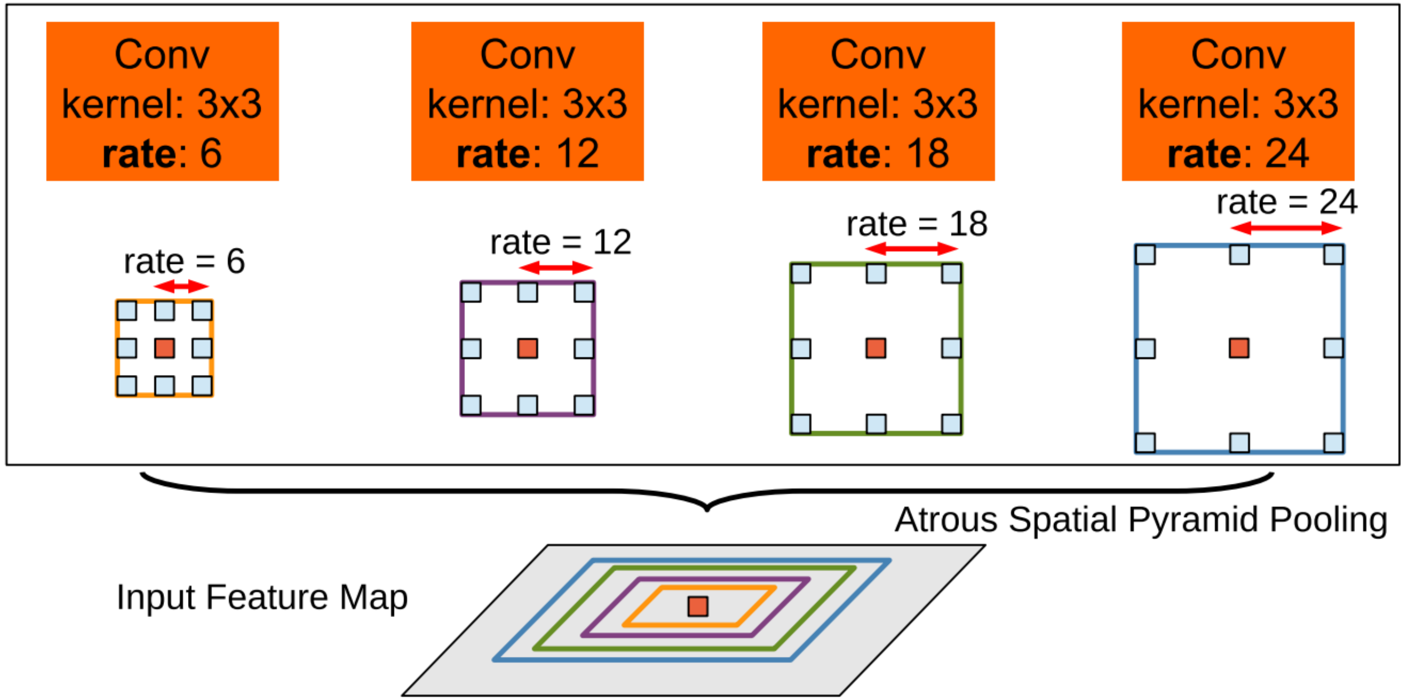


$$y[i] = \sum_{k=1}^{K} x[i + r \cdot k] \cdot w[k]$$



DeepLab: Atrous Convolution

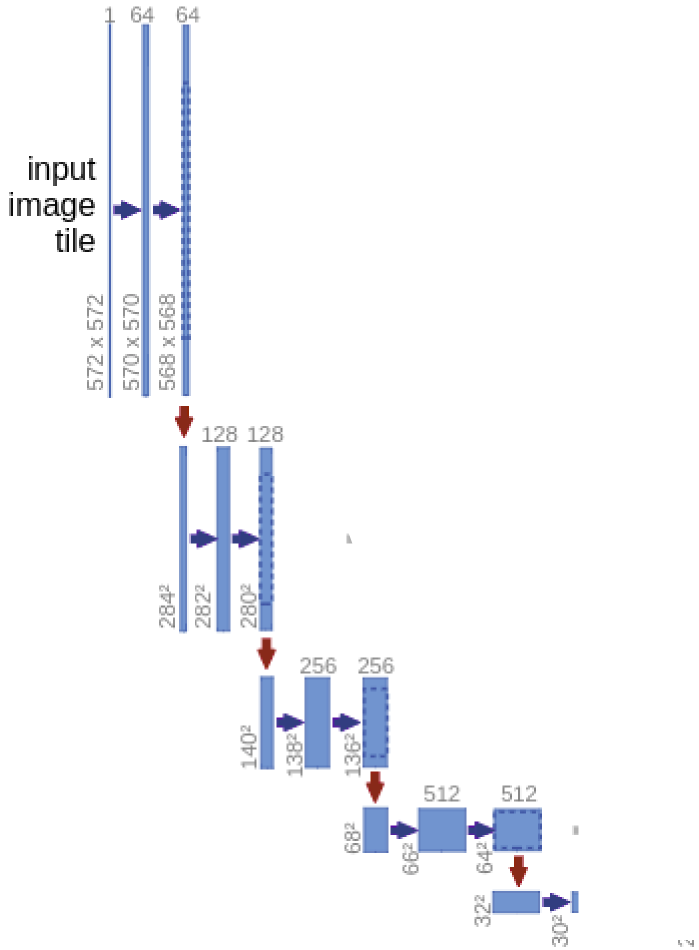
Chen, Papandreou, Kokkinos, Murphy and Yuille
[Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs]() (*PAMI 2018)*



DeepLab: Atrous Convolution

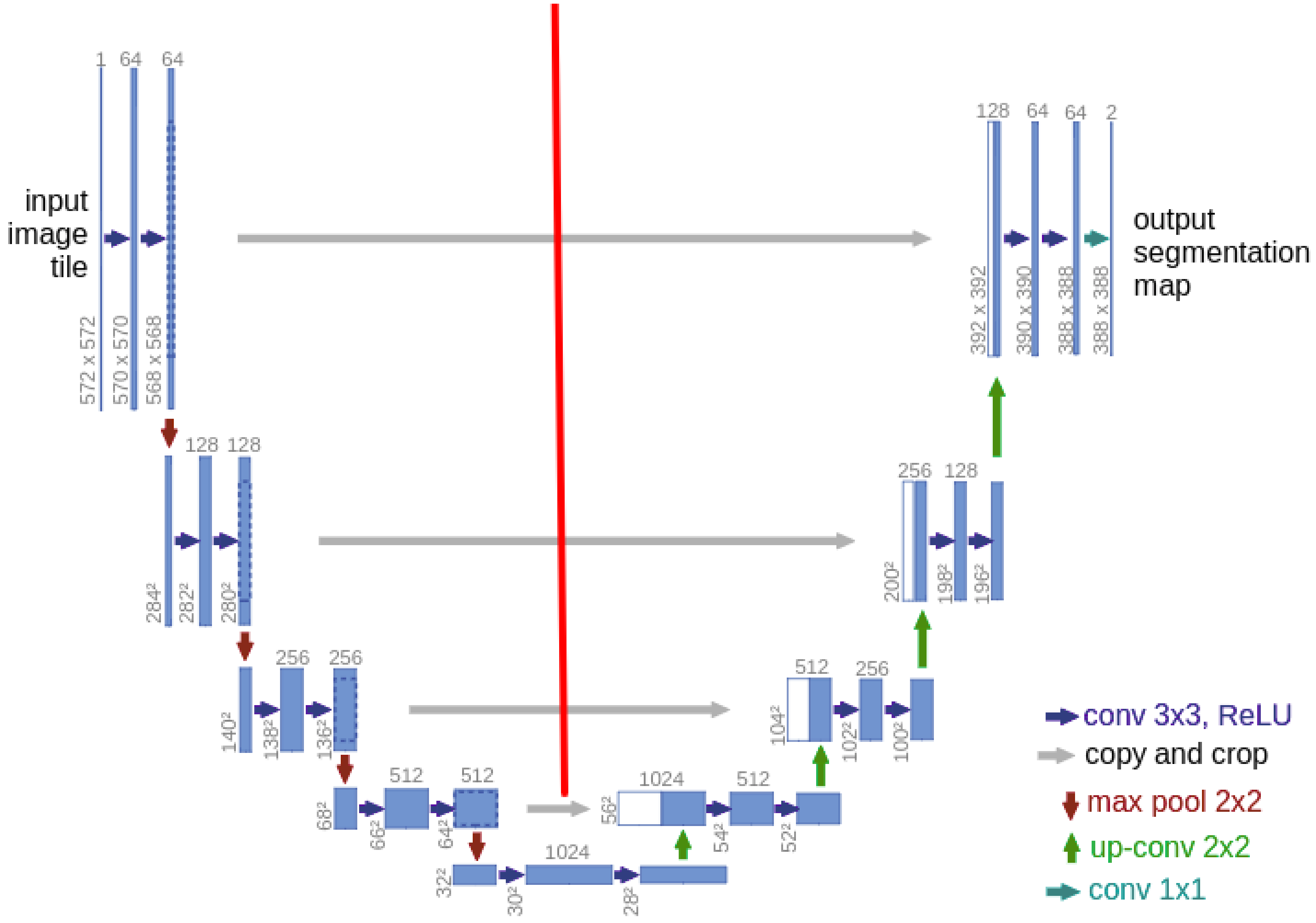
- Trade stride/pooling with “dilation” of kernel
- Increase receptive field without increase in parameters

Semantic Segmentation – U-net



Olaf, Fischer and Brox **U-net: Convolutional networks for biomedical image segmentation** (2015)

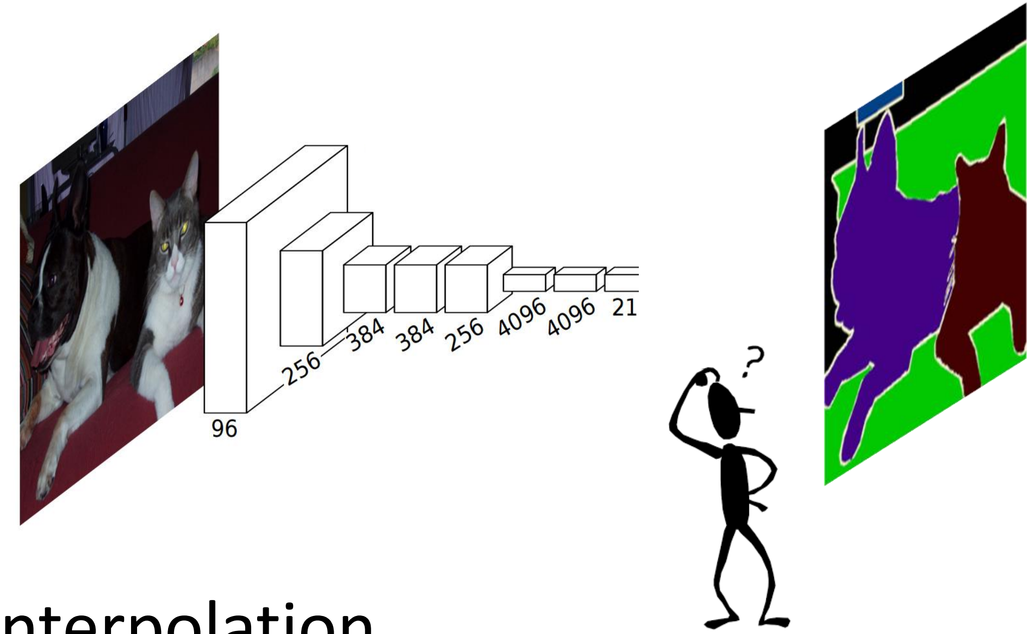
Semantic Segmentation – U-net



Olaf, Fischer and Brox [U-net: Convolutional networks for biomedical image segmentation]() (2015)

Semantic Segmentation

Resolution vs. Semantic information



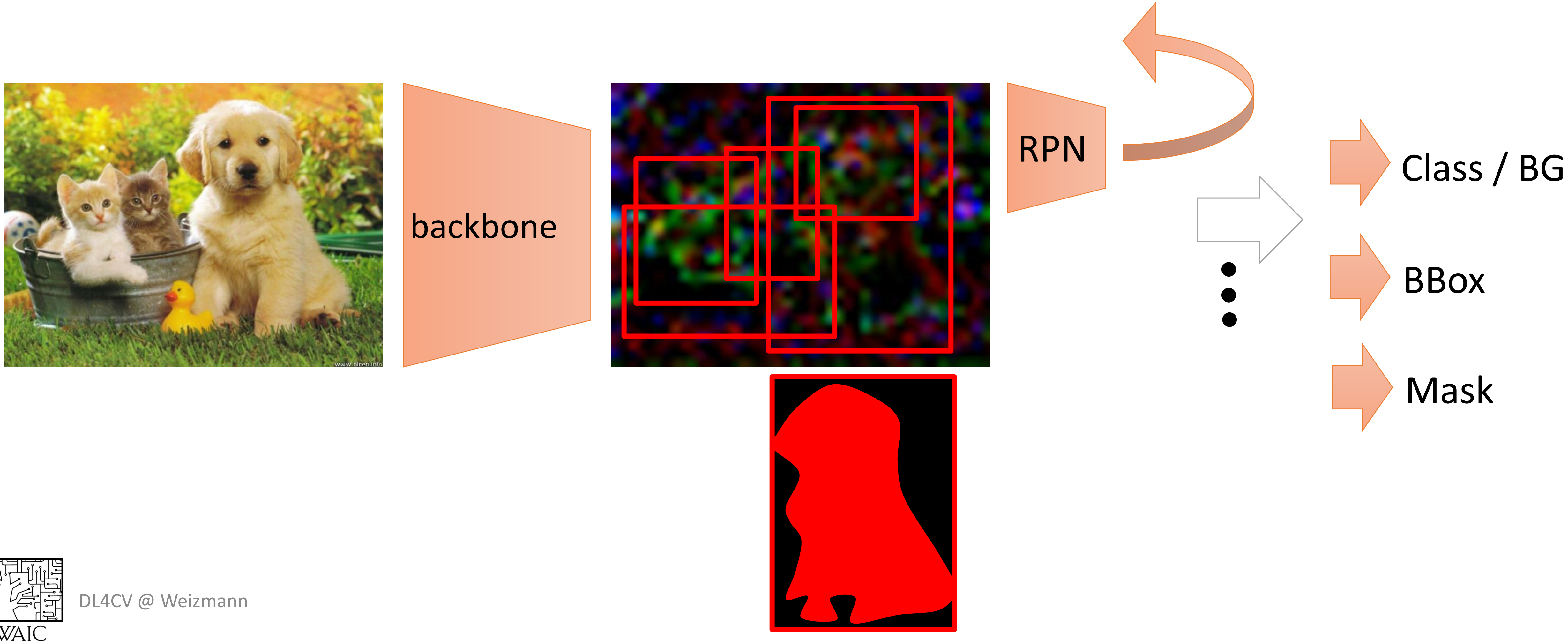
- FCN: using "transposed convolution"
- DeepLab: dilated convolution + simple interpolation
- U-net: skip connections

Instance Segmentation

Instance Segmentation: Detection or Segmentation?

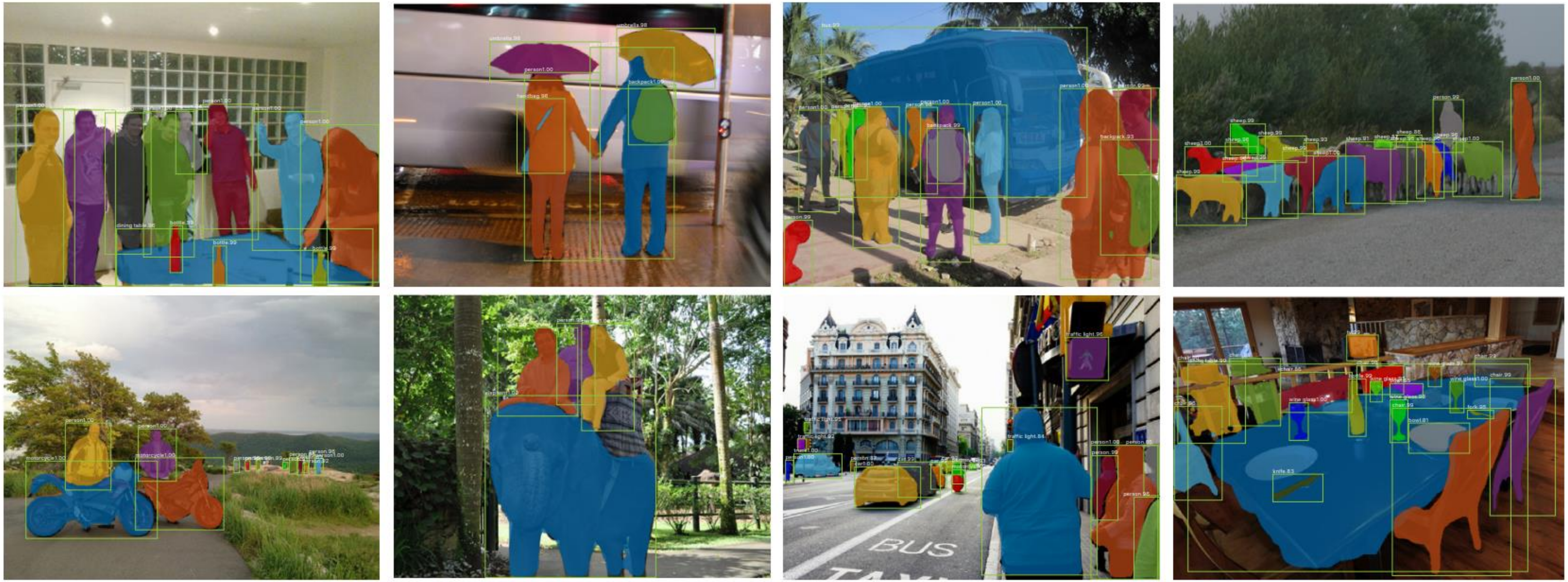
Instance Segmentation: Mask R-CNN

He, Gkioxari, Dollar and Girshick “[Mask R-CNN]” (ICCV 2017)



Instance Segmentation: Mask R-CNN

He, Gkioxari, Dollar and Girshick “**Mask R-CNN**” (ICCV 2017)



Summary

- Deep learning beyond image classification
- Classification “backbones” = “transfer learning”
- Same features - multitasking
- Handling varying number of predictions
- Coping with RF/resolution trade-offs

What's next?

- Tutorial – Final project!
- Next week - GANs