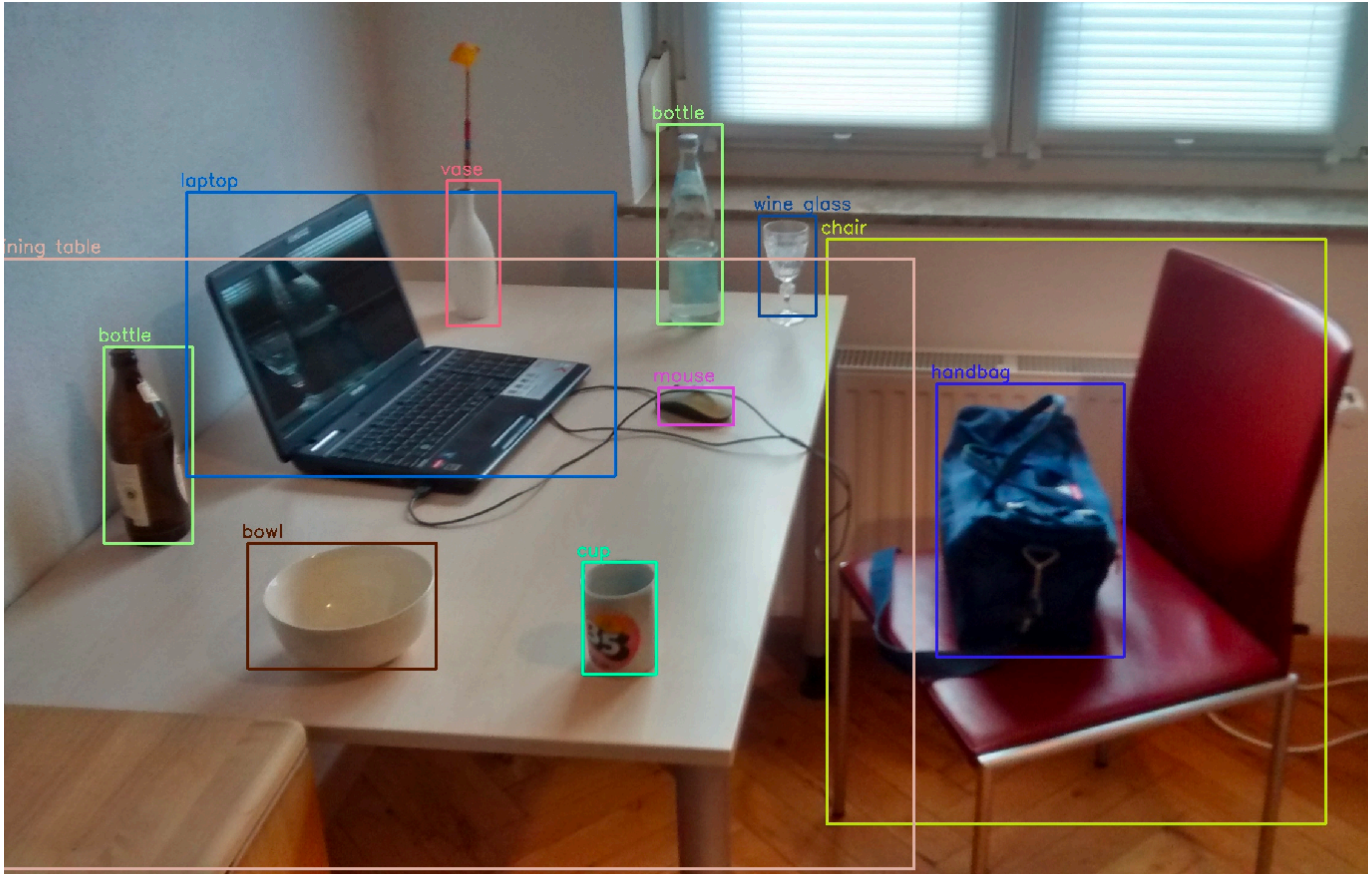
Open Vocabulary Object Detection

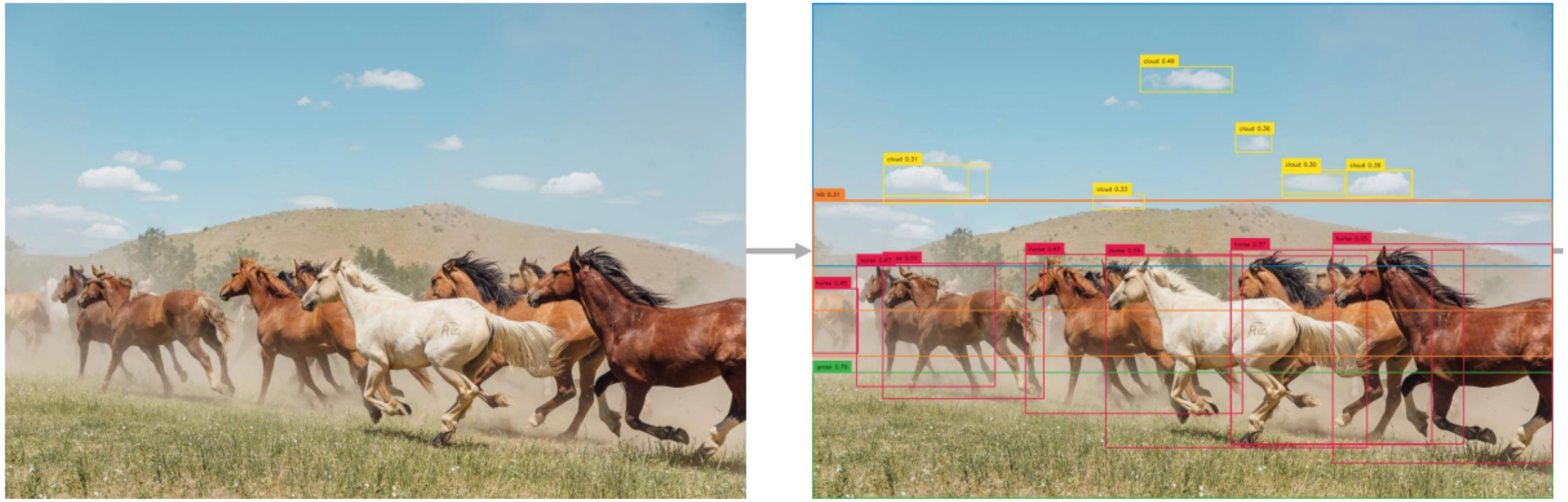
Sanjiban Choudhury

Cornell Bowers CIS
Computer Science

What is an object? Why should robots detect them?



Rise of Open-Vocabulary Object Detectors

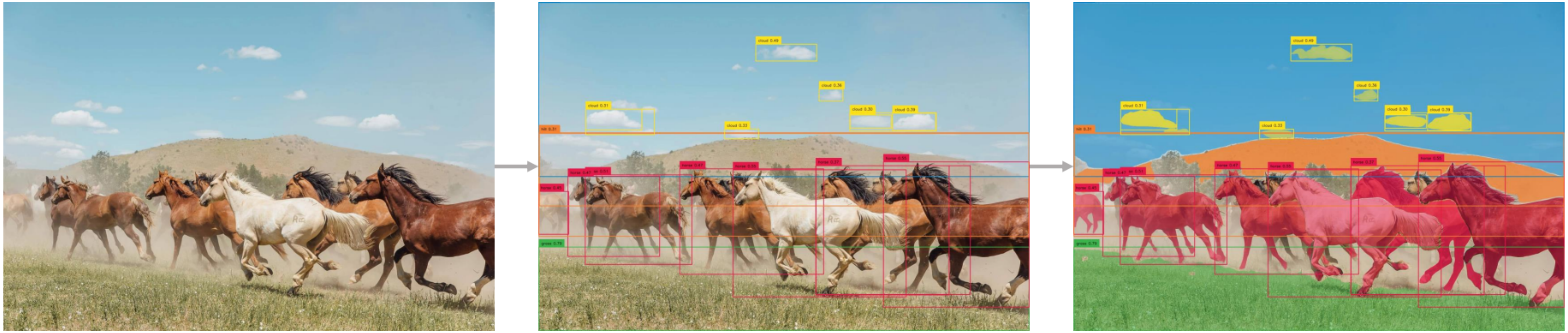


Text Prompt:
"Horse. Clouds. Grasses. Sky. Hill."

Grounding DINO:
Detect Everything

Pre-trained models like **OWL-ViT** and **Grounding DINO** can take any image and text queries, and output bounding boxes with scores

Rise of Open-Vocabulary Object Detectors



Text Prompt:
"Horse. Clouds. Grasses. Sky. Hill."

Grounding DINO:
Detect Everything

Grounded-SAM:
Detect and Segment Everything

Pre-trained models like **Segment Anything (SAM)** can segment individual pixels to precisely identify where the object is

Let's try it out!

https://huggingface.co/spaces/johko/OWL-ViT

https://huggingface.co/spaces/merve/Grounding_DINO_demo

Robots now use these models to detect and manipulate objects without requiring any further training!

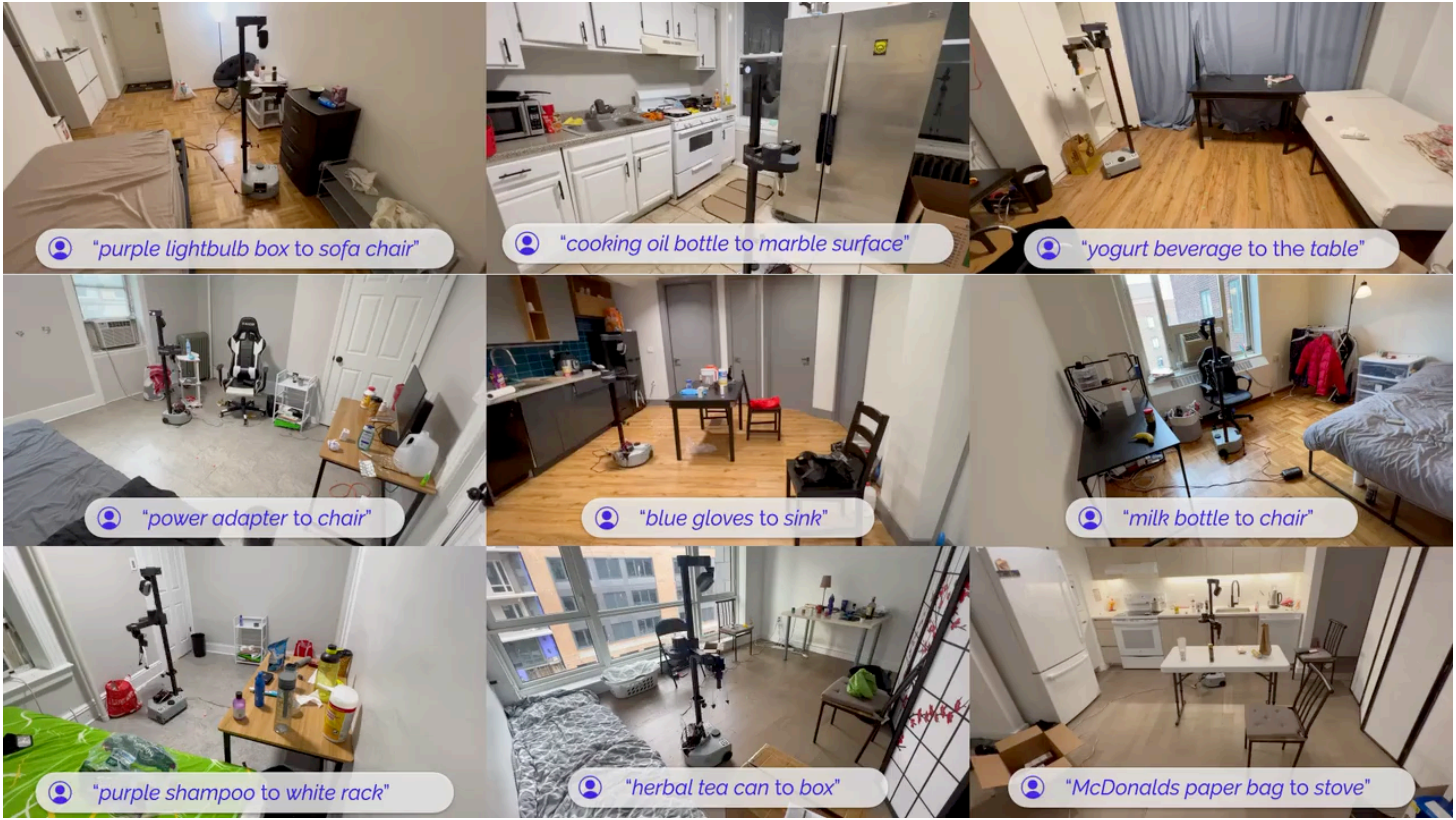
MOSAIC

A Modular System for Assistive and Interactive Cooking

https://portal-cornell.github.io/MOSAIC/

OK-Robot

An open, modular framework for zero-shot, language conditioned pick-and-drop tasks in arbitrary homes.


"purple lightbulb box to sofa chair"
"cooking oil bottle to marble surface"
"yogurt beverage to the table"
"power adapter to chair"
"blue gloves to sink"
"milk bottle to chair"
"purple shampoo to white rack"
"herbal tea can to box"
"McDonalds paper bag to stove"

Goal for Today's Class

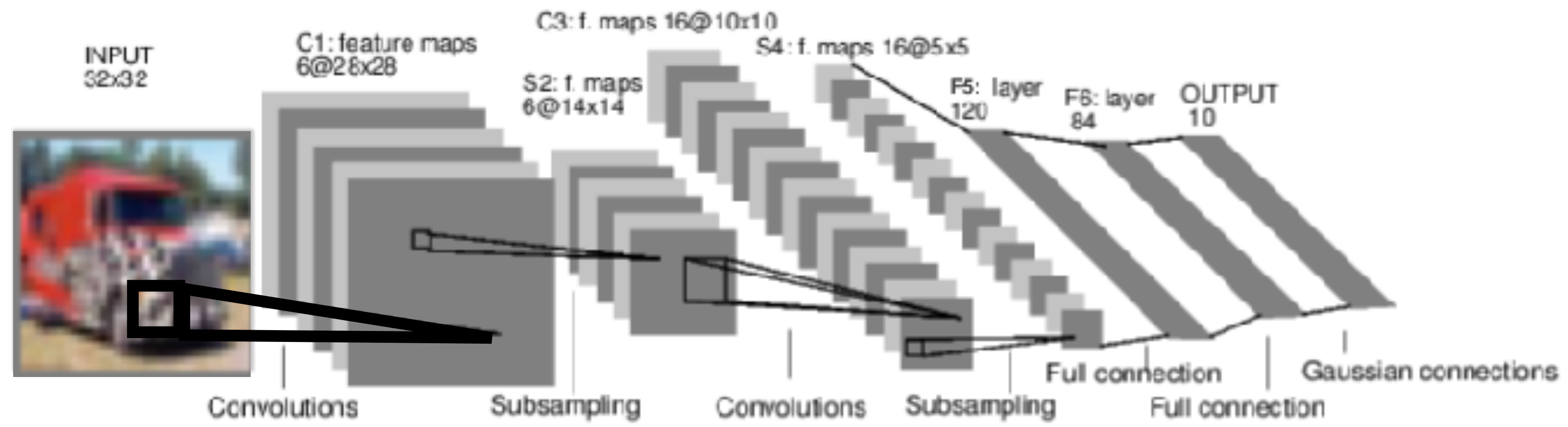
Build fundamental understanding for
object detection and semantic segmentation

Activity!

Let’s assume we have a really good image *classifier*

(assume given a set of possible labels)
{dog, cat, truck, plane, ...}

$\longrightarrow$ cat

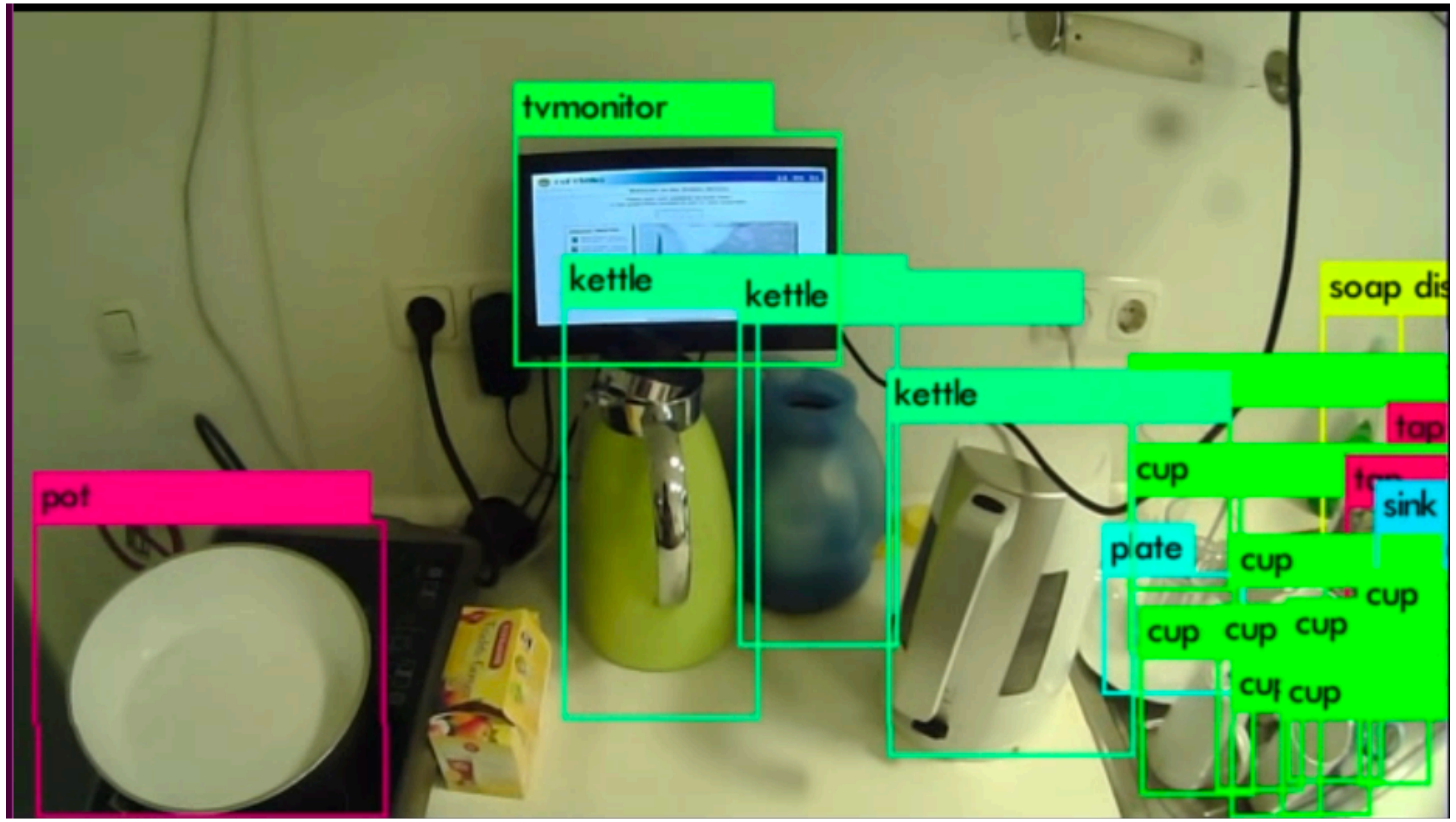


Think-Pair-Share!

Think (30 sec): How can we extend our image classifiers to detect and classify objects in an image?

Pair: Find a partner

Share (45 sec): Partners exchange ideas



Increasing complexity of computer vision tasks

Slides from Stanford CS231N: Object Detection and Image Segmentation

Increasing complexity of computer vision tasks

Classification

CAT

No spatial extent

Slides from Stanford CS231N: Object Detection and Image Segmentation

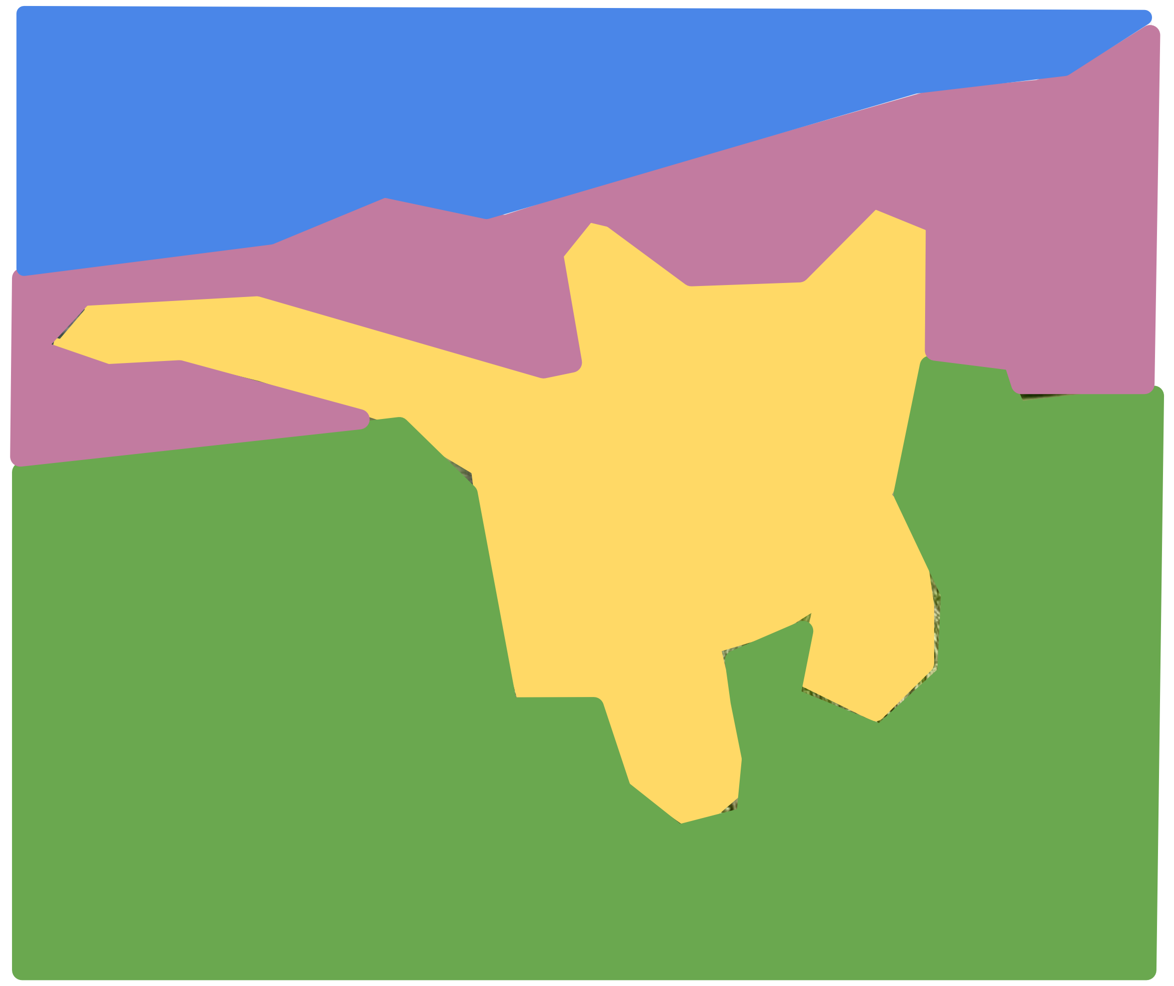
Increasing complexity of computer vision tasks

Classification

CAT

No spatial extent

Semantic Segmentation



GRASS, CAT, TREE, SKY

No objects, just pixels

Slides from Stanford CS231N: Object Detection and Image Segmentation

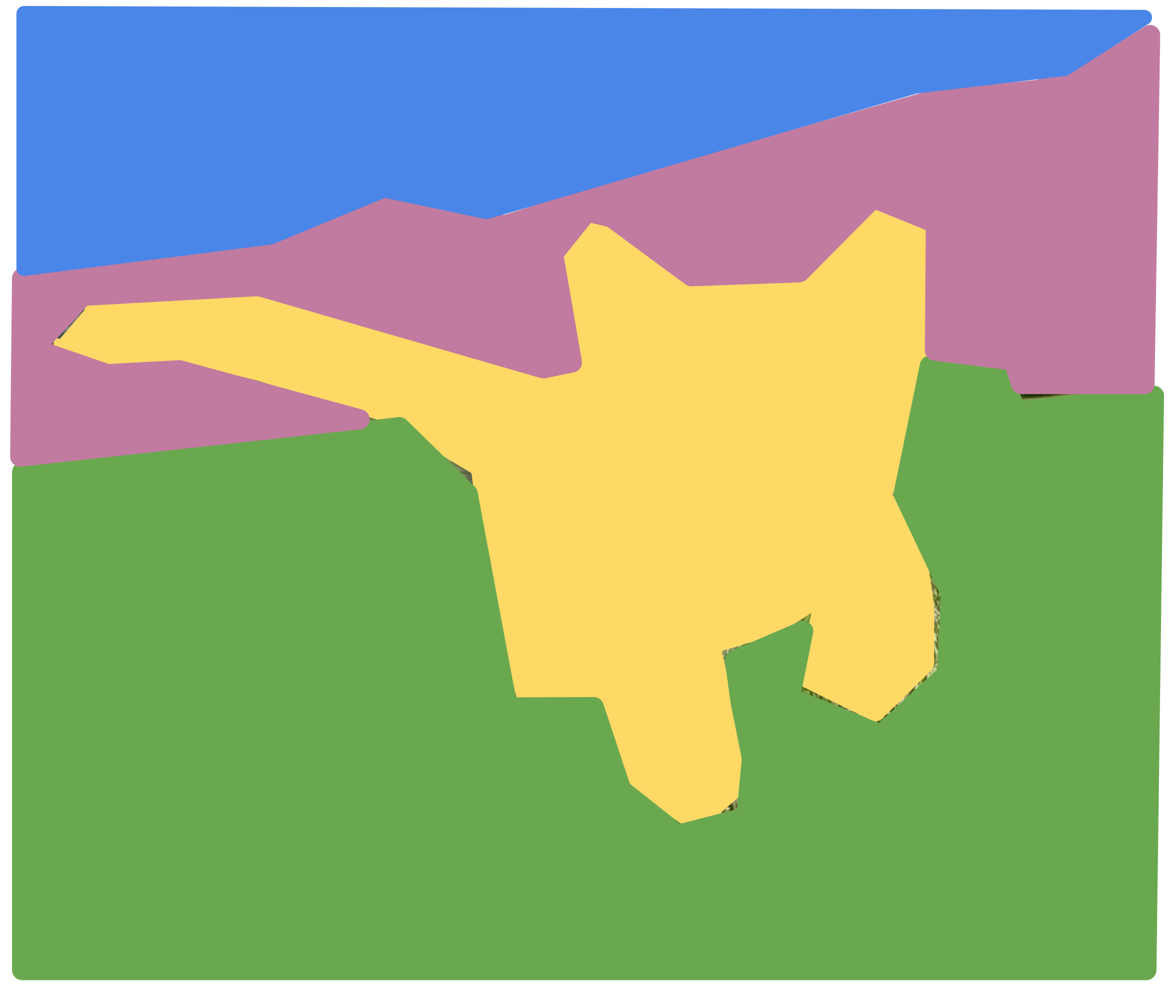
Increasing complexity of computer vision tasks

Classification

CAT

No spatial extent

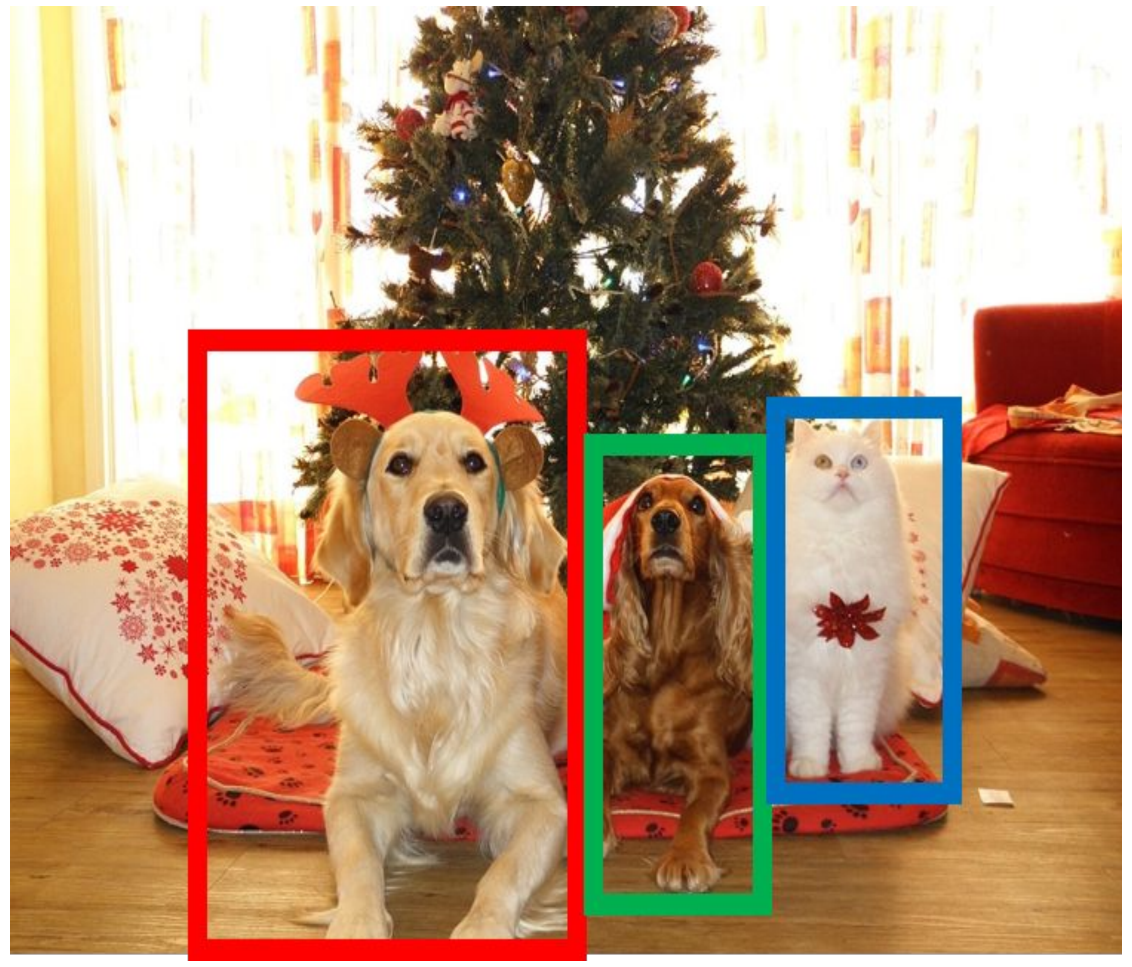
Semantic Segmentation



GRASS, CAT, TREE, SKY

No objects, just pixels

Object Detection



DOG, DOG, CAT

Multiple Object

Slides from Stanford CS231N: Object Detection and Image Segmentation

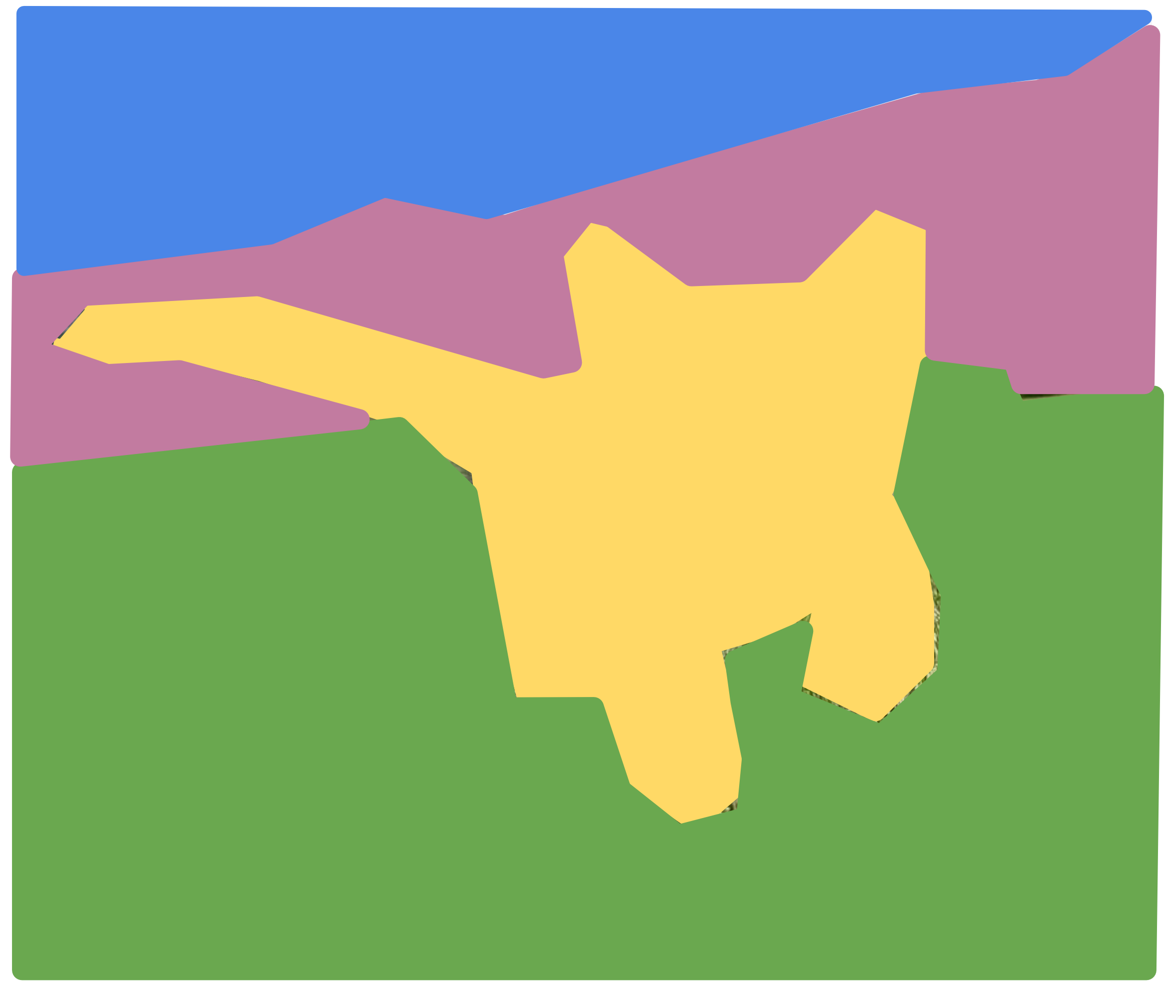
Increasing complexity of computer vision tasks

Classification

CAT

No spatial extent

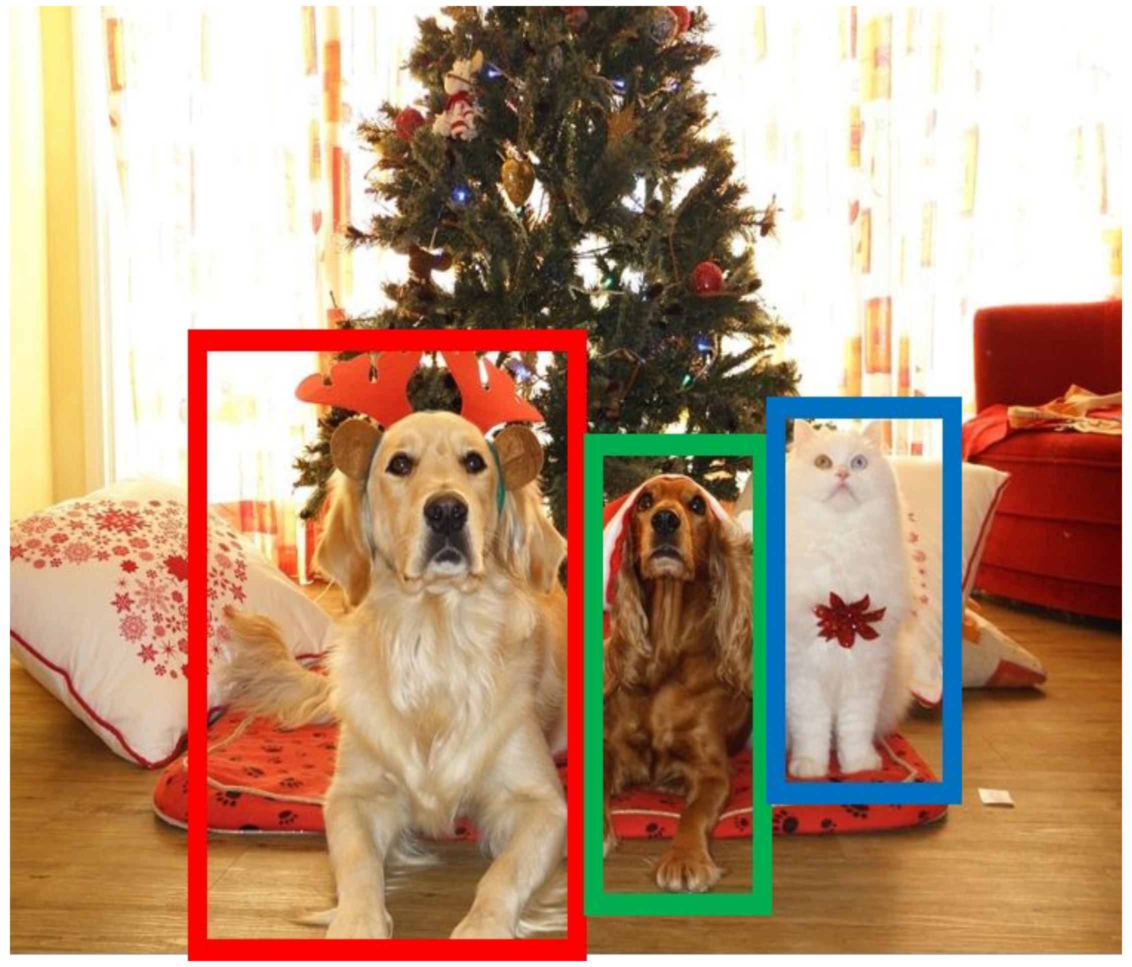
Semantic Segmentation



GRASS, CAT, TREE, SKY

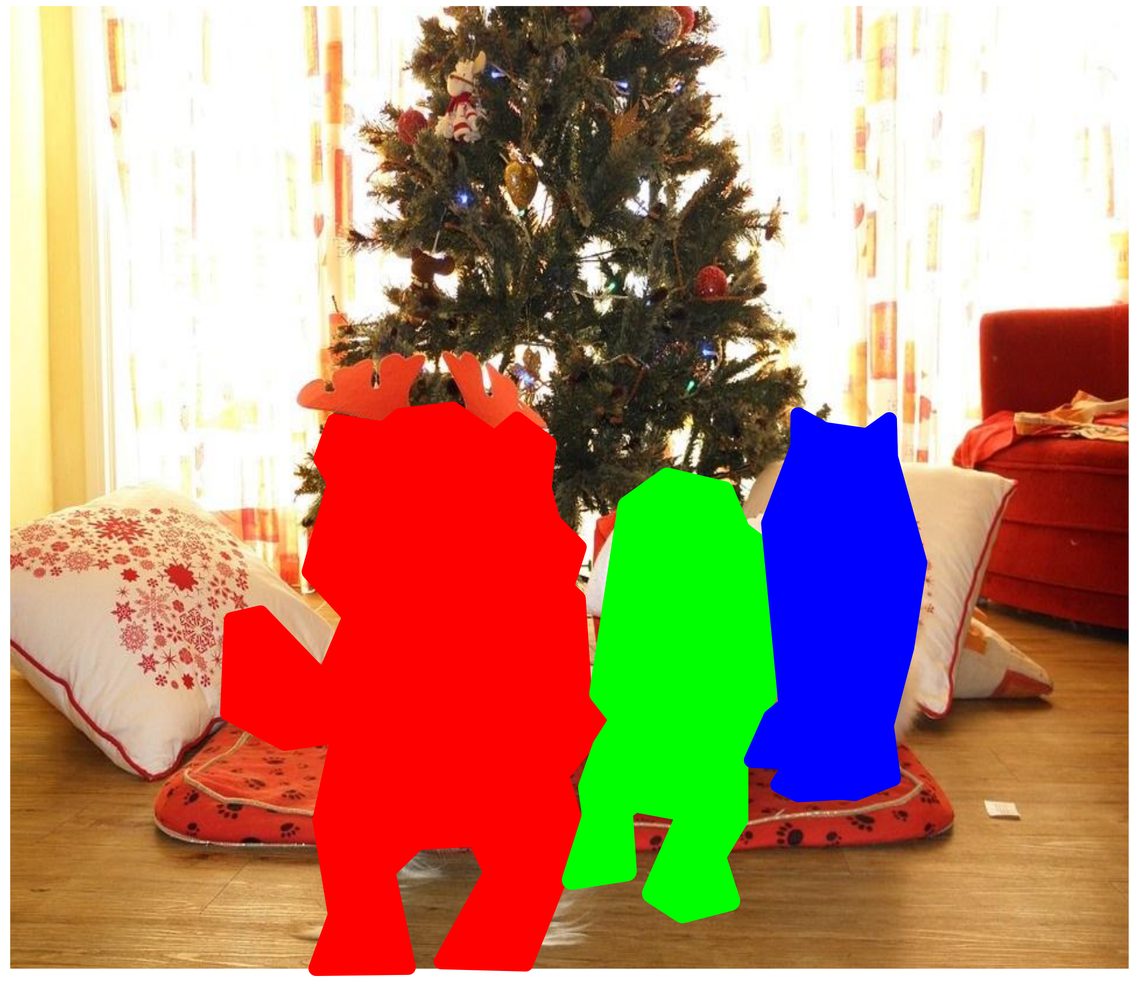
No objects, just pixels

Object Detection



DOG, DOG, CAT

Instance Segmentation

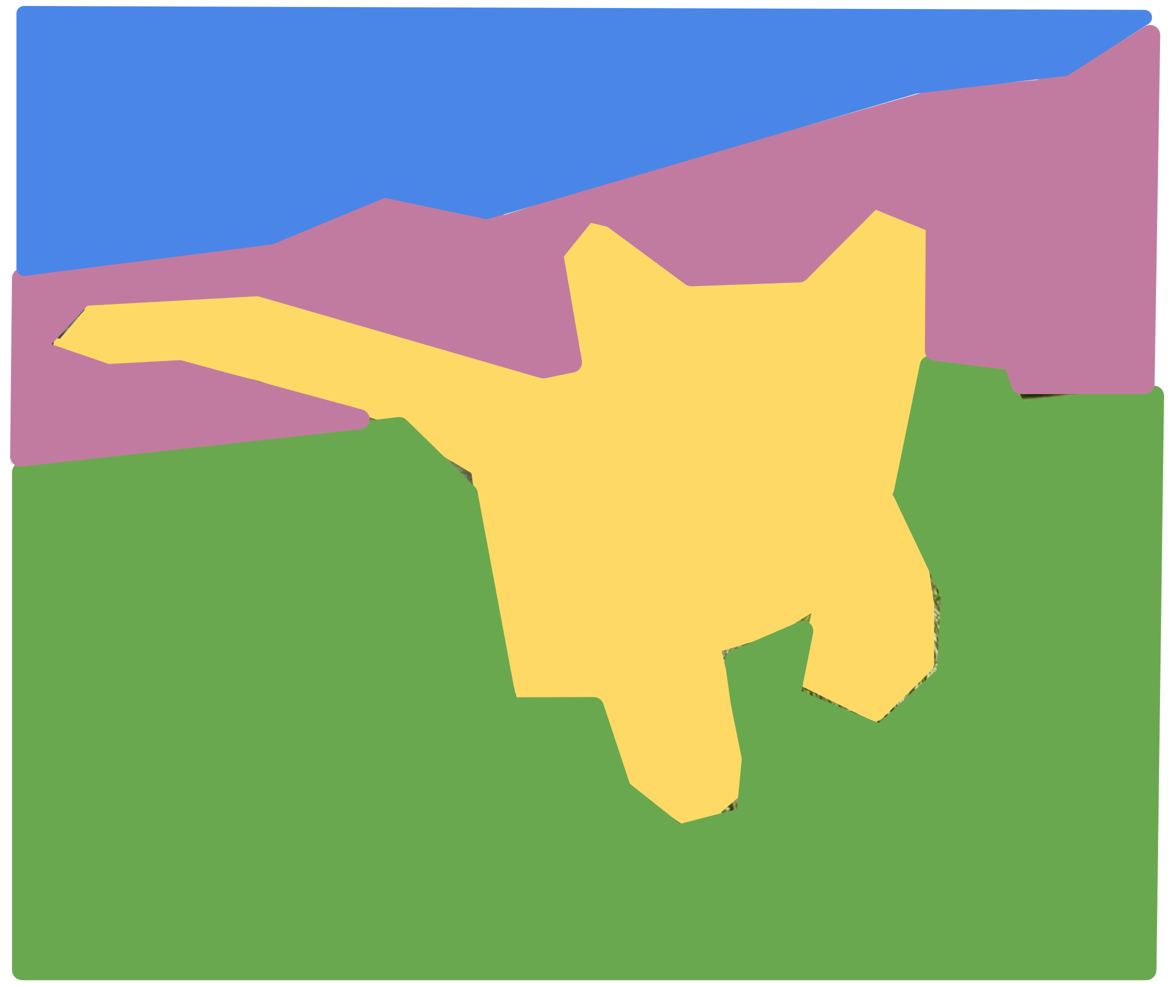


DOG, DOG, CAT

Multiple Object

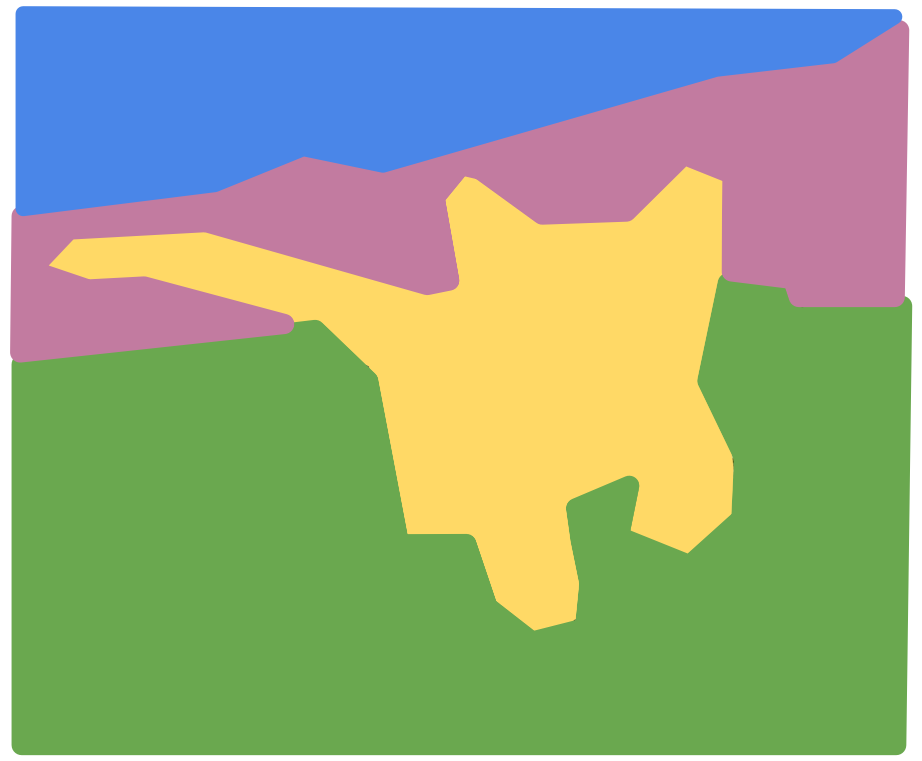
Slides from Stanford CS231N: Object Detection and Image Segmentation

Increasing complexity of computer vision tasks

Classification	Semantic Segmentation	Object Detection	Instance Segmentation
			
CAT	GRASS, CAT, TREE, SKY	DOG, DOG, CAT	DOG, DOG, CAT
No spatial extent	No objects, just pixels	Multiple Object	

Slides from Stanford CS231N: Object Detection and Image Segmentation

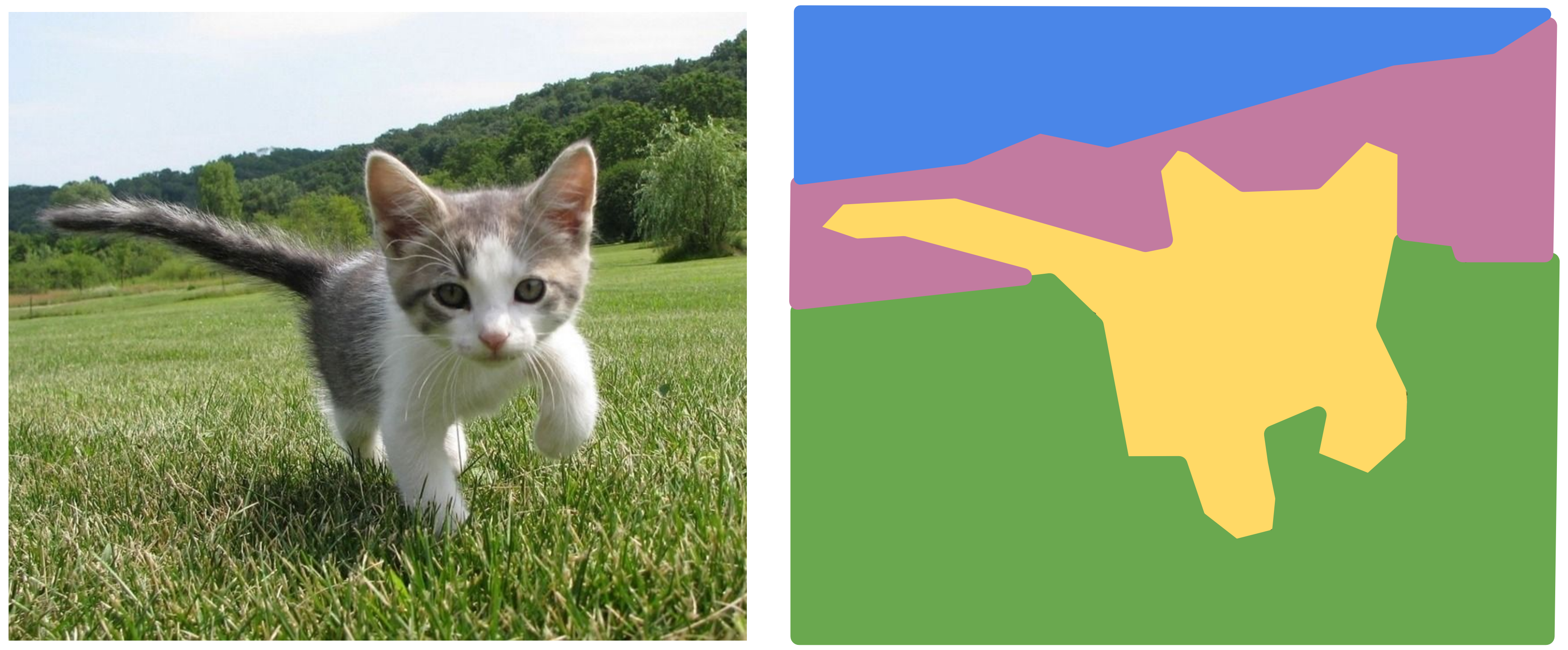
Semantic Segmentation: The Problem



GRASS, **CAT**,
TREE, **SKY**, **...**

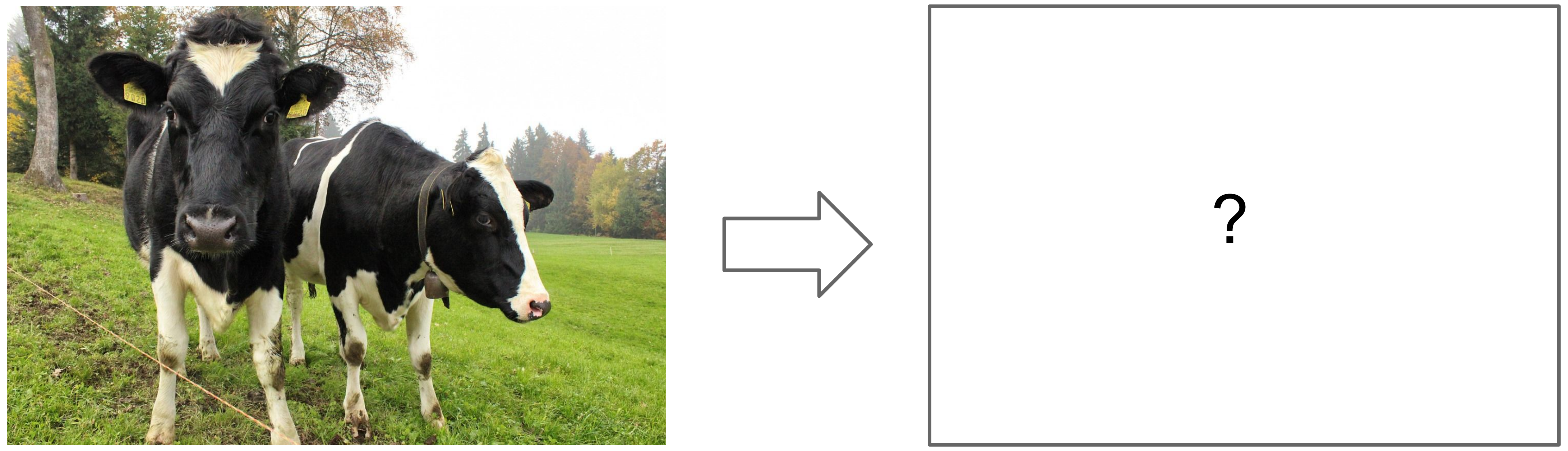
Paired training data: for each training image, each pixel is labeled with a semantic category.

Semantic Segmentation: The Problem



GRASS, CAT,
TREE, SKY, ...

Paired training data: for each training image, each pixel is labeled with a semantic category.



At test time, classify each pixel of a new image.

Slides from Stanford CS231N: Object Detection and Image Segmentation

Semantic Segmentation Idea: Sliding Window

Full image

Can you classify this pixel?

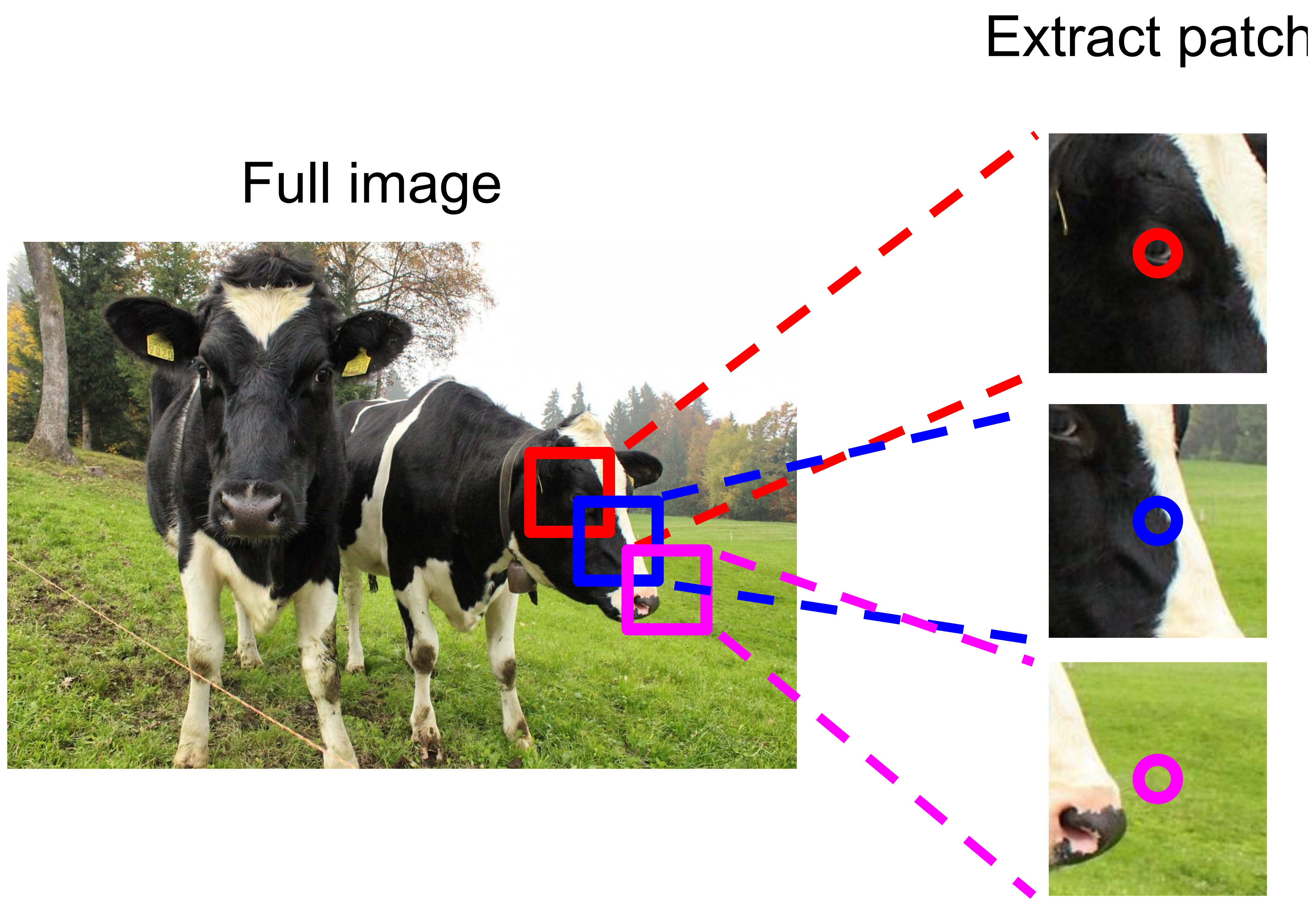
Semantic Segmentation Idea: Sliding Window

Full image

Can you classify this pixel?

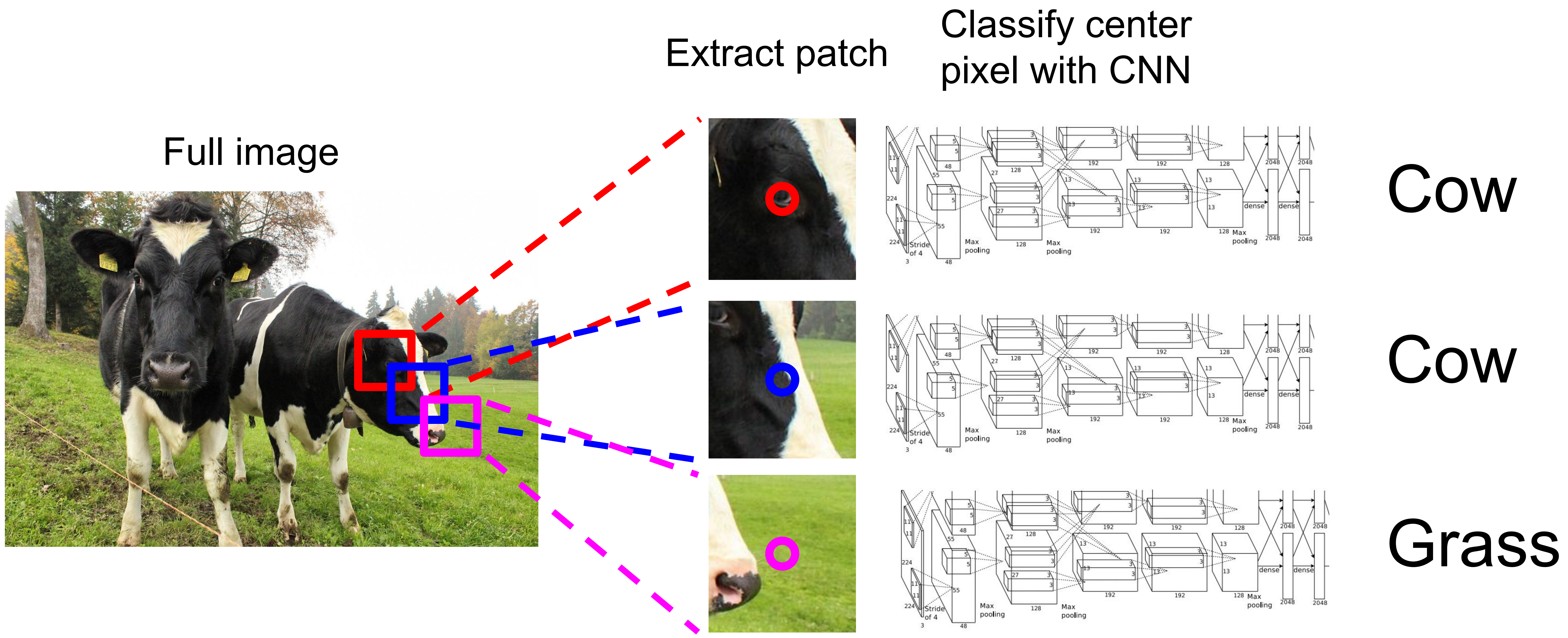
Pretty hard without context!

Semantic Segmentation Idea: Sliding Window



Slides from Stanford CS231N: Object Detection and Image Segmentation

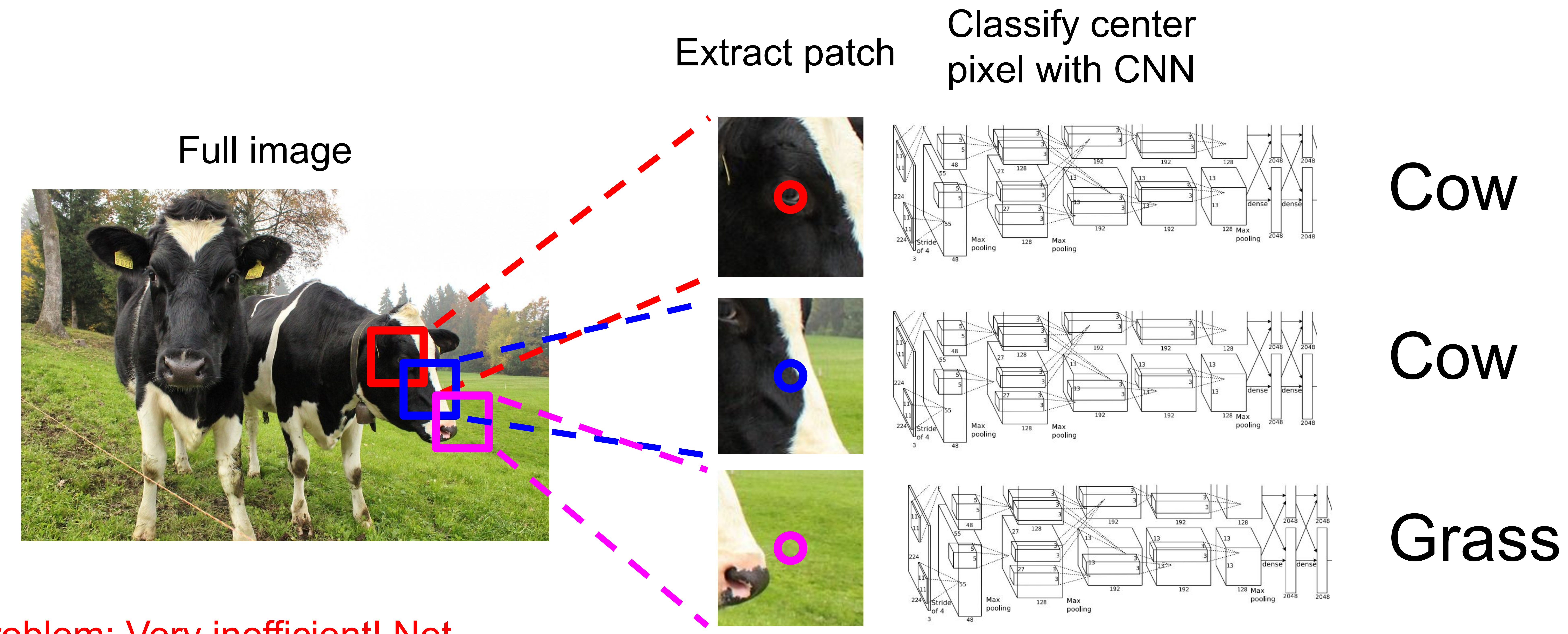
Semantic Segmentation Idea: Sliding Window



Classify each patch!

Farabet et al, “Learning Hierarchical Features for Scene Labeling,” TPAMI 2013
Pinheiro and Collobert, “Recurrent Convolutional Neural Networks for Scene Labeling”, ICML 2014

Semantic Segmentation Idea: Sliding Window



Problem: Very inefficient! Not reusing shared features between overlapping patches

Farabet et al, “Learning Hierarchical Features for Scene Labeling,” TPAMI 2013
Pinheiro and Collobert, “Recurrent Convolutional Neural Networks for Scene Labeling”, ICML 2014

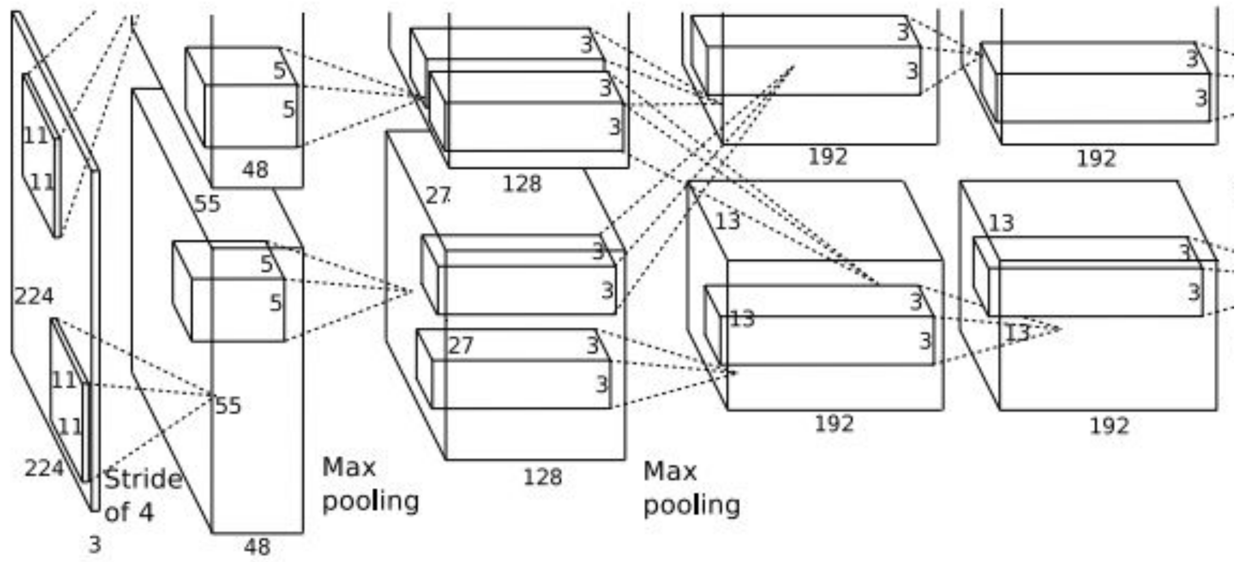
Semantic Segmentation Idea: Convolution

Full image

Slides from Stanford CS231N: Object Detection and Image Segmentation

Semantic Segmentation Idea: Convolution

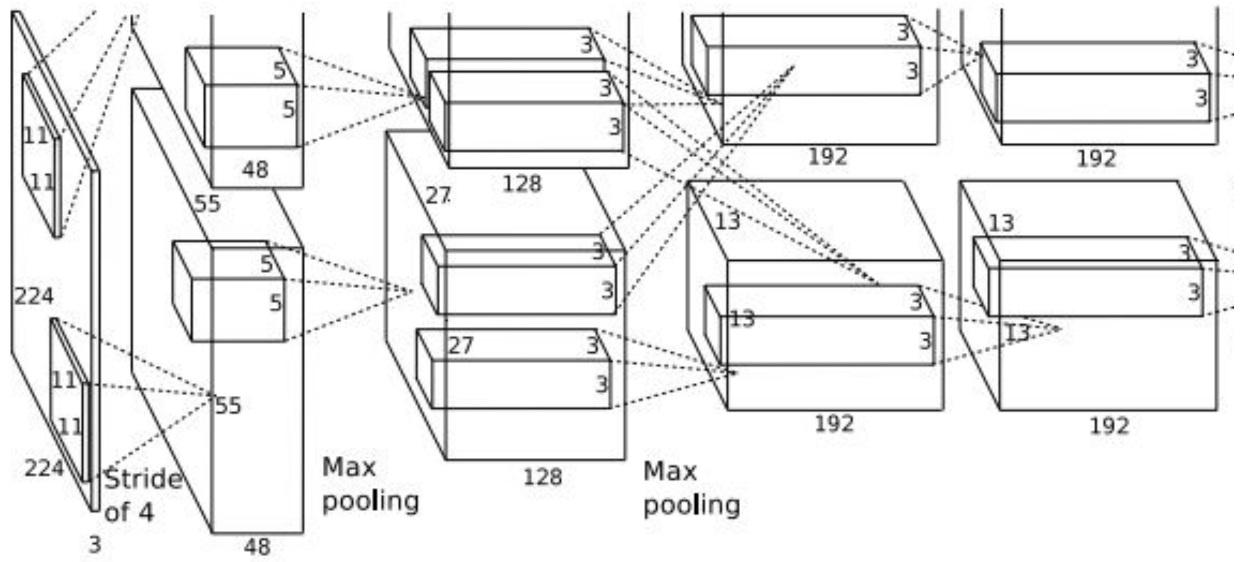
Full image



An intuitive idea: encode the entire image with conv net, and do semantic segmentation on top.

Semantic Segmentation Idea: Convolution

Full image



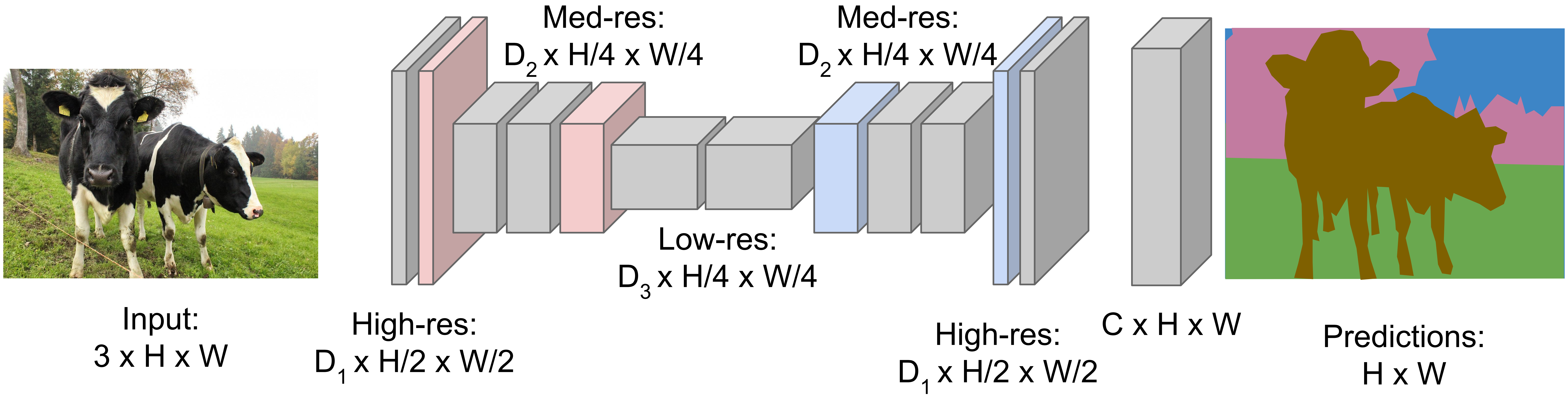
An intuitive idea: encode the entire image with conv net, and do semantic segmentation on top.

Problem: classification architectures often reduce feature spatial sizes to go deeper, but semantic segmentation requires the output size to be the same as input size.

Slides from Stanford CS231N: Object Detection and Image Segmentation

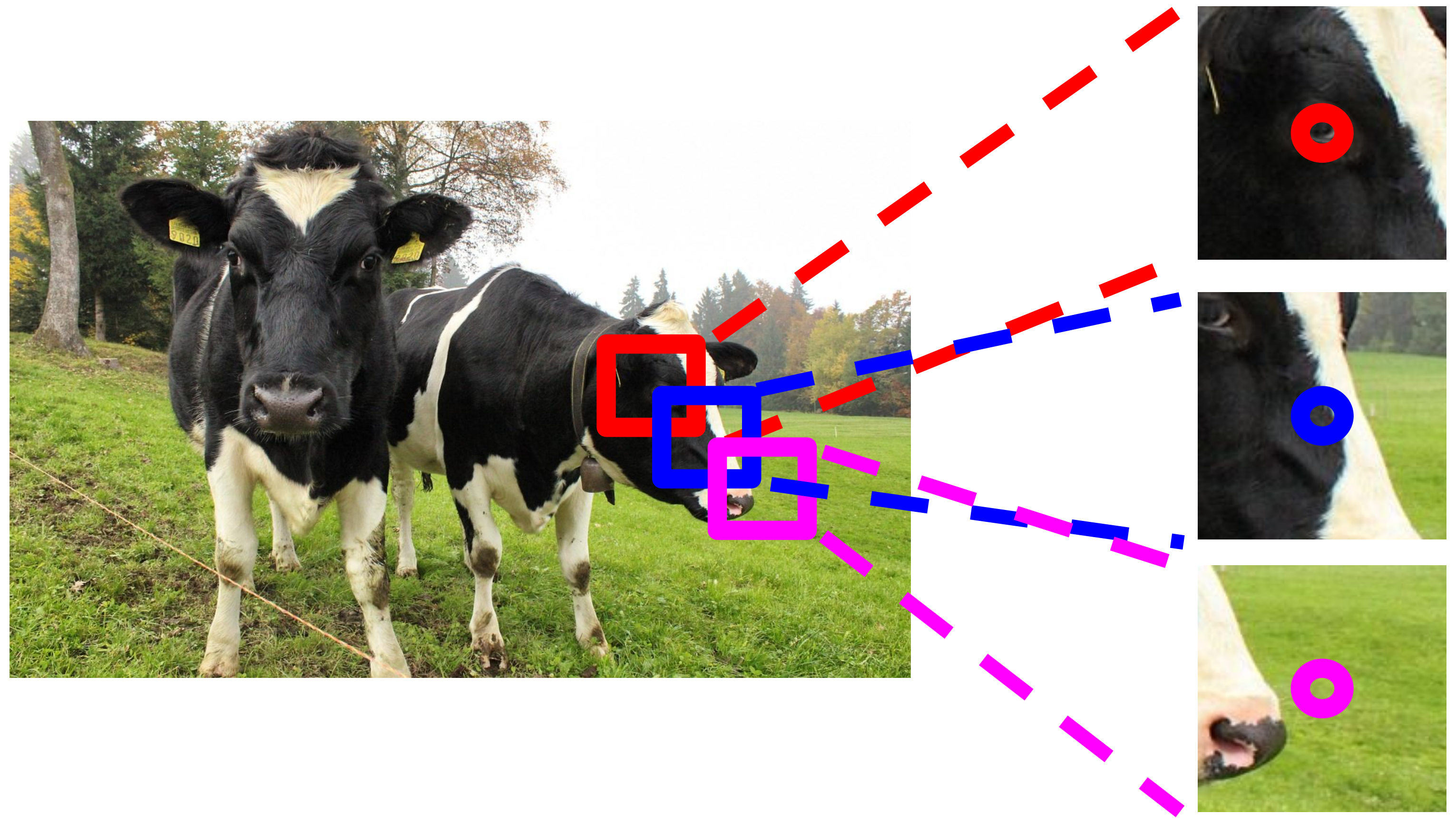
Semantic Segmentation Idea: Fully Convolutional

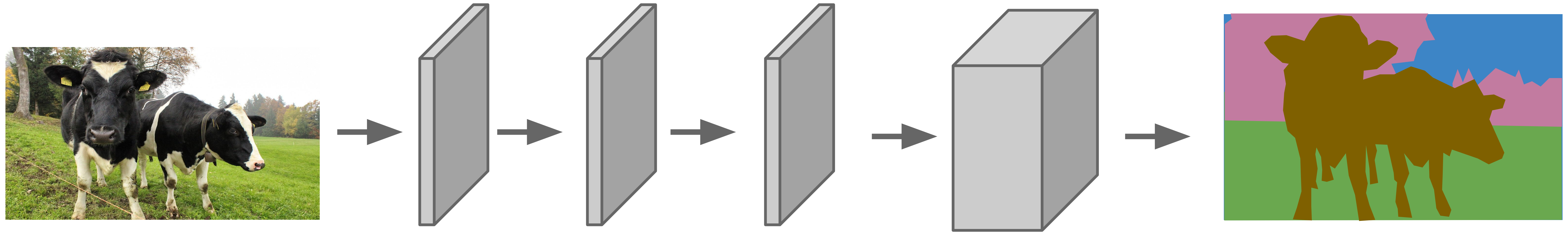
Design network as a bunch of convolutional layers, with **downsampling** and **upsampling** inside the network!

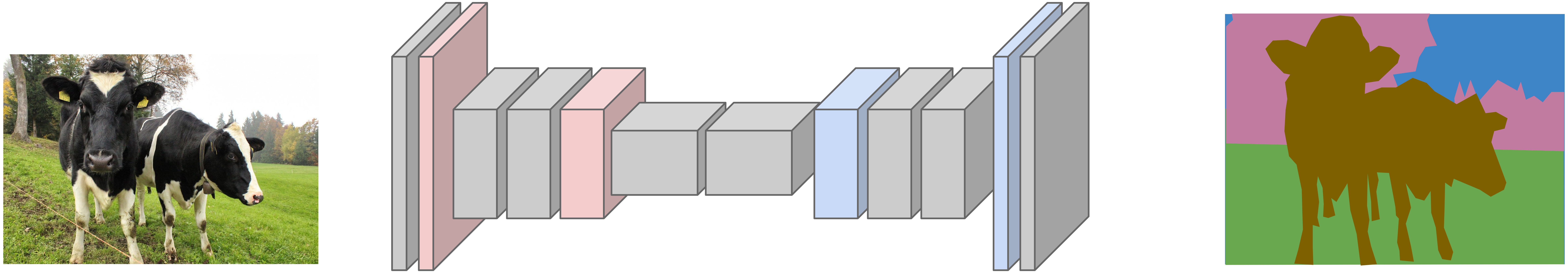


Long, Shelhamer, and Darrell, "Fully Convolutional Networks for Semantic Segmentation", CVPR 2015
Noh et al, "Learning Deconvolution Network for Semantic Segmentation", ICCV 2015

Semantic Segmentation: Summary





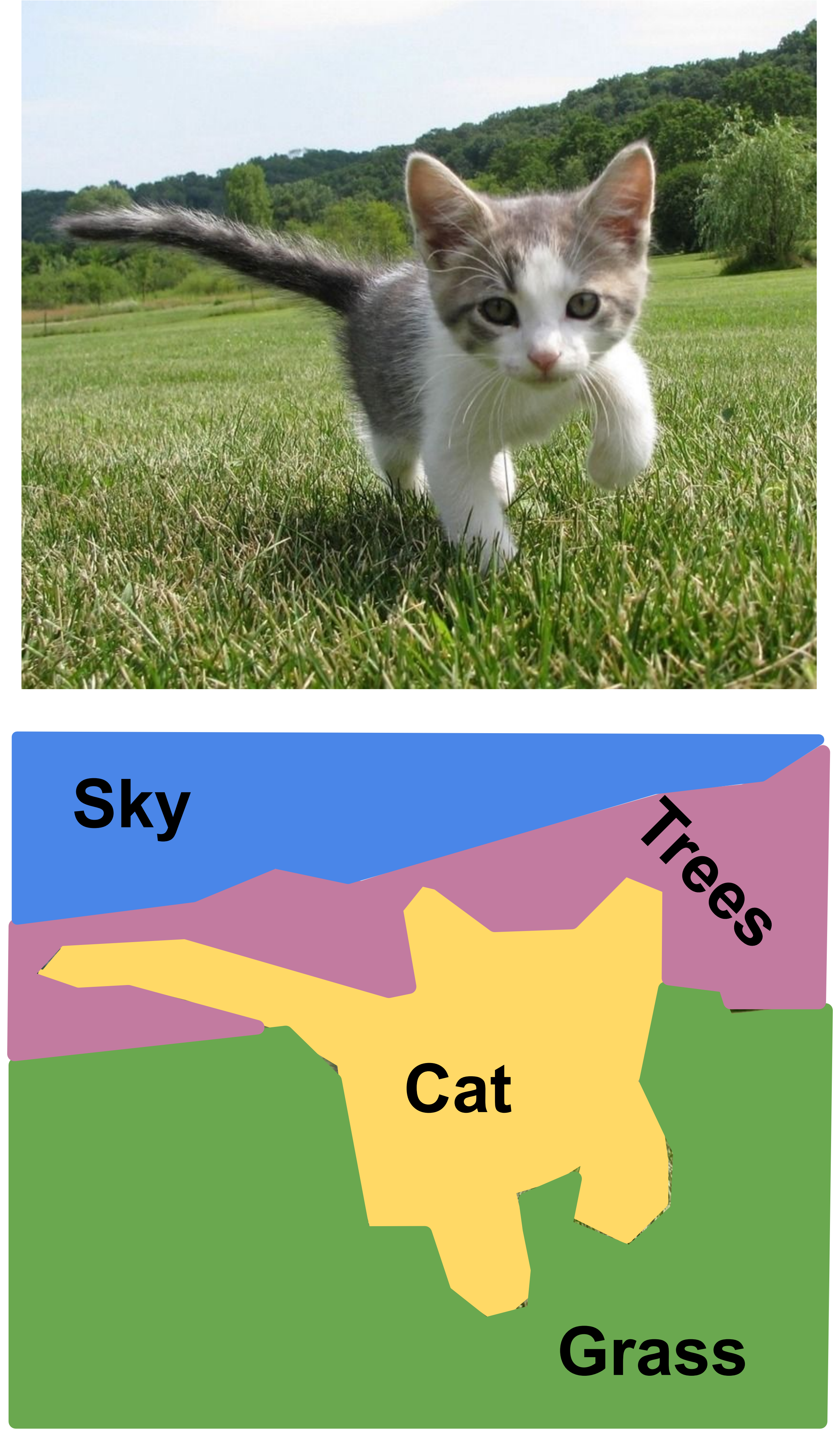


Slides from Stanford CS231N: Object Detection and Image Segmentation

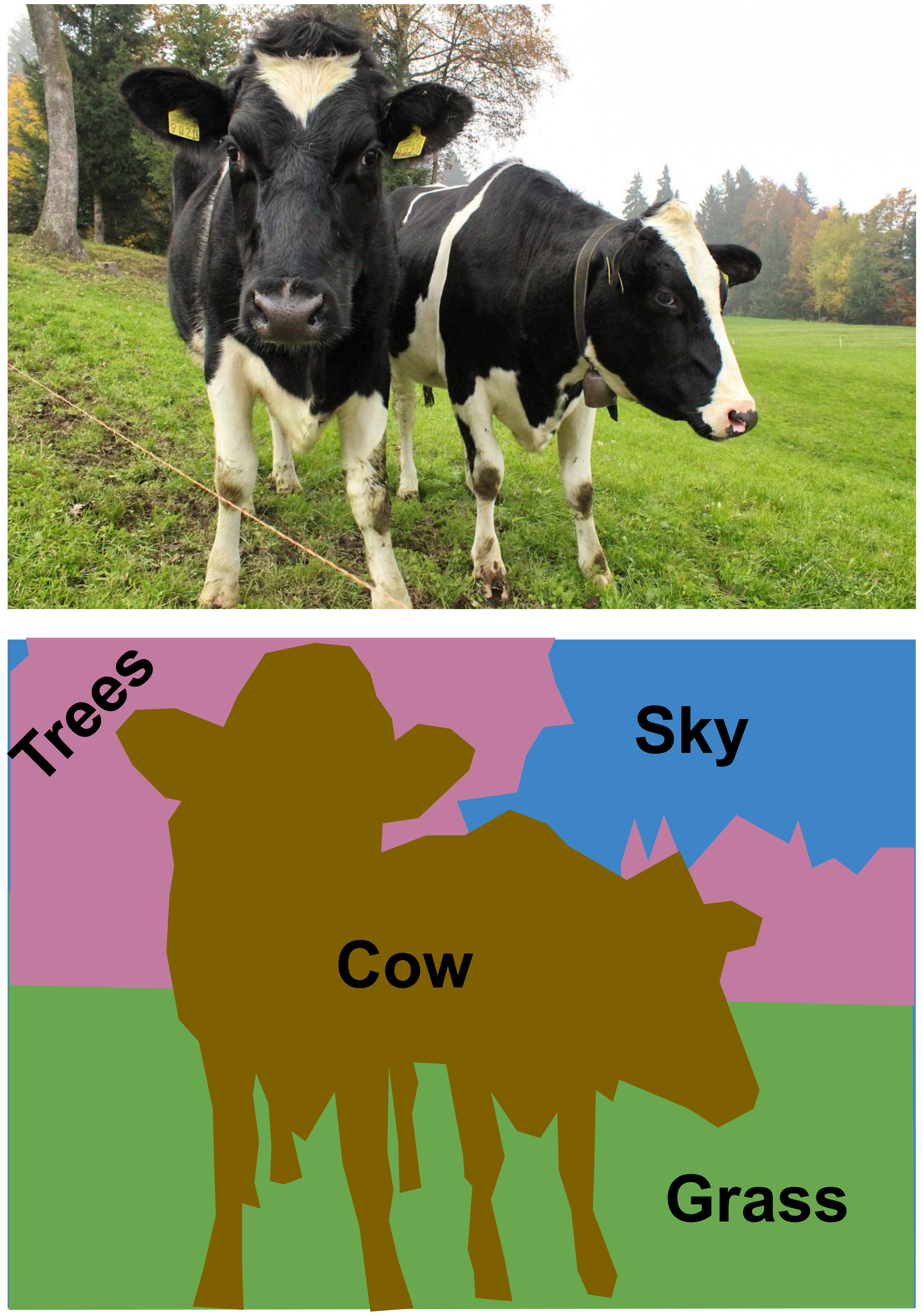
Semantic Segmentation

Label each pixel in the image with a category label

Don't differentiate instances, only care about pixels



This image is CC0 public domain



Increasing complexity of computer vision tasks

Classification

CAT

No spatial extent

Semantic Segmentation

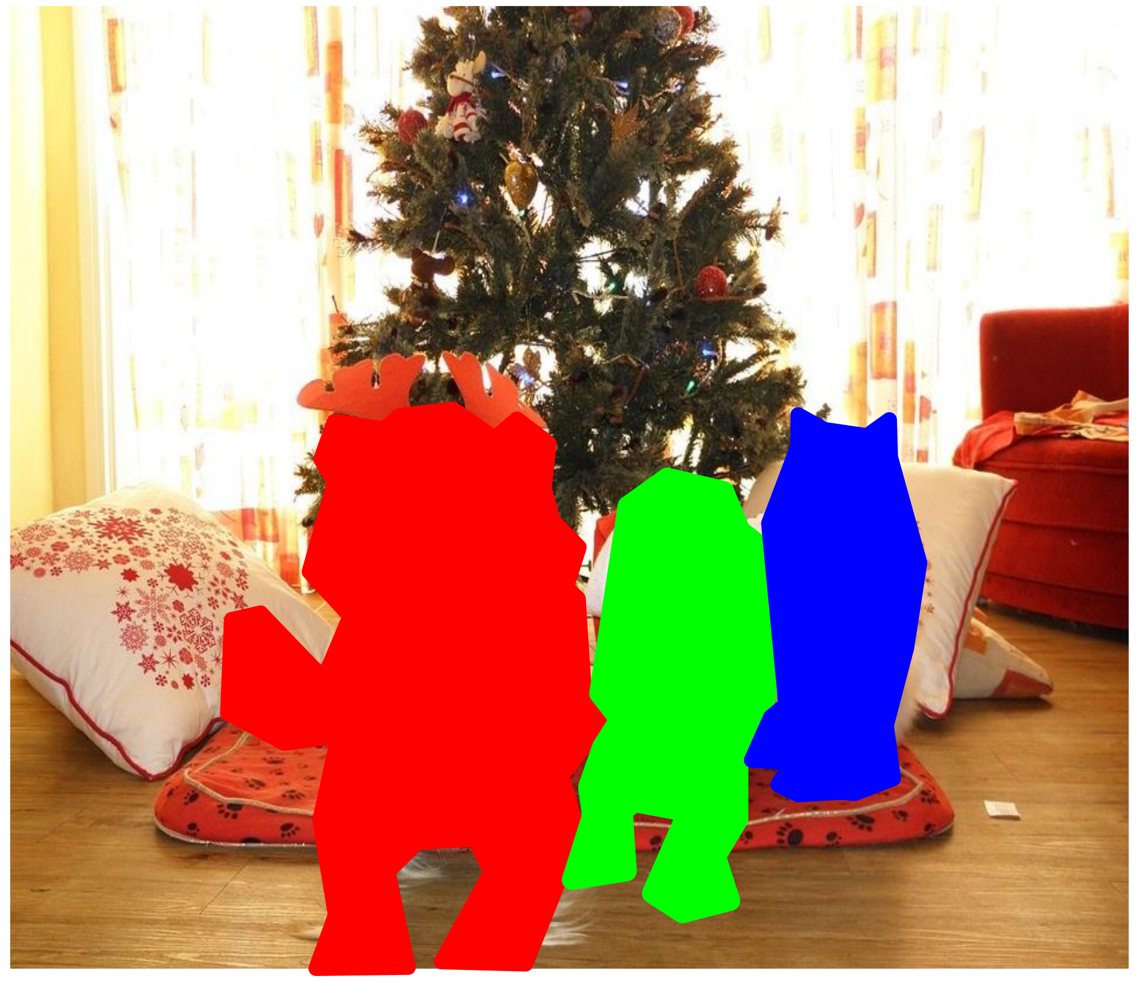
GRASS, CAT, TREE, SKY

No objects, just pixels

Object Detection

DOG, DOG, CAT

Instance Segmentation



DOG, DOG, CAT

Multiple Object

Slides from Stanford CS231N: Object Detection and Image Segmentation

Increasing complexity of computer vision tasks

Classification

CAT

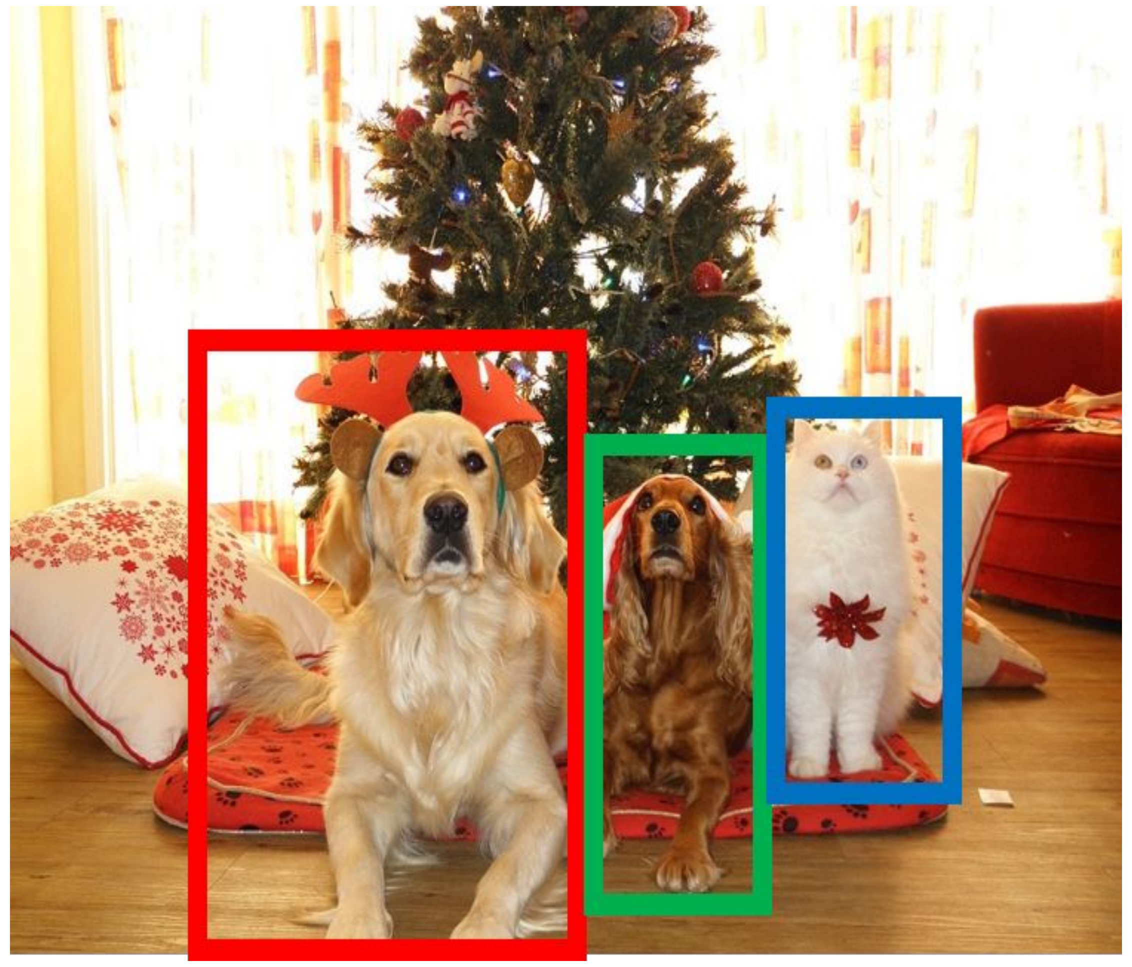
No spatial extent

Semantic Segmentation

GRASS, CAT, TREE, SKY

No objects, just pixels

Object Detection



DOG, DOG, CAT

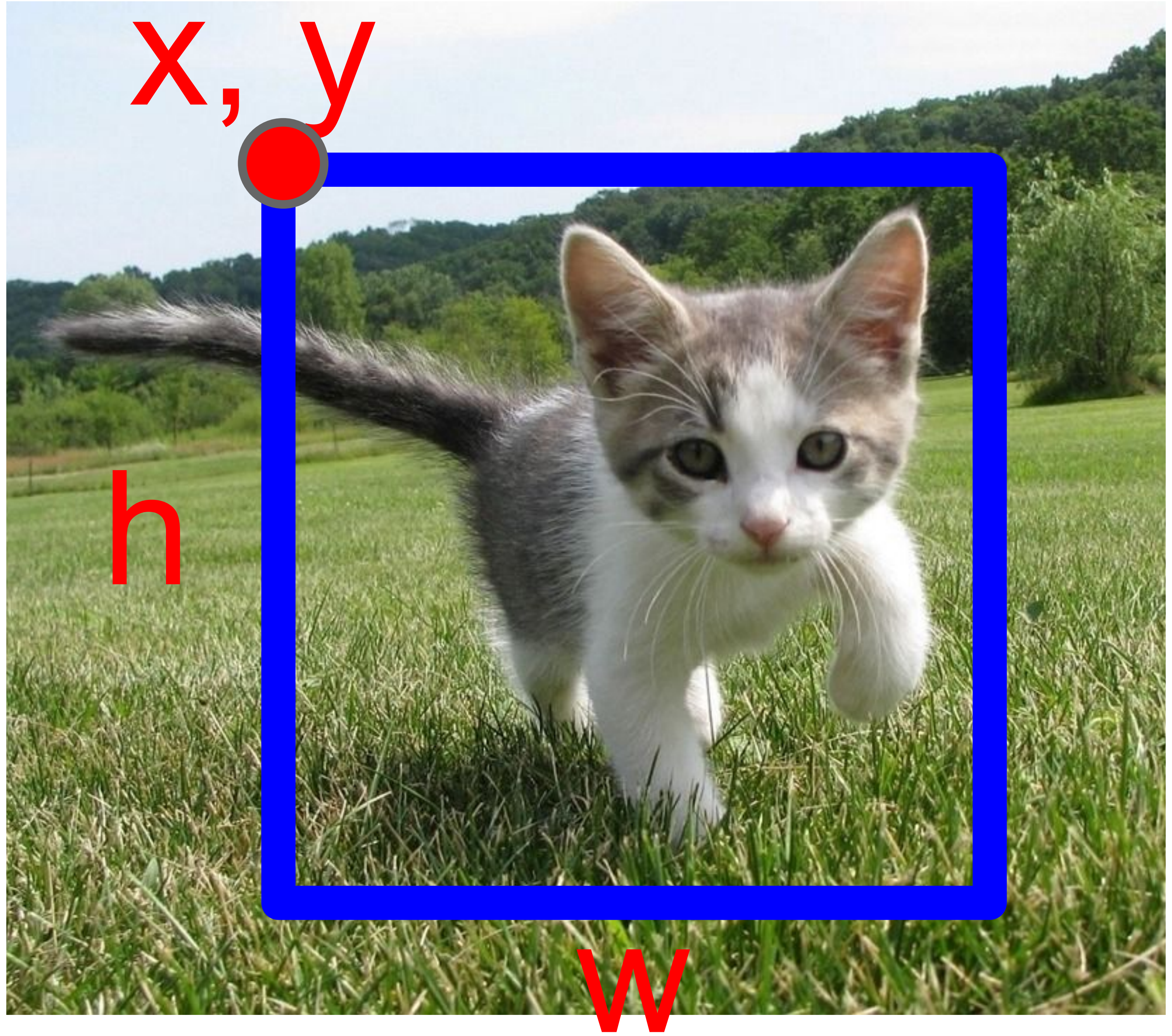
Instance Segmentation

DOG, DOG, CAT

Multiple Object

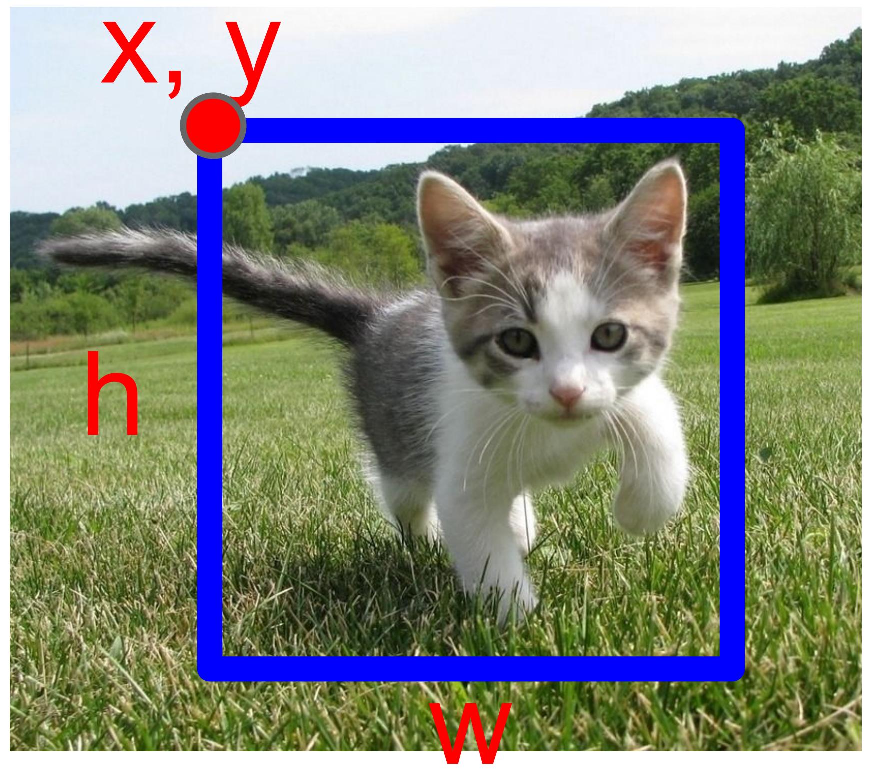
Slides from Stanford CS231N: Object Detection and Image Segmentation

Object Detection: Single Object
(Classification + Localization)



Activity!

Poll



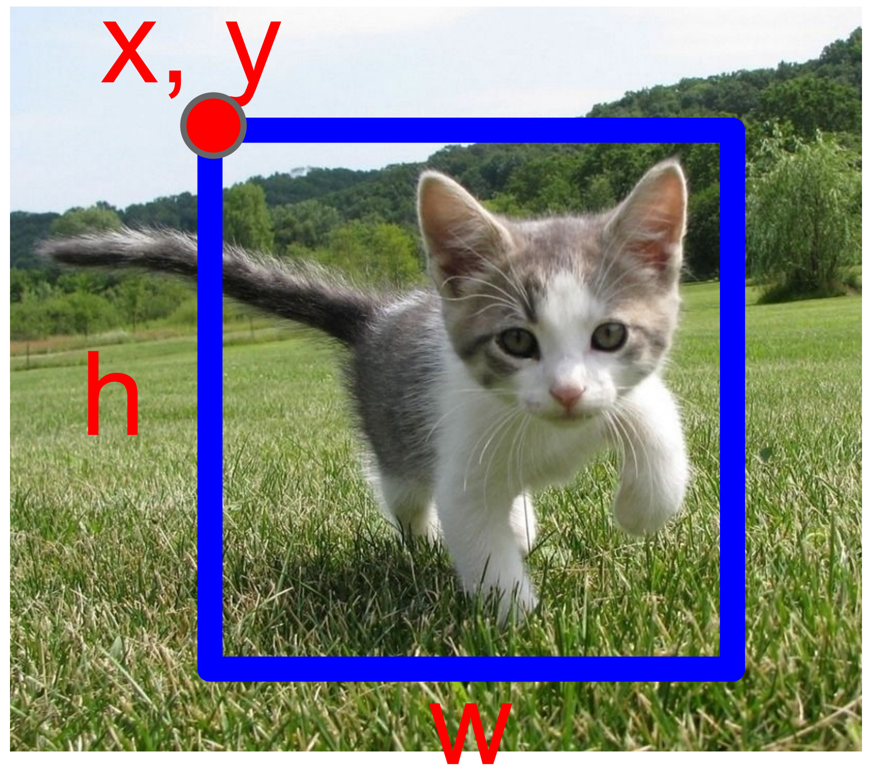
Assume you have a dataset of images.

For each image, you have a target object and a bounding box.

You have a model to predict target objects and bounding boxes.

What loss will you use?

Poll



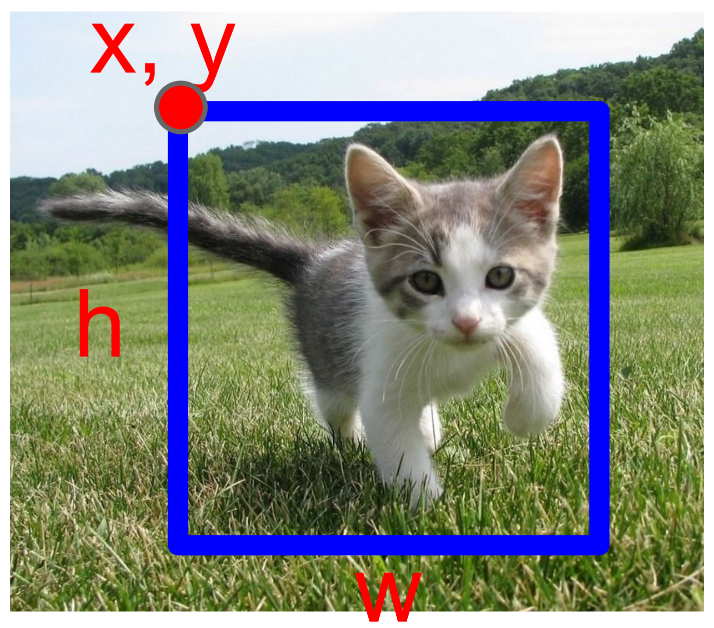
What loss will you use?

When poll is active respond at **PollEv.com/sc2582**

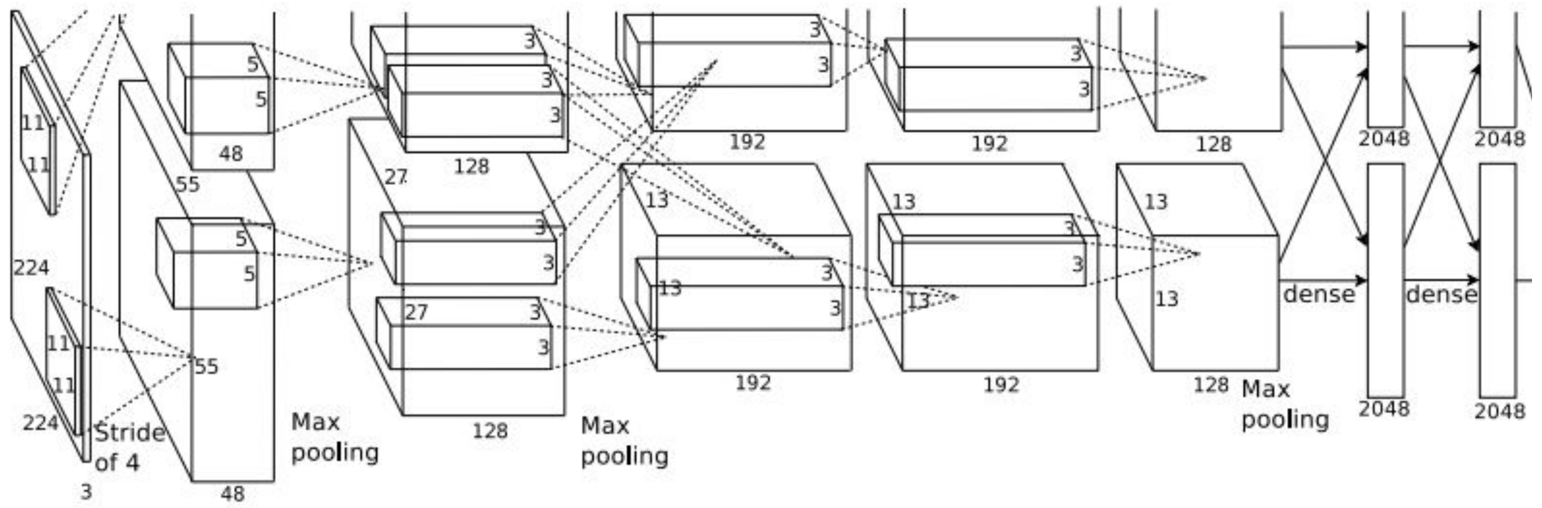
Send **sc2582** to **22333**

Object Detection: Single Object

(Classification + Localization)



This image is CC0 public domain



Vector: 4096

Fully Connected: 4096 to 1000

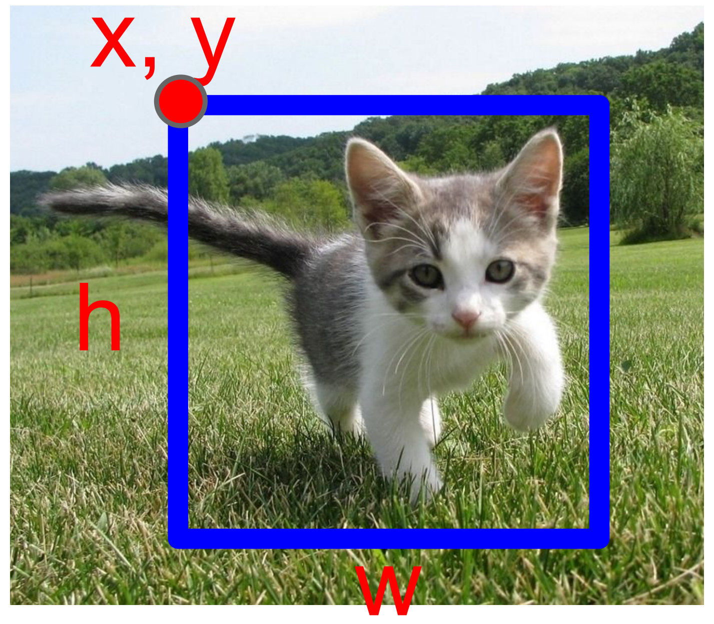
Class Scores
Cat: 0.9
Dog: 0.05
Car: 0.01
...

Fully Connected: 4096 to 4

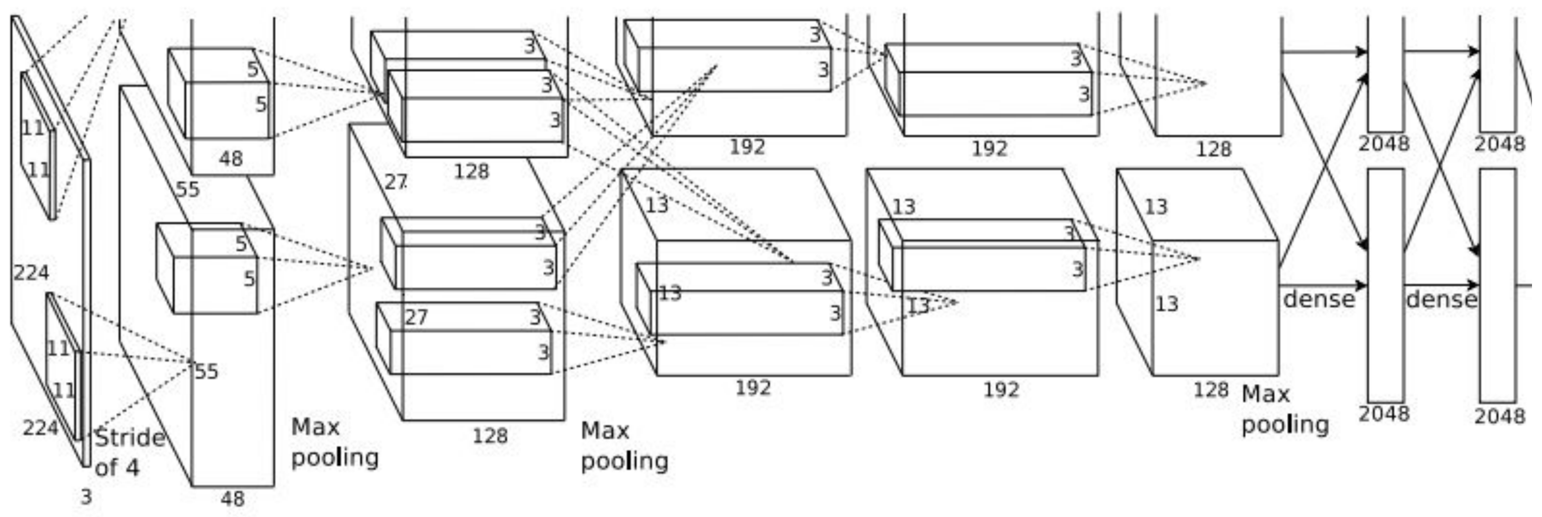
Box Coordinates
(x, y, w, h)

Object Detection: Single Object

(Classification + Localization)



[This image](#) is [CC0 public domain](#)



Vector: 4096

Fully Connected: 4096 to 1000

Class Scores
Cat: 0.9
Dog: 0.05
Car: 0.01
...

→ **Softmax Loss** ← **Correct label:** Cat

Fully Connected: 4096 to 4

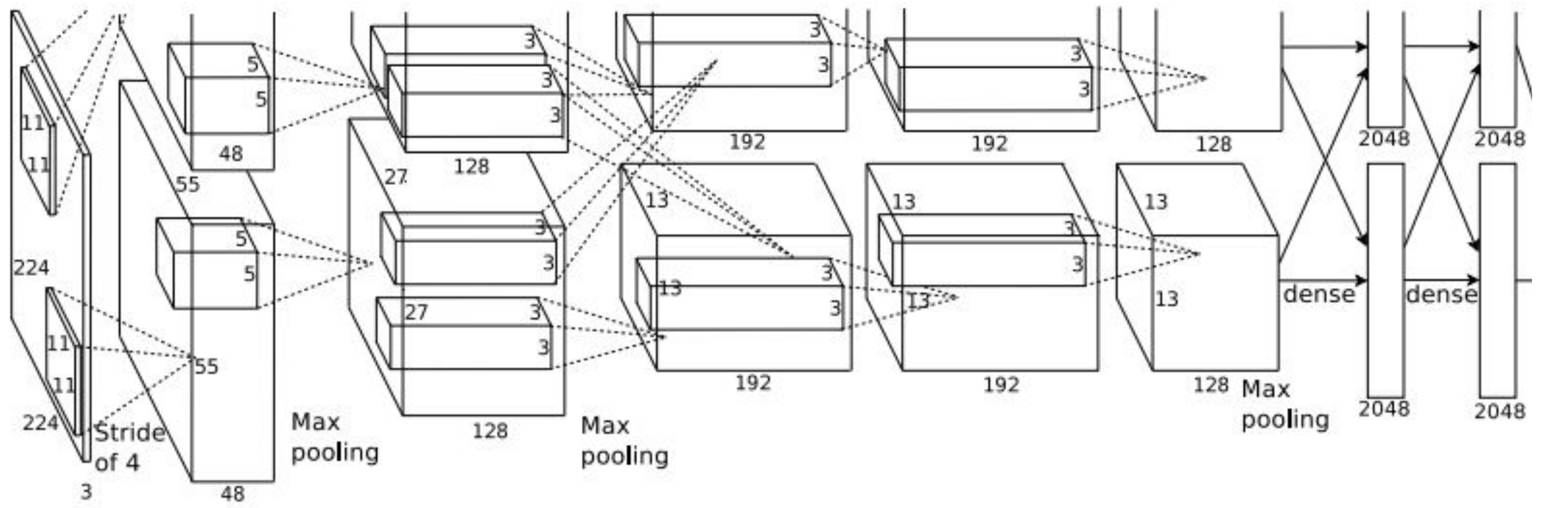
Box Coordinates
(x, y, w, h)

→ **L2 Loss** ← **Correct box**: (x’, y’, w’, h’)

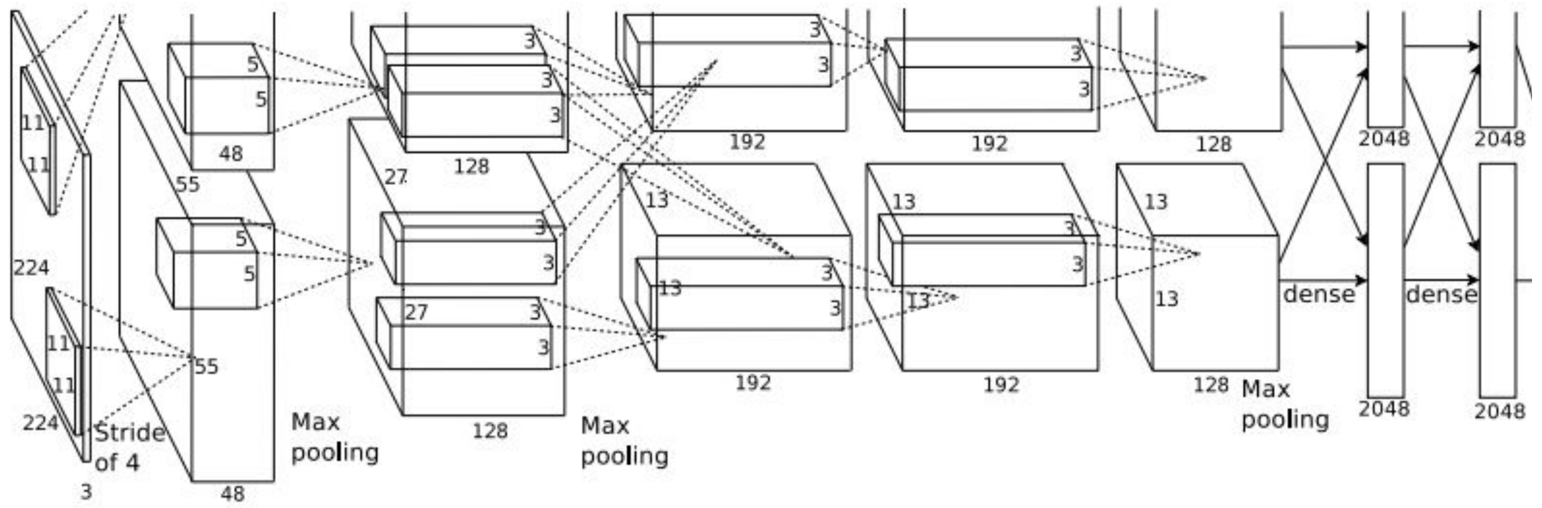
Treat localization as a regression problem!

What about multiple objects? Would this idea work?

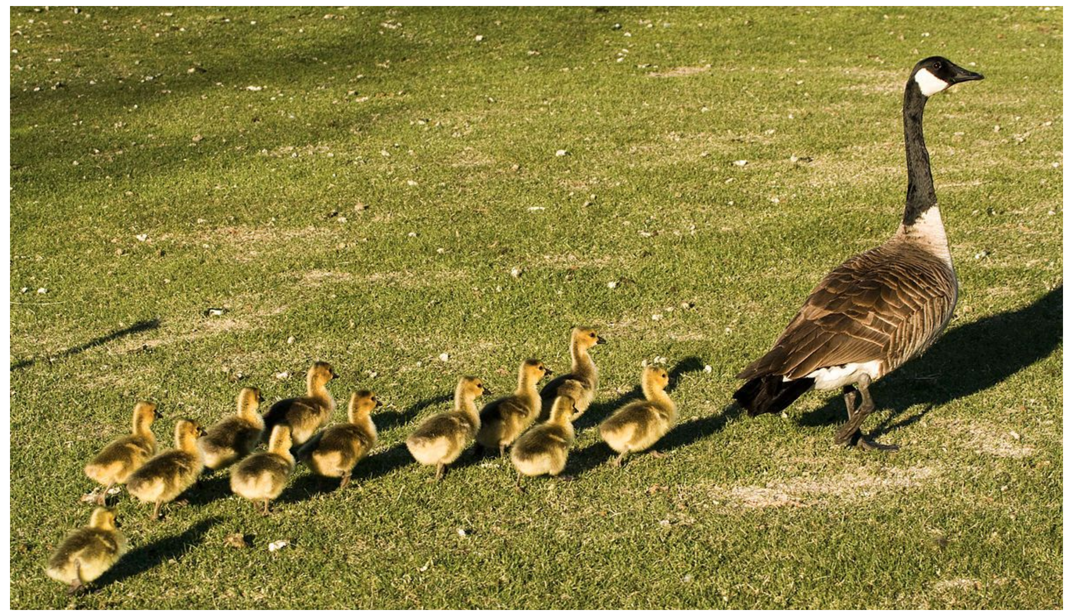
Object Detection: Multiple Objects

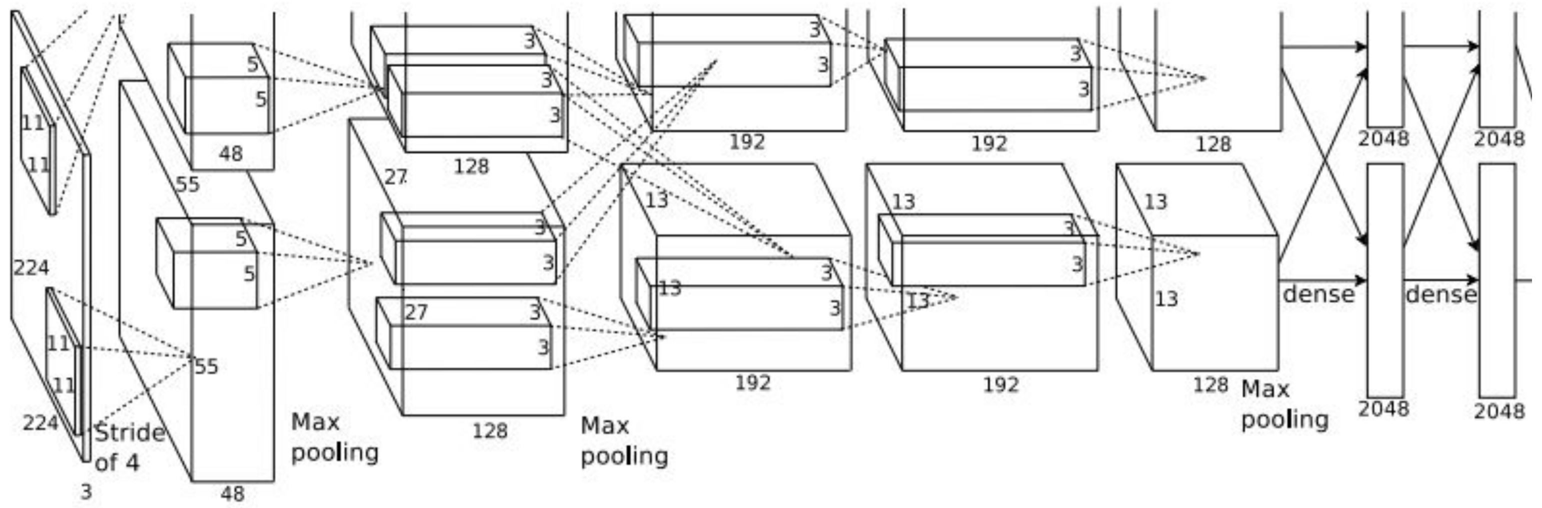


CAT: (x, y, w, h)



DOG: (x, y, w, h)
DOG: (x, y, w, h)
CAT: (x, y, w, h)

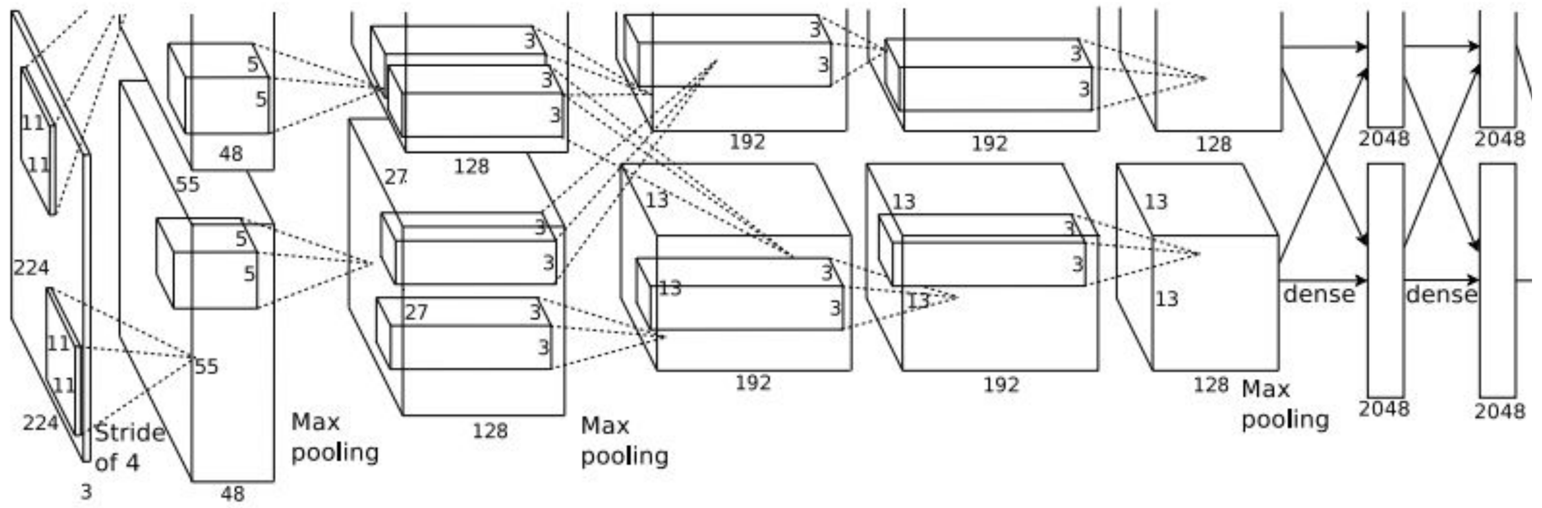




DUCK: (x, y, w, h)
DUCK: (x, y, w, h)
….

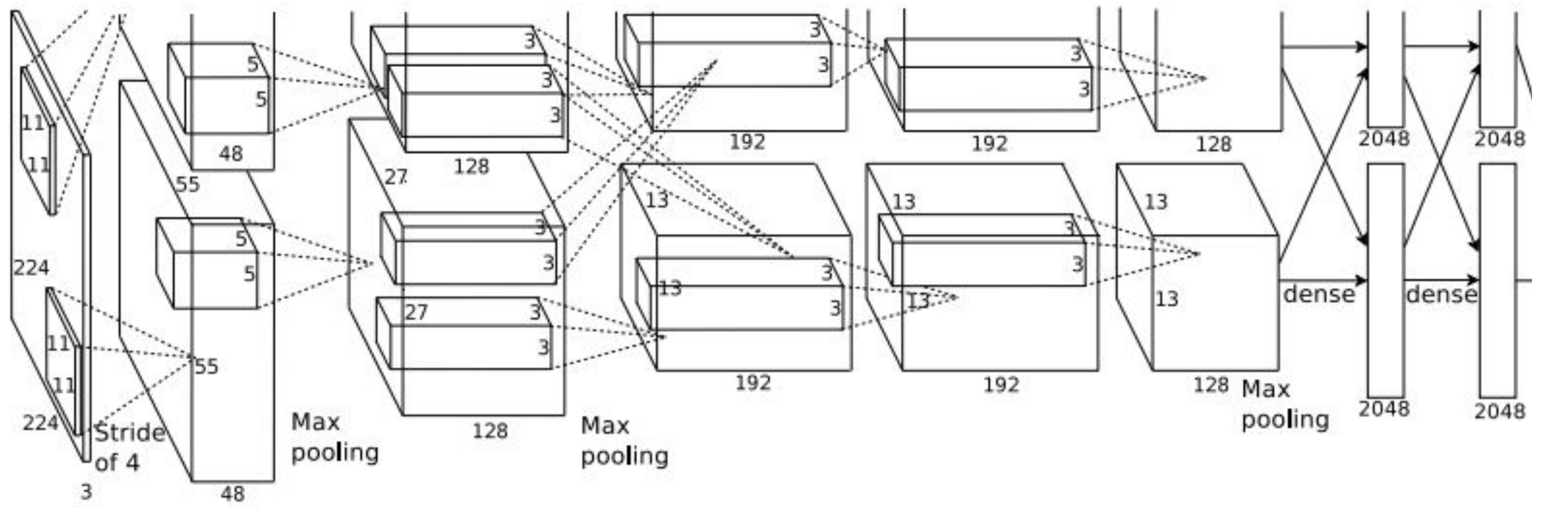
Object Detection: Multiple Objects

Each image needs a different number of outputs!



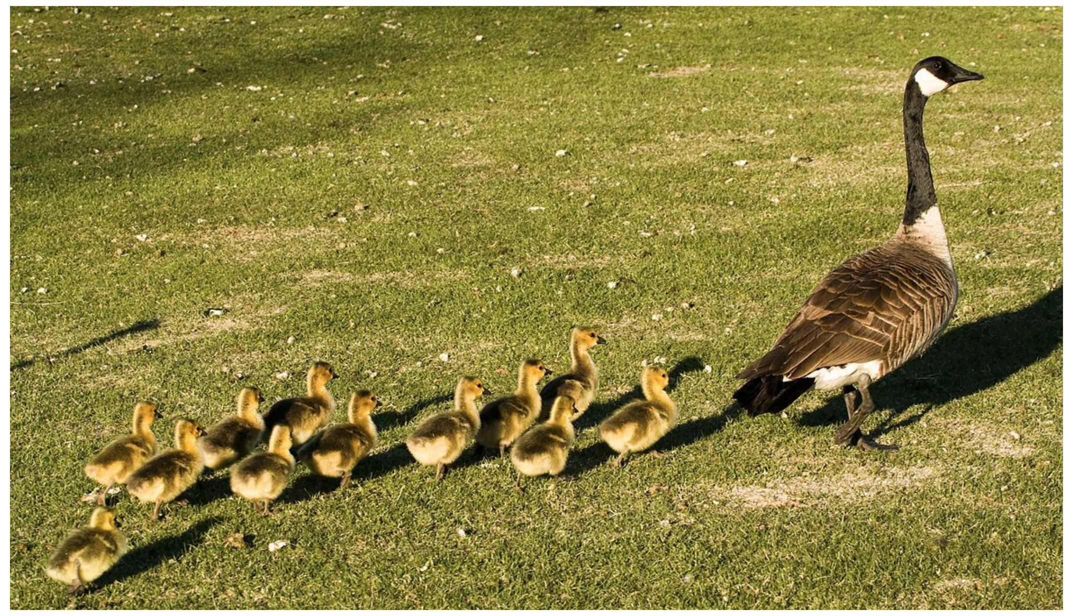
CAT: (x, y, w, h)

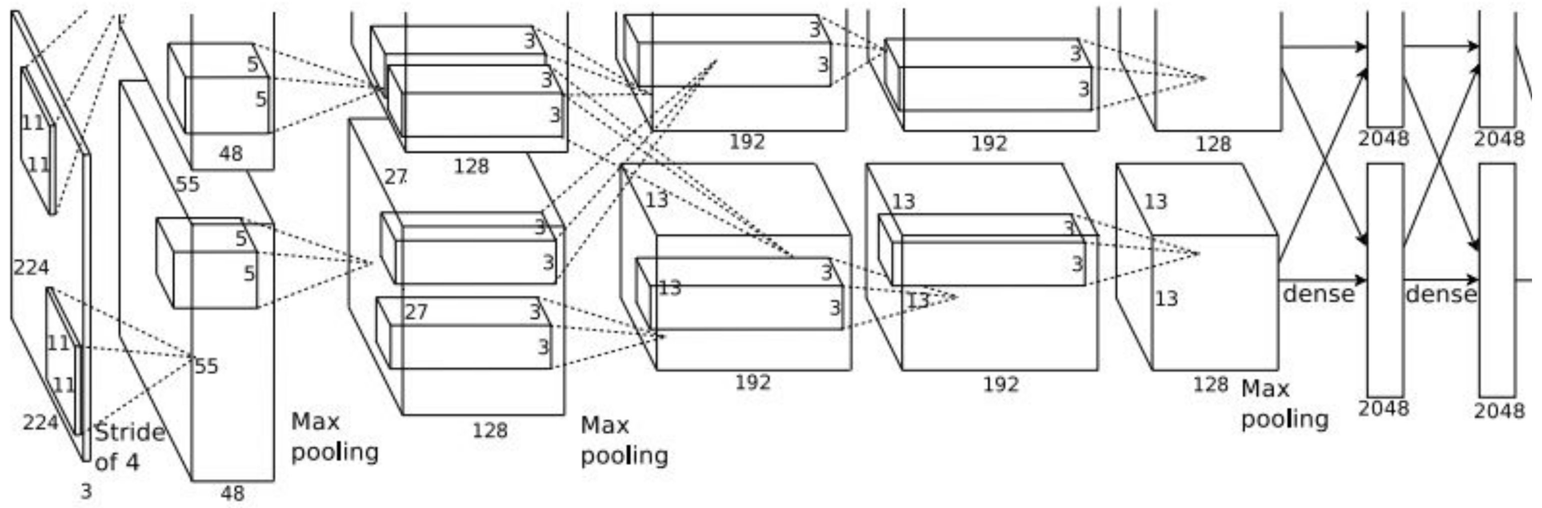
4 numbers



DOG: (x, y, w, h)
DOG: (x, y, w, h)
CAT: (x, y, w, h)

12 numbers





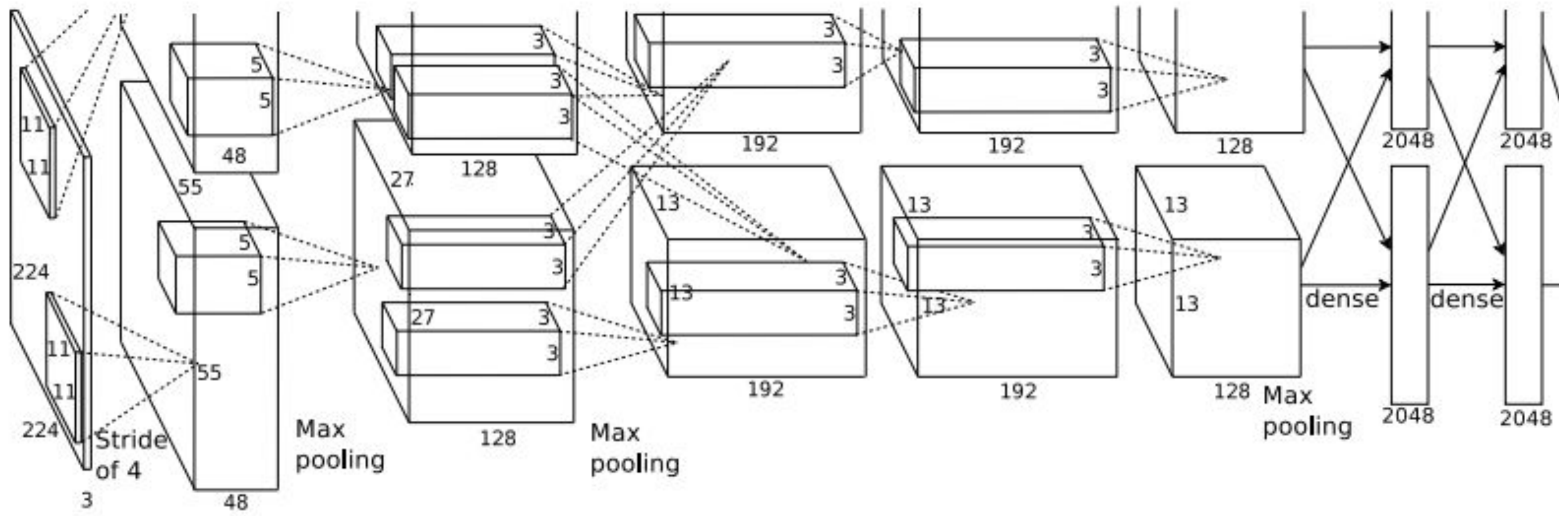
DUCK: (x, y, w, h)
DUCK: (x, y, w, h)
....

Many numbers!

What if we tried to detect a SINGLE object in a PATCH?

Object Detection: Multiple Objects

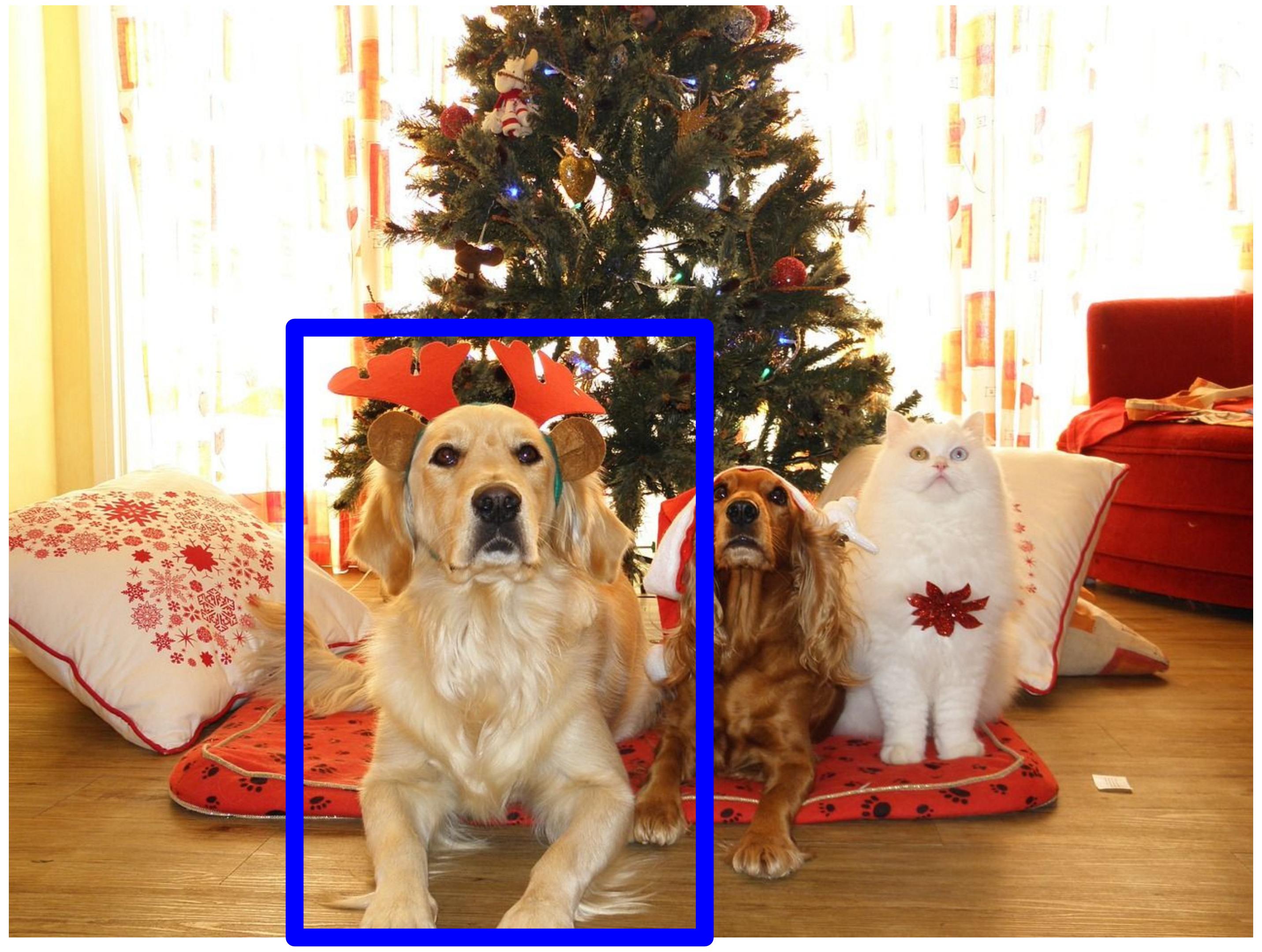
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background

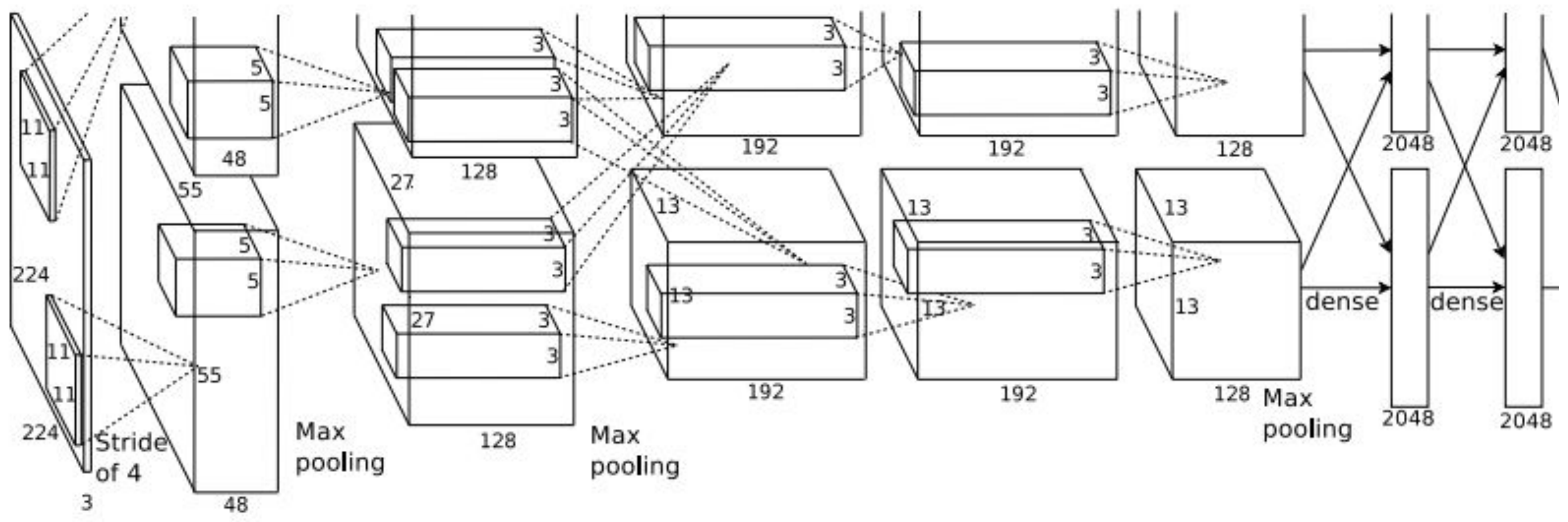


Dog? NO
Cat? NO
Background? YES

Object Detection: Multiple Objects

Apply a CNN to many different crops of the image, CNN classifies each crop as object or background

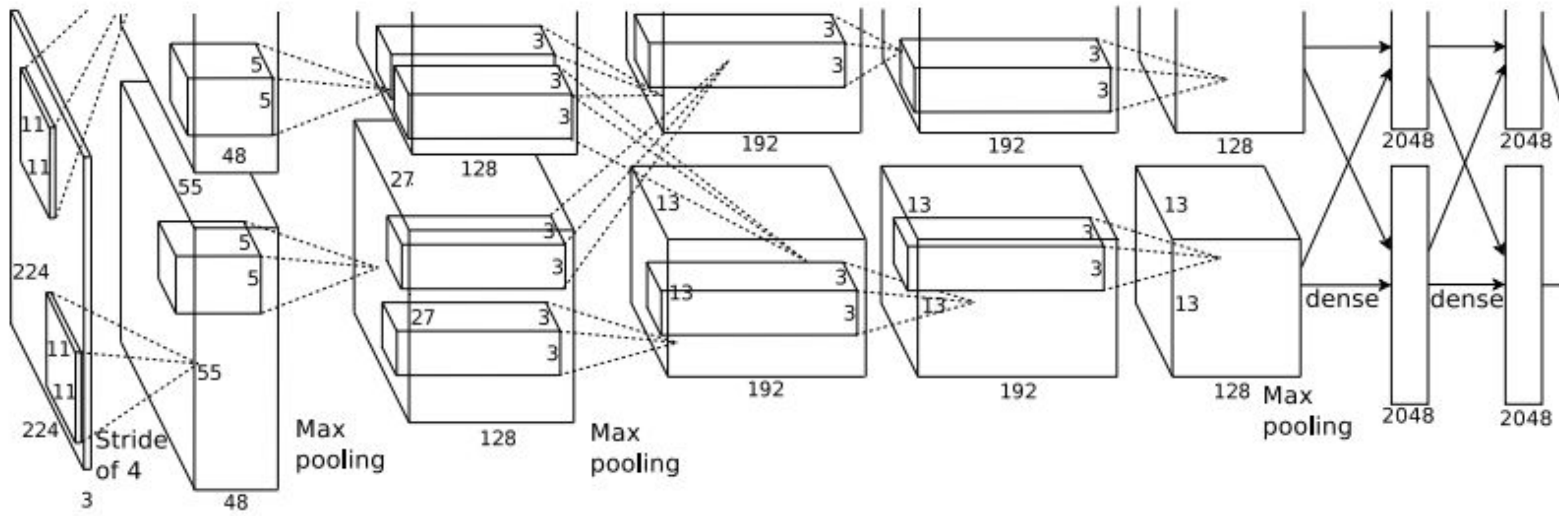




Dog? YES
Cat? NO
Background? NO

Object Detection: Multiple Objects

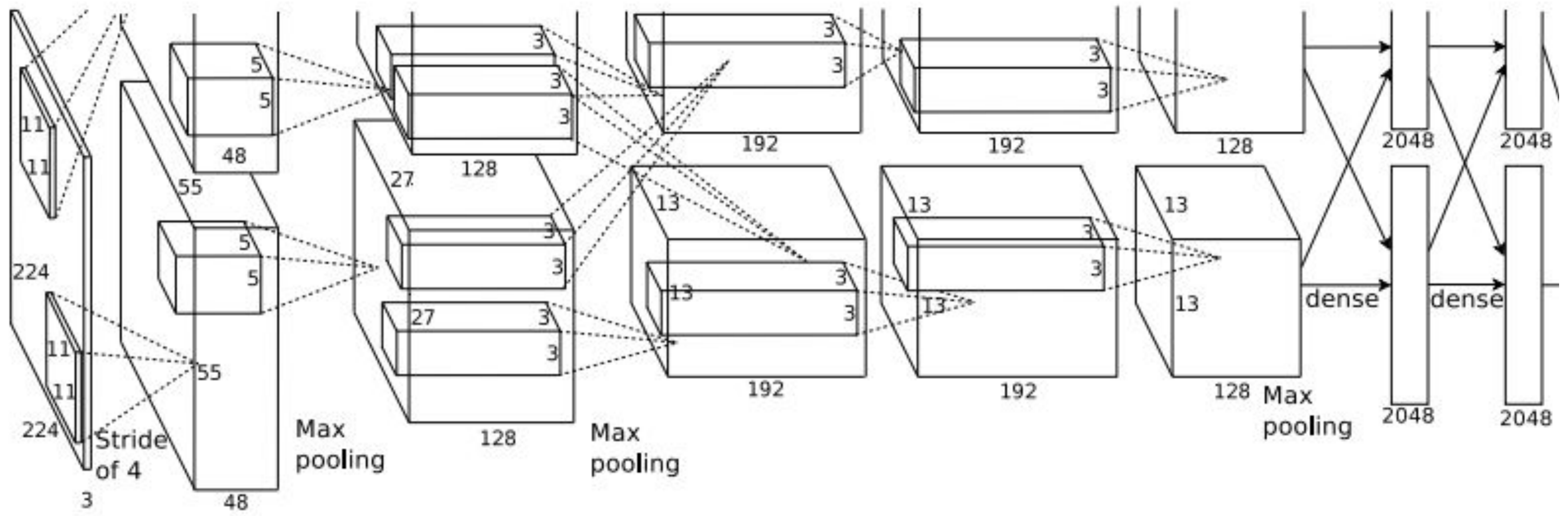
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background



Dog? YES
Cat? NO
Background? NO

Object Detection: Multiple Objects

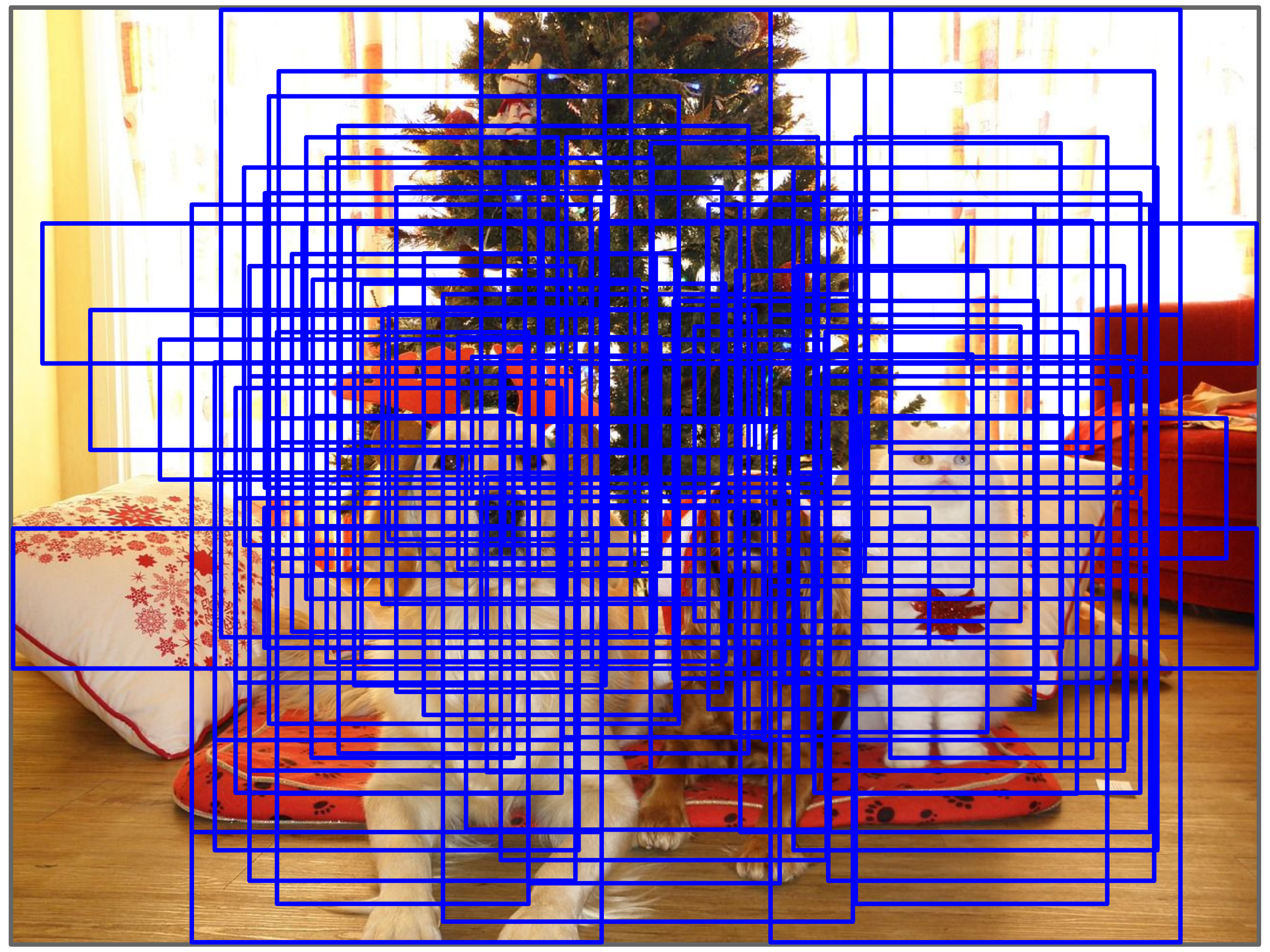
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background



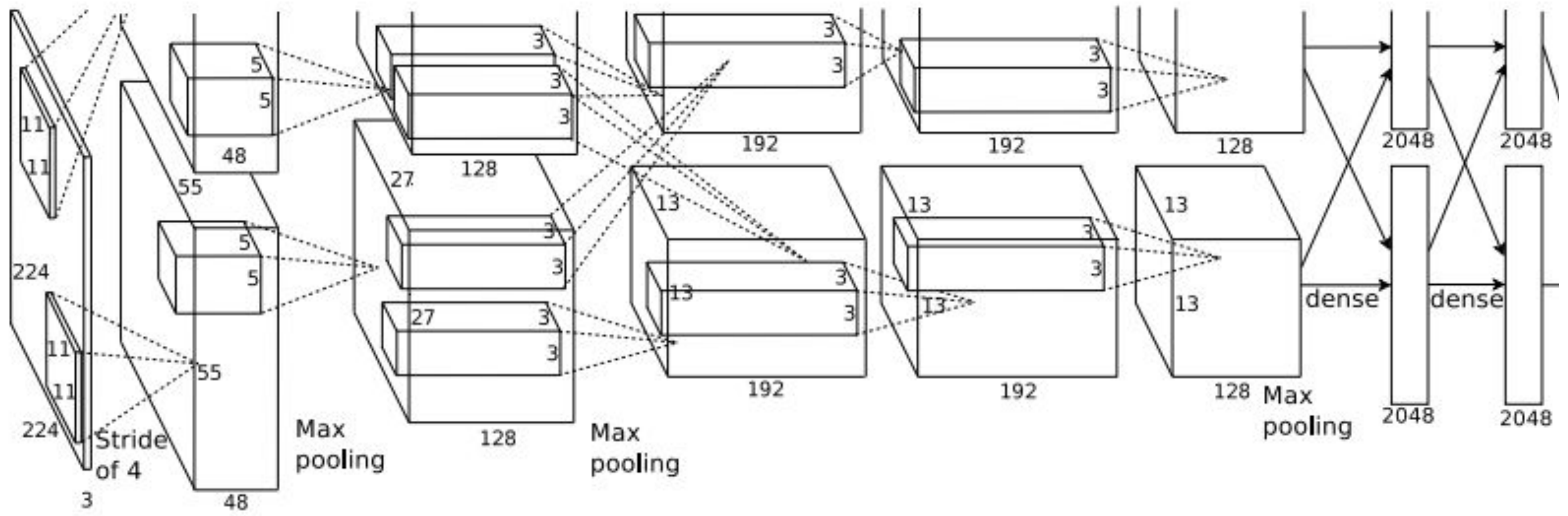
Dog? NO
Cat? YES
Background? NO

Q: What's the problem with this approach?

Object Detection: Multiple Objects



Apply a CNN to many different crops of the image, CNN classifies each crop as object or background



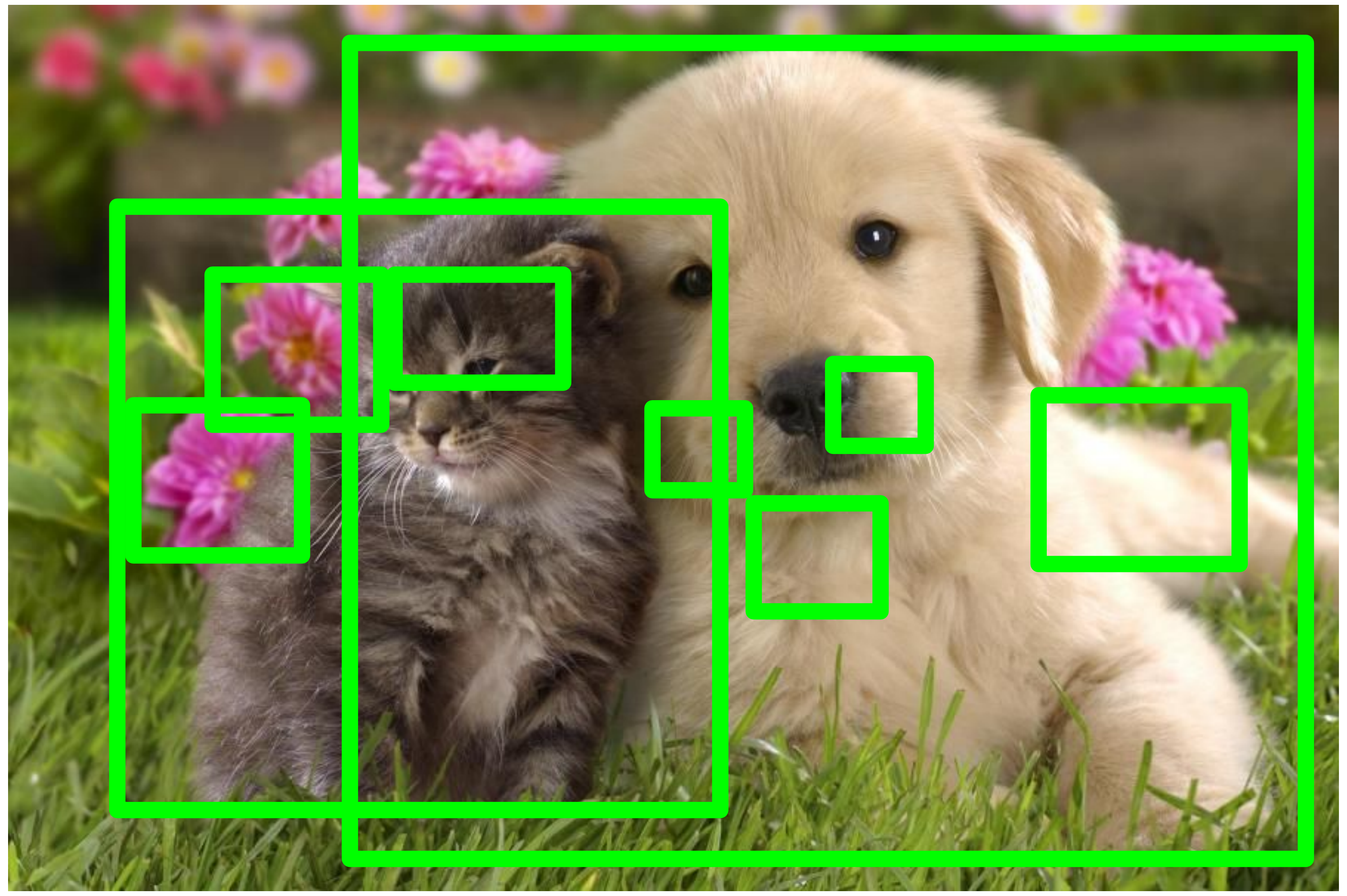
Dog? NO
Cat? YES
Background? NO

Problem: Need to apply CNN to huge number of locations, scales, and aspect ratios, very computationally expensive!

What if we had a
SMART patch proposer?

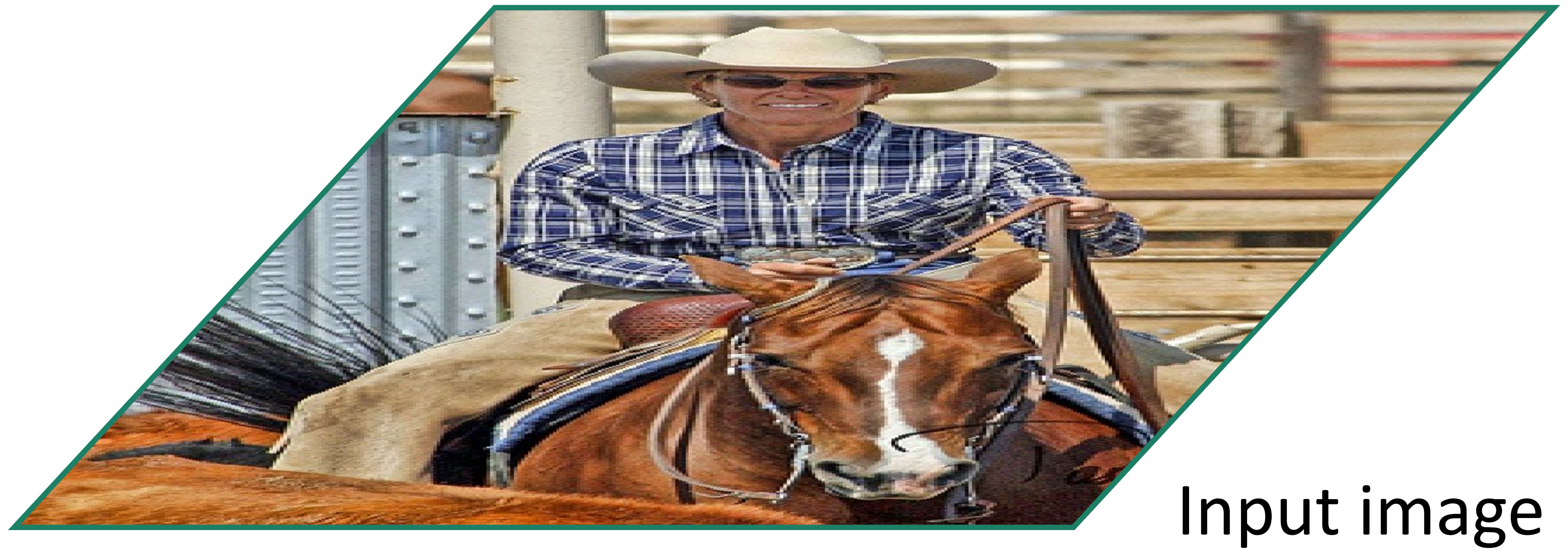
Region Proposals: Selective Search

- Find "blobby" image regions that are likely to contain objects
- Relatively fast to run; e.g. Selective Search gives 2000 region proposals in a few seconds on CPU



Alexe et al, "Measuring the objectness of image windows", TPAMI 2012
Uijlings et al, "Selective Search for Object Recognition", IJCV 2013
Cheng et al, "BING: Binarized normed gradients for objectness estimation at 300fps", CVPR 2014
Zitnick and Dollar, "Edge boxes: Locating object proposals from edges", ECCV 2014

R-CNN



Input image

Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.

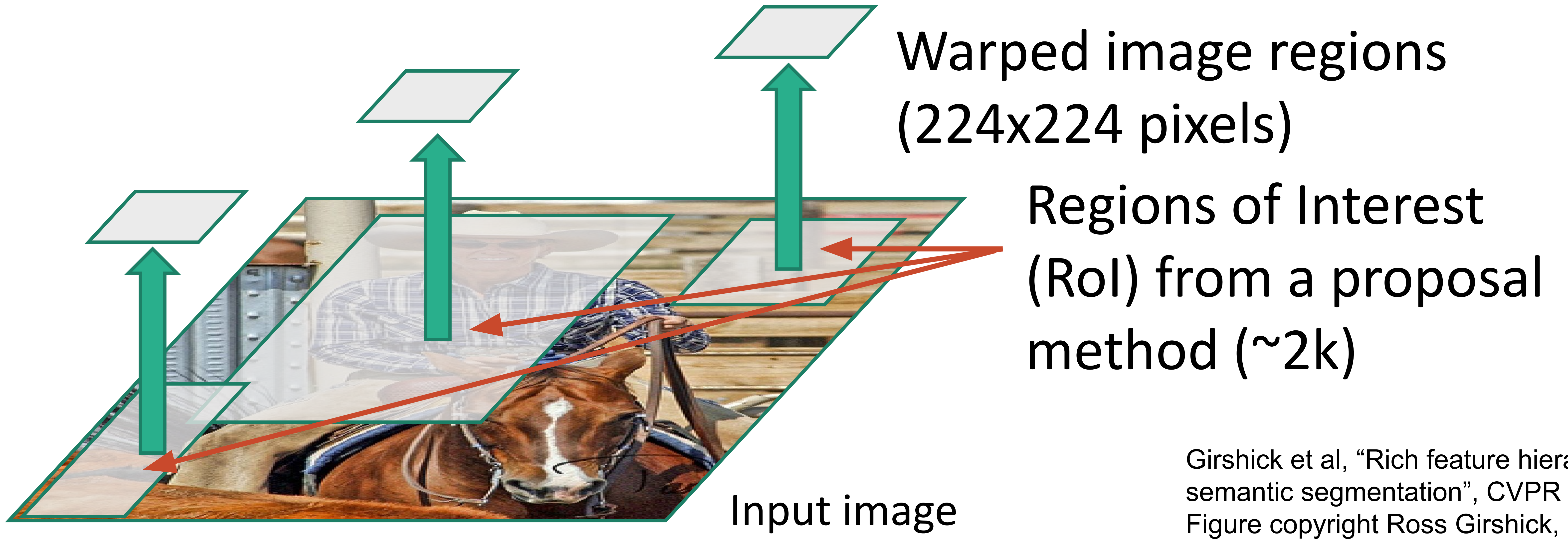
R-CNN

Regions of Interest (RoI) from a proposal method (~2k)

Input image

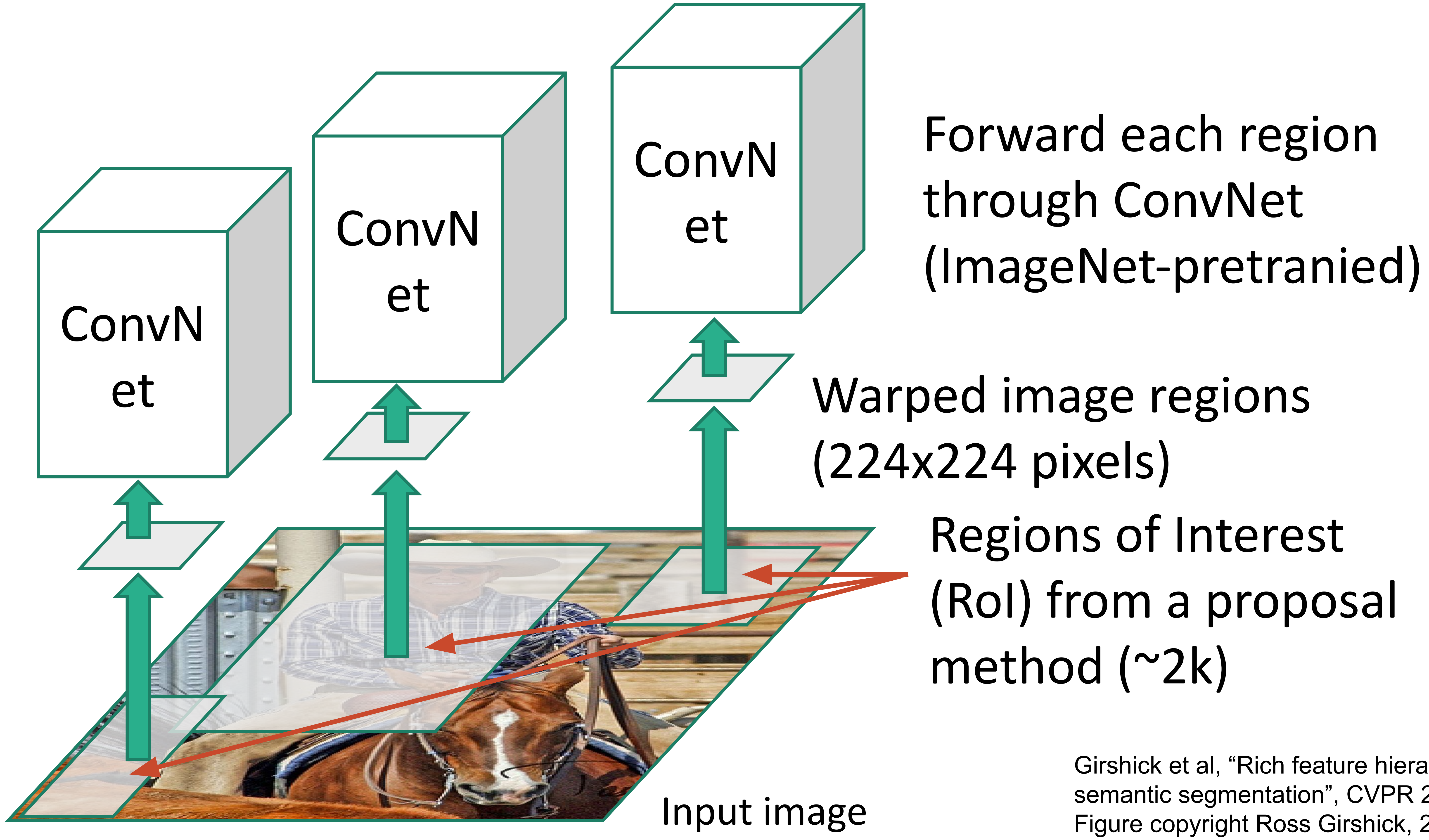
Girshick et al, “Rich feature hierarchies for accurate object detection and semantic segmentation”, CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

R-CNN



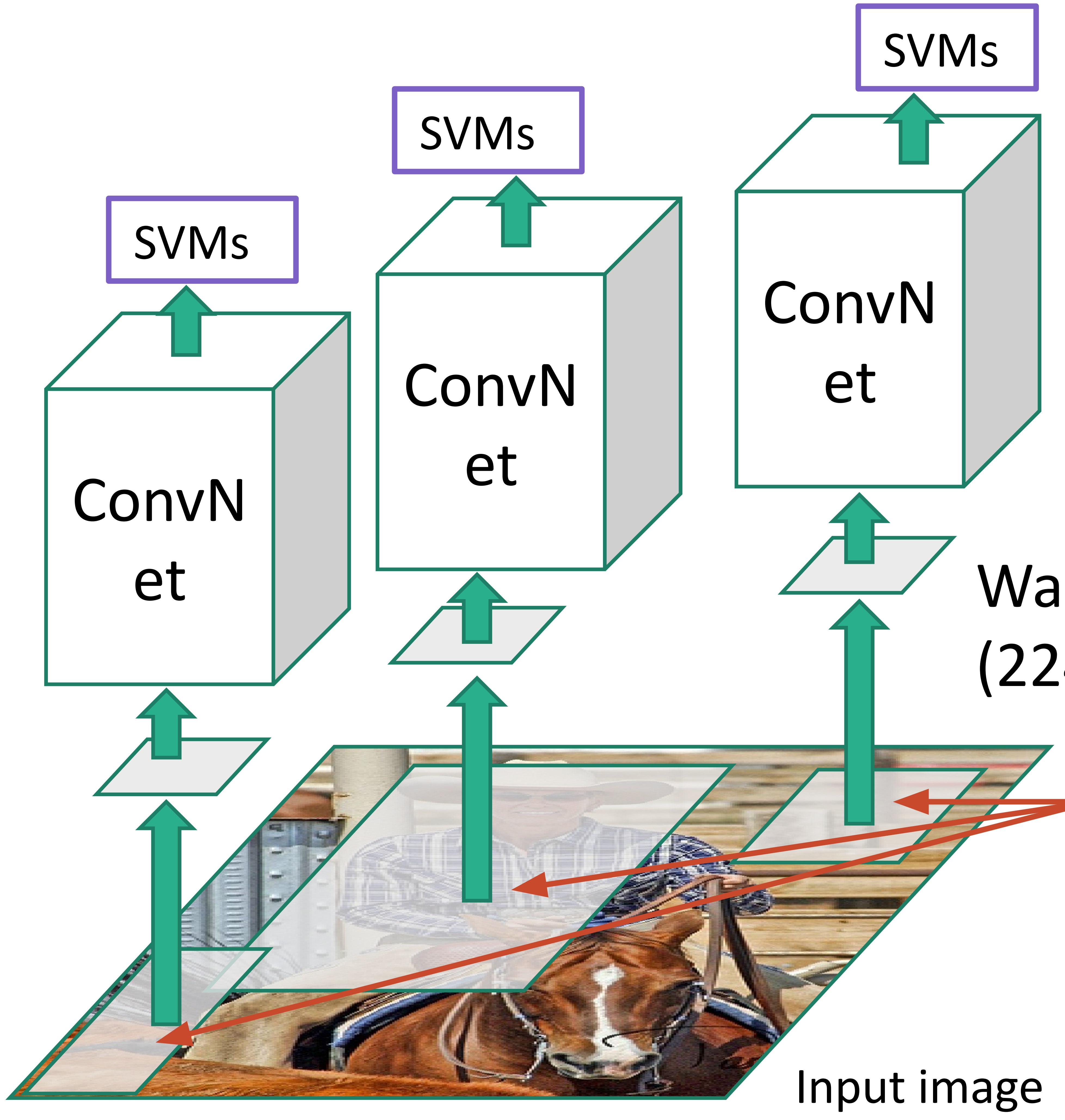
Girshick et al, “Rich feature hierarchies for accurate object detection and semantic segmentation”, CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

R-CNN



Girshick et al, “Rich feature hierarchies for accurate object detection and semantic segmentation”, CVPR 2014.

R-CNN



Classify regions with SVMs

Forward each region through ConvNet (ImageNet-pretranied)

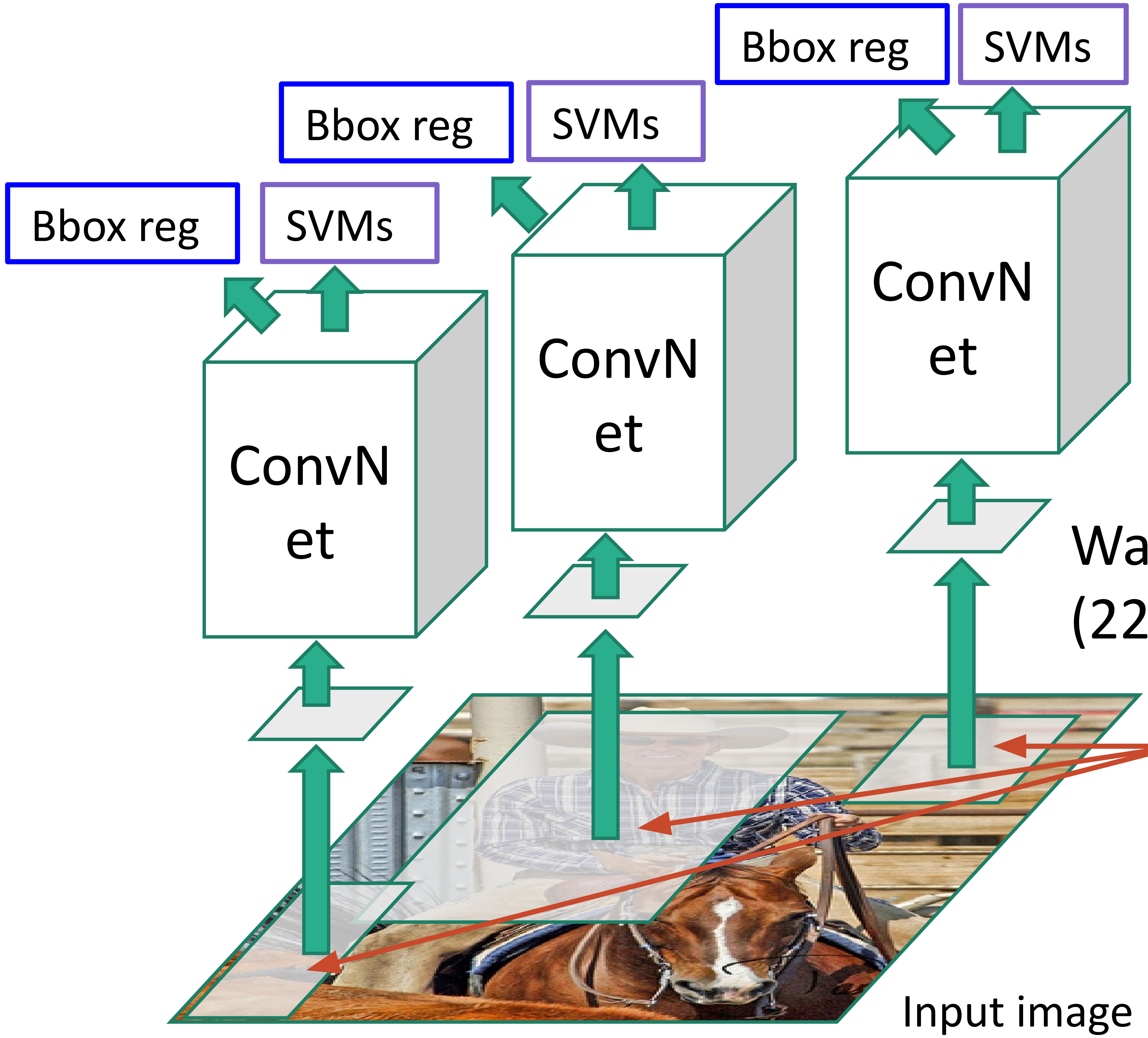
Warped image regions (224x224 pixels)

Regions of Interest (RoI) from a proposal method (~2k)

Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.

R-CNN

Predict “corrections” to the RoI: 4 numbers: (dx, dy, dw, dh)



Classify regions with SVMs

Forward each region through ConvNet (ImageNet-pretranied)

Warped image regions (224x224 pixels)

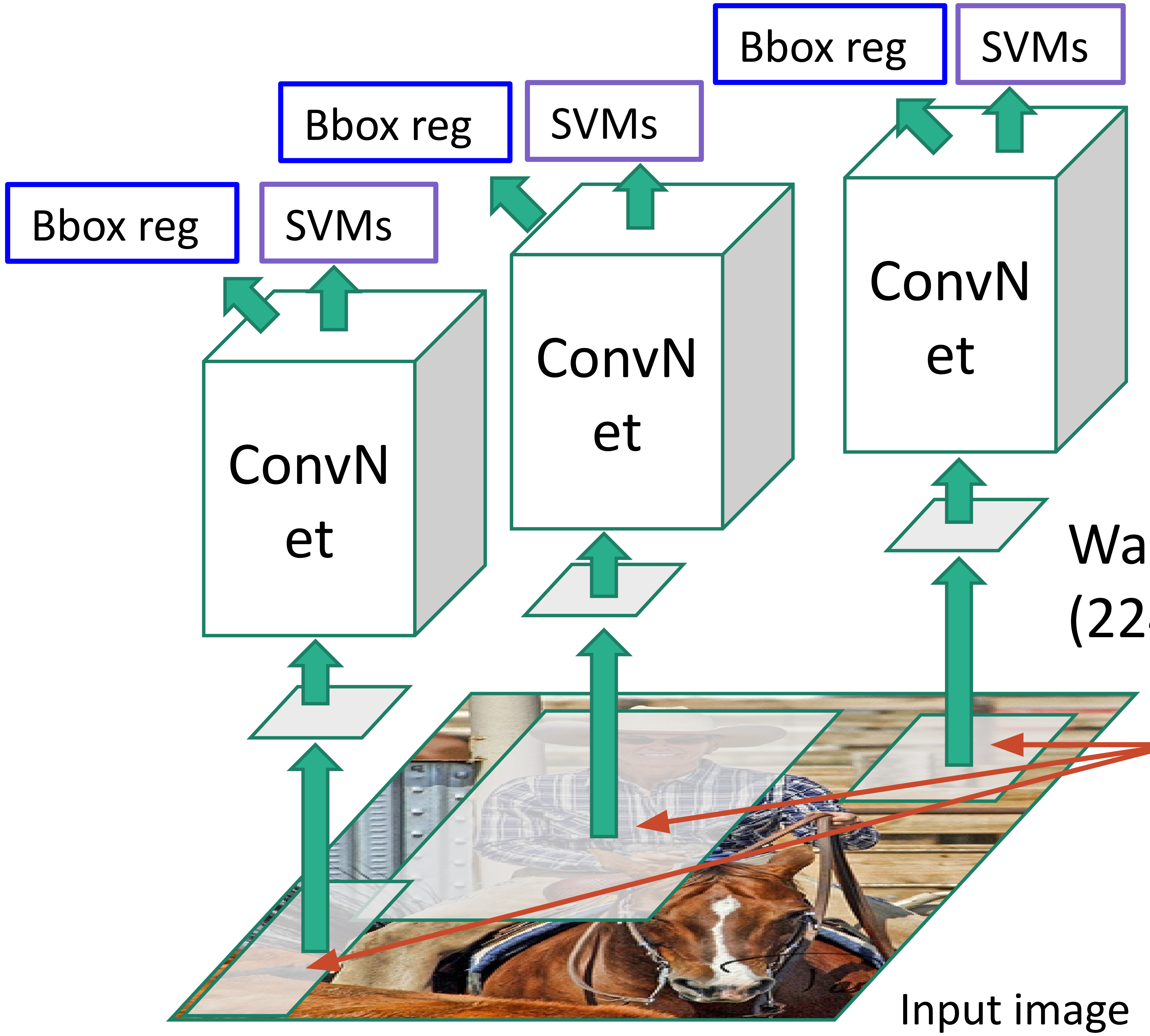
Regions of Interest (RoI) from a proposal method (~2k)

Girshick et al, “Rich feature hierarchies for accurate object detection and semantic segmentation”, CVPR 2014.

Isn't calling a CNN for each patch super duper slow?

"Slow" R-CNN

Predict "corrections" to the RoI: 4 numbers: (dx, dy, dw, dh)



Classify regions with SVMs

Forward each region through ConvNet

Warped image regions (224x224 pixels)

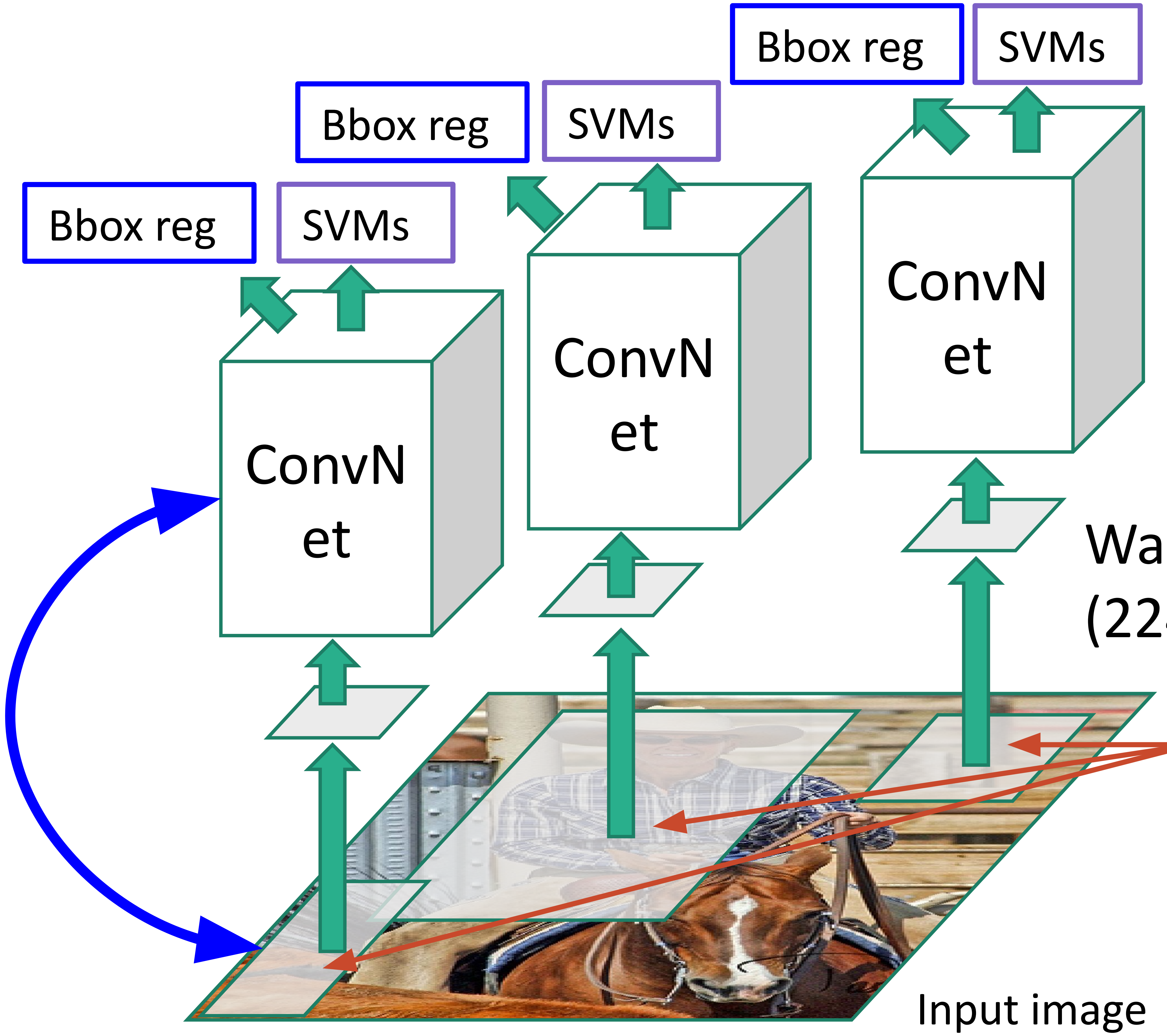
Regions of Interest (RoI) from a proposal method (~2k)

Problem: Very slow! Need to do ~2k independent forward passes for each image!

Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.

"Slow" R-CNN

Predict "corrections" to the RoI: 4 numbers: (dx, dy, dw, dh)



Classify regions with SVMs

Forward each region through ConvNet

Warped image regions (224x224 pixels)

Regions of Interest (RoI) from a proposal method (~2k)

Problem: Very slow! Need to do ~2k independent forward passes for each image!

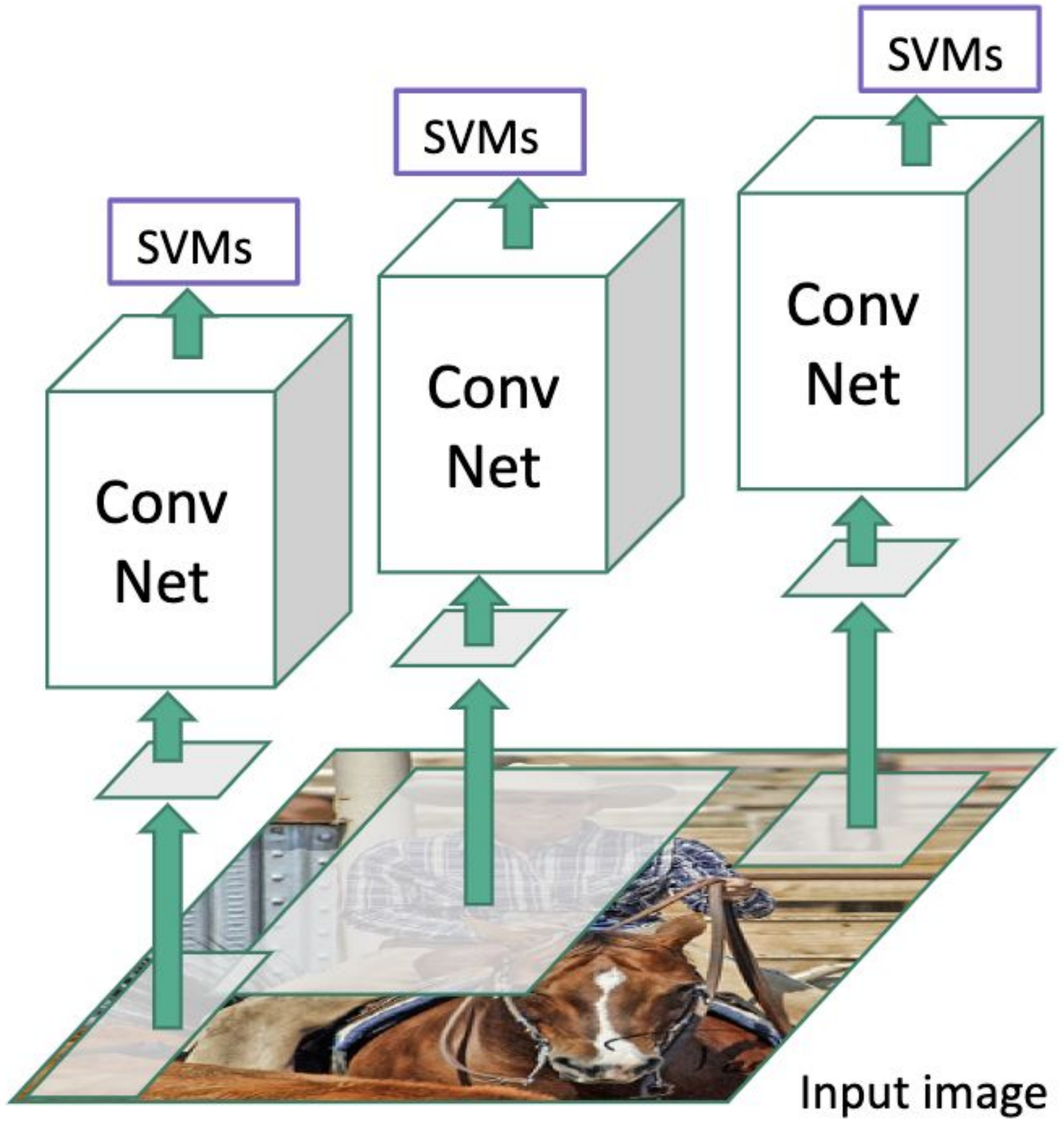
Idea: Pass the image through convnet before cropping! Crop the conv feature instead!

Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.

Instead of running N ConvNets, run just ONE!

Fast R-CNN

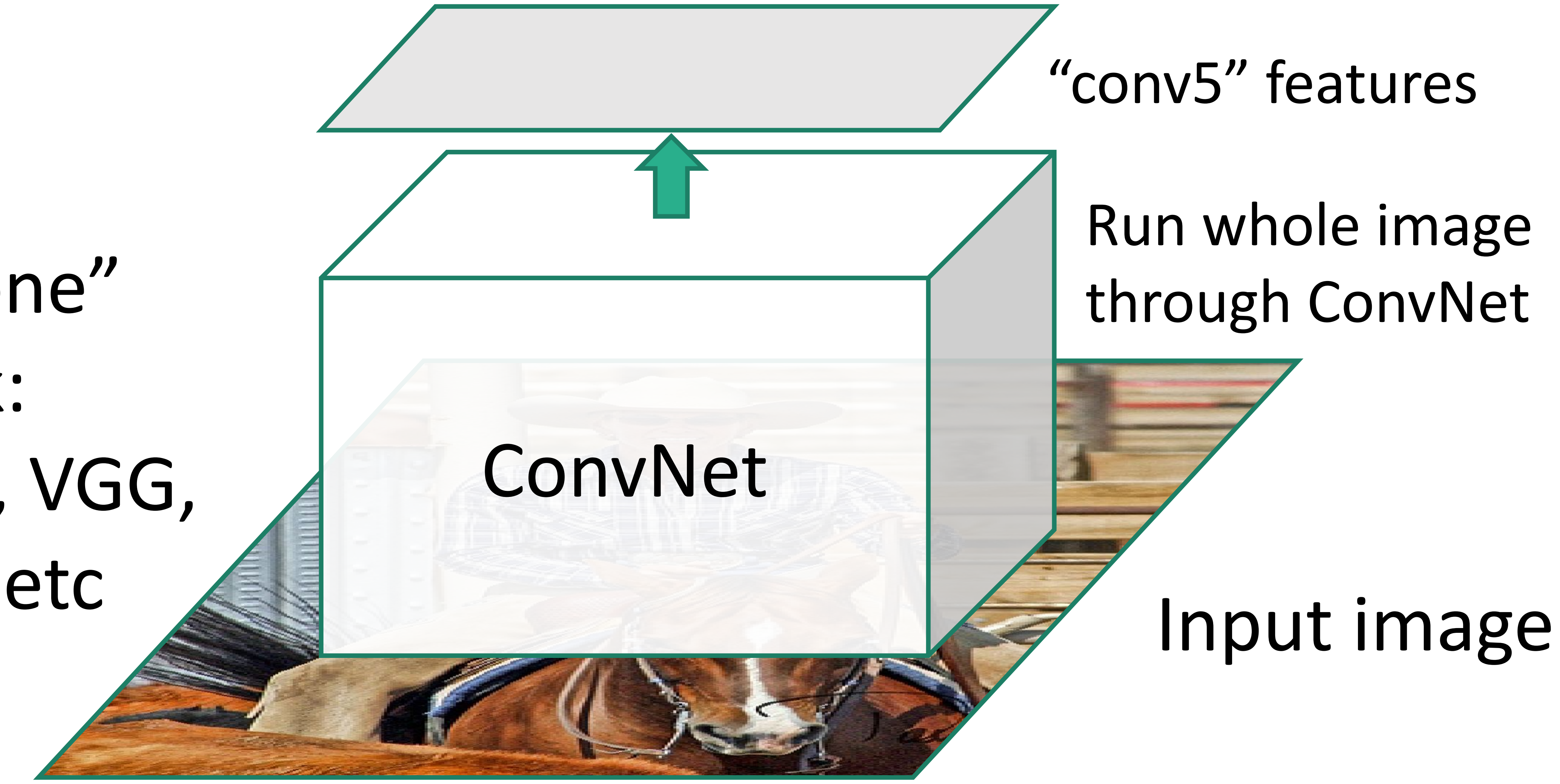
"Slow" R-CNN



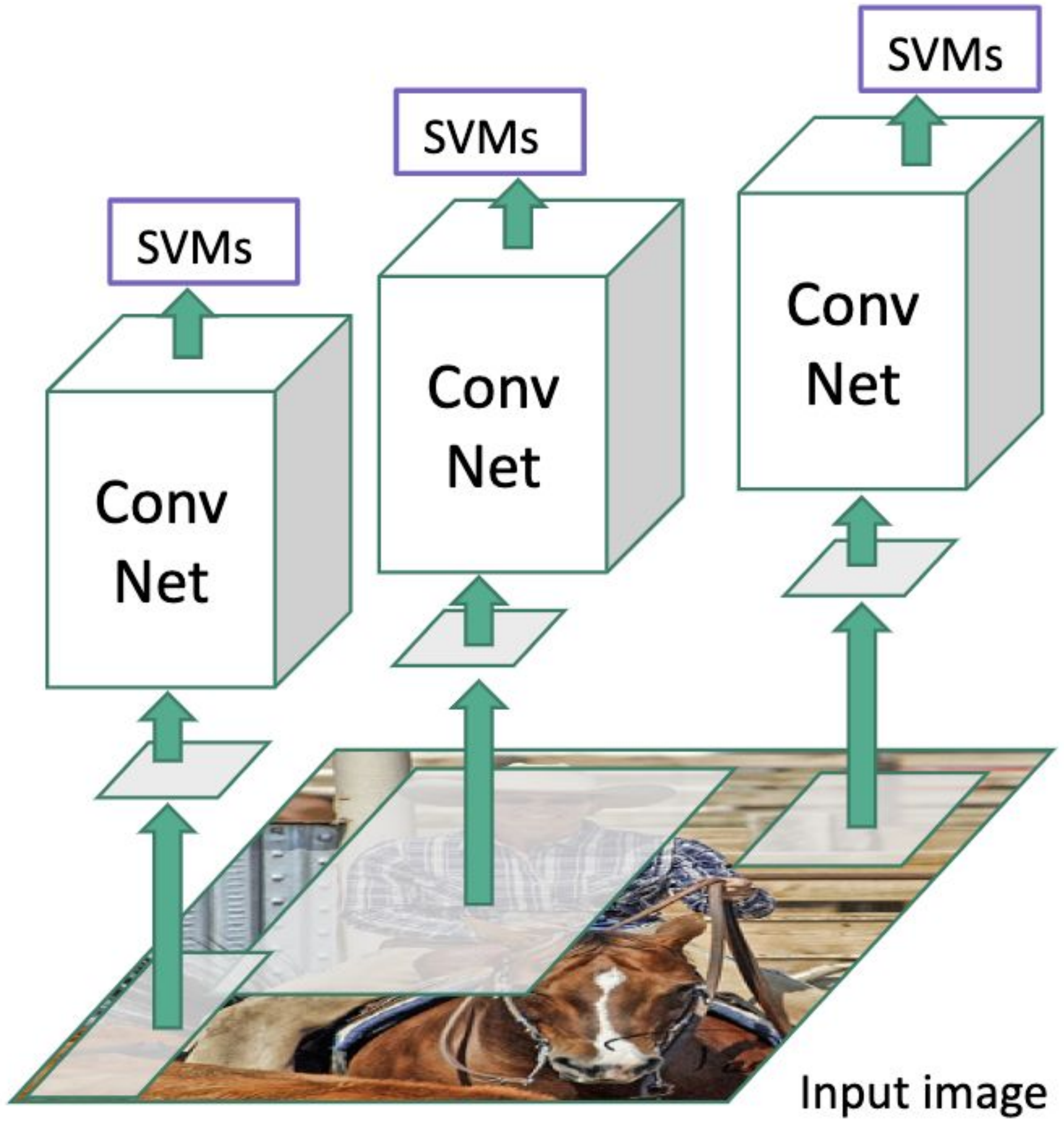
Girshick, "Fast R-CNN", ICCV 2015. Figure copyright Ross Girshick, 2015; [source](). Reproduced with permission.

Fast R-CNN

"Backbone" network: AlexNet, VGG, ResNet, etc



"Slow" R-CNN

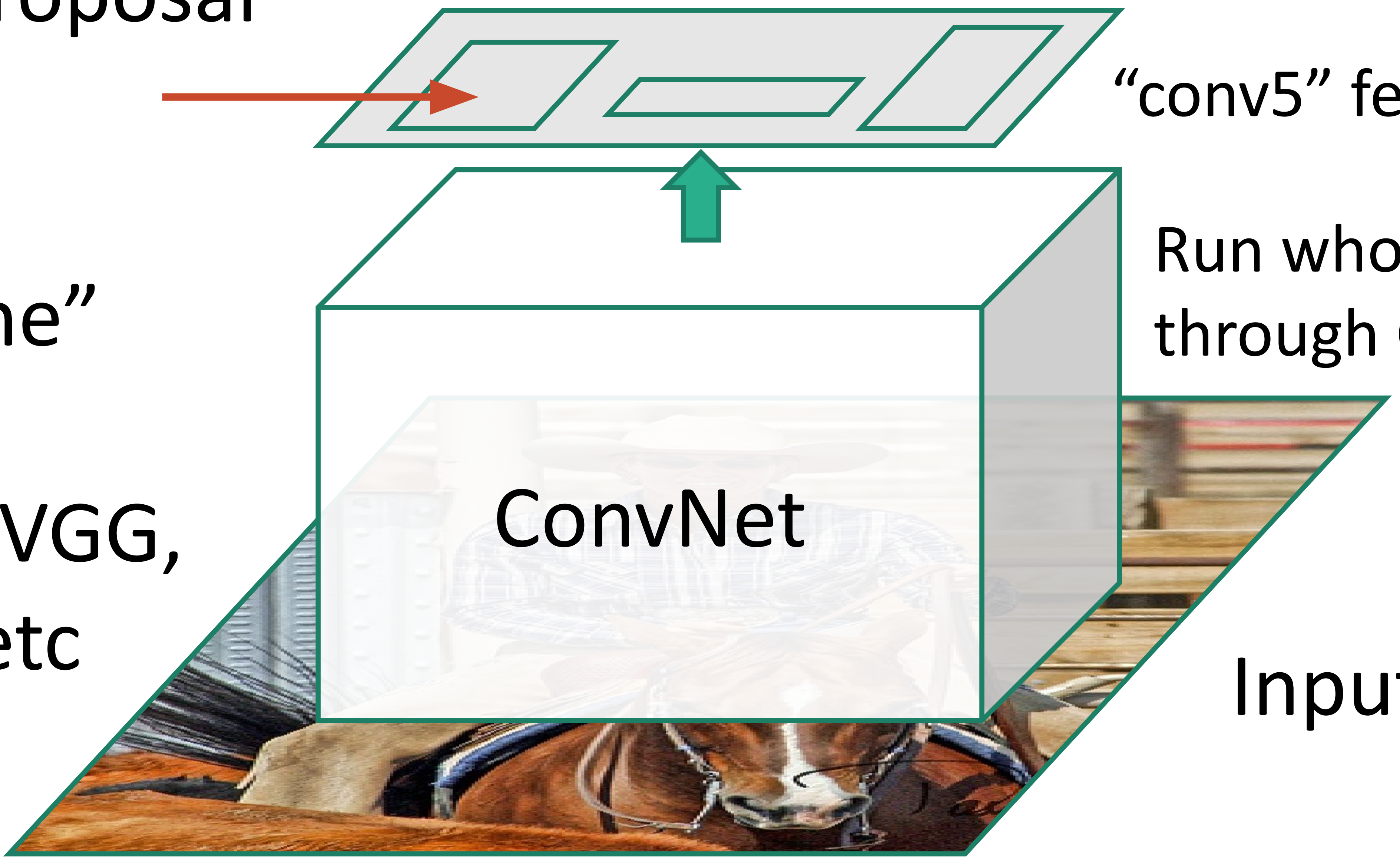


Girshick, "Fast R-CNN", ICCV 2015. Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

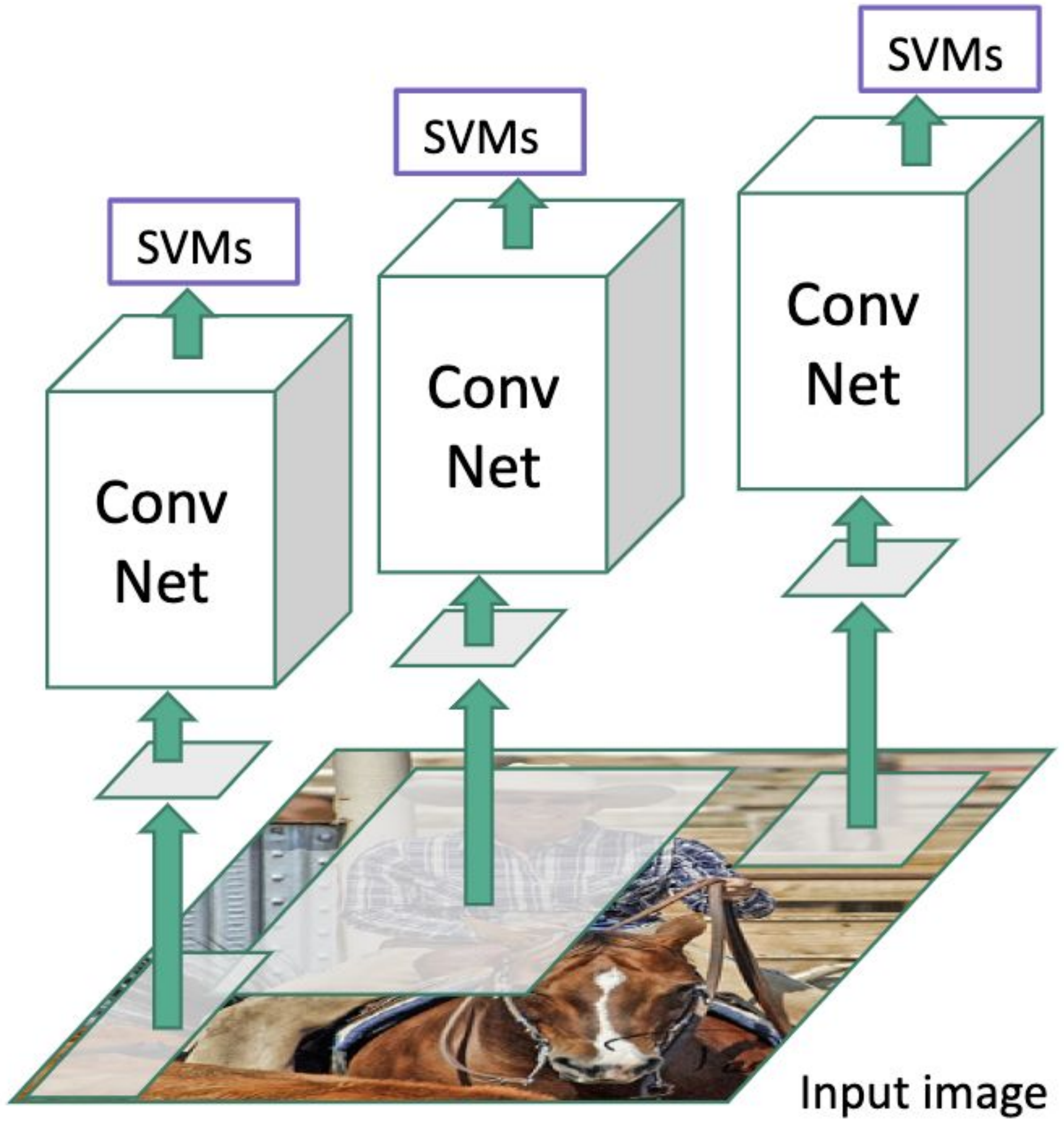
Fast R-CNN

Regions of Interest (RoIs) from a proposal method

"Backbone" network: AlexNet, VGG, ResNet, etc



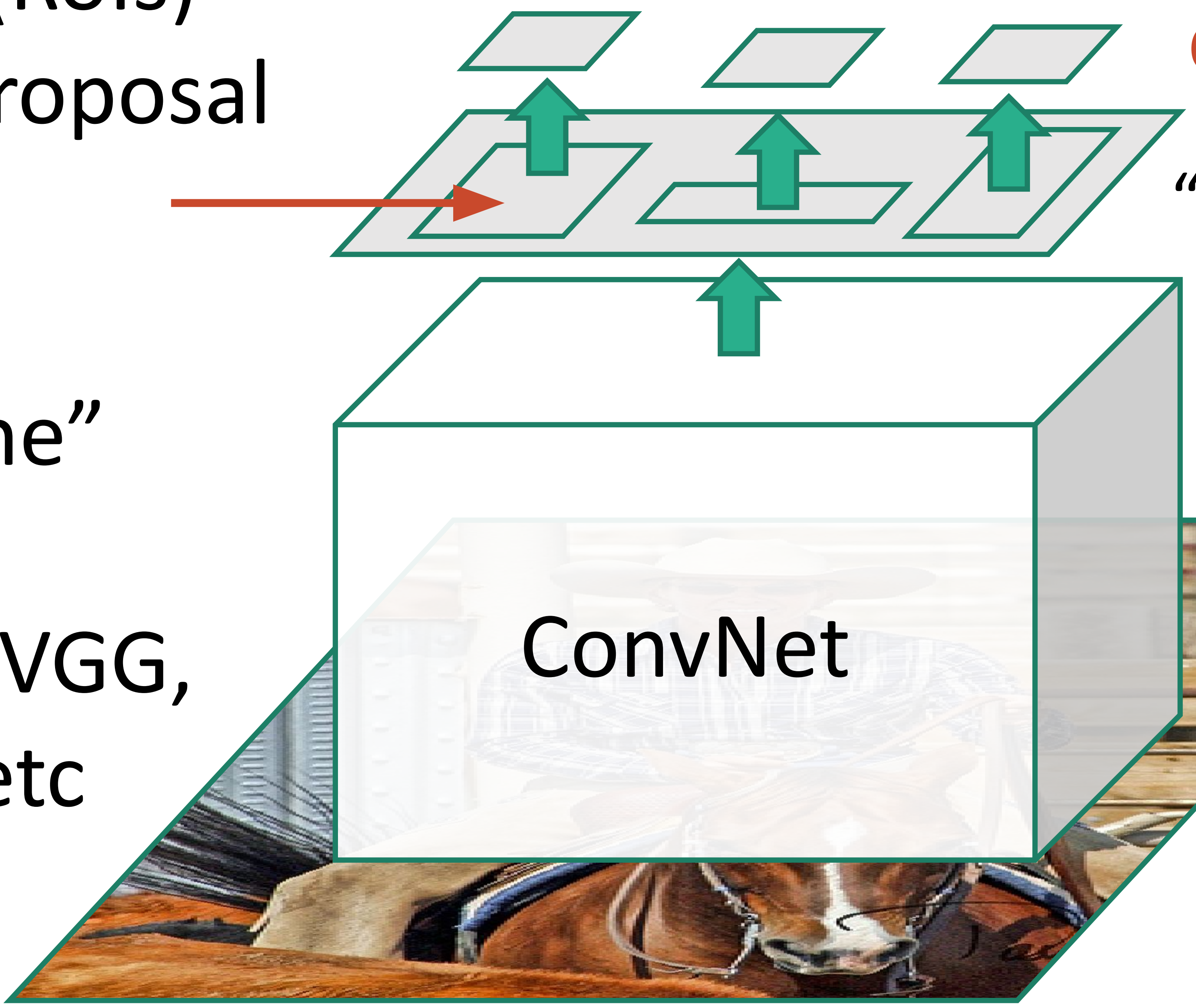
"Slow" R-CNN



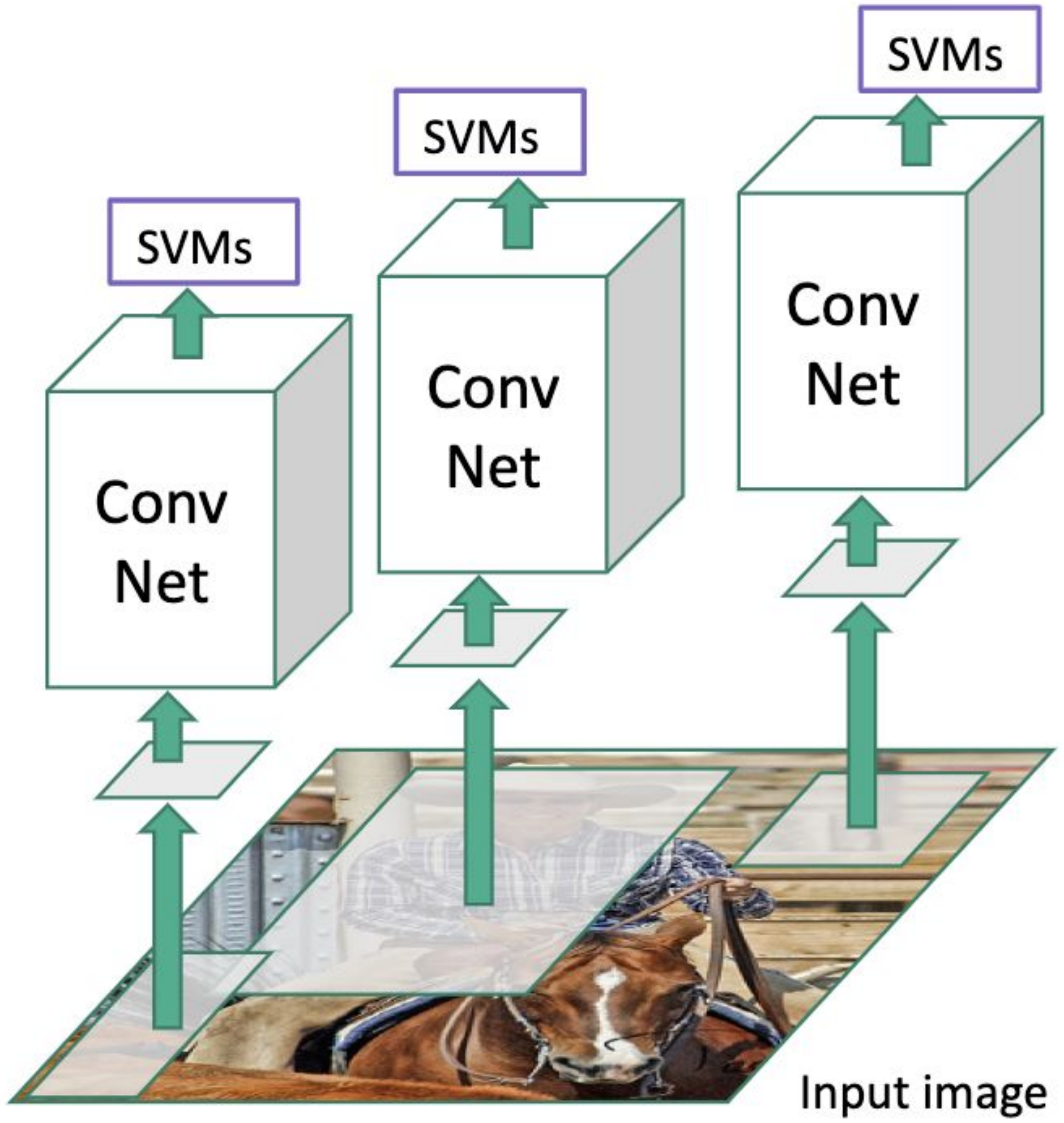
Fast R-CNN

Regions of Interest (RoIs) from a proposal method

"Backbone" network: AlexNet, VGG, ResNet, etc

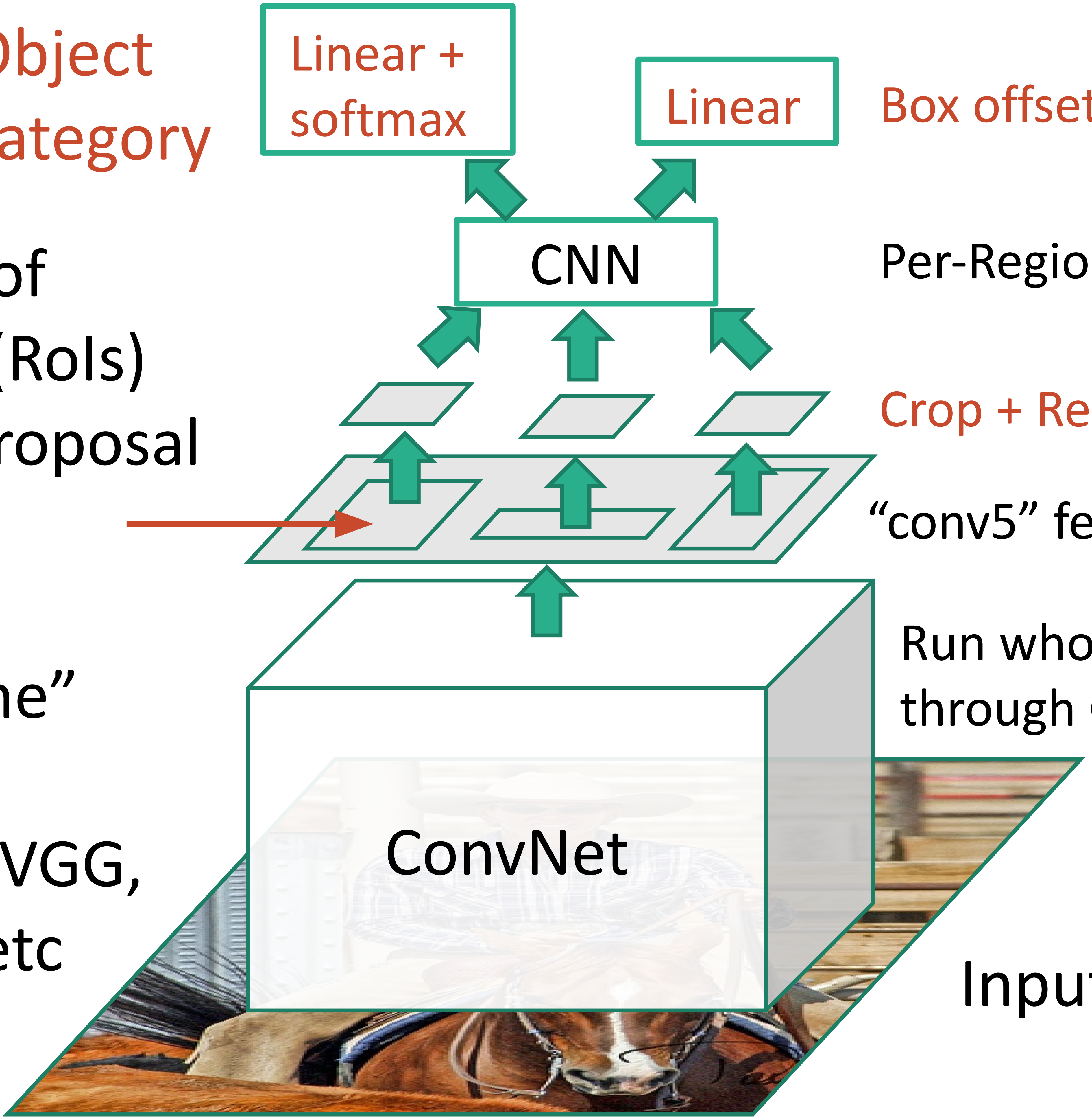


"Slow" R-CNN



Girshick, "Fast R-CNN", ICCV 2015. Figure copyright Ross Girshick, 2015; [source](). Reproduced with permission.

Fast R-CNN

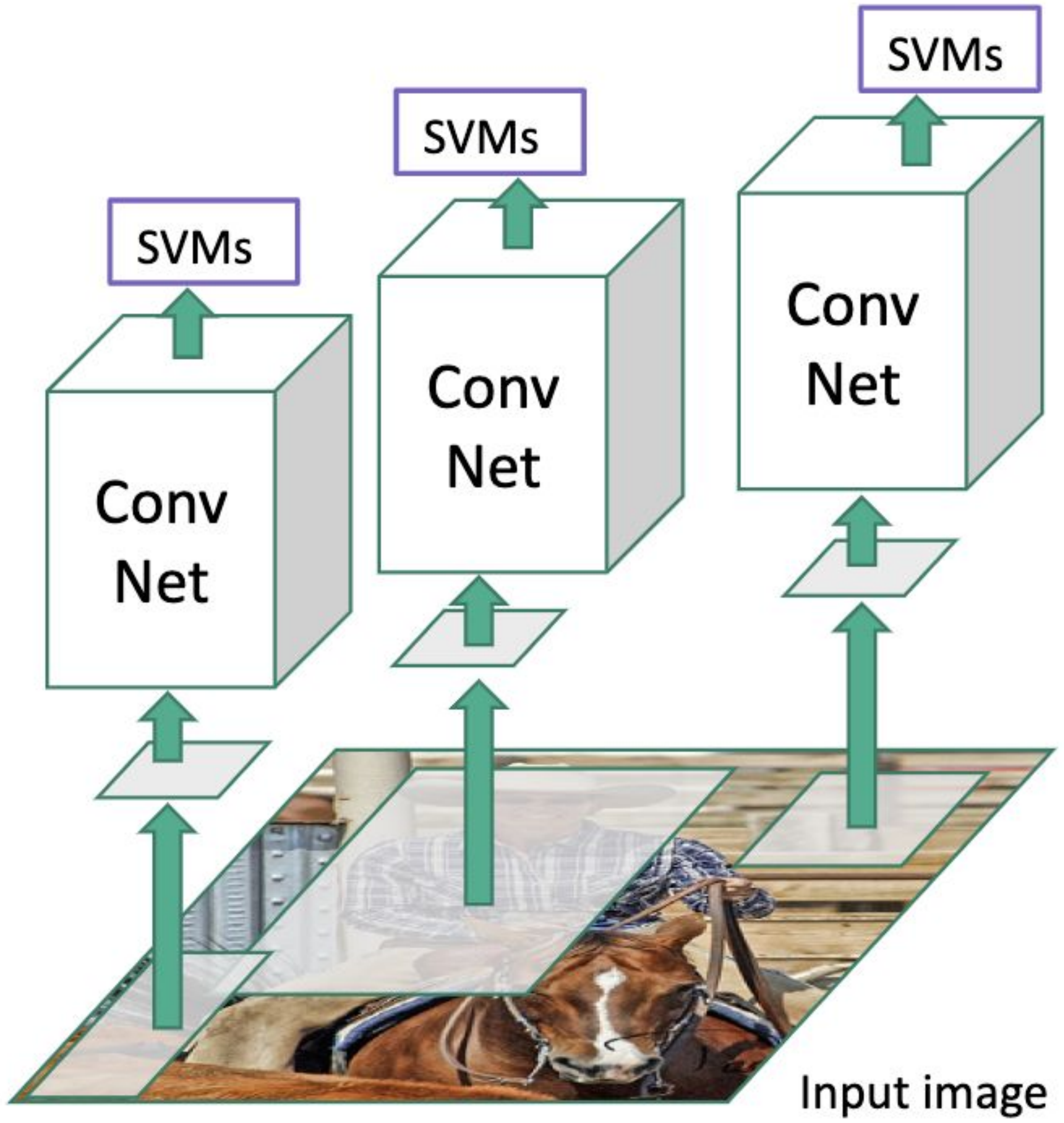


Regions of Interest (RoIs) from a proposal method

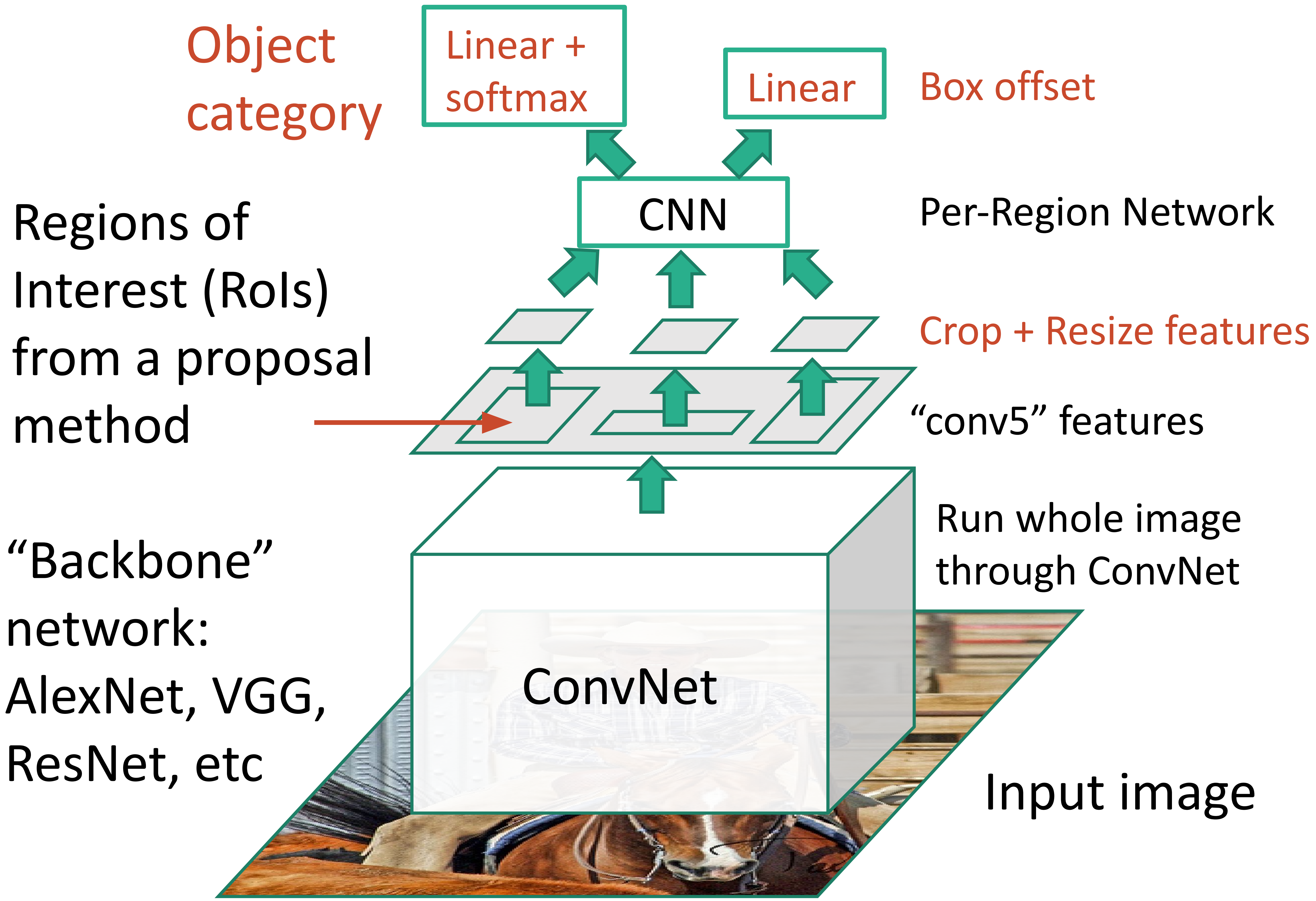
"Backbone" network: AlexNet, VGG, ResNet, etc

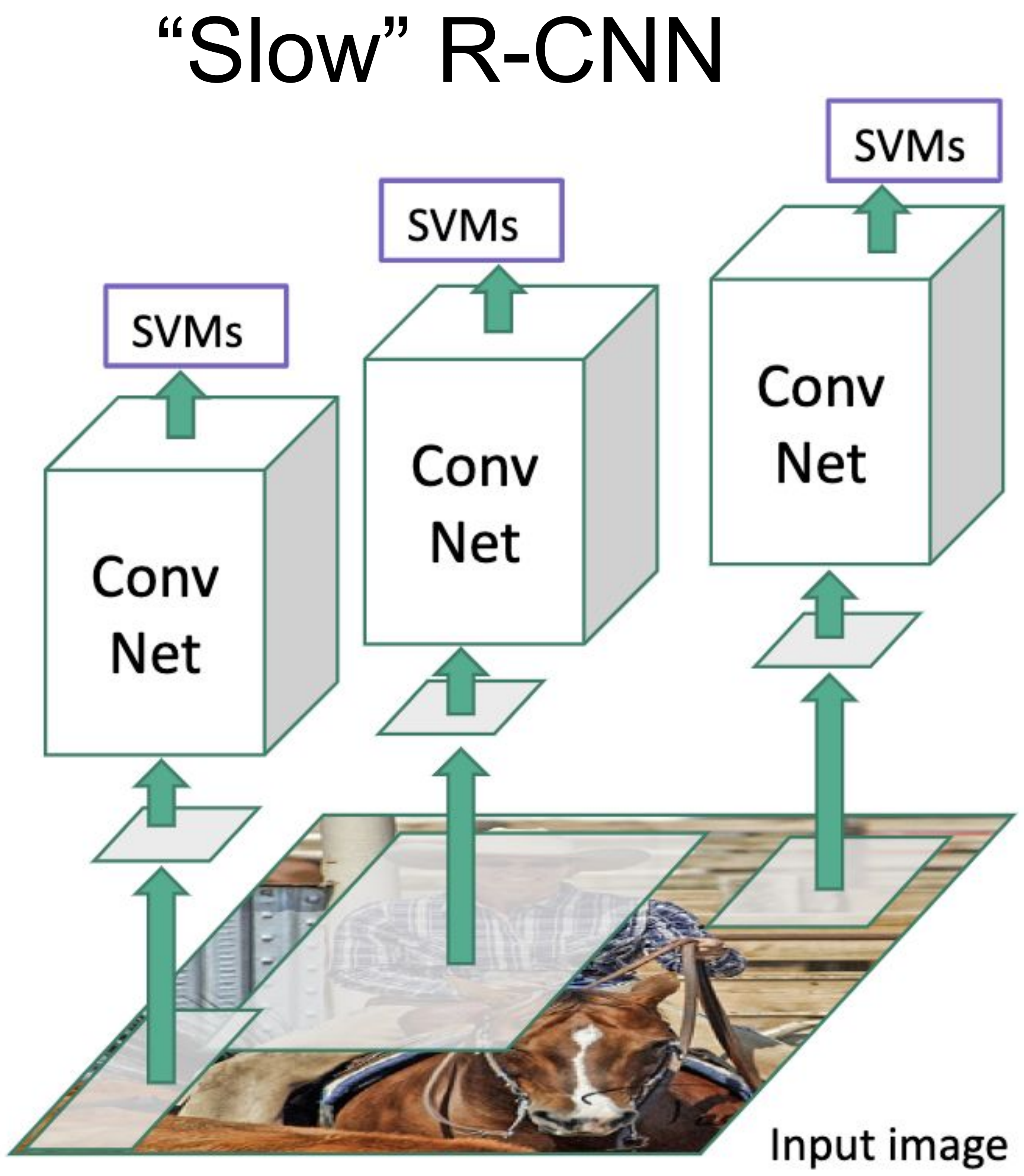
Girshick, "Fast R-CNN", ICCV 2015. Figure copyright Ross Girshick, 2015; [source](). Reproduced with permission.

"Slow" R-CNN

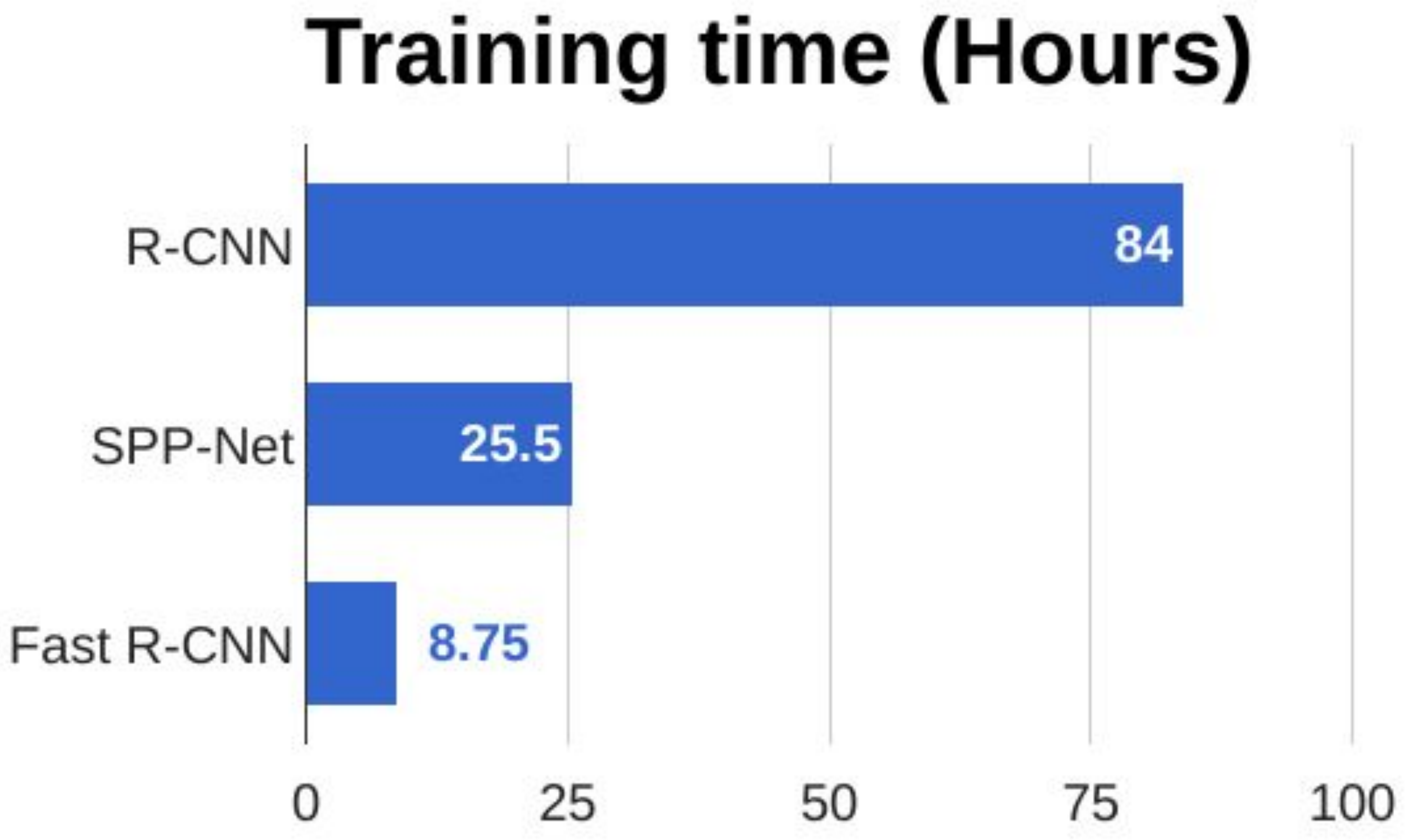


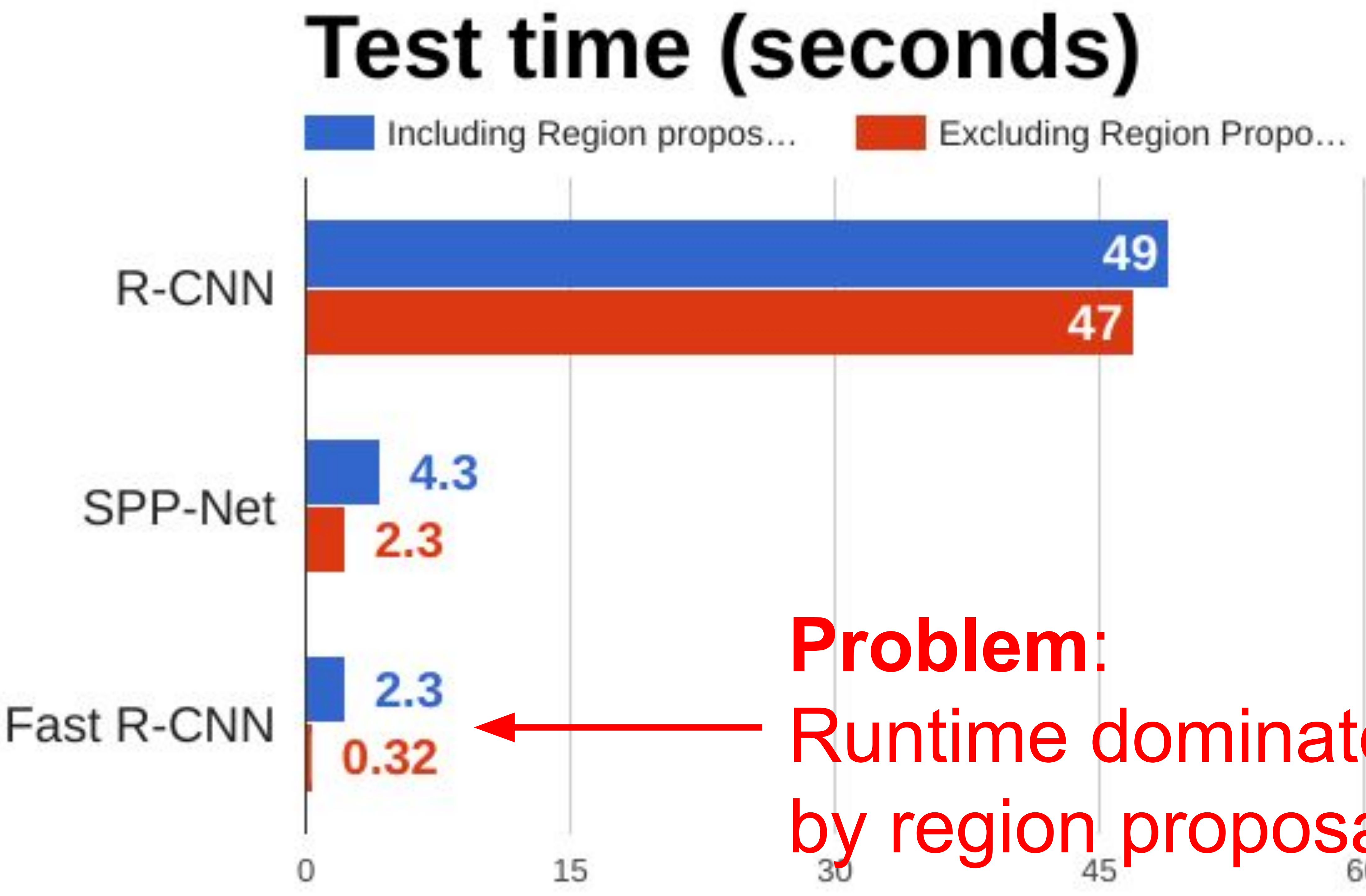
Fast R-CNN


Object category
Linear + softmax
Linear
Box offset
CNN
Per-Region Network
Regions of Interest (RoIs) from a proposal method
Crop + Resize features
"conv5" features
Run whole image through ConvNet
"Backbone" network: AlexNet, VGG, ResNet, etc
ConvNet
Input image


"Slow" R-CNN
SVMs
SVMs
SVMs
Conv Net
Conv Net
Conv Net
Input image

R-CNN vs Fast R-CNN





Problem:
Runtime dominated by region proposals!

Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
He et al, "Spatial pyramid pooling in deep convolutional networks for visual recognition", ECCV 2014
Girshick, "Fast R-CNN", ICCV 2015

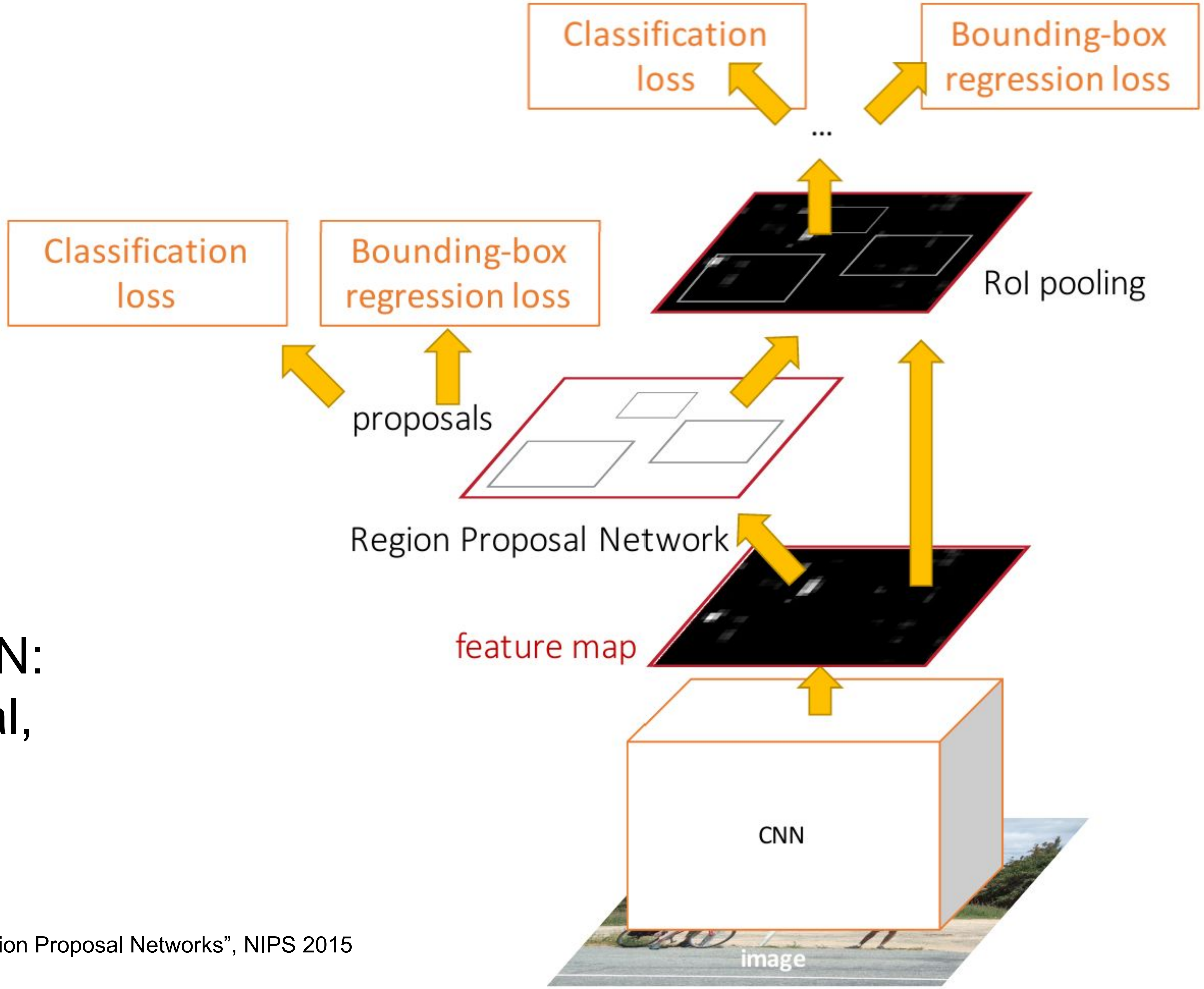
Can we get rid of the hacky region proposal algorithm?

Learn region proposal in an end to end manner!

Fast**er** R-CNN:
Make CNN do proposals!

Insert **Region Proposal Network (RPN)** to predict proposals from features

Otherwise same as Fast R-CNN: Crop features for each proposal, classify each one



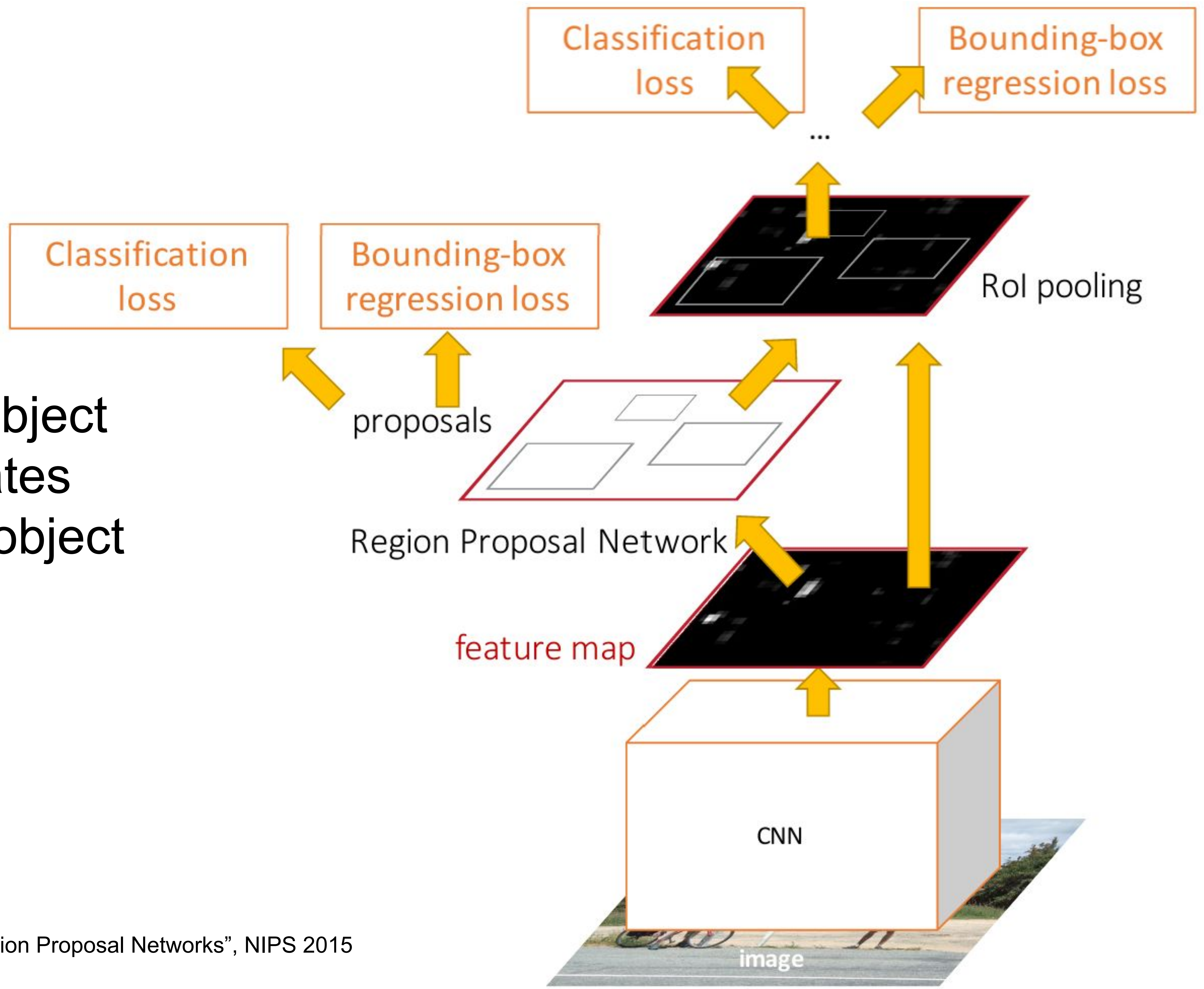
Ren et al, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NIPS 2015

Fast<u>er</u> R-CNN:

Make CNN do proposals!

Jointly train with 4 losses:

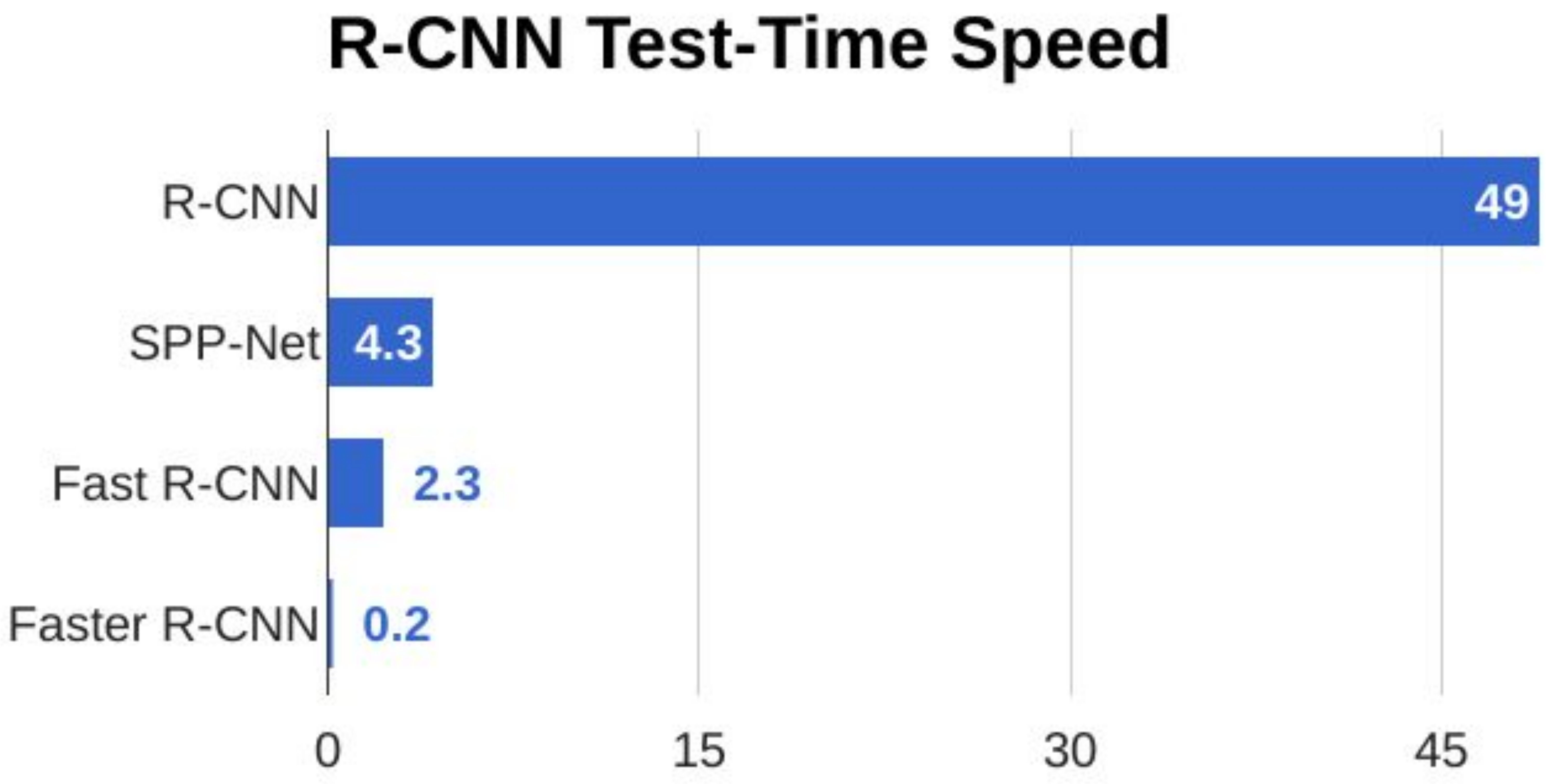
1. RPN classify object / not object
2. RPN regress box coordinates
3. Final classification score (object classes)
4. Final box coordinates



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

Faster R-CNN:

Make CNN do proposals!



Instance Segmentation

Classification

CAT

No spatial extent

Semantic Segmentation

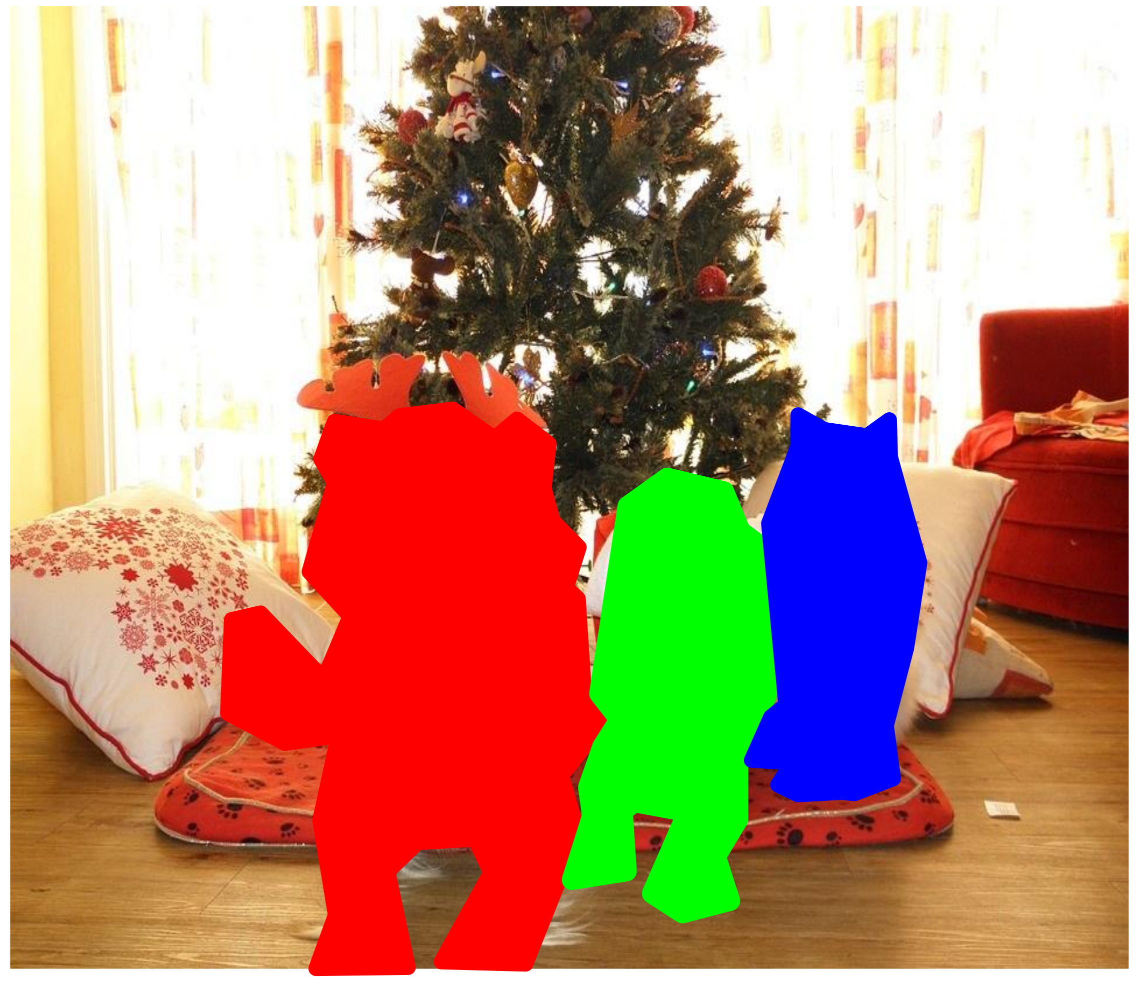
GRASS, CAT, TREE, SKY

No objects, just pixels

Object Detection

DOG, DOG, CAT

Instance Segmentation



DOG, DOG, CAT

Multiple Object

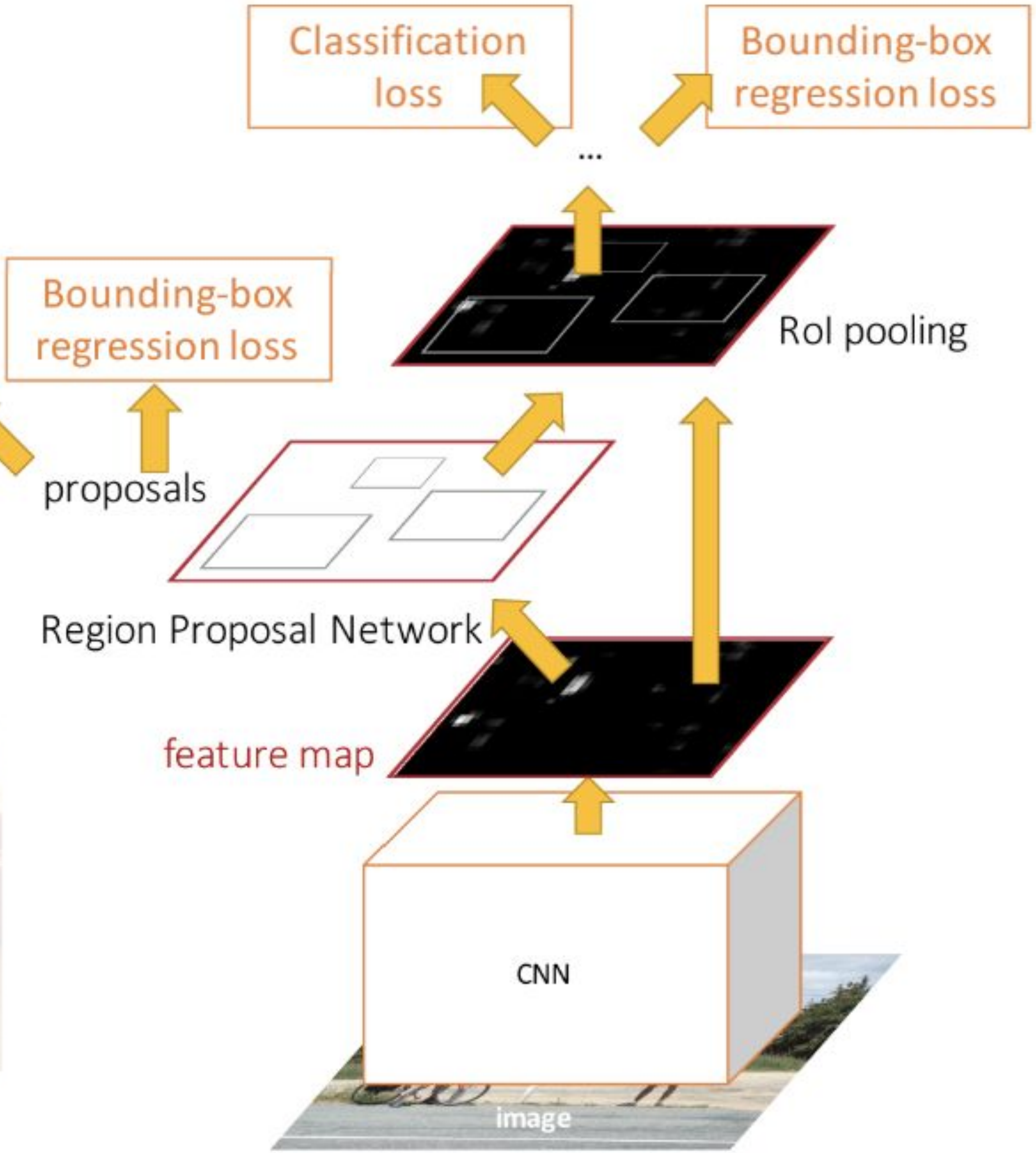
Object Detection:
Faster R-CNN

Object Detection

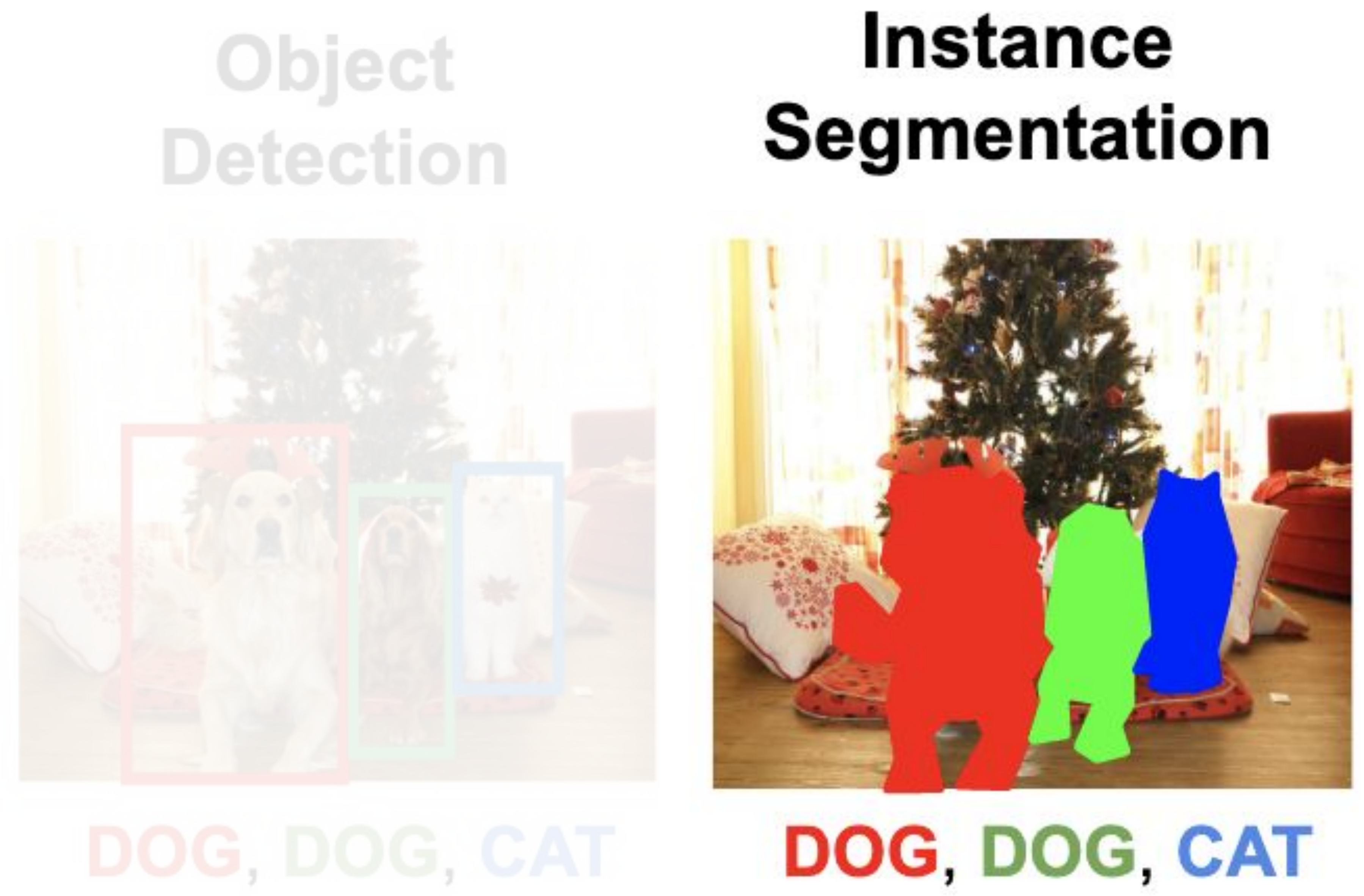
DOG, DOG, CAT

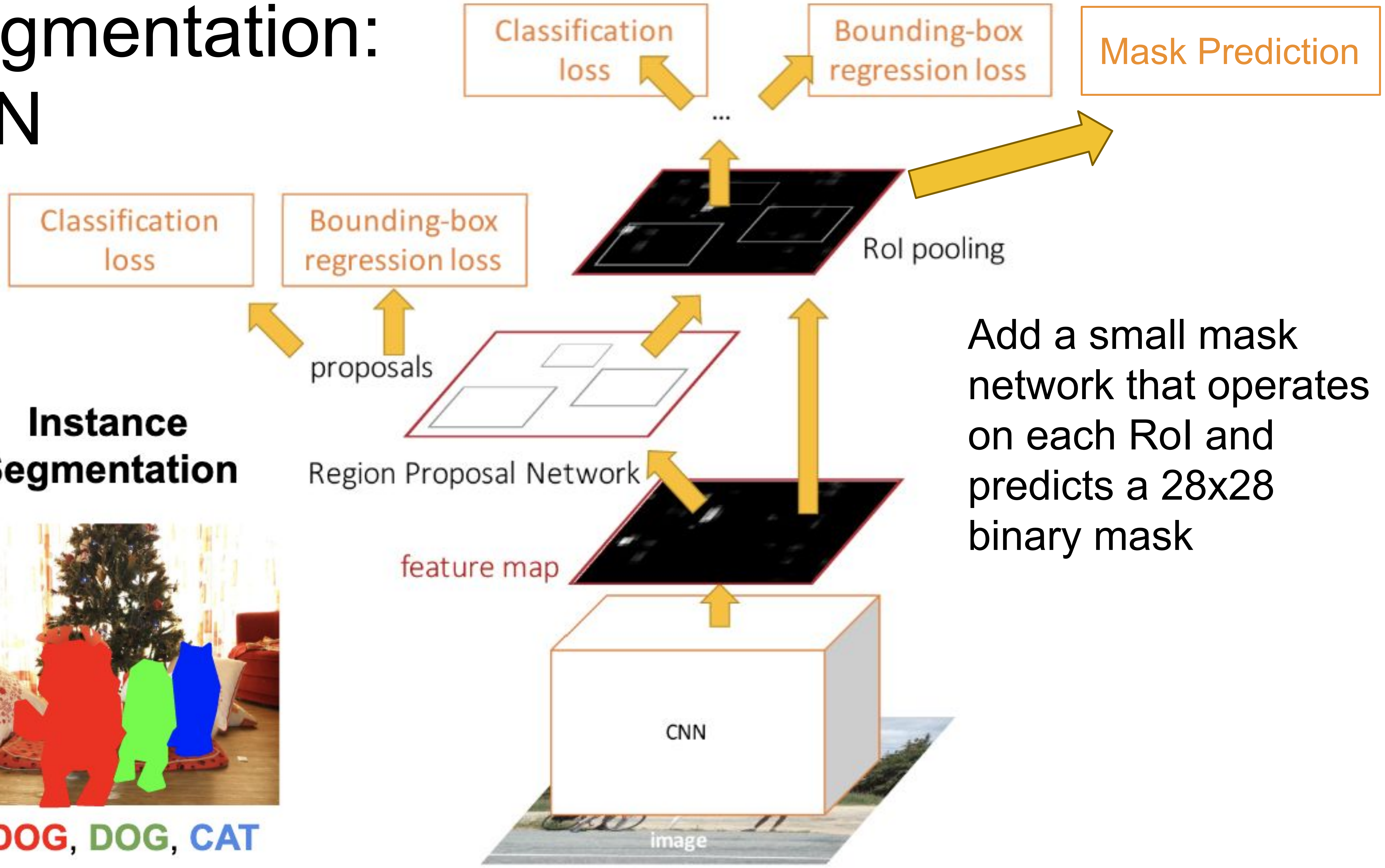
Instance Segmentation

DOG, DOG, CAT



Instance Segmentation: Mask R-CNN

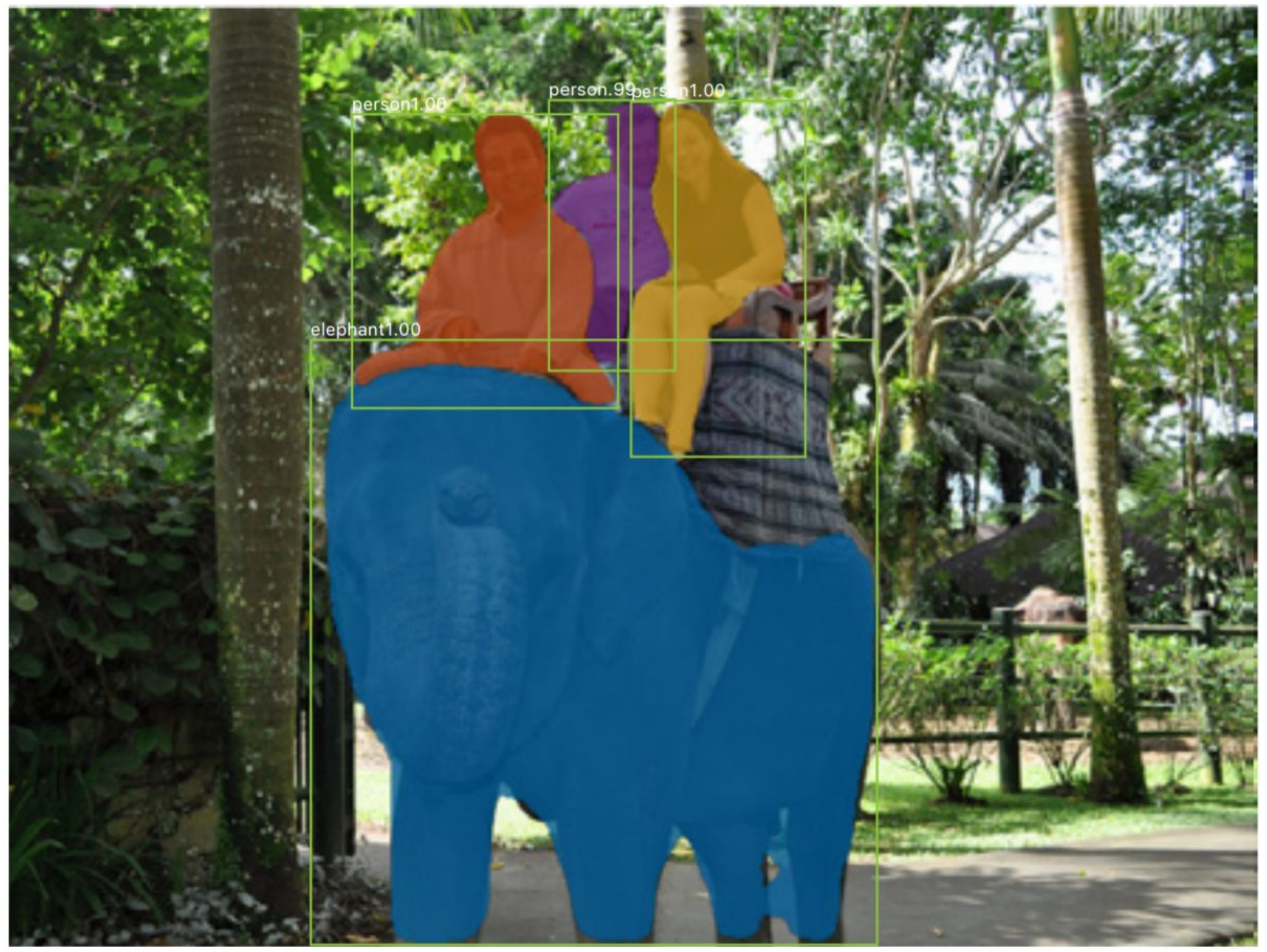


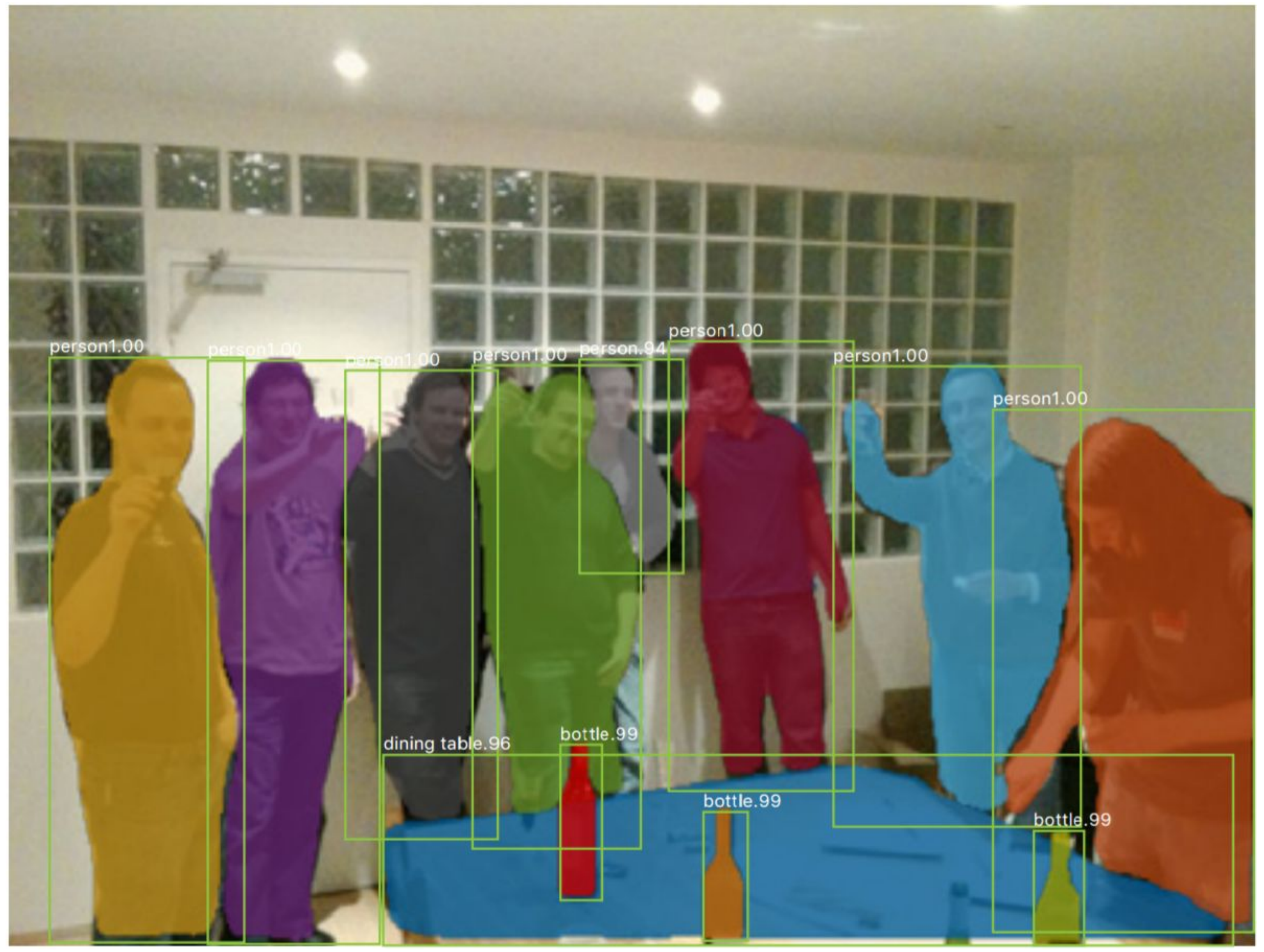


Add a small mask network that operates on each RoI and predicts a 28x28 binary mask

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

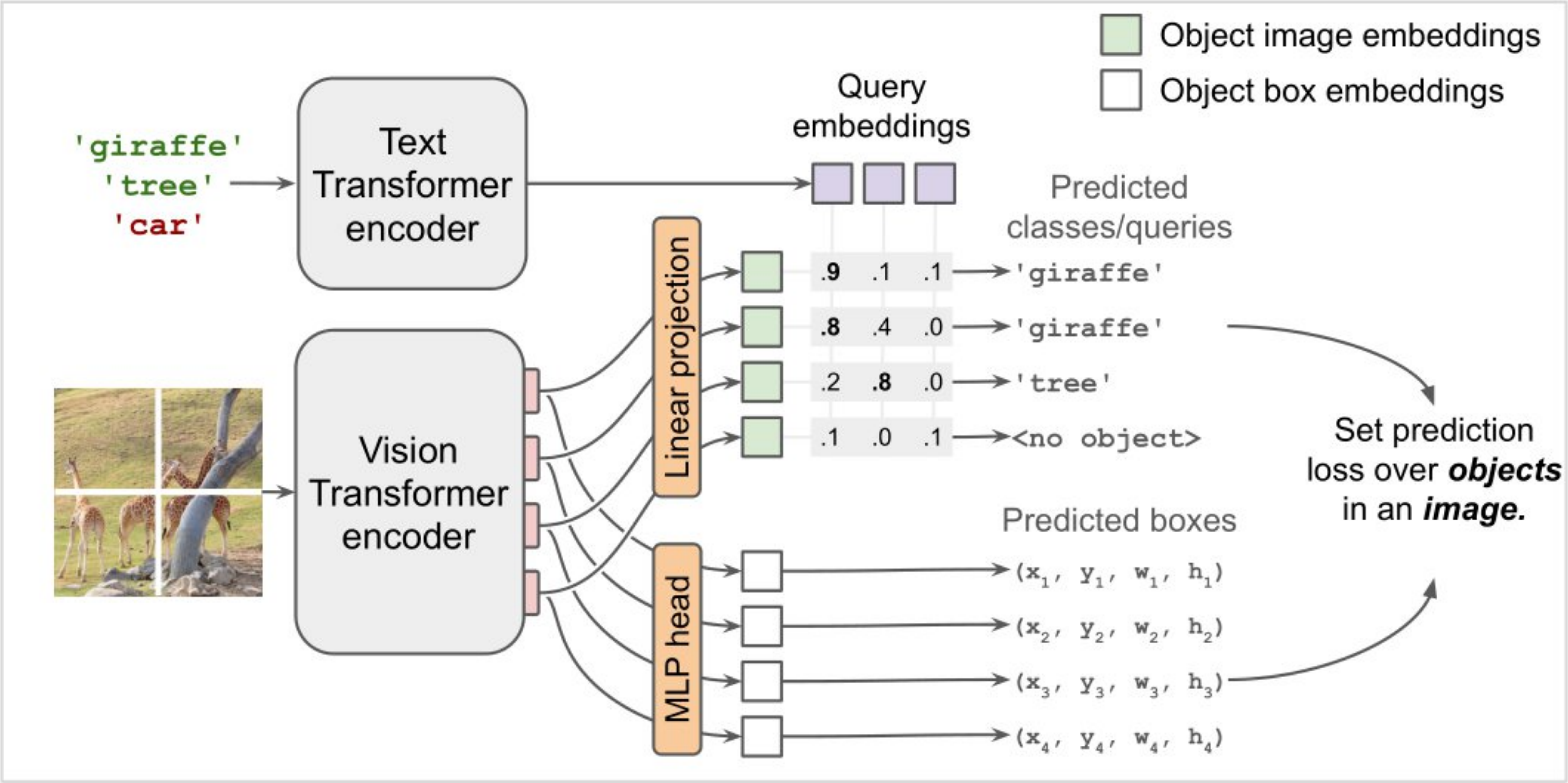
Mask R-CNN: Very Good Results!





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

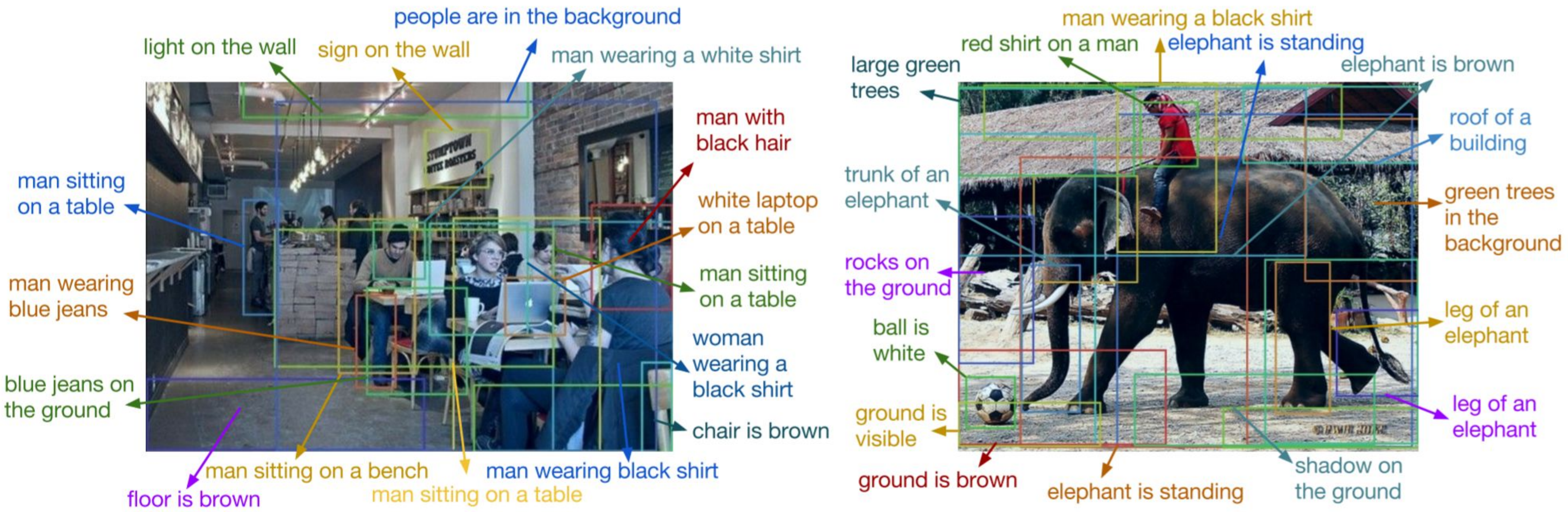
Modern Architectures (OWL-ViT)



Is 2D instance segmentation enough for robots?

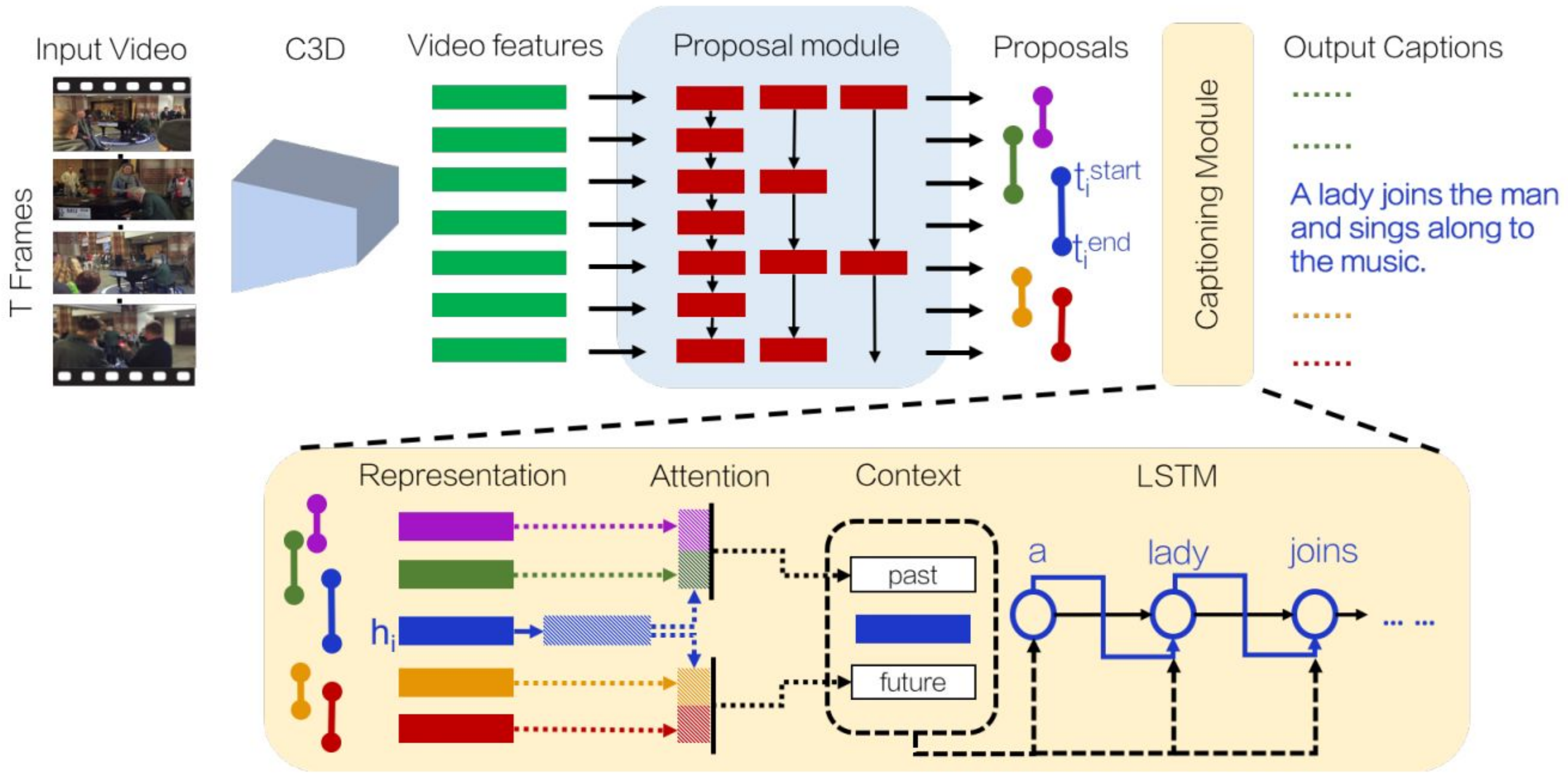
No!

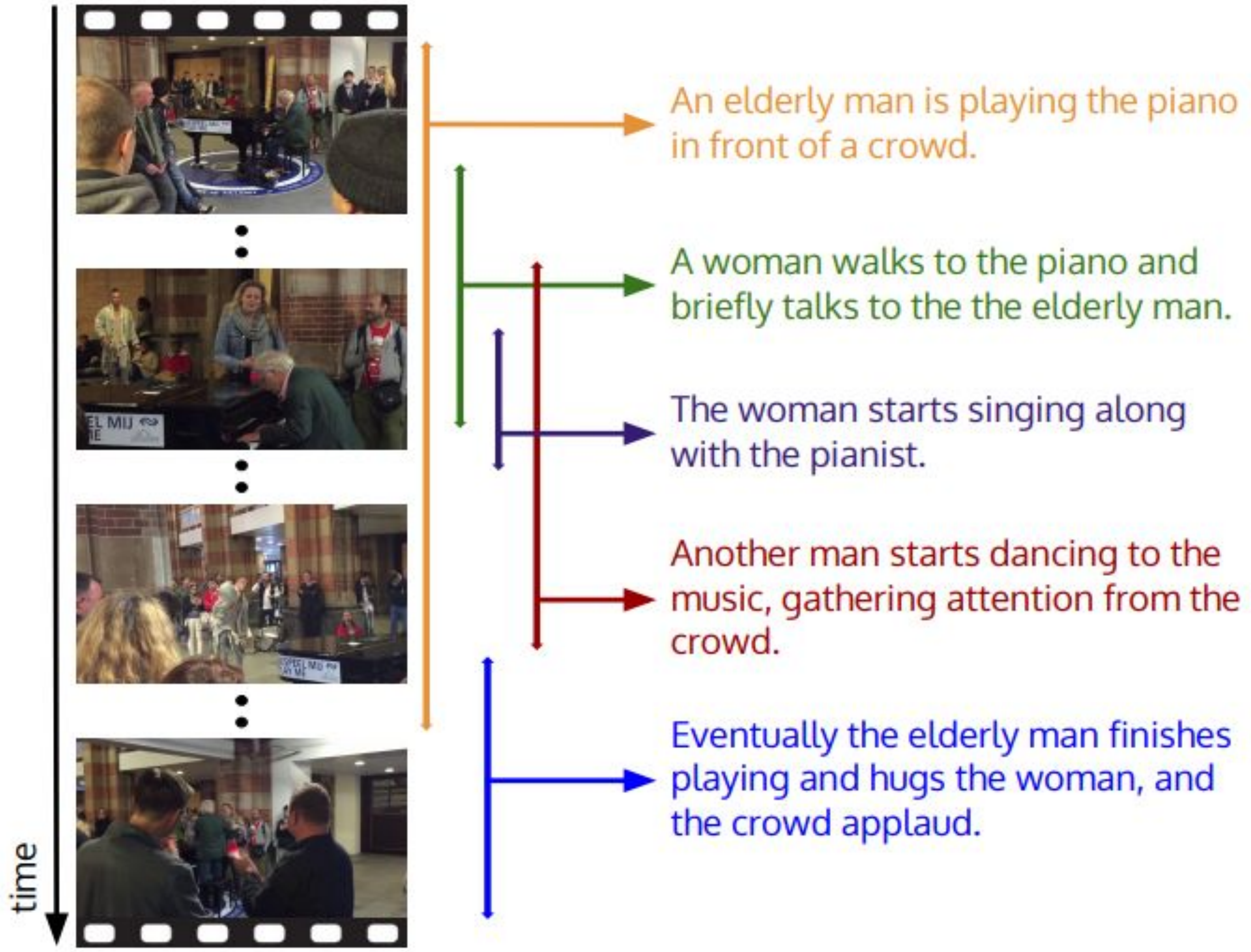
Object Detection + Captioning = Dense Captioning



Johnson, Karpathy, and Fei-Fei, “DenseCap: Fully Convolutional Localization Networks for Dense Captioning”, CVPR 2016

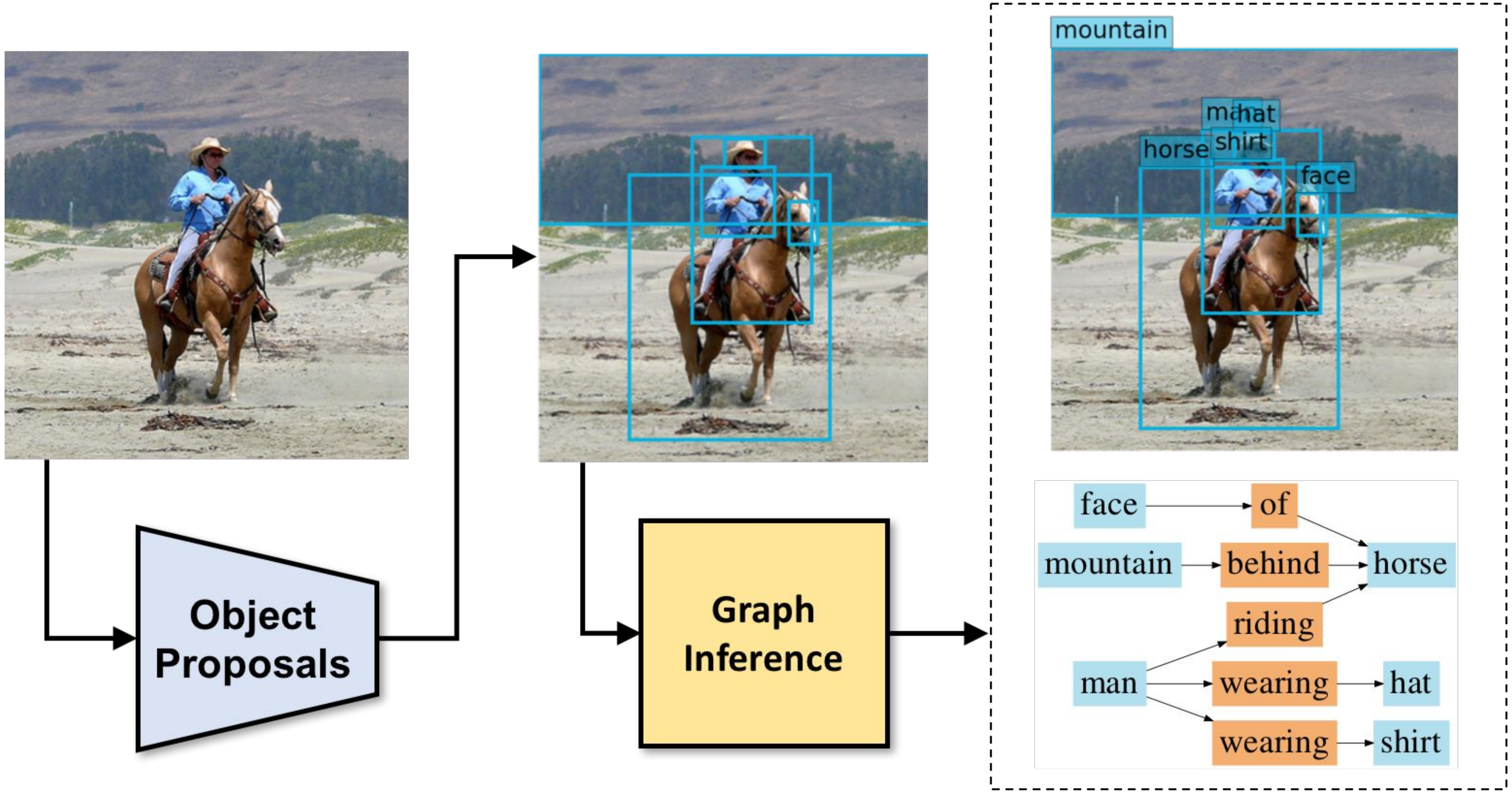
Dense Video Captioning





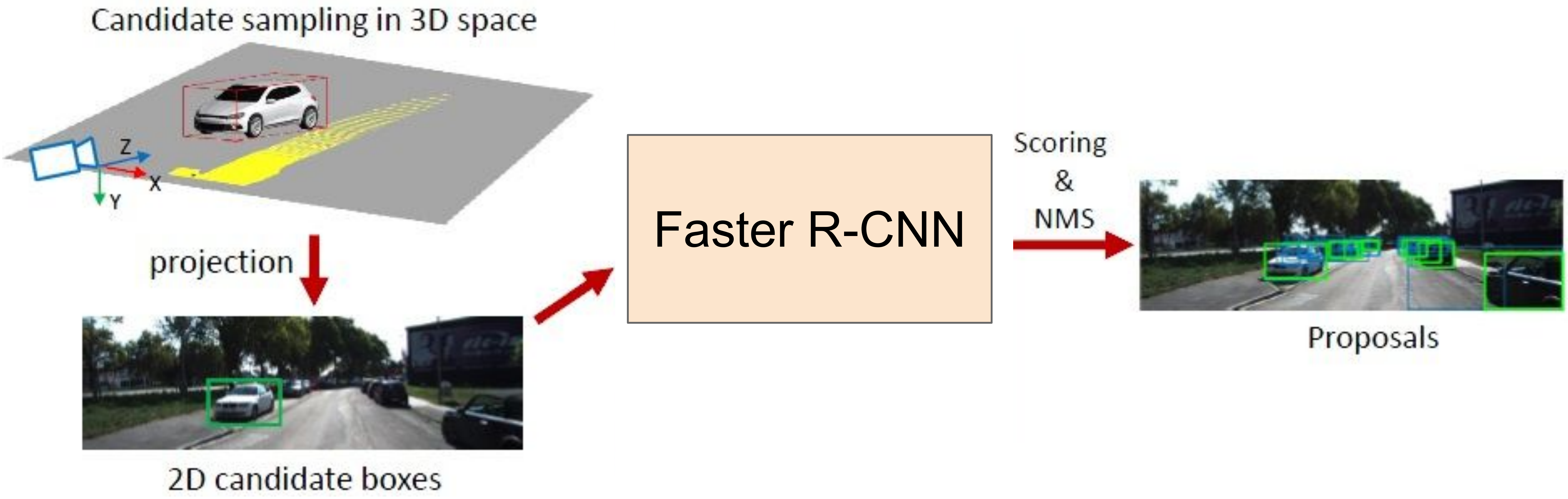
Ranjay Krishna et al., "Dense-Captioning Events in Videos", ICCV 2017

Scene Graph Prediction



Xu, Zhu, Choy, and Fei-Fei, “Scene Graph Generation by Iterative Message Passing”, CVPR 2017

3D Object Detection: Monocular Camera



- Same idea as Faster RCNN, but proposals are in 3D
- 3D bounding box proposal, regress 3D box parameters + class score

Chen, Xiaozhi, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. "Monocular 3d object detection for autonomous driving." CVPR 2016.