Deep Learning for Instance-level Object Understanding

CVPR 2017 Tutorial on Deep Learning for Objects and Scenes

Ross Girshick

in collaboration with

Kaiming He, Georgia Gkioxari, Tsung-Yi Lin (Cornell Tech),

Bharath Hariharan, Piotr Dollár

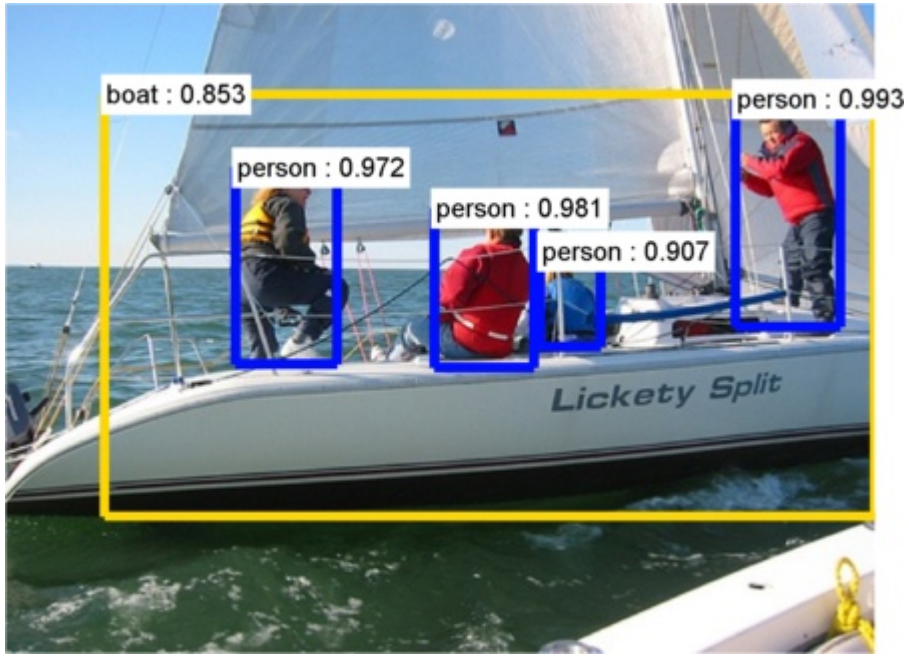
Facebook AI Research (FAIR)

… Now you're Experts in Deep Representations

- This talk: object detection and instance-level object understanding

Object Detection

Image classification
(what? *I don't care where*)



Object detection
(what + where?)

Object Detection circa 2007

Felzenszwalb, Ramanan, McAllester. A Discriminatively Trained, Multiscale, Deformable Part Model. CVPR 2008 (DPM v1)

Object Detection Today

He, Gkioxari, Dollár, Girshick. Mask R-CNN. In ICCV 2017

Instance-level Object Understanding Today

He, Gkioxari, Dollár, Girshick. Mask R-CNN. In ICCV 2017

Instance-level Object Understanding Today

He, Gkioxari, Dollár, Girshick. Mask R-CNN. In ICCV 2017

Instance-level Object Understanding Today

Gkioxari, Girshick, Dollár, He. Detecting and Recognizing Human-Object Interactions. arXiv 2017

Outline

Mask R-CNN [25 min]

- Task
- Model
- Training
- Inference

Deep Learning Methods for Object Detection [10 min *brief* survey]

- Landscape of methods
- Speed/accuracy tradeoffs
- How we got here (2007 vs. 2017)
- Some open directions in instance-level understanding

Task: Instance Segmentation

Object detection

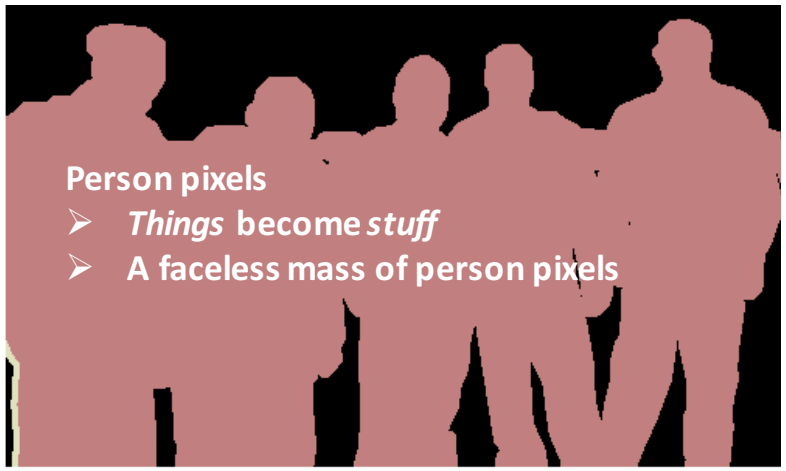
Image from PASCAL VOC

Task: Instance Segmentation

Object detection

Image from PASCAL VOC

Semantic segmentation

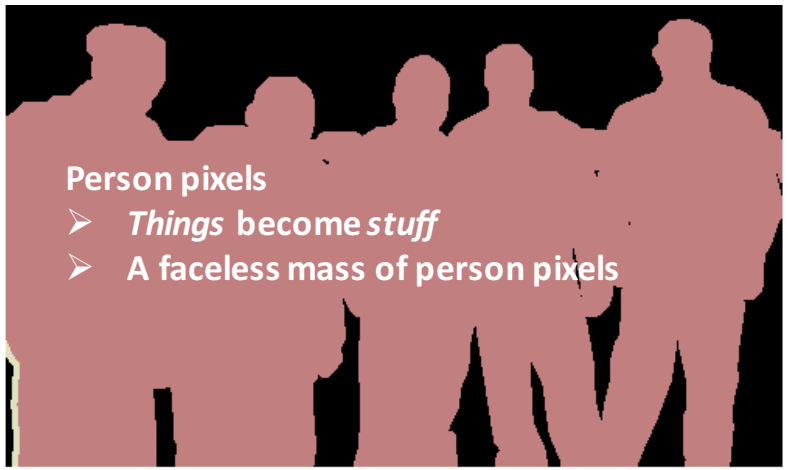


Task: Instance Segmentation

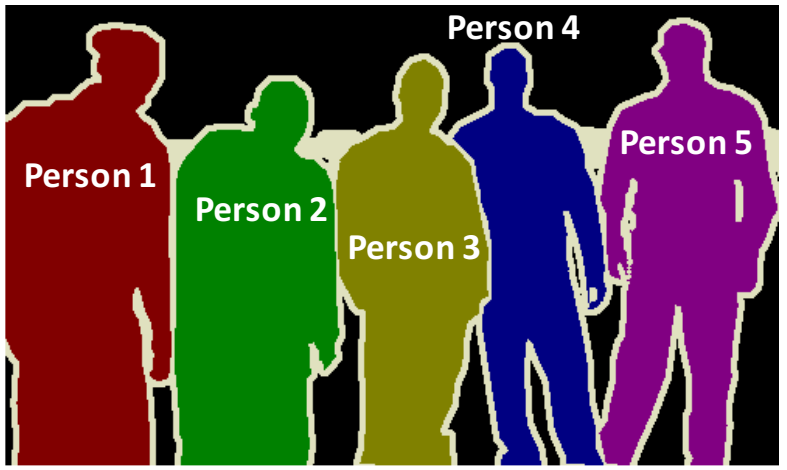
Object detection

Image from PASCAL VOC

Semantic segmentation



Instance segmentation



Best of both tasks

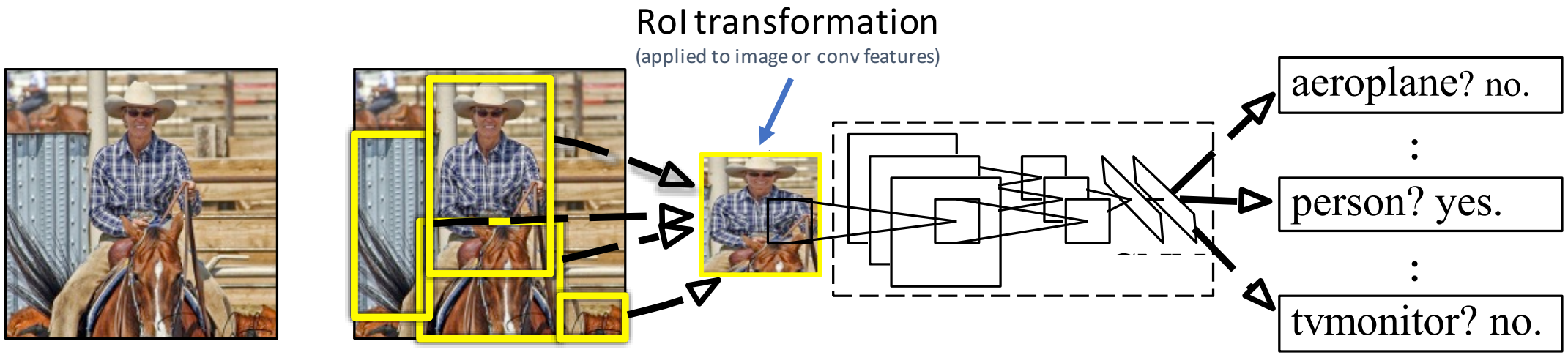
Mask R-CNN: Model Overview

R-CNN-style detection system

1. Backbone architecture
2. Feature Pyramid Network (FPN)
3. Region Proposal Network (RPN)
4. Region of interest feature alignment (RoIAlign)
5. Multi-task network head
   - Box classifier
   - Box regressor
   - Mask predictor
   - Keypoint predictor

Modular composition of many recent ideas

0. R-CNN-style Approach to Object Detection



Input image

Object / region proposals (RoIs)
(external or internal to network)

Deep Learning-based region classifier

Detection = Region classification + box regression + …

General formula for **R**egion-based **CNN** models

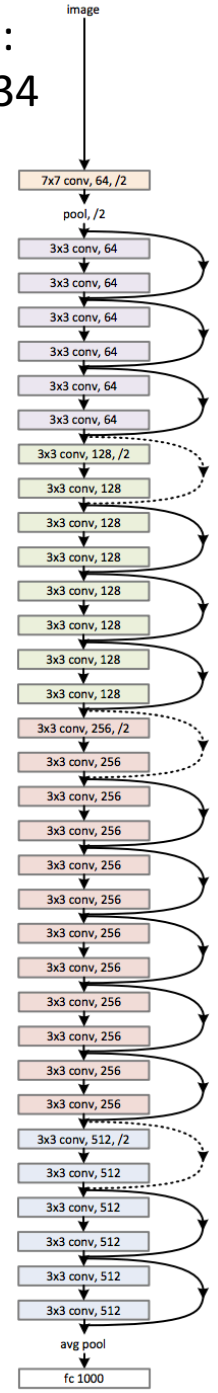
Girshick, Donahue, Darrell, Malik. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. CVPR 2014
Uijlings, van de Sande, Gevers, Smeulders. Selective Search for Object Recognition. IJCV 2013

1. Backbone ConvNet

Use any standard ConvNet as a “backbone architecture”

- AlexNet, VGG, ResNet, Inception, Inception-ResNet, ResNeXt (poster 67 Tuesday morning), …
- Use “same” padding everywhere (preserves integer scales)
- Prefer fully convolutional networks (ignoring cls head; see next slide)
- Pre-train on ImageNet classification (or similar)

Example:
ResNet-34

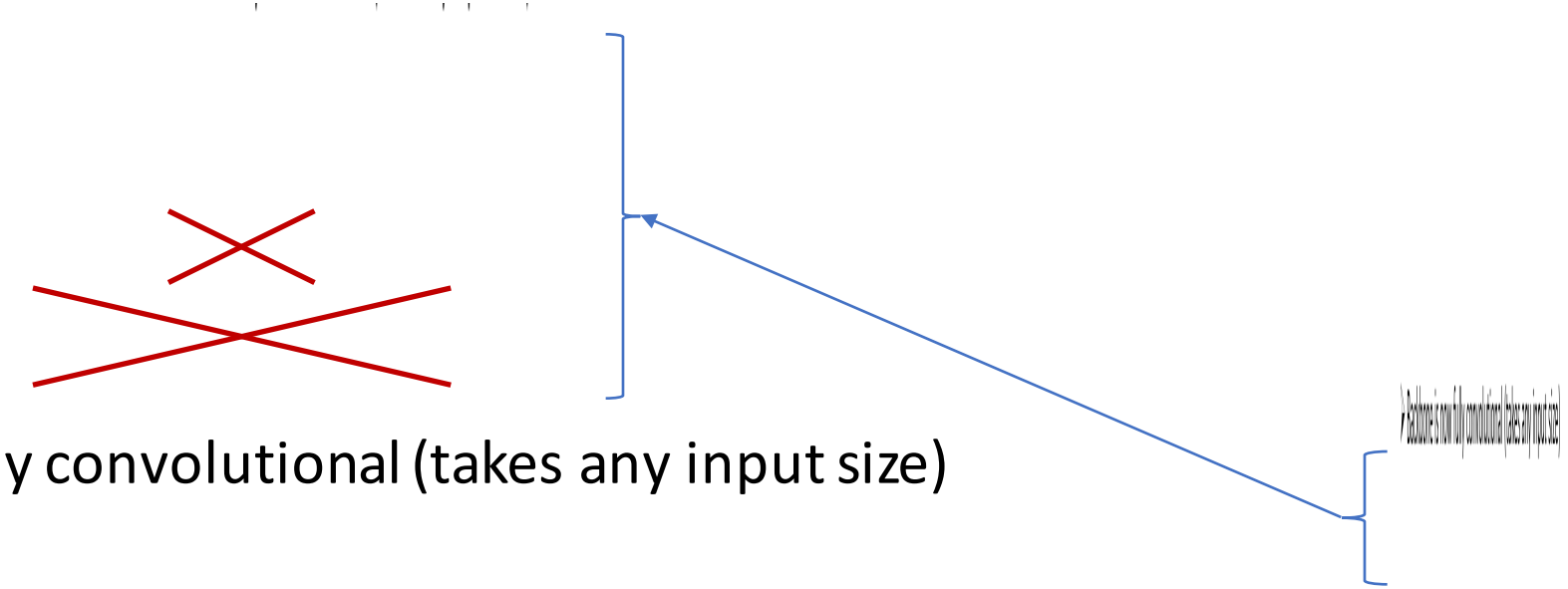


1. Prepare for Detection! (Minor Surgery)

Replace test-time batch normalization with fixed affine transform

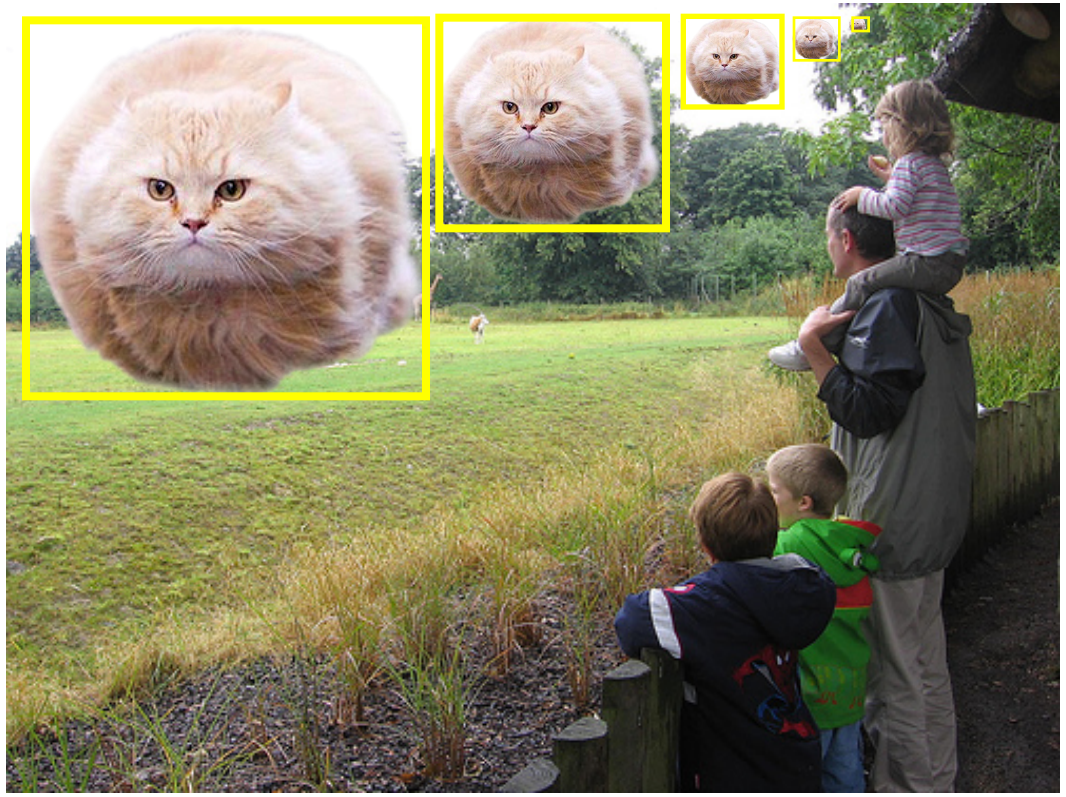
- $\mathrm{BN}_{\mathrm{test}}(x) = \gamma(x - \mu)/\sigma + \beta$, treat $\gamma, \beta, \mu, \sigma$ as constants

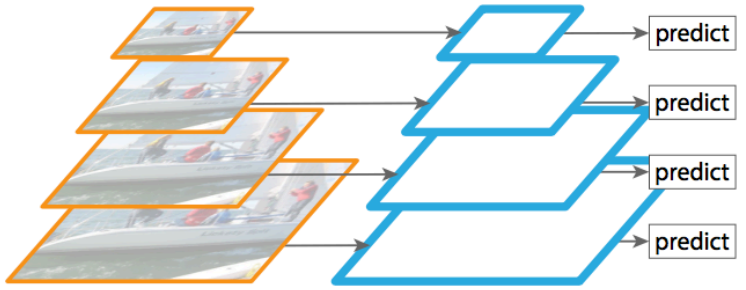
Remove pre-trained classification head



- Backbone is now fully convolutional (takes any input size)

2. Scale Invariant Detection

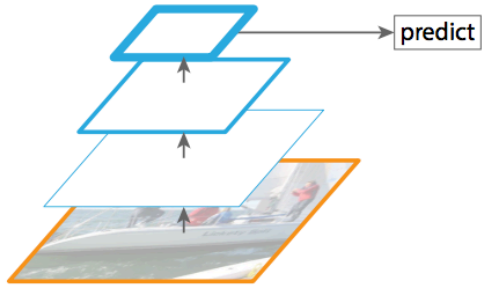




(a) Featurized image pyramid

SLOW!

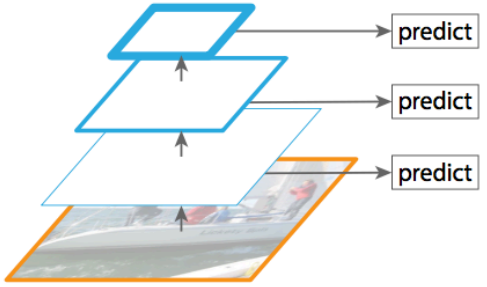
(Viola & Jones, HOG, DPM, multiscale Fast R-CNN, …)



(b) Single feature map

Do nothing; give up; Fast

(Fast R-CNN, YOLO, …)



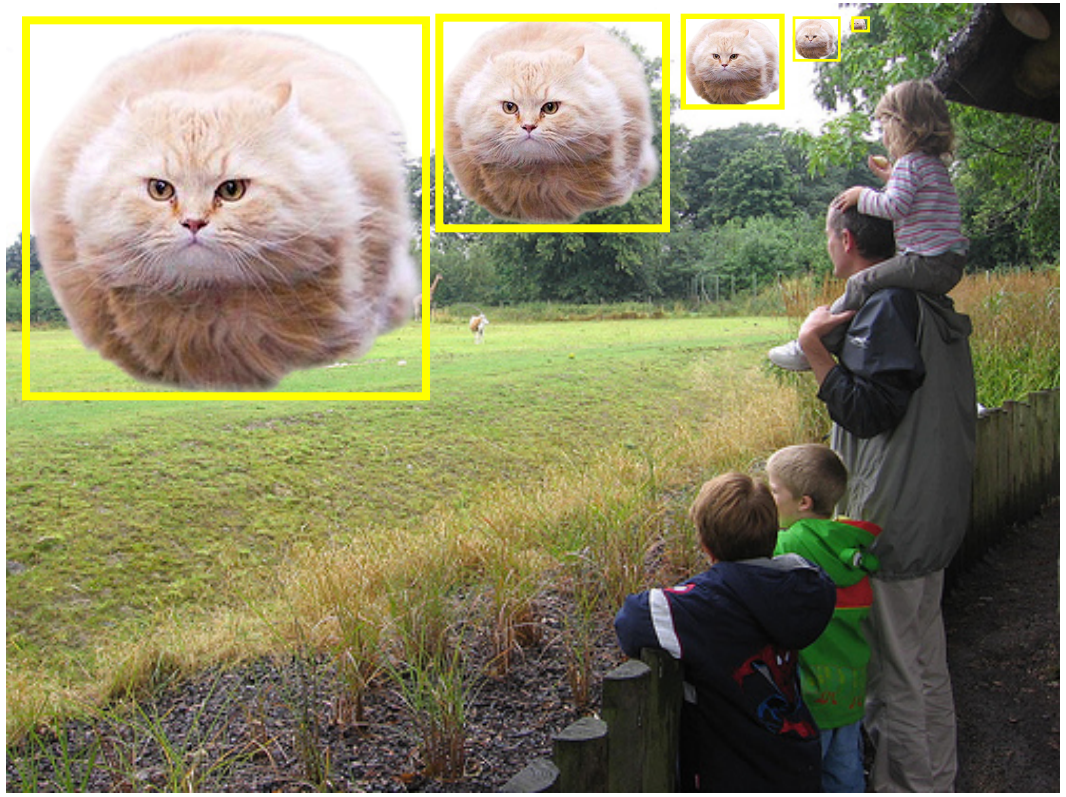
(c) Pyramidal feature hierarchy

Fast!

(≈ SSD, …)

Popular strategies for detecting objects over large scale changes

2. Scale Invariant Detection



(a) Featurized image pyramid

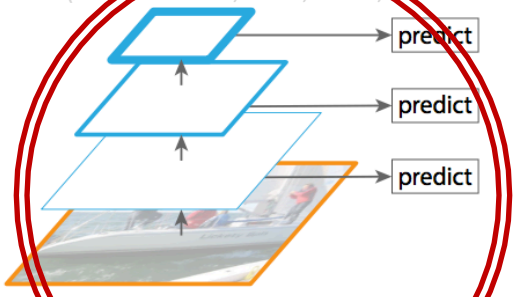
SOW!!

(Viola & Jones, HOG, DPM, multiscale Fast R-CNN, …)

(b) Single feature map

Do nothing; give up; Fast!

(Fast R-CNN, YOLO, …)



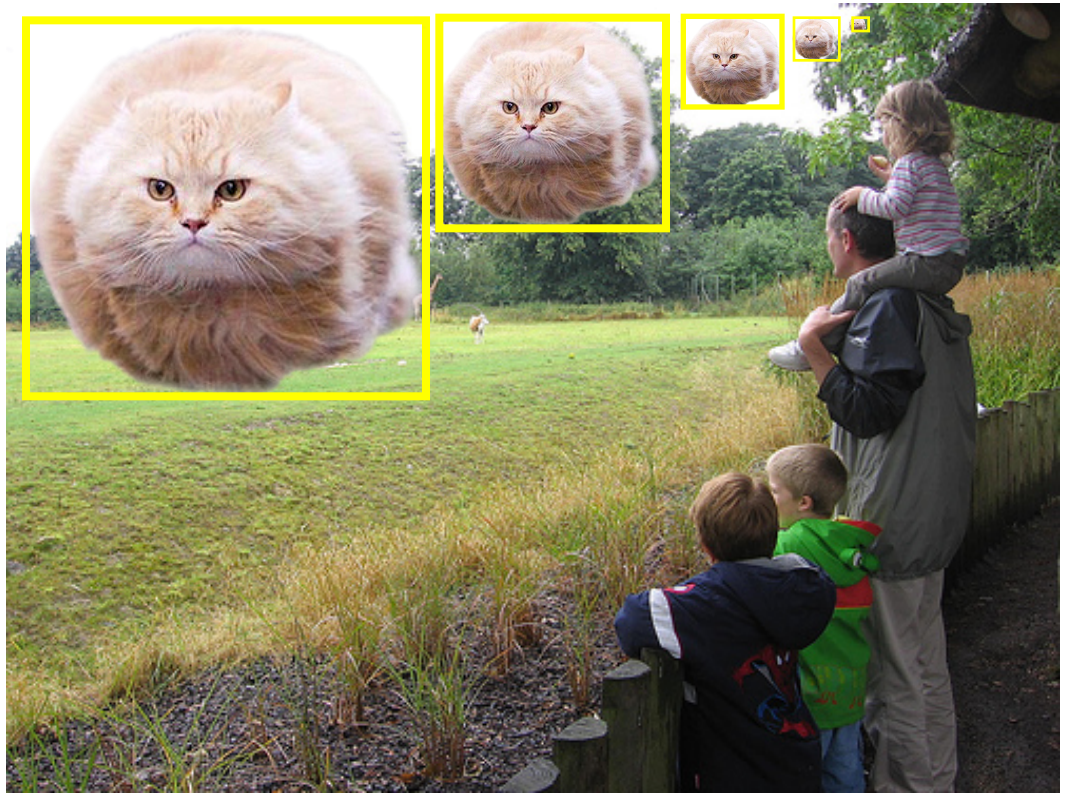
(c) Pyramidal feature hierarchy

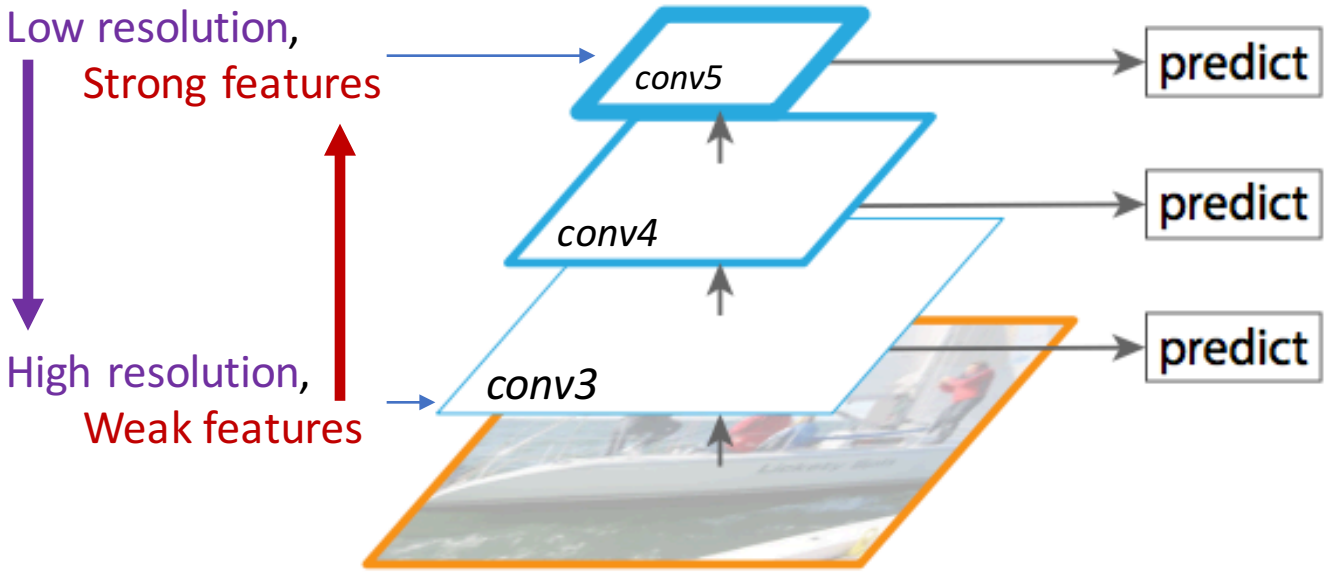
Fast!

(≈ SSD, …)

Let's examine this one

Compromise Feature Quality, but Fast (“Free”)

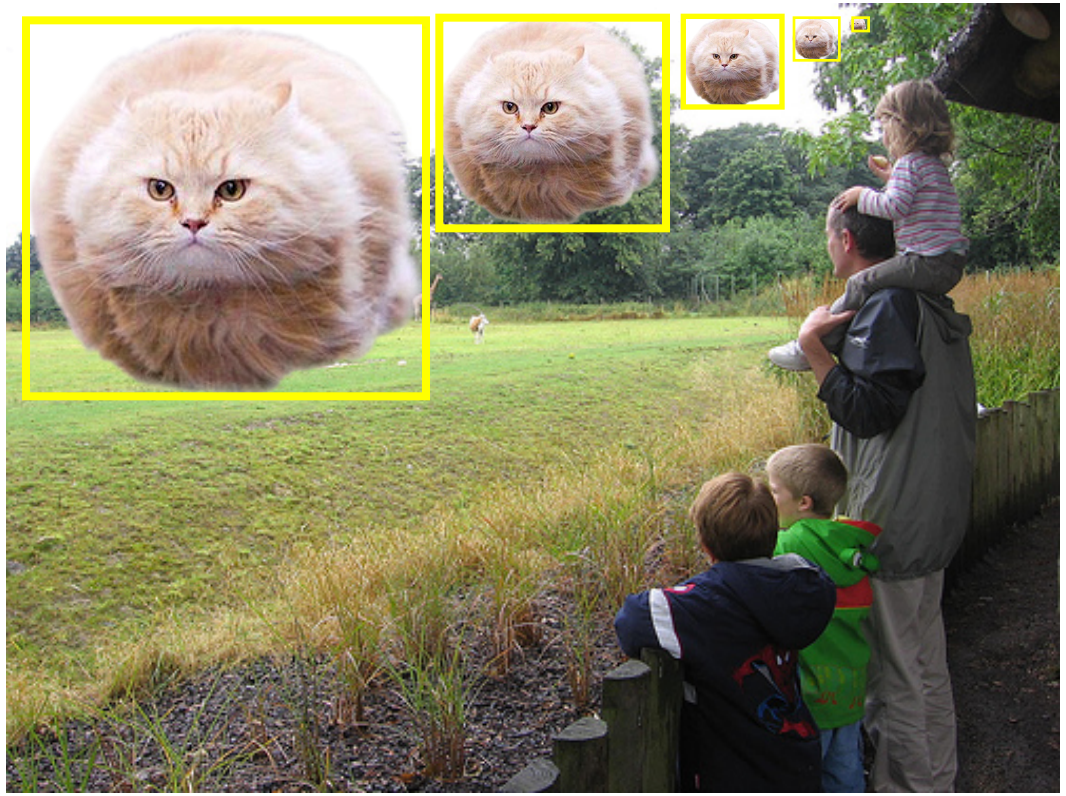


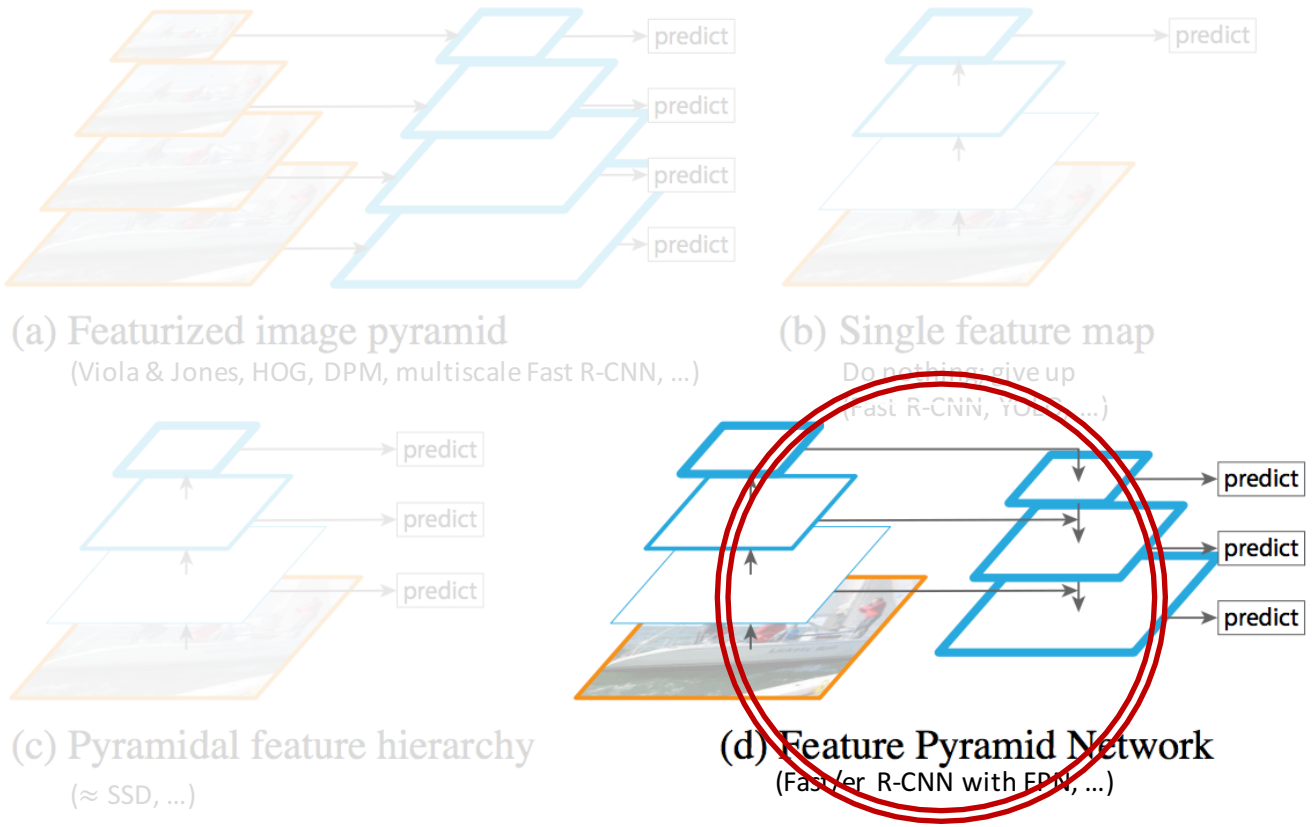


The “native” in-network feature pyramid poses an inherent tradeoff

2. … + Feature Pyramid Network (FPN)

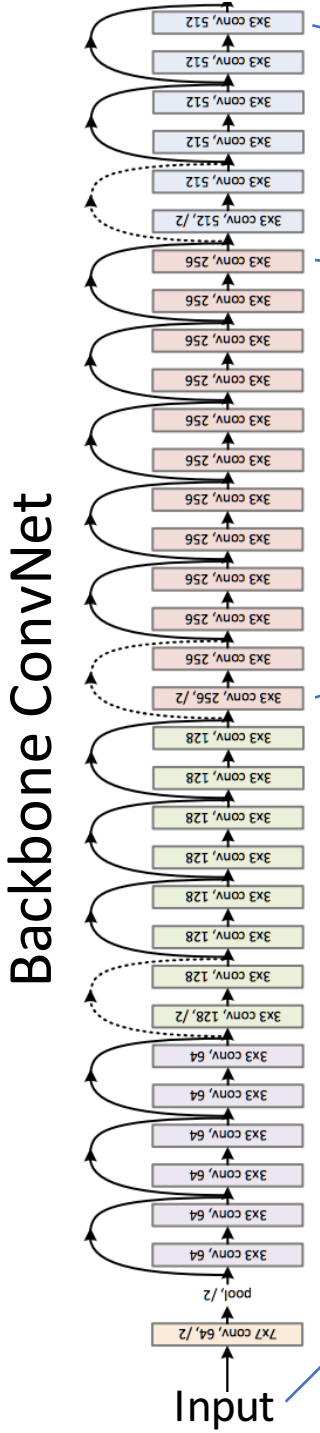
[Optional, but recommended]





Lin et al. Feature Pyramid Networks for Object Detection. In CVPR 2017. Poster 99 Saturday morning. See also: Shrivastava's TDM

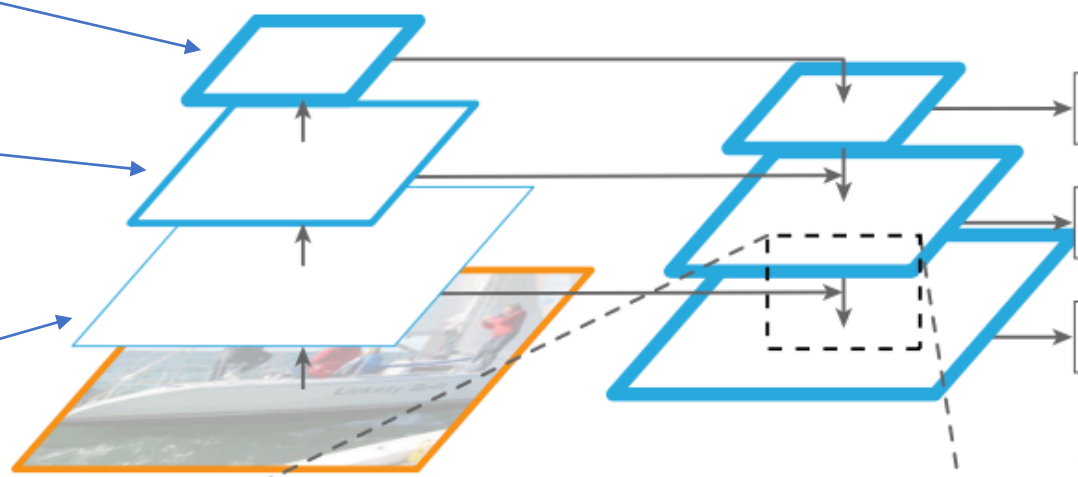
FPN



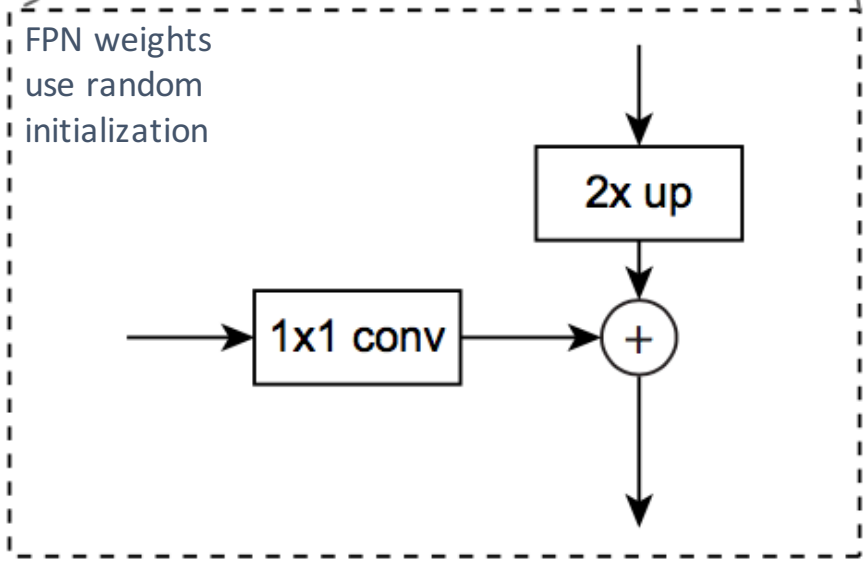
1/16 image resolution
Res4 stage output

1/8 image resolution
Res3 stage output

1/4 image resolution
Res2 stage output

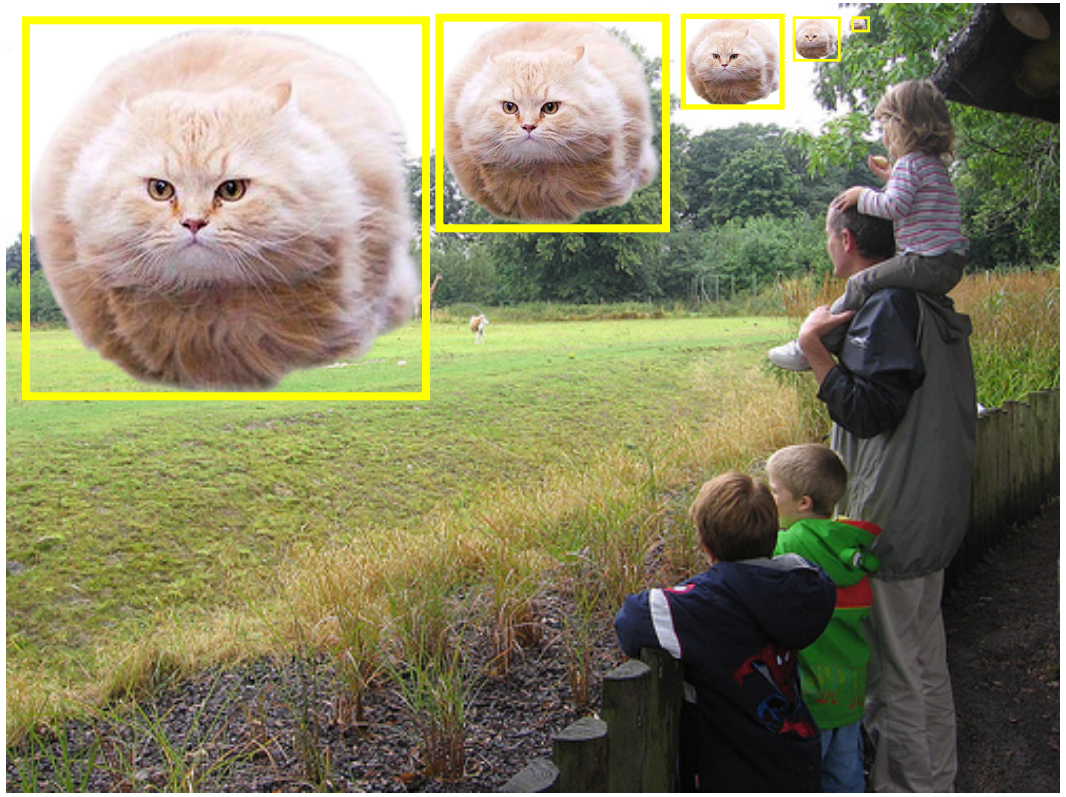


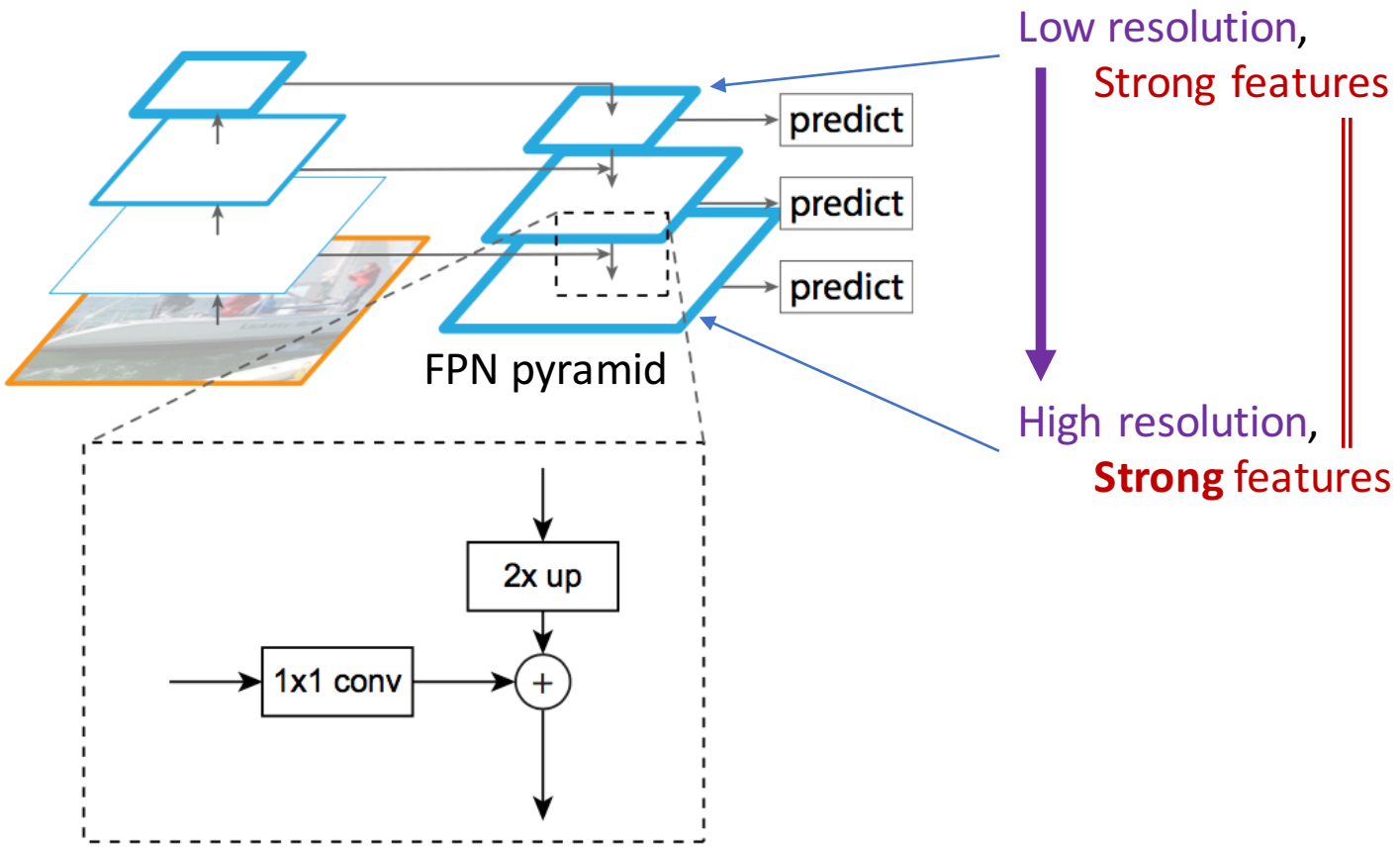
(Illustrated with
3 levels for simplicity;
typically use 5 or 6)



Lin et al. Feature Pyramid Networks for Object Detection. In CVPR 2017. Poster 99 Saturday morning.

No Compromise on Feature Quality, still Fast

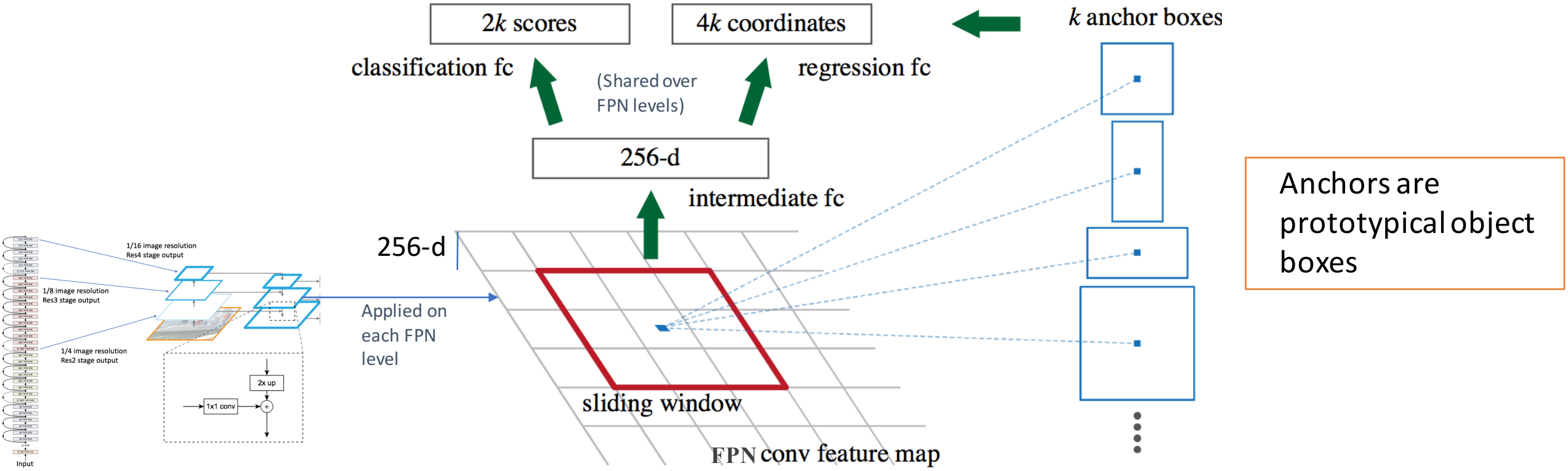




Lin et al. Feature Pyramid Networks for Object Detection. In CVPR 2017. Poster 99 Saturday morning. See also: Shrivastava's TDM

3. … + Region Proposal Network (RPN)

Proposals = sliding window object/not-object classifier + box regression
inside the same network

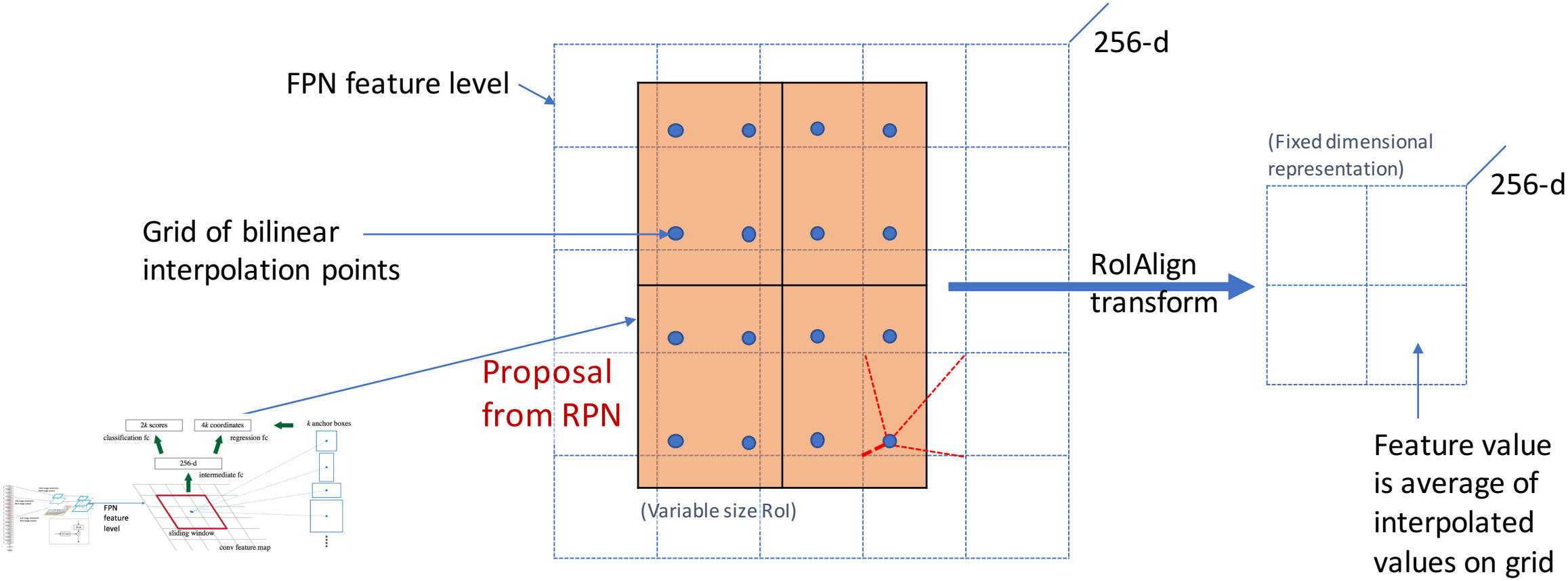


Anchors are prototypical object boxes

Ren et al. Faster R-CNN: Region Proposal Networks for Real-Time Object Detection. In NIPS 2015

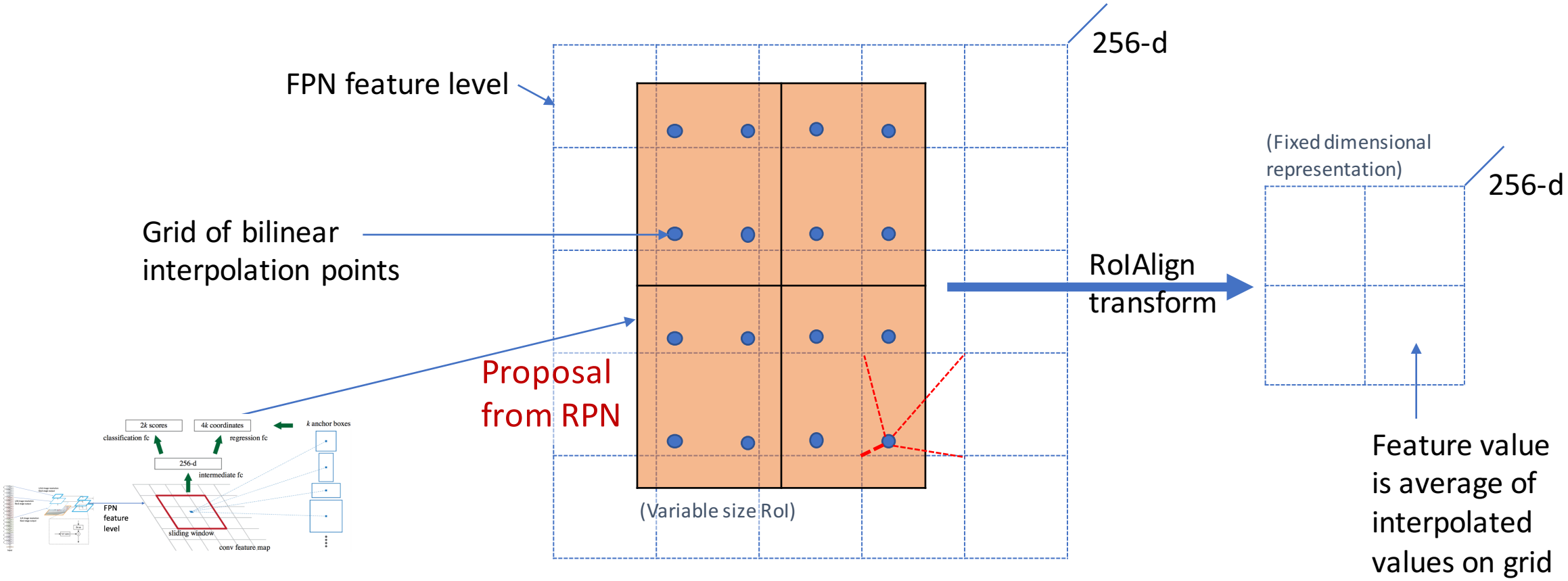
4. … + RoIAlign Transform (on each Proposal)

Smoothly normalize features and predictions into coordinate frame
free of scale and aspect ratio



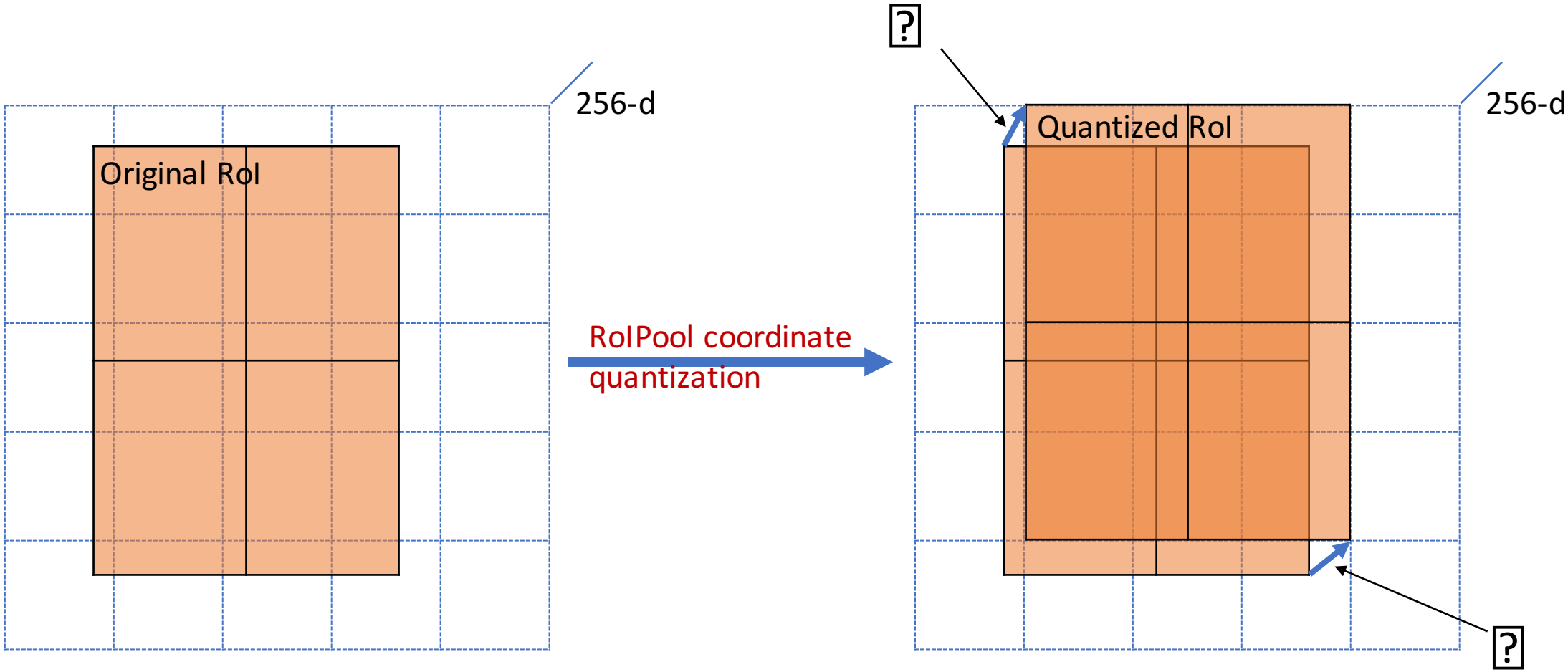
4. … + RoIAlign Transform (on each Proposal)

Key: No coordinate quantization (cf. RoIPool in Fast R-CNN, etc.)



Compare to RoIPool, RoIWarp, and others

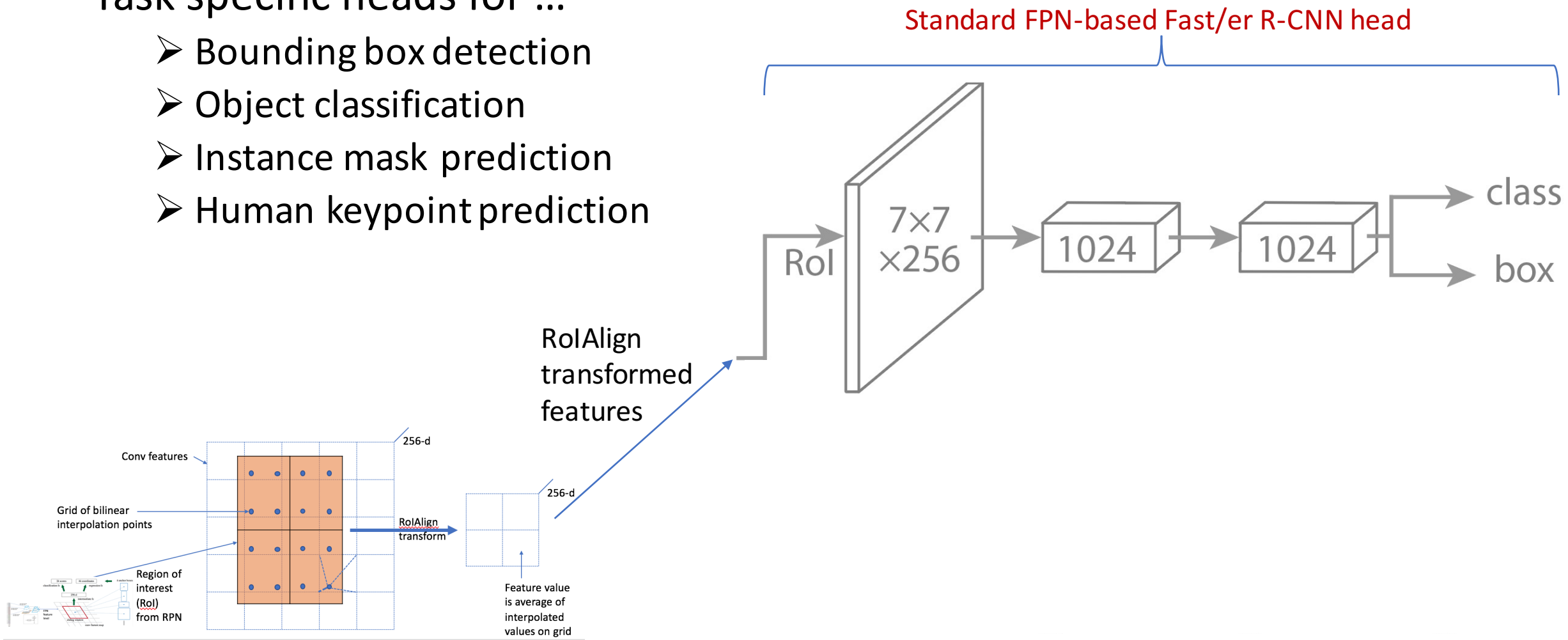
Quantization breaks pixel-to-pixel alignment



5. … + Task-specific Heads (on each Proposal)

Task specific heads for …

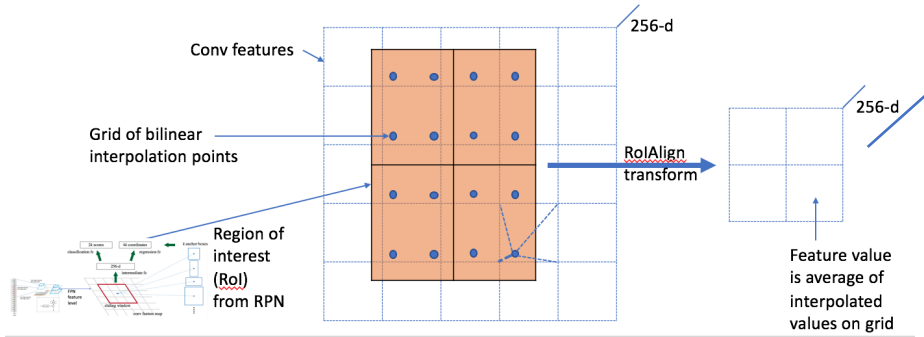
- Bounding box detection
- Object classification
- Instance mask prediction
- Human keypoint prediction



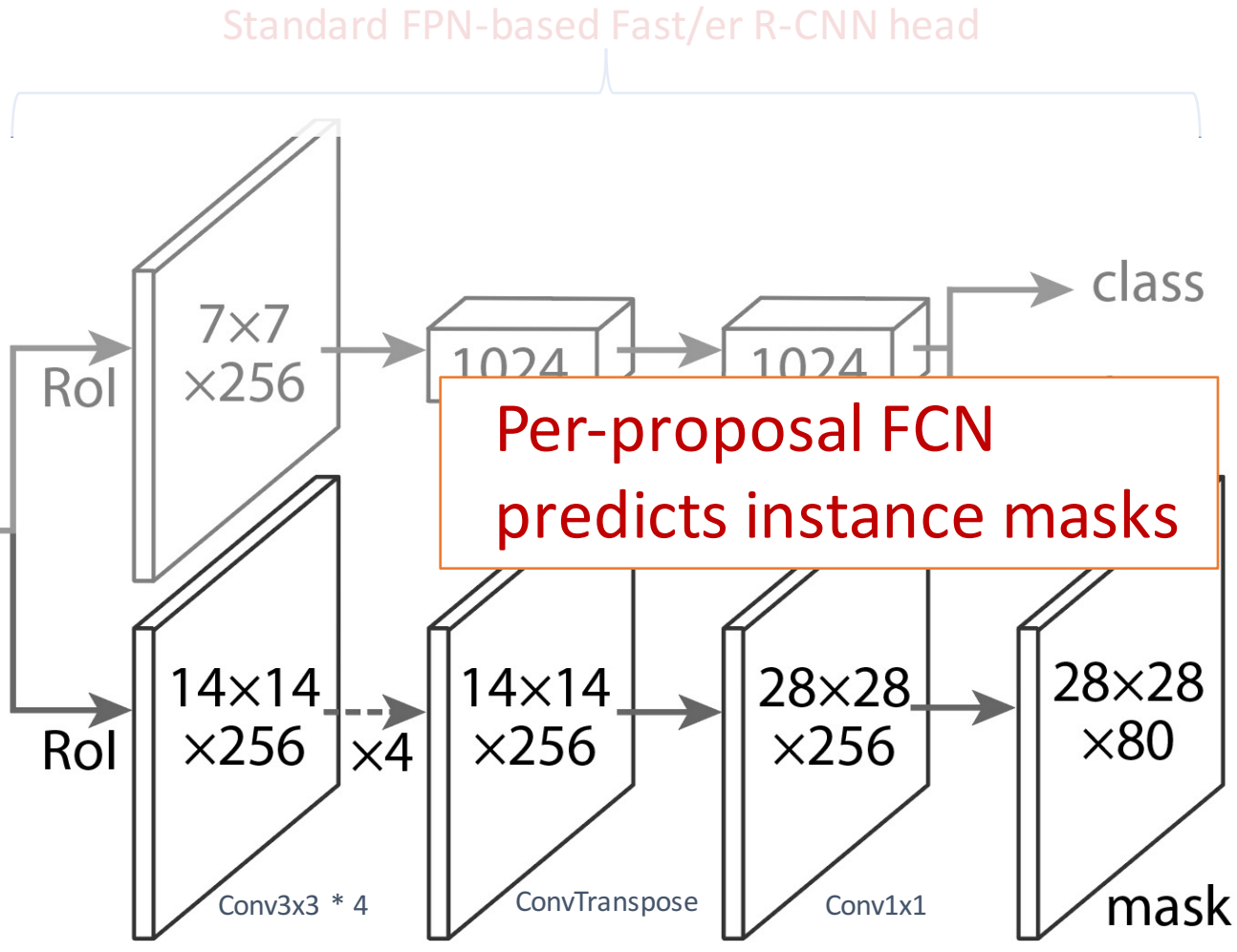
5. … + Task-specific Heads (on each Proposal)

Task specific heads for …

- Object classification
- Bounding box regression
- Instance mask prediction
- Human keypoint prediction



RoIAlign transformed features



Mask R-CNN: Training

Not enough time for details (sorry!)

Same as “image centric” Fast/er R-CNN training

- Use precomputed proposals for faster experimentation
- Use joint / end-to-end training for sharing features

But with training targets for masks

Example Mask Training Targets

Image with training proposal

28x28 mask target

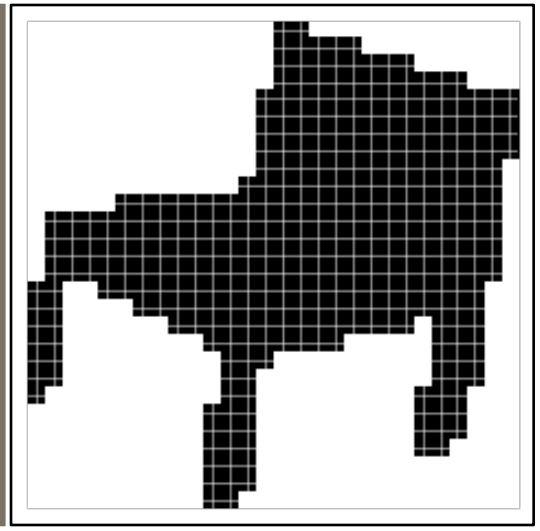
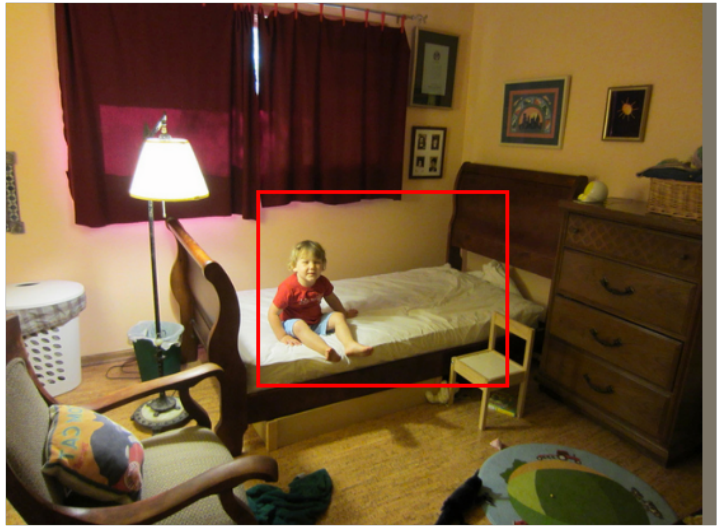
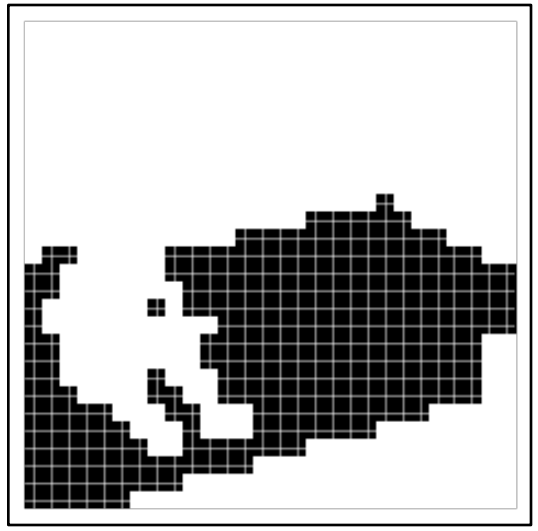
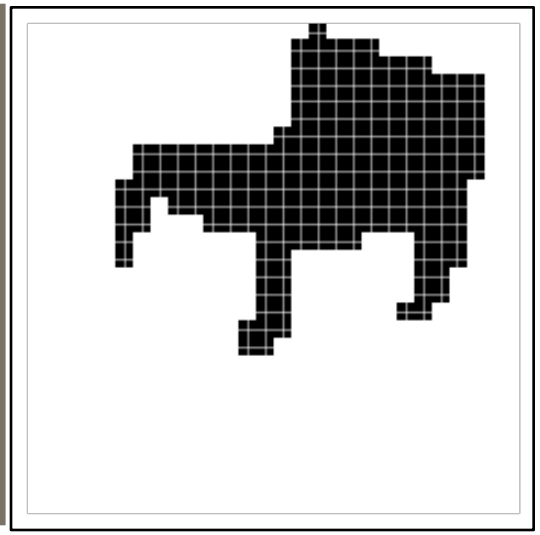


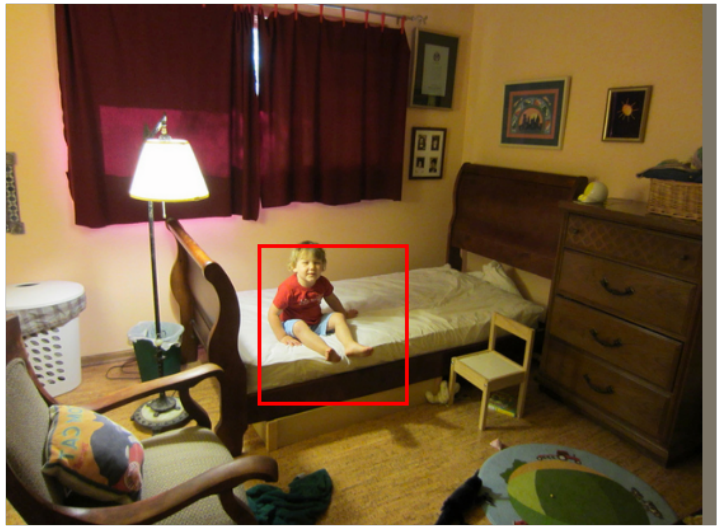
Image with training proposal

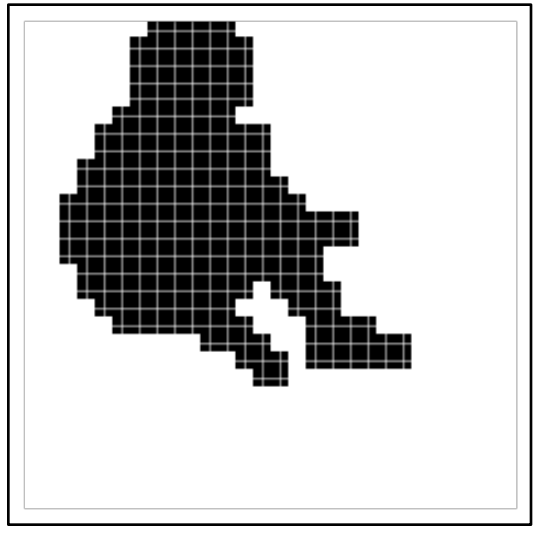


28x28 mask target









Mask R-CNN: Inference

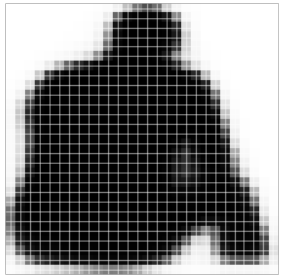
1. Perform Faster R-CNN inference
   - Generate proposals (RPN)
   - Score the proposals
   - Regress from proposals to refined detection boxes
   - Apply NMS and take the top *K* (= 100, e.g.)

2. Run RoIAlign and mask head on top-*K* refined, post-NMS boxes
   - Fast (only compute masks for top-*K* detections)
   - Improves accuracy (uses *refined* detection boxes, not proposals)

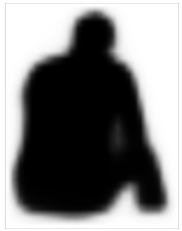
Mask Prediction

Validation image with box detection shown in red

28x28 soft prediction from Mask R-CNN
(enlarged)



Soft prediction resampled to image coordinates
(bilinear and bicubic interpolation work equally well)

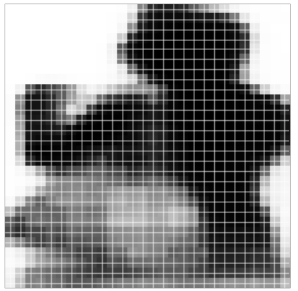


Final prediction (threshold at 0.5)

Mask Prediction

Validation image with box detection shown in red

28x28 soft prediction



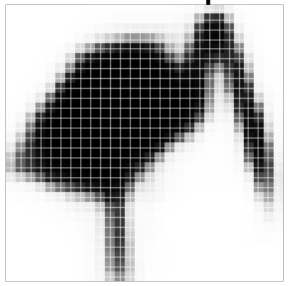
Resized soft prediction

Final mask

Mask Prediction

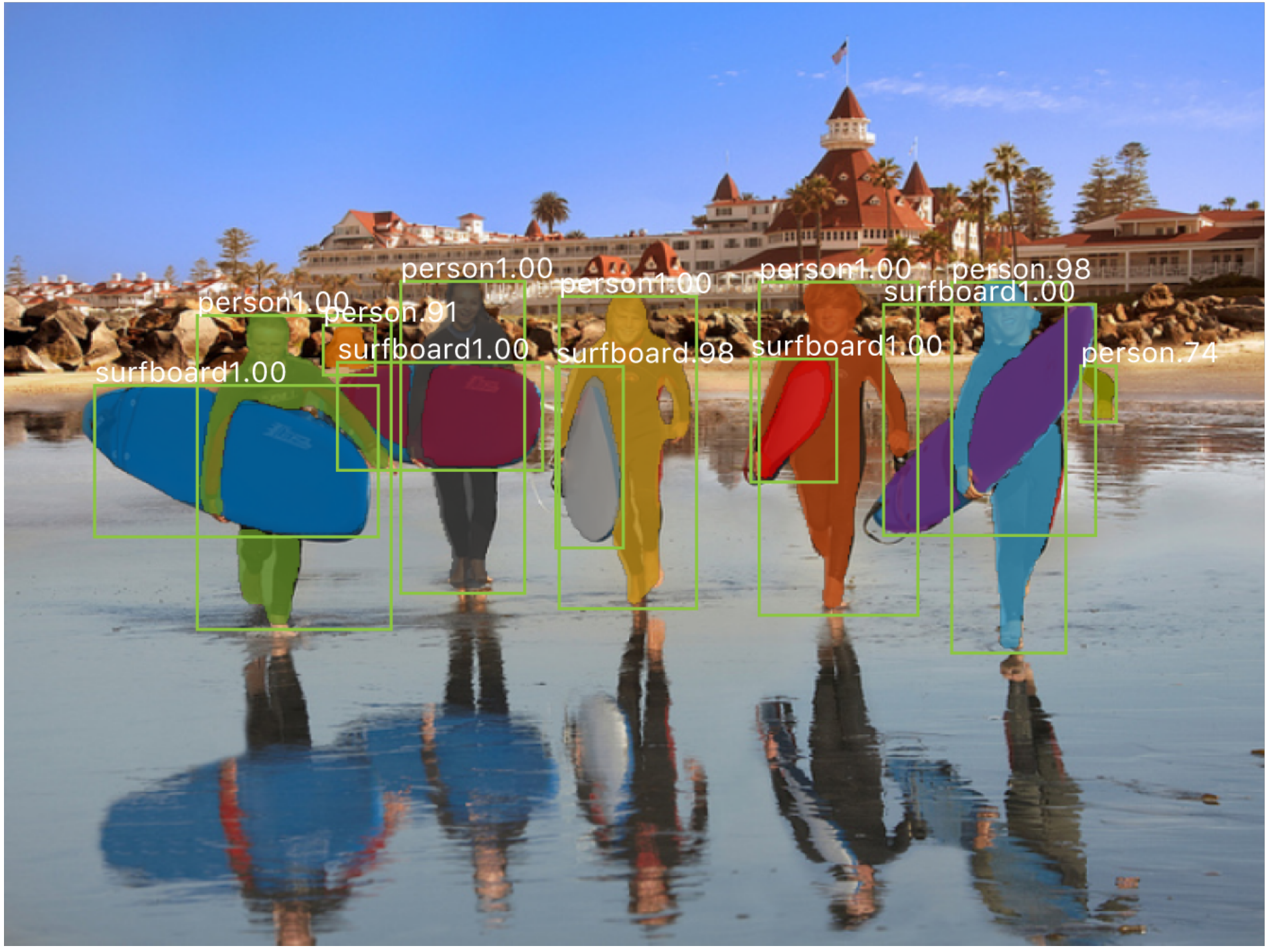
Validation image with box detection shown in red

28x28 soft prediction



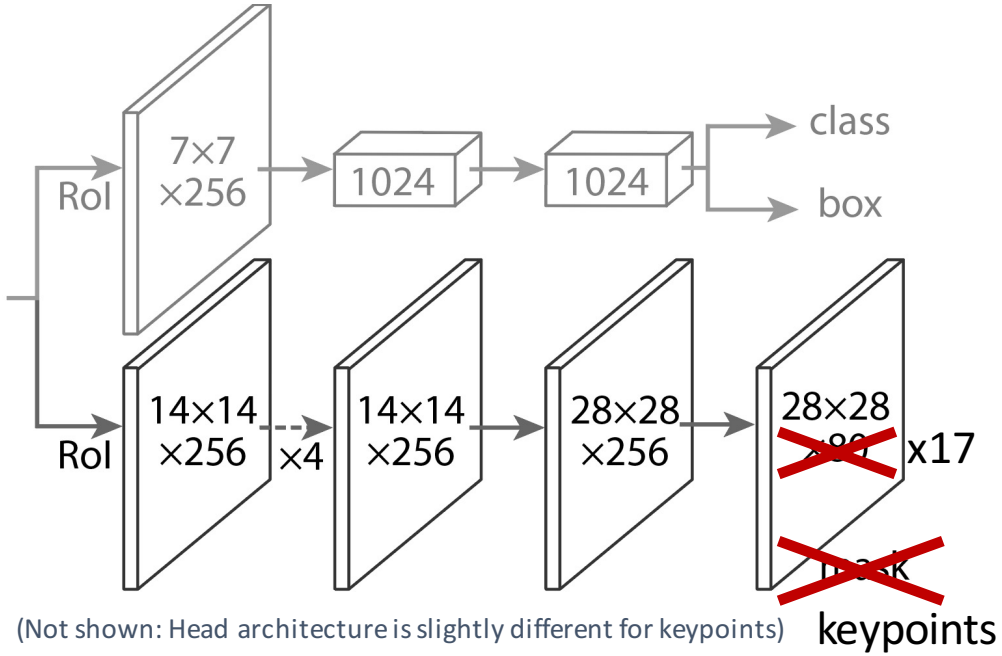
Resized Soft prediction

Final mask

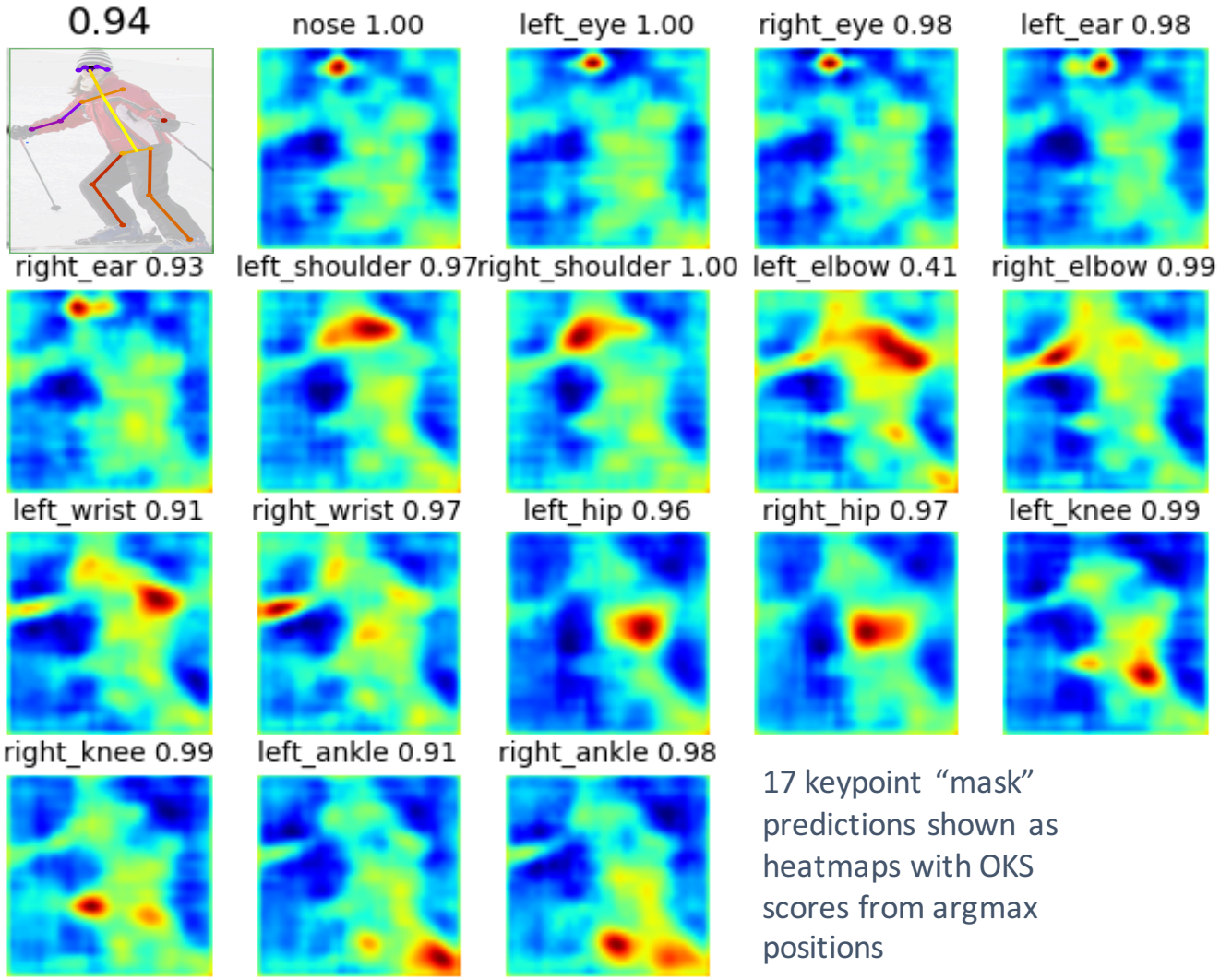

person1.00
person1.00
person1.00
person.98
surfboard1.00
person1.00
person.91
surfboard1.00
surfboard.98
surfboard1.00
person.74
surfboard1.00


person.88
person1.00
tv.98
tv.84
person1.00
person1.00
bottle.97
wine glass.99
dining table.95
wine glass1.00
wine glass1.00

Human Pose



(Not shown: Head architecture is slightly different for keypoints)



17 keypoint "mask" predictions shown as heatmaps with OKS scores from argmax positions

- Add keypoint head (28x28x17)
- Predict one "mask" for each keypoint
- Softmax over spatial locations (encodes one keypoint per mask "prior")

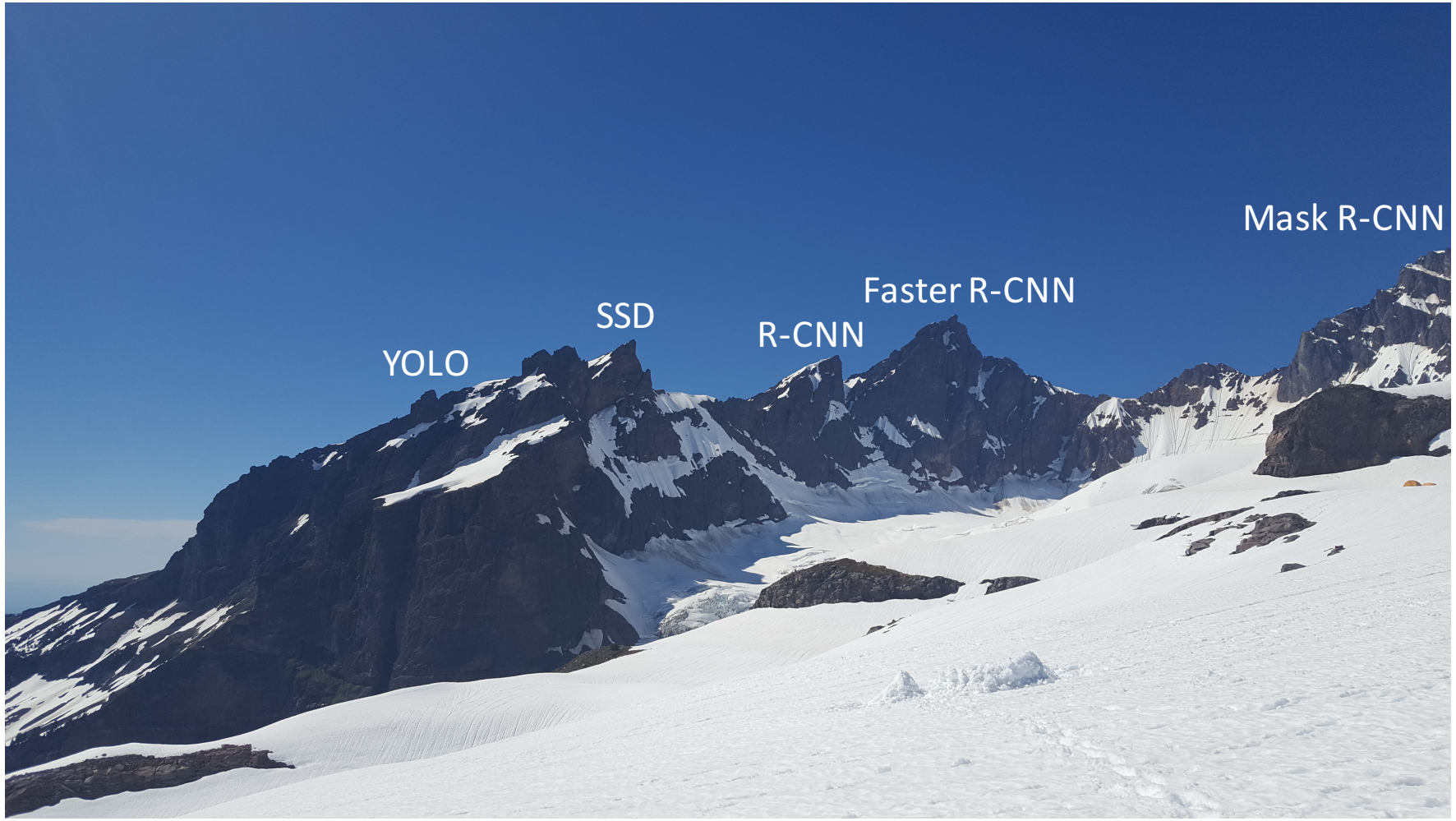

www.tfrancis.co.uk

Deep Learning for Object Detection:
A Bewildering Word Salad

R-CNN OverFeat DetectorNet DeepMultibox SPP-net Fast R-CNN MR-CNN SSD YOLO YOLOv2 G-CNN AttractioNet Mask R-CNN R-FCN RPN FPN Faster R-CNN ...

and many more words

Let’s Organize the Landscape



A random landscape scene on Mt. Baker, just because I like mountains.

Photo credit: Ross Girshick

Common to all Methods

Start by modifying a classification network

Since R-CNN, this network is pre-trained, typically using ImageNet (cf. DetectorNet)

R-CNN: Girshick et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. CVPR 2014
DetectorNet: Szegedy et al. Deep Neural Networks for Object Detection. NIPS 2013

Highest Information Gain Split: “Stage” Count

More than one stage

- DetectorNet (Szegedy et al.)
- R-CNN (Girshick et al.)
- SPP-net (He et al.)
- Fast R-CNN (Girshick)
- Faster R-CNN (Ren et al.)
- R-FCN (Dai et al.)
- Mask R-CNN (He et al.)

One stage

- OverFeat (Sermanet et al.)
- YOLO, YOLOv2 (Redmon et al.)
- SSD (Wei et al.)
- RetinaNet (Lin et al.) [Poster at WICV on Wed.]

Stages

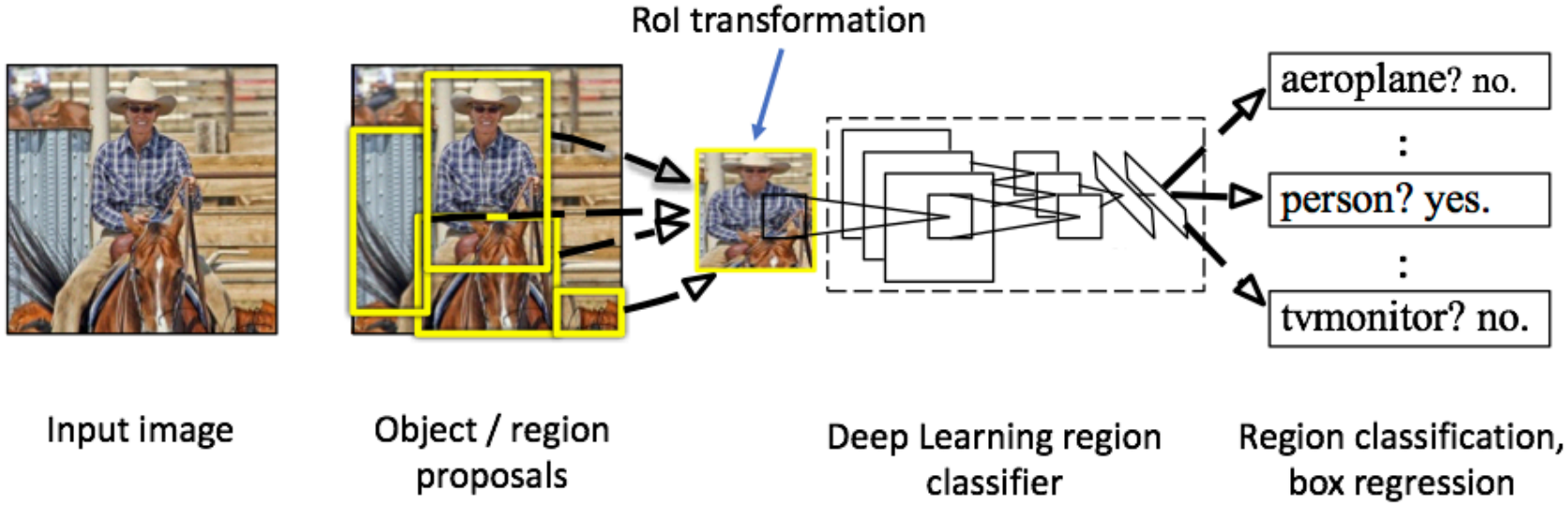
Detection Output space: $N = H \times W$ pixel image has $O(N^2)$ windows

Output space is HUGE, even for small images

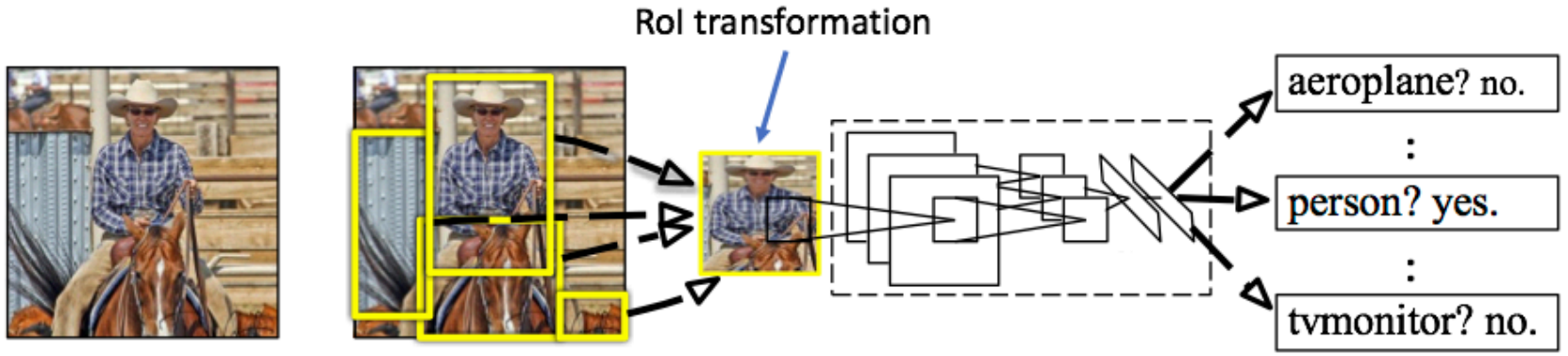
Classic approaches to dealing with this issue …

- Sliding window with subsampled aspect ratios, translation, and scale (reduces window count to $\approx$ 100,000)
- Multiple stages of *cascaded classification* (helps deal with extreme foreground-vs-background class imbalance)

More than one “stage” (≈ proposal based; but doesn’t require proposals)



More than one “stage” (≈ proposal based; but doesn’t require proposals)



Cascade-like reduction in output space →

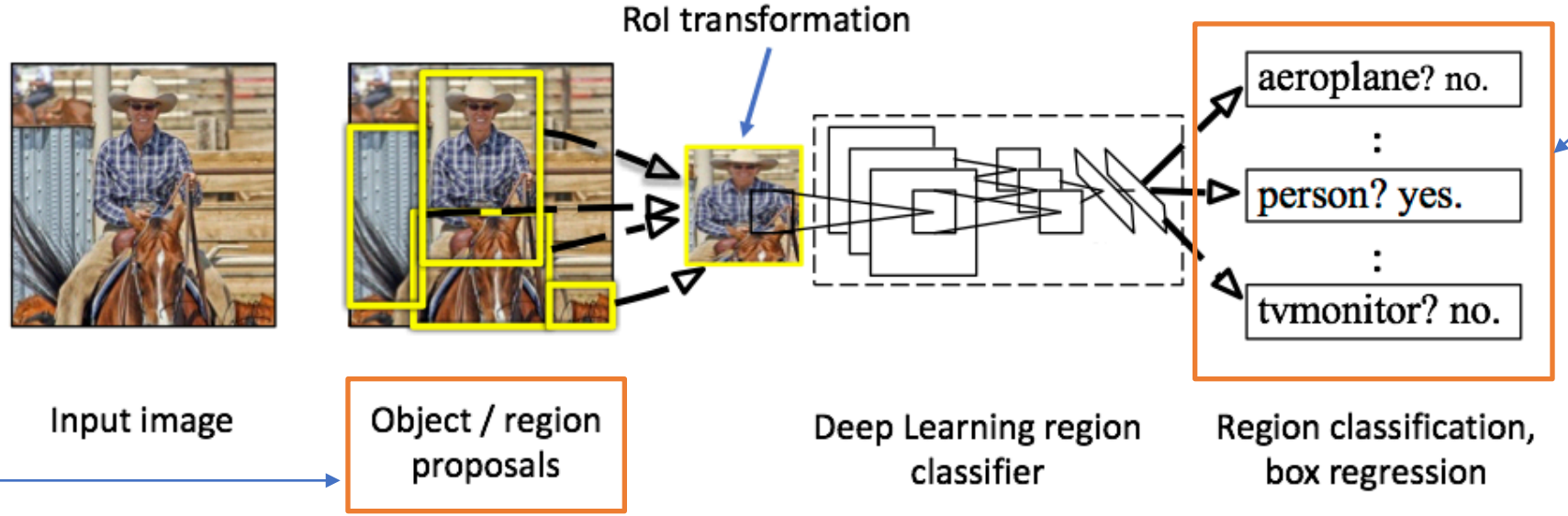
Input image

Object / region proposals

Deep Learning region classifier

Region classification, box regression

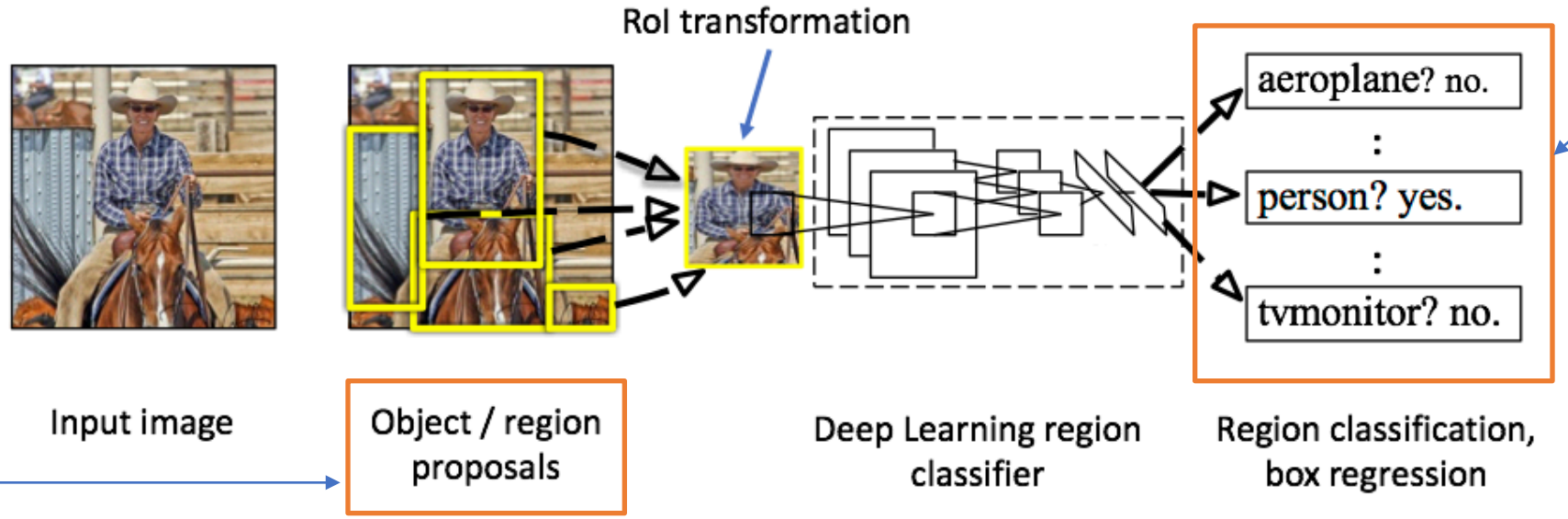
More than one “stage” (≈ proposal based; but doesn’t require proposals)



Classification of *reduced* output space elements

Cascade-like reduction in output space

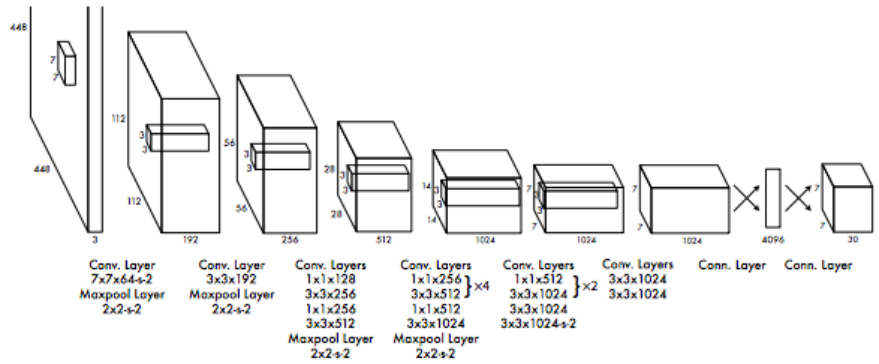
More than one "stage" (≈ proposal based; but doesn't require proposals)



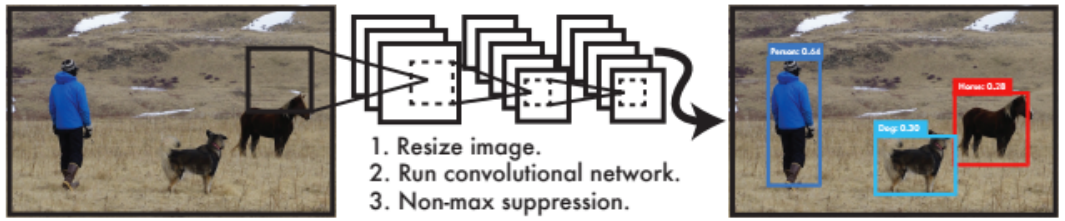
Classification of *reduced* output space elements

Cascade-like reduction in output space

One stage

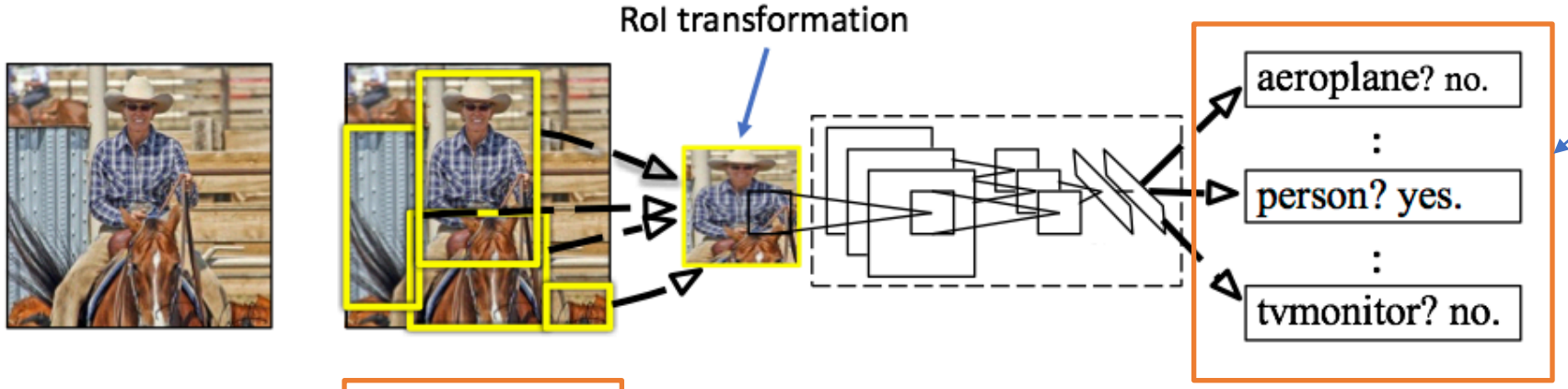


Redmond et al. You Only Look Once: Unified Real-time Object Detection. In CVPR 2016



More than one "stage" (≈ proposal based; but doesn't require proposals)

Classification of *reduced* output space elements



Cascade-like reduction in output space

Input image

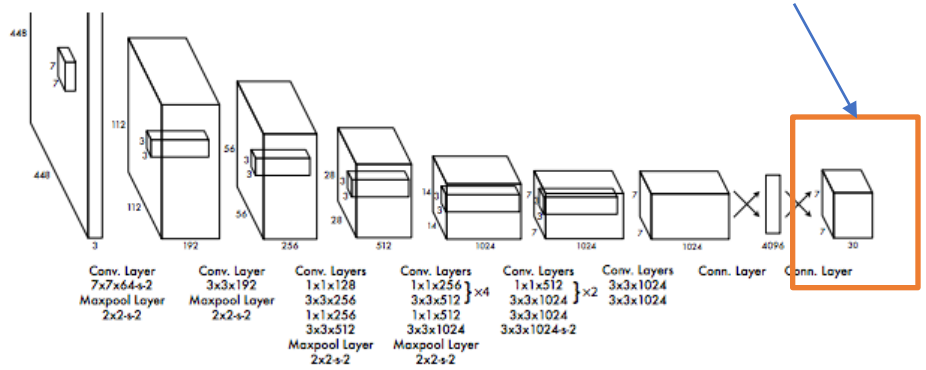
Object / region proposals

Deep Learning region classifier

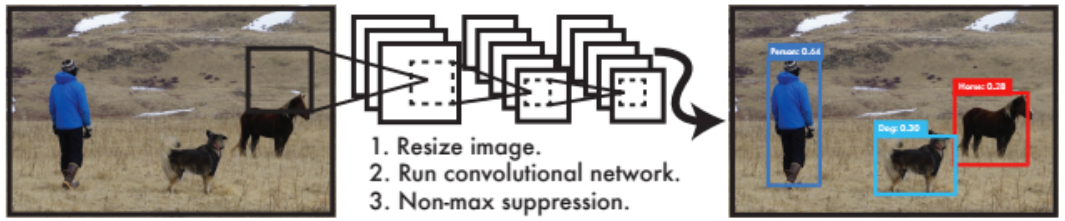
Region classification, box regression

One stage

Direct classification of *all* output space elements

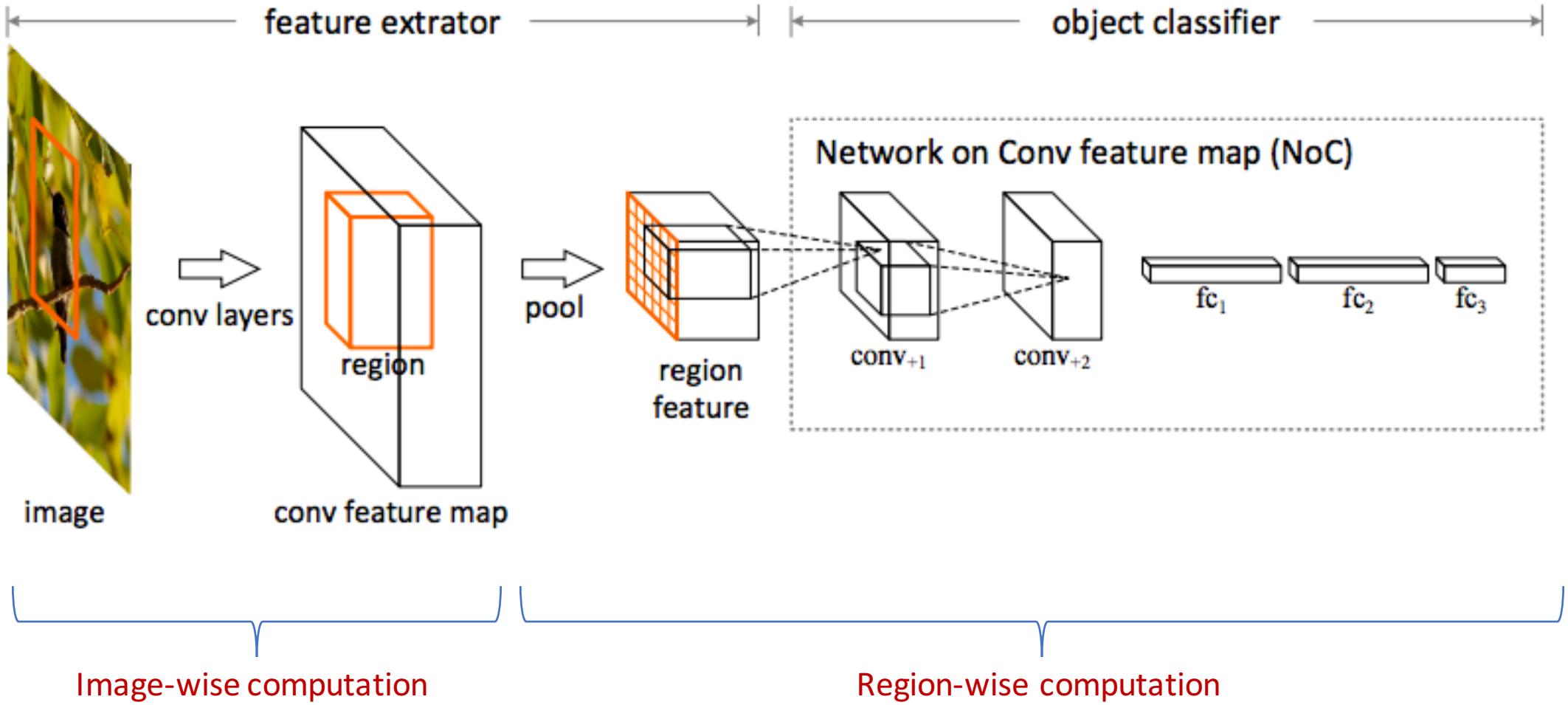


Redmond et al. You Only Look Once: Unified Real-time Object Detection. In CVPR 2016



"You only look once"
"Single shot"

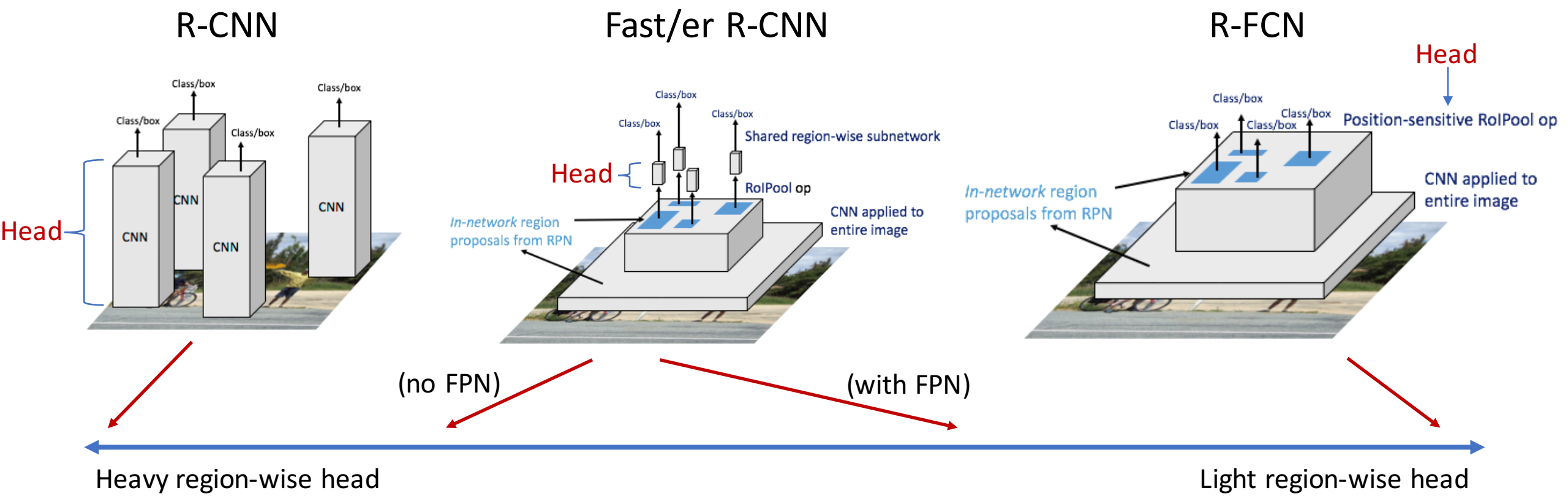
Thinking about Multi-stage Detectors



Ren et al. Object Detection Networks on Convolutional Feature Maps. PAMI 2017.

Networks on Convolutional Feature Maps

Shifting computation between the “head” and the “trunk”



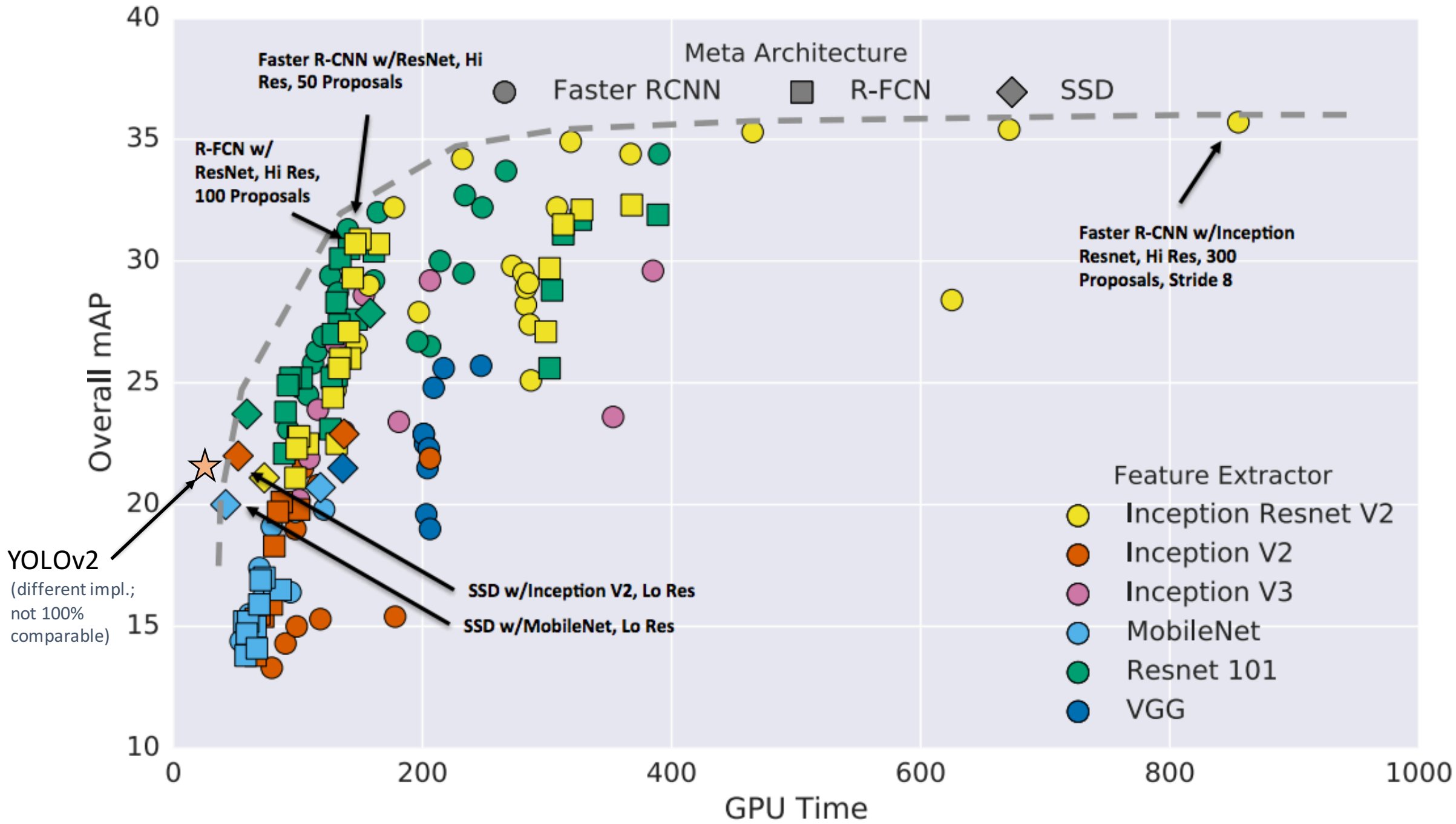
Speed / Accuracy Tradeoffs

Speed is mainly a function of:

- Input image resolution
- Network complexity
- Number of proposals (if applicable)

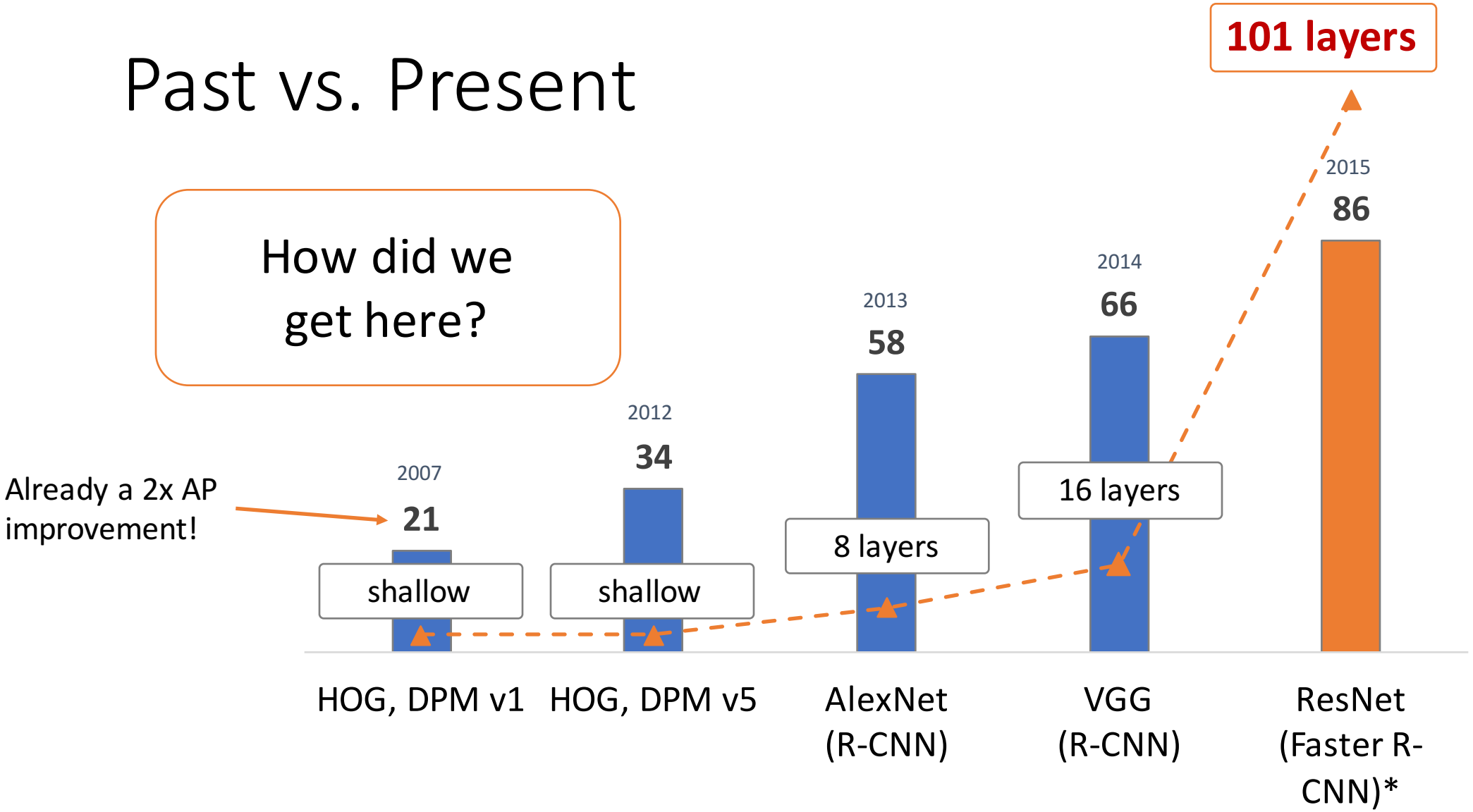
Optimizing AP on PASCAL VOC has lead to confusion about speed / accuracy tradeoffs

COCO provides a clear picture of tradeoffs



Huang et al. Speed/Accuracy Tradeoffs for Modern Convolutional Object Detectors. CVPR 2017

Past vs. Present



*w/ other improvements & more data

Slide credit: Kaiming He

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

False in 2012 and earlier, but True since 2013

State-of-the-art detectors

- Improve with more data
- Improve with increased model capacity
- Improve from transfer learning
- Immediately benefit from image classification research
- Share a common modeling framework with speech and NLP

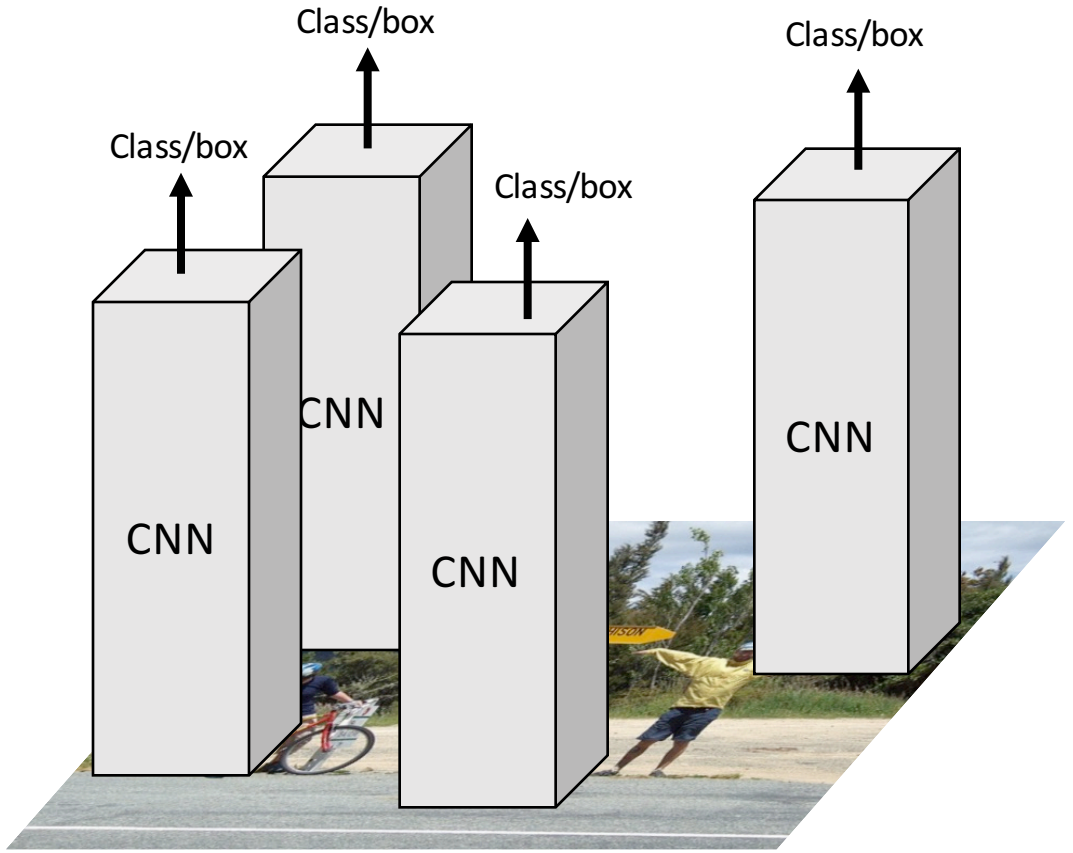
Conclusions. Thank you! Questions?

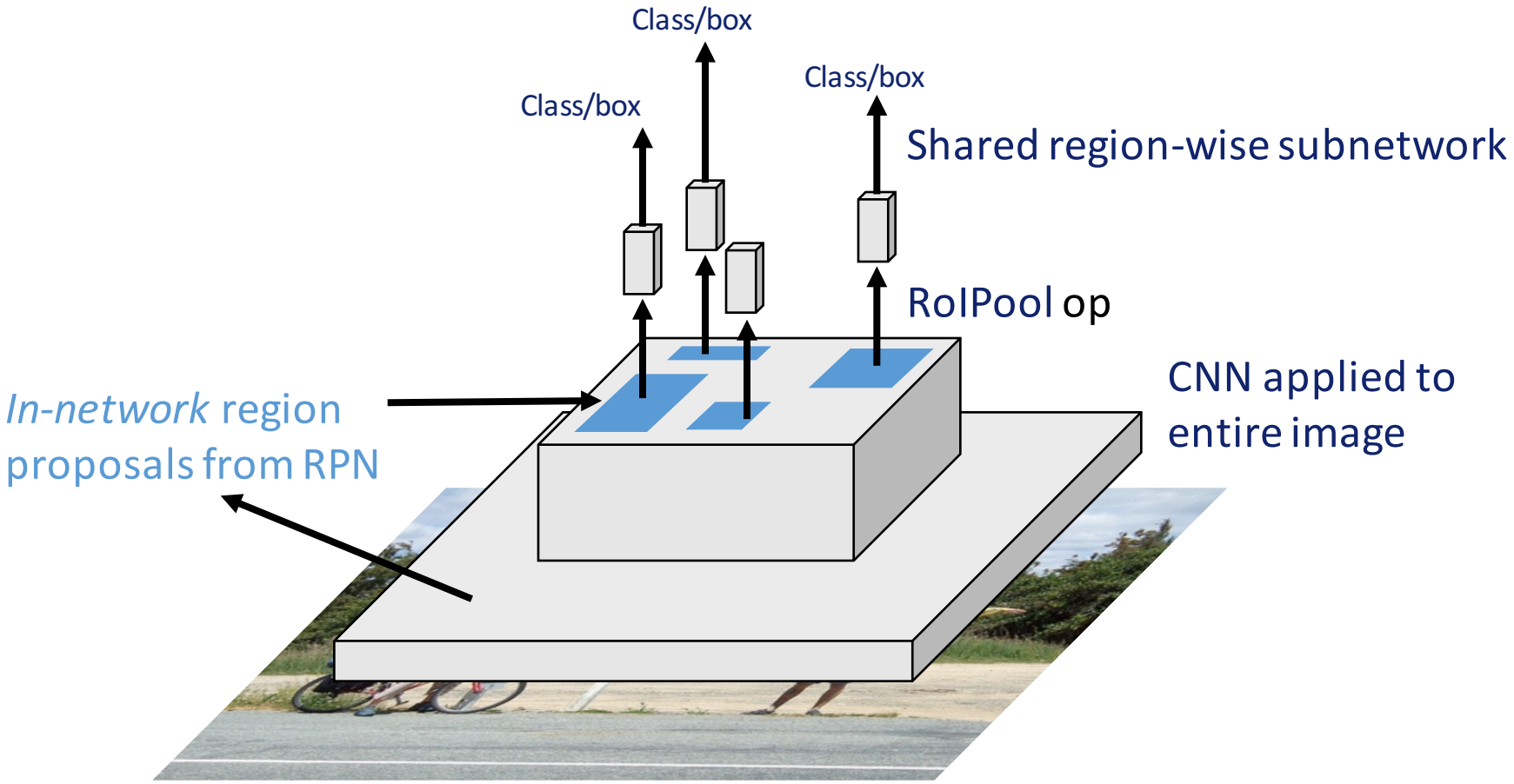
Object detection has come a very long way in a short time!

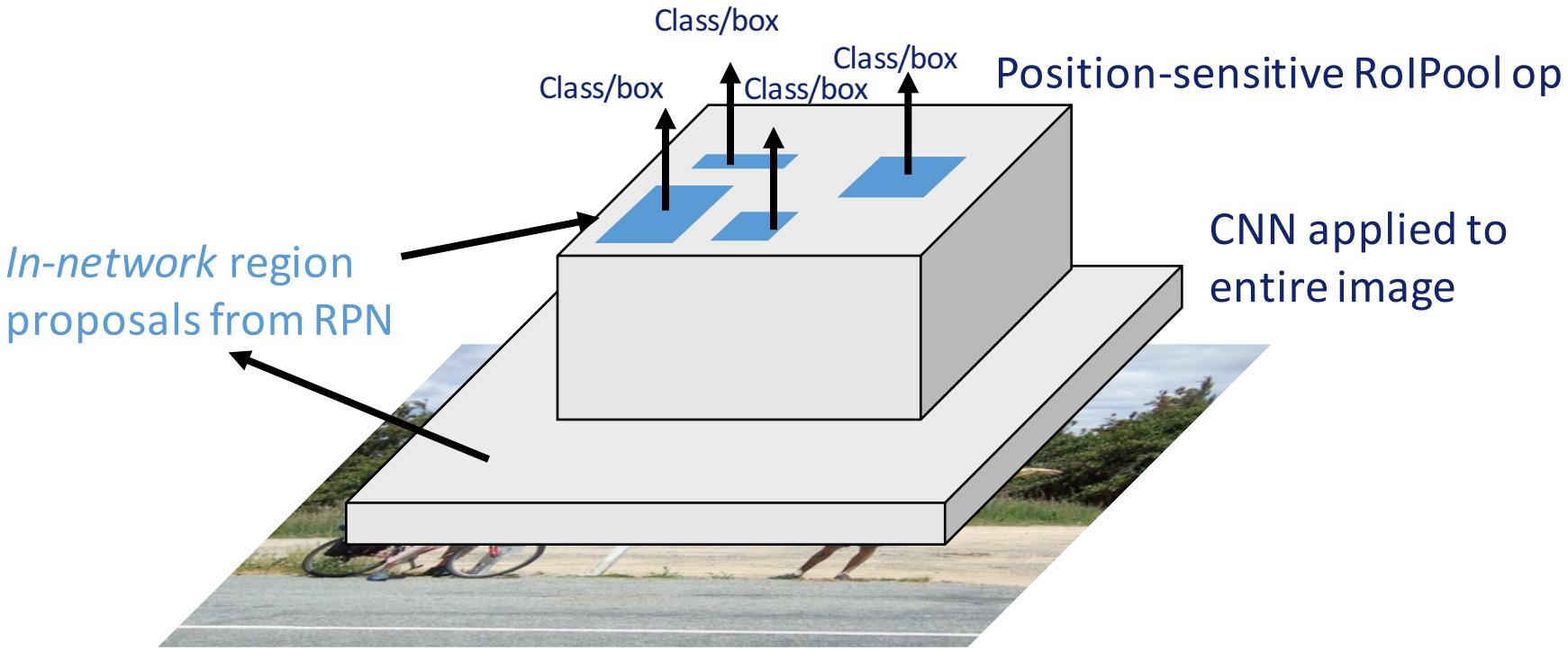
We're moving from box detection to instance-level understanding

Major challenges remain:

- Long-tailed examples-per-category distribution (implies low-shot learning)
- Open vocabulary recognition (implies benchmarking challenges)
- Low accuracy with very heavy occlusion / clutter (esp. for keypoints; "reasoning")
- Bottleneck may still be the raw engine of recognition (improve backbone, data)


Class/box
Class/box
Class/box
Class/box
CNN
CNN
CNN
CNN


Class/box
Class/box
Class/box
Shared region-wise subnetwork
RoIPool op
CNN applied to entire image
In-network region proposals from RPN


Class/box
Class/box
Class/box
Class/box
Position-sensitive RoIPool op
CNN applied to entire image
In-network region proposals from RPN