Referred by lecture notes of CSC486B/CSC586B @UVIC

ML Coding Practice
Lecture 04-1

Real-time Object Detection

Prof. Jongwon Choi
Chung-Ang University
Fall 2022

Today's Lecture

- **What's Object Detection?**
- **R-CNN / Fast RCNN / Faster RCNN**
- SSD / YOLO

So far: Image classification

This image is CC0 public domain

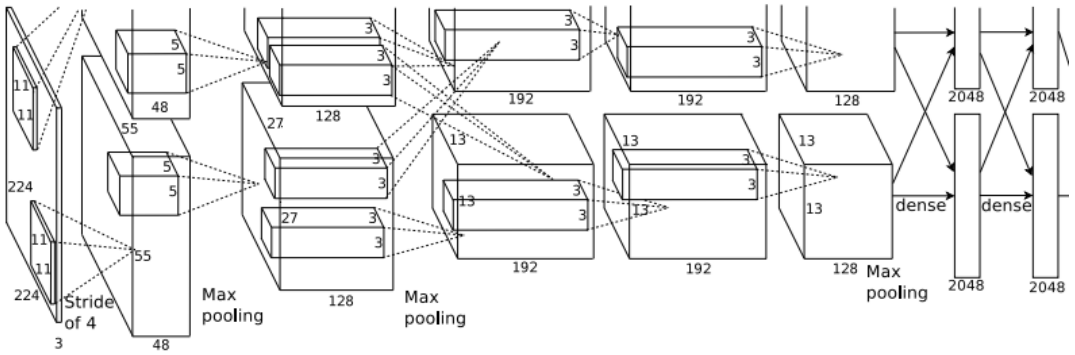
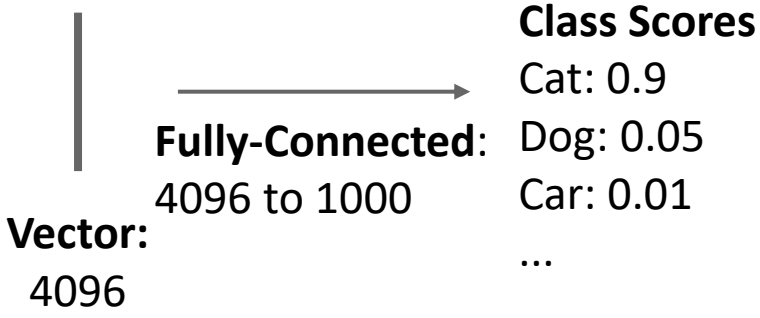
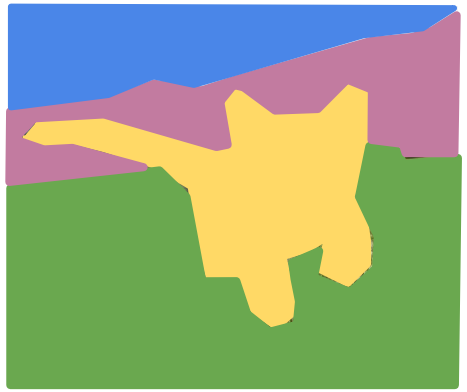


Figure copyright Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton, 2012. Reproduced with permission.



Other Computer Vision tasks

Semantic Segmentation



GRASS, CAT, TREE, SKY

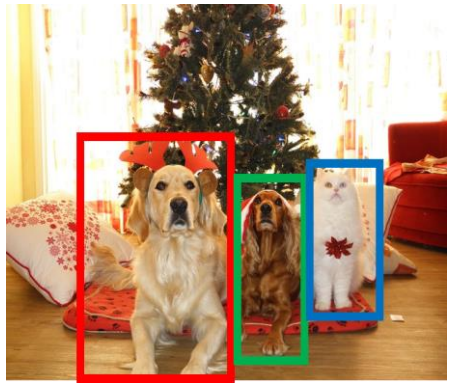
No objects, just pixels

Classification + Localization

CAT

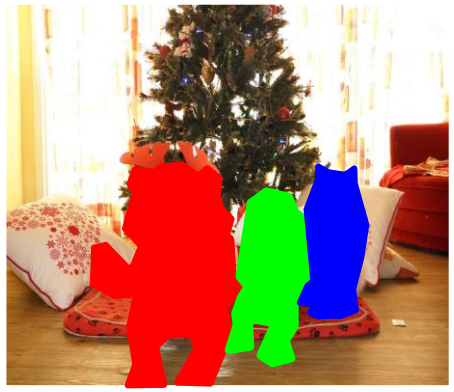
Single Object

Object Detection



DOG, DOG, CAT

Instance Segmentation

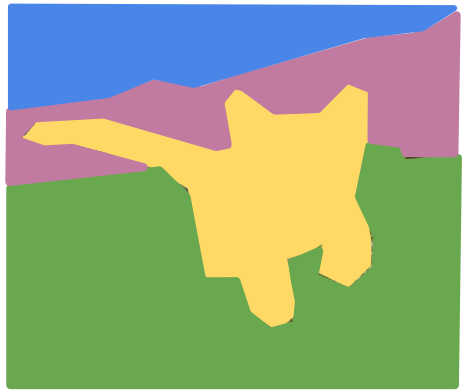


DOG, DOG, CAT

Multiple Object

Other Computer Vision tasks

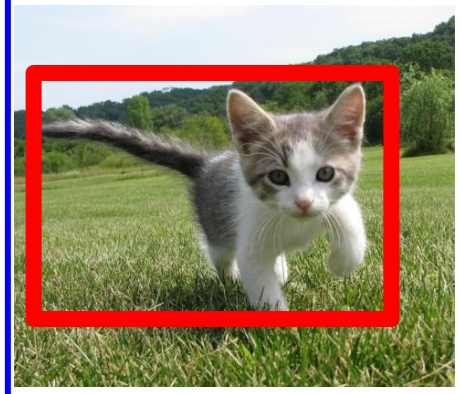
Semantic Segmentation



GRASS, CAT, TREE, SKY

No objects, just pixels

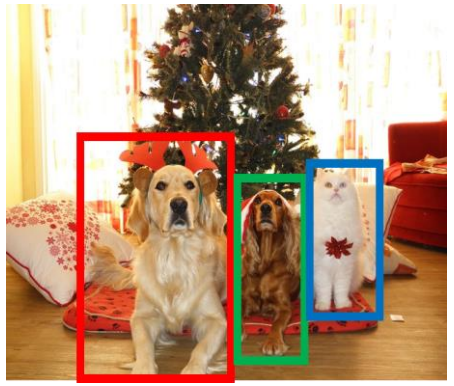
Classification + Localization



CAT

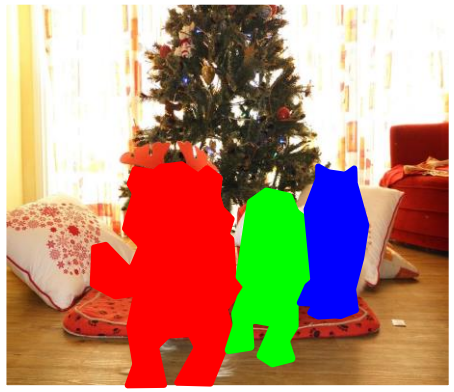
Single Object

Object Detection



DOG, DOG, CAT

Instance Segmentation



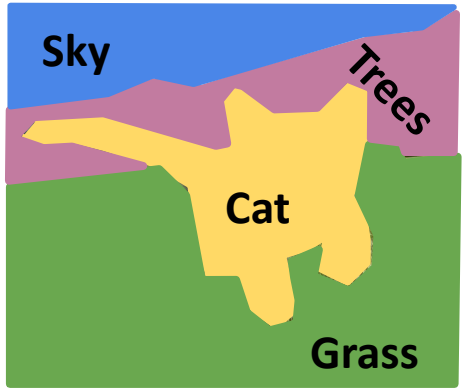
DOG, DOG, CAT

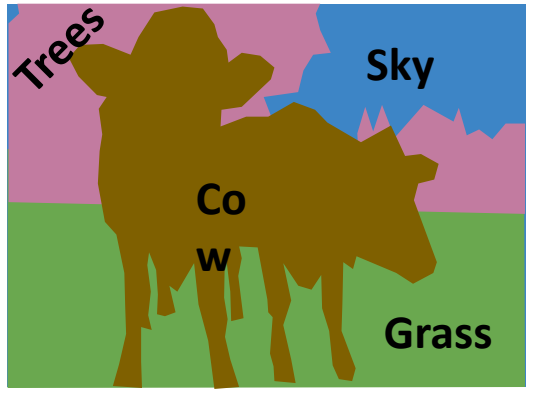
Multiple Object

Semantic segmentation

Label each pixel in the image with a category label

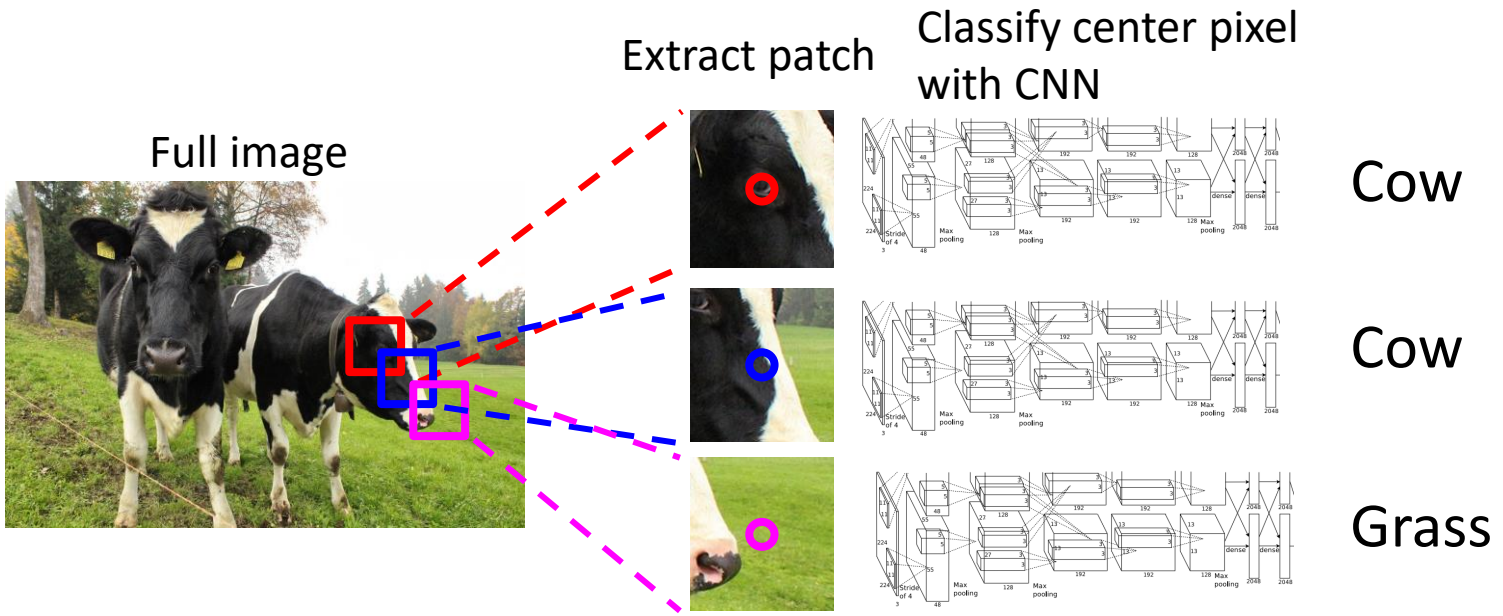
Don’t differentiate instances, only care about pixels





Semantic segmentation

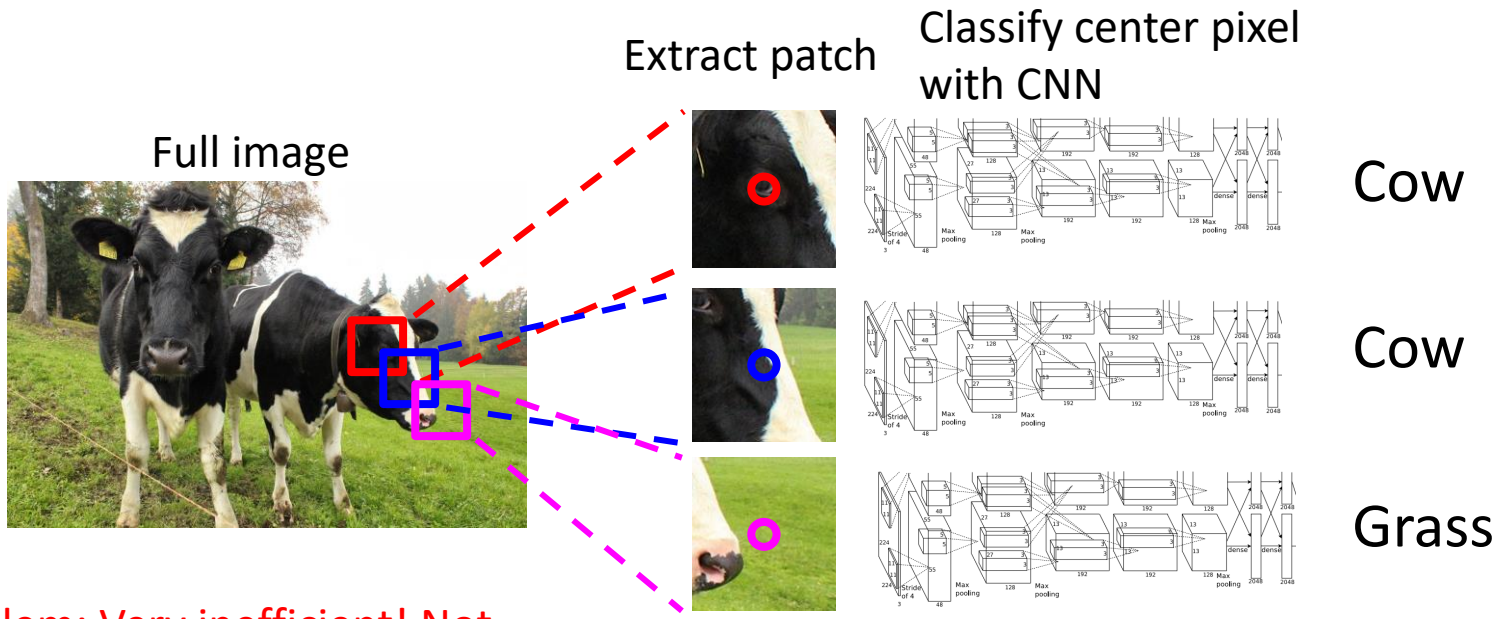
- Sliding window approach



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

- Sliding window approach



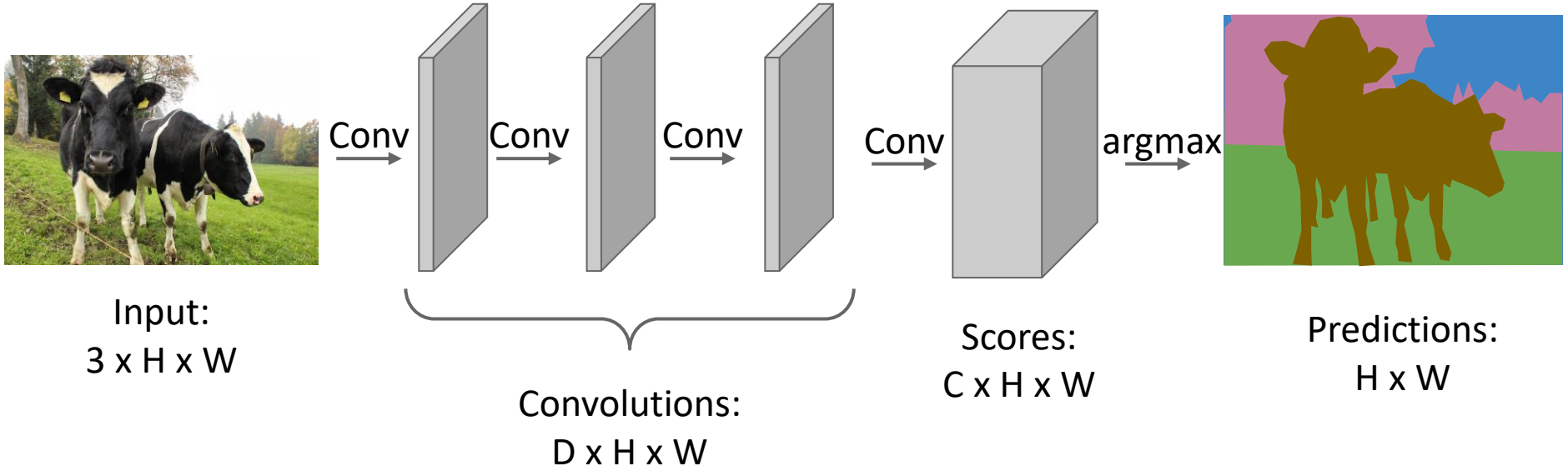
Problem: Very inefficient! Not reusing shared features between overlapping patches

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

- Fully convolutional

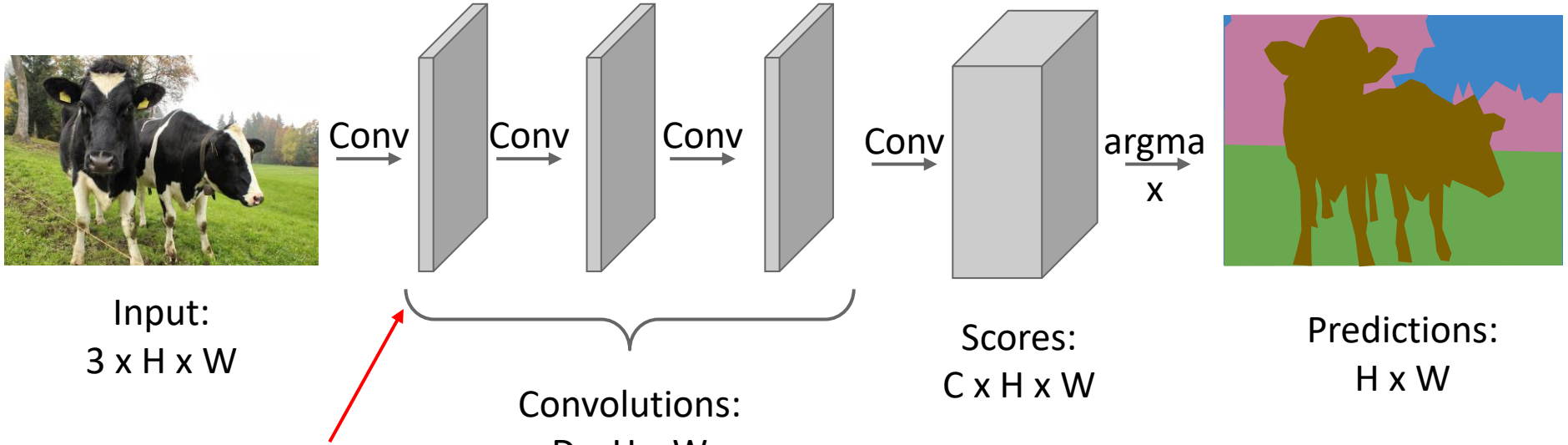
Design a network as a bunch of convolutional layers to make predictions for pixels all at once!



Semantic segmentation

- Fully convolutional

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



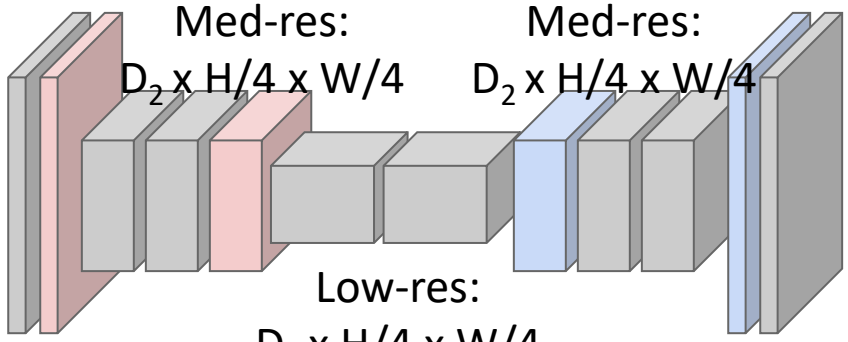
Problem: convolutions at original image resolution will be very expensive ...

Semantic segmentation

- Fully convolutional

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

Input:
3 x H x W



High-res:
D1 x H/2 x W/2

High-res:
D1 x H/2 x W/2

Predictions:
H x W

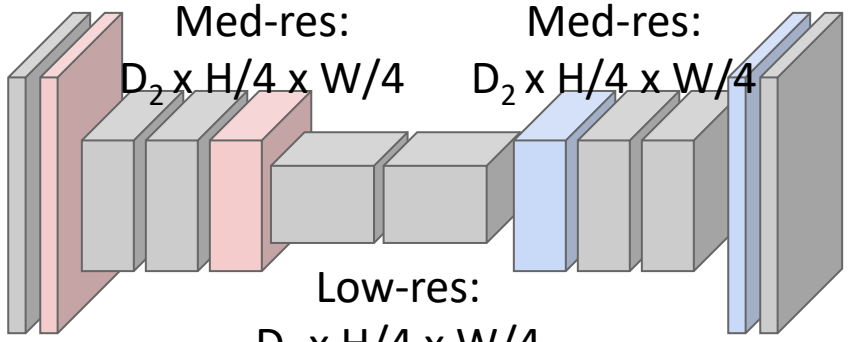
Long, Shelhamer, and Darrell, "Fully Convolutional Networks for Semantic Segmentation", CVPR 2015
Noh et al, "Learning Deconvolution Network for Semantic Segmentation", ICCV 2015

Semantic segmentation

- Fully convolutional

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

Input:
3 x H x W



High-res:
D_1 x H/2 x W/2

High-res:
D_1 x H/2 x W/2

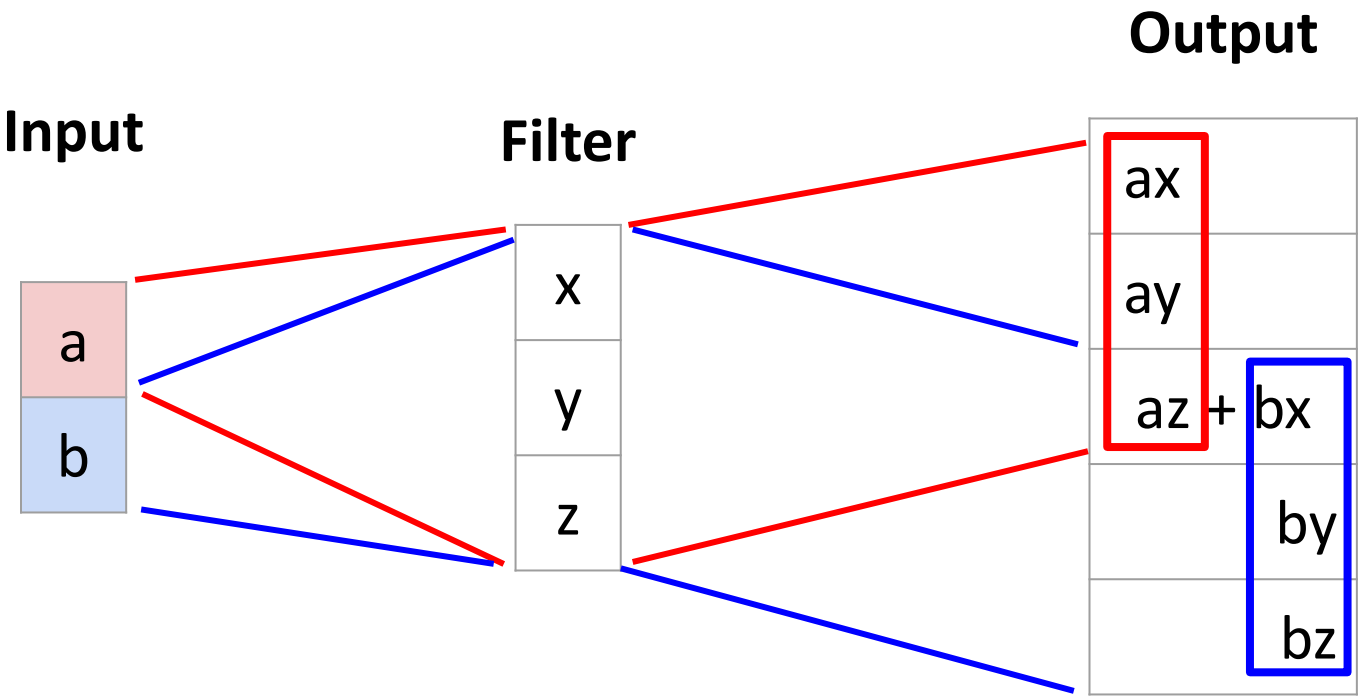
Predictions:
H x W

Downsampling:
Pooling, strided convolution

Upsampling:
????

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

Recall: Transposed convolution



Output contains copies of the filter weighted by the input, summing at where at overlaps in the output

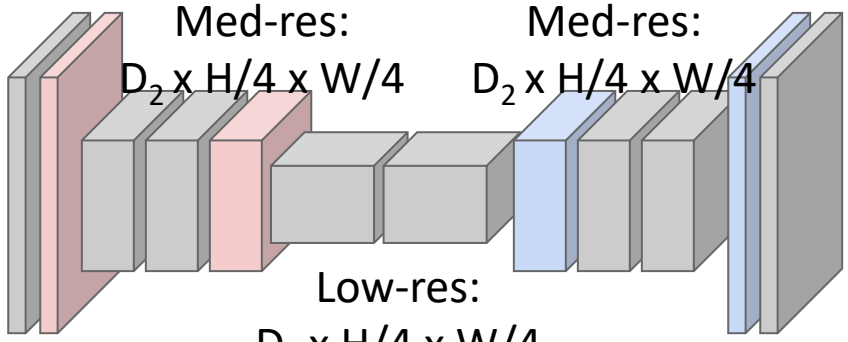
Need to crop one pixel from output to make output exactly 2x input

Semantic segmentation

- Fully convolutional

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

Input:
3 x H x W



High-res:
D_1 x H/2 x W/2

High-res:
D_1 x H/2 x W/2

Predictions:
H x W

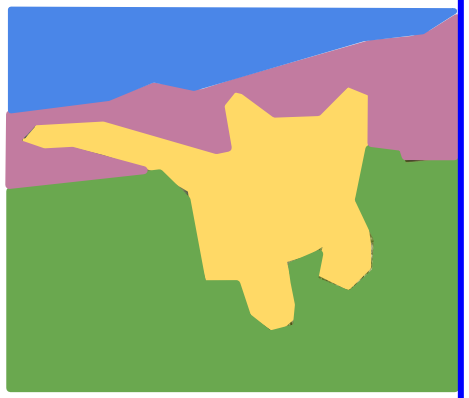
Downsampling:
Pooling, strided convolution

Upsampling:
Unpooling or strided transpose convolution

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

Other Computer Vision tasks

Semantic Segmentation



GRASS, CAT, TREE, SKY

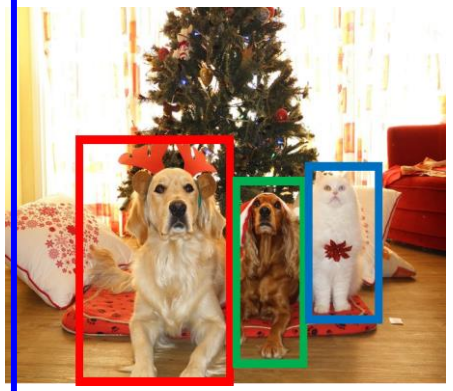
No objects, just pixels

Classification + Localization

CAT

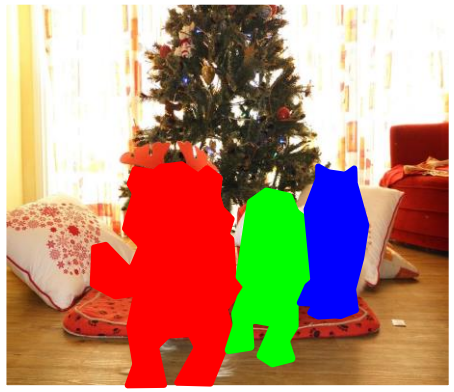
Single Object

Object Detection



DOG, DOG, CAT

Instance Segmentation

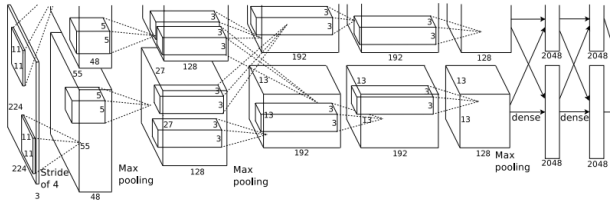


DOG, DOG, CAT

Multiple Object

Classification + Localization

This image is CC0 public domain



Fully Connected: 4096 to 1000

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

Vector: 4096

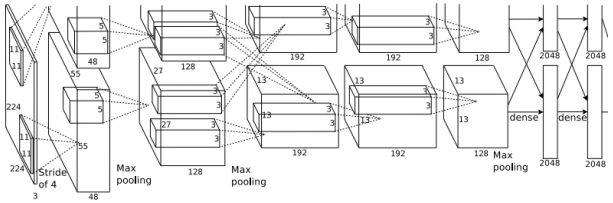
Fully Connected: 4096 to 4

Box Coordinates
(x, y, w, h)

Treat localization as a regression problem!

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)

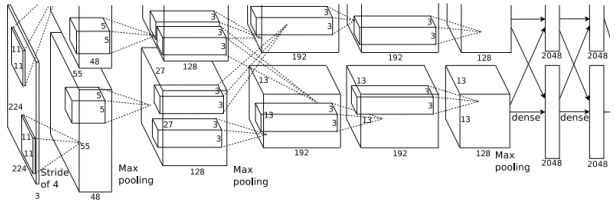
L2 Loss

Correct box:
(x', y', w', h')

Treat localization as a regression problem!

Classification + Localization

This image is CC0 public domain



Vector: 4096

Fully Connected: 4096 to 1000

Fully Connected: 4096 to 4

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

Box Coordinates
(x, y, w, h)

Correct label:
Cat

Softmax Loss

Multitask Loss

+

Loss

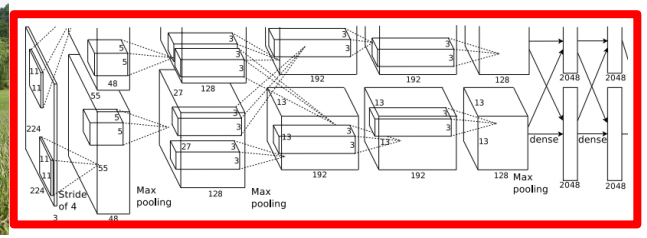
L2 Loss

Correct box:
(x', y', w', h')

Treat localization as a regression problem!

Classification + Localization

This image is CC0 public domain



Often pretrained on ImageNet (Transfer learning)

Vector: 4096

Fully Connected: 4096 to 1000

Fully Connected: 4096 to 4

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

Box Coordinates
(x, y, w, h)

Correct label:
Cat

Softmax Loss

L2 Loss

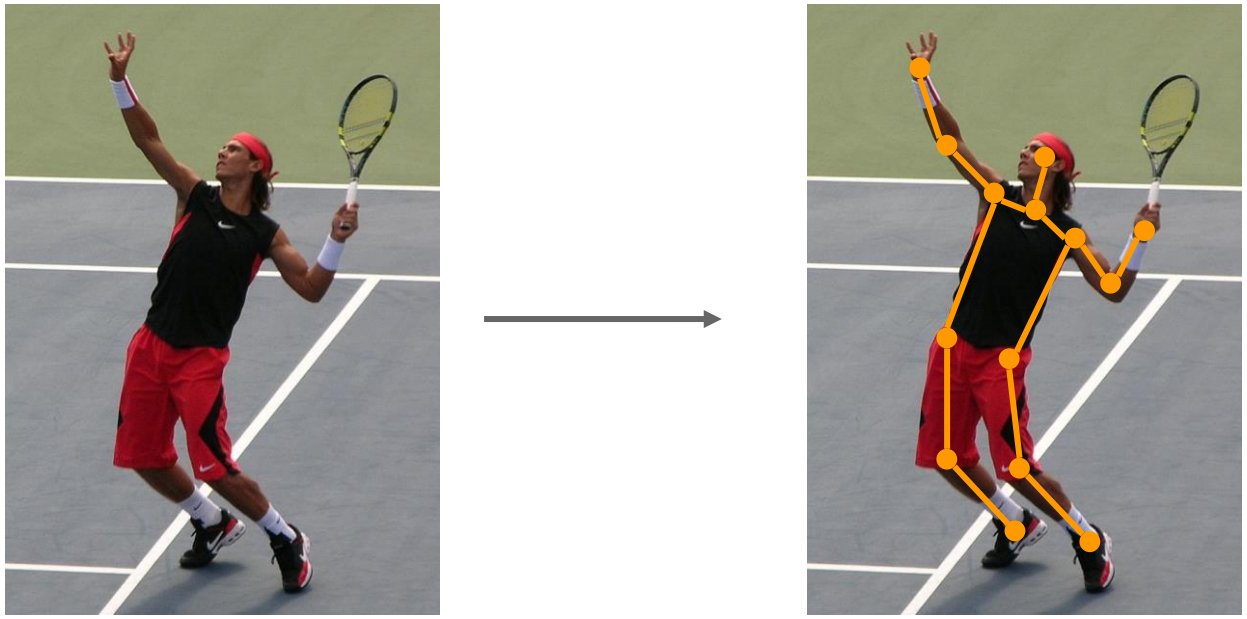
+

Loss

Correct box:
(x', y', w', h')

Treat localization as a regression problem!

Aside: Human pose estimation

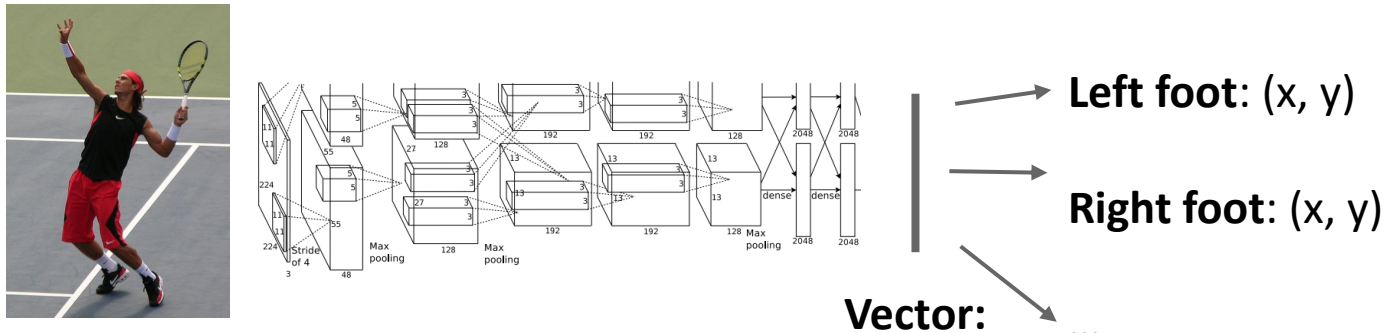


Represent pose as a set of 14 joint positions:

Left / right foot
Left / right knee
Left / right hip
Left / right shoulder
Left / right elbow
Left / right hand
Neck
Head top

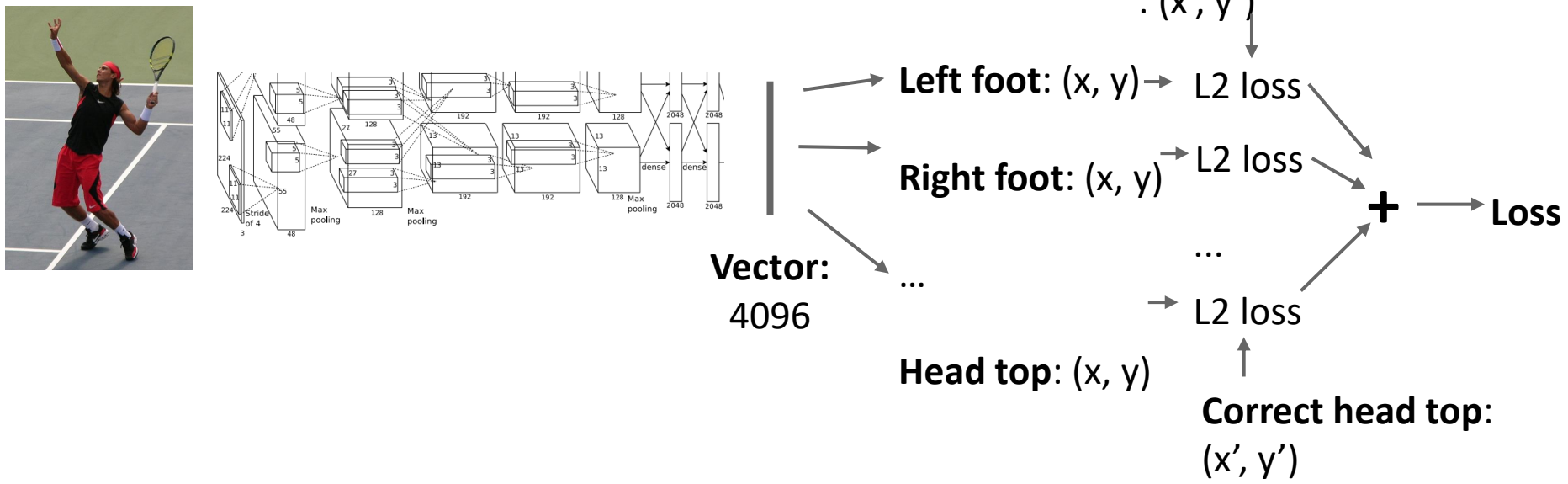
Johnson and Everingham, "Clustered Pose and Nonlinear Appearance Models for Hum an Pose Estimation", BMVC 2010

Aside: Human pose estimation



Toshev and Szegedy, "DeepPose: Human Pose Estimation via Deep Neural Networks", CVPR 2014

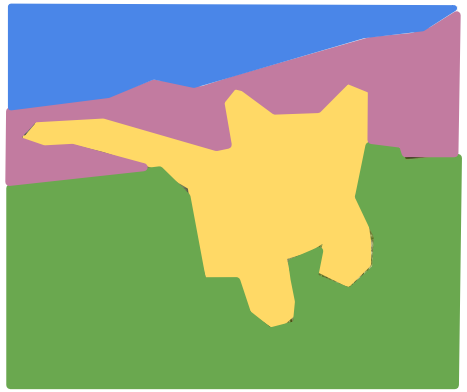
Aside: Human pose estimation



Toshev and Szegedy, "DeepPose: Human Pose Estimation via Deep Neural Networks", CVPR 2014

Other Computer Vision tasks

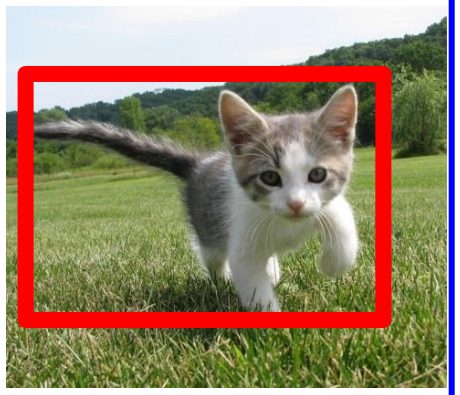
Semantic Segmentation



GRASS, CAT, TREE, SKY

No objects, just pixels

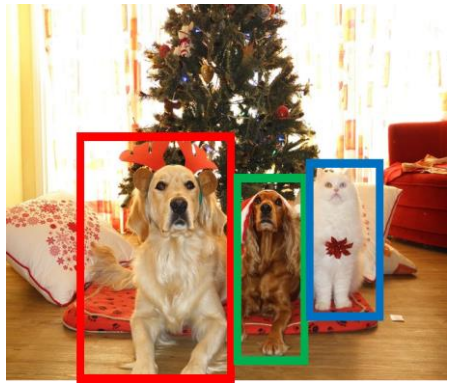
Classification + Localization



CAT

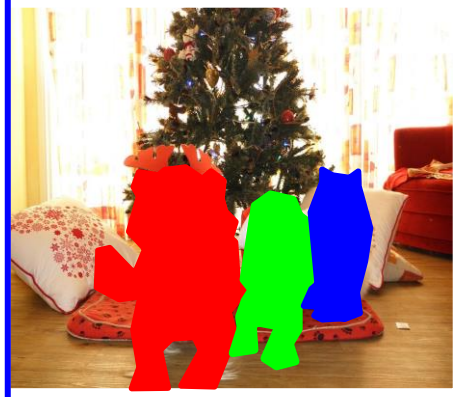
Single Object

Object Detection



DOG, DOG, CAT

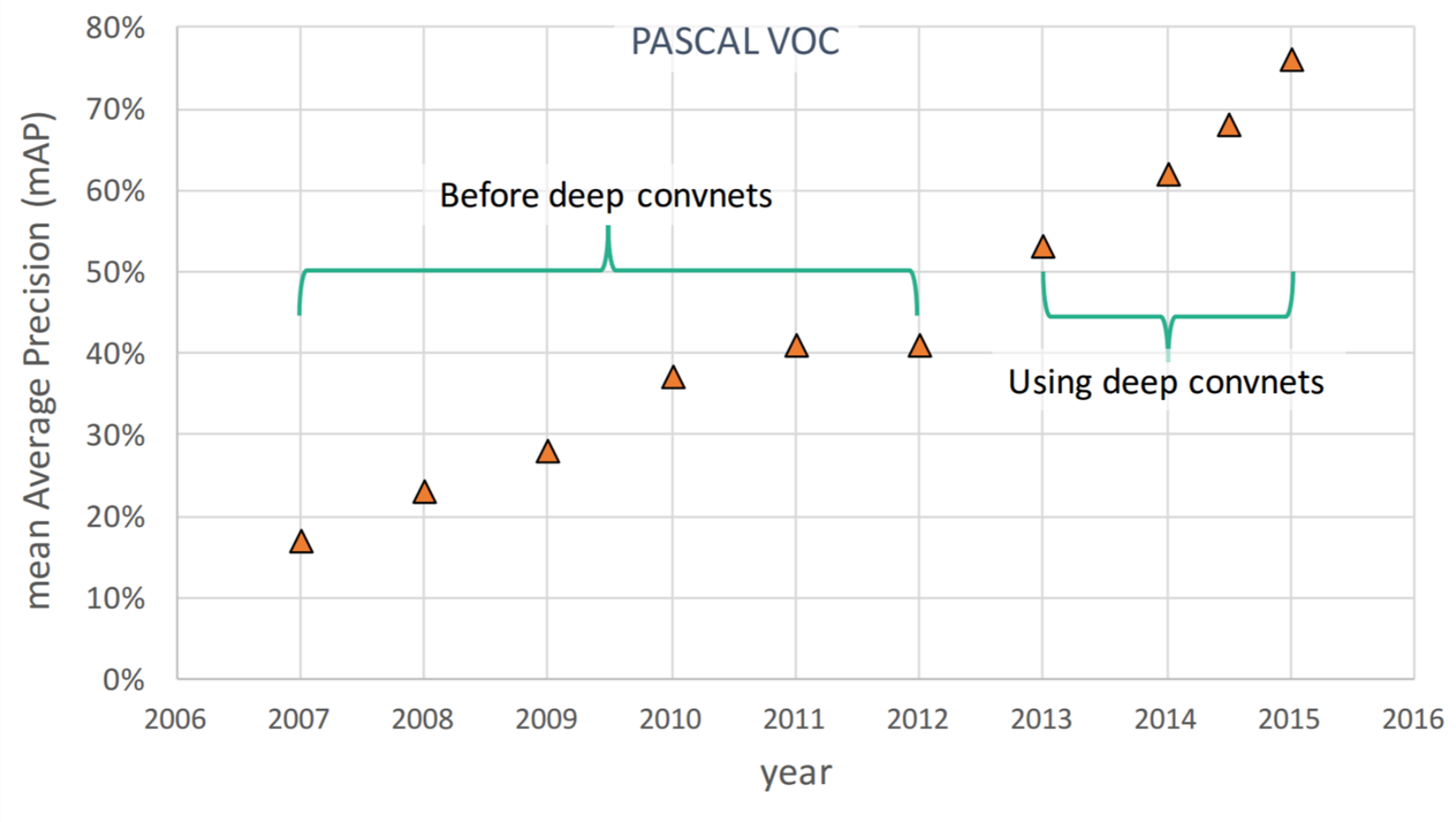
Instance Segmentation



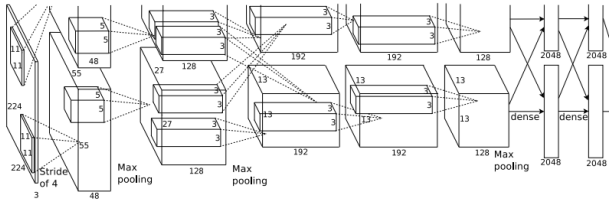
DOG, DOG, CAT

Multiple Object

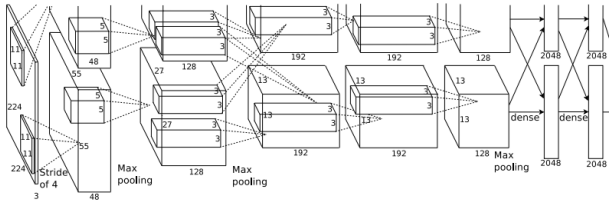
Object detection: preface



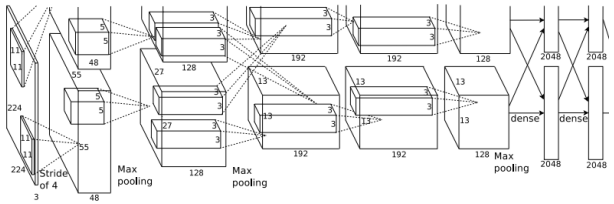
Object detection: as regression?



CAT: (x, y, w, h)



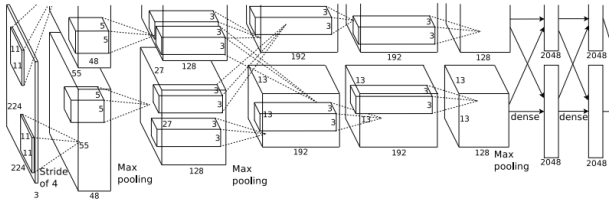
DOG: (x, y, w, h)
DOG: (x, y, w, h)
CAT: (x, y, w, h)



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

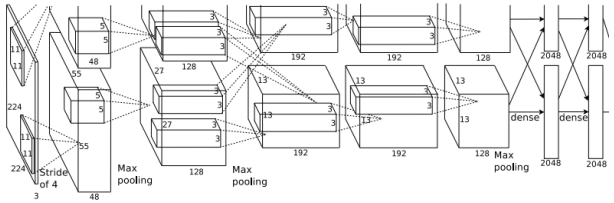
Object detection: as regression?

Each image needs a different number of outputs!



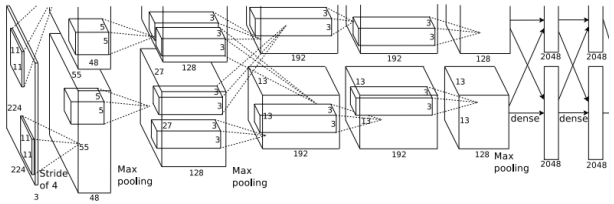
CAT: (x, y, w, h)

4 numbers



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

16 numbers

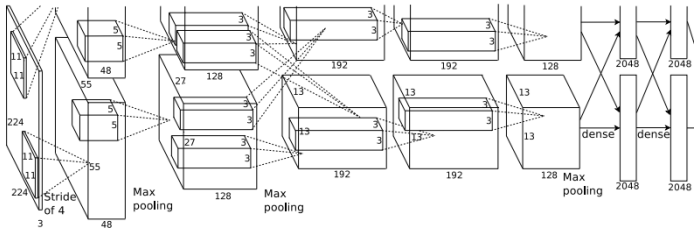


DUCK: (x, y, w, h)
DUCK: (x, y, w, h)
....

Many numbers!

Object detection: as classification!

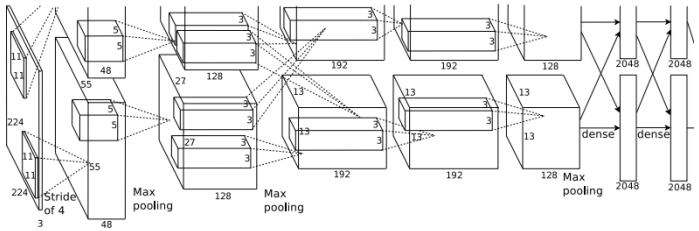
Apply a CNN to many different crops of the image,
CNN classifies each crop as object or background



Dog? NO
Cat? NO
Background? YES

Object detection: as classification!

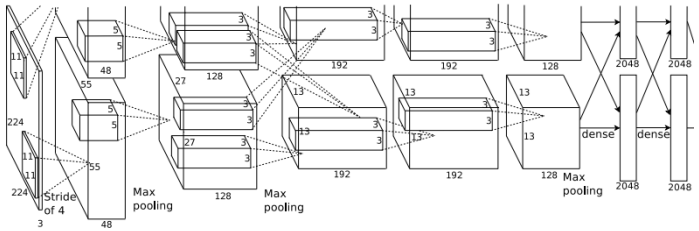
Apply a CNN to many different crops of the image,
CNN classifies each crop as object or background



Dog? YES
Cat? NO
Background? NO

Object detection: as classification!

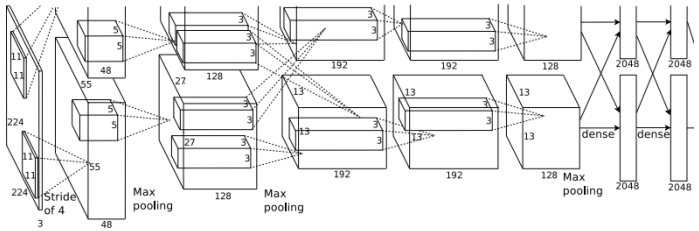
Apply a CNN to many different crops of the image,
CNN classifies each crop as object or background



Dog? YES
Cat? NO
Background? NO

Object detection: as classification!

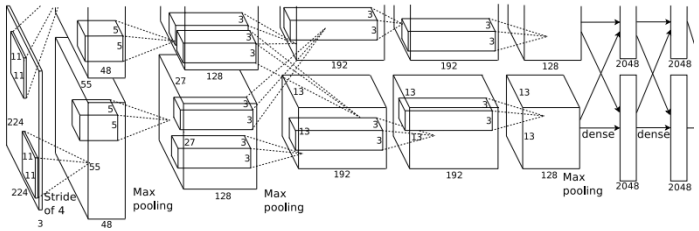
Apply a CNN to many different crops of the image,
CNN classifies each crop as object or background



Dog? NO
Cat? YES
Background? NO

Object detection: as classification!

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

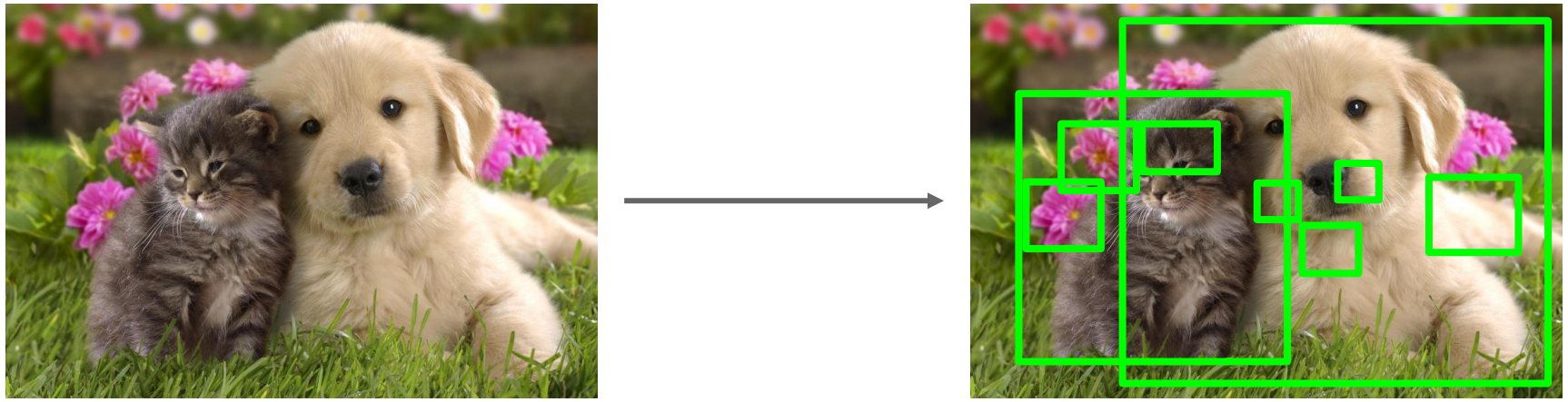


Dog? NO
Cat? YES
Background? NO

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

Region proposals

- Find “blobby” image regions that are likely to contain objects
- Relatively fast to run; e.g. Selective Search gives 1000 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 s egmentation”, CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

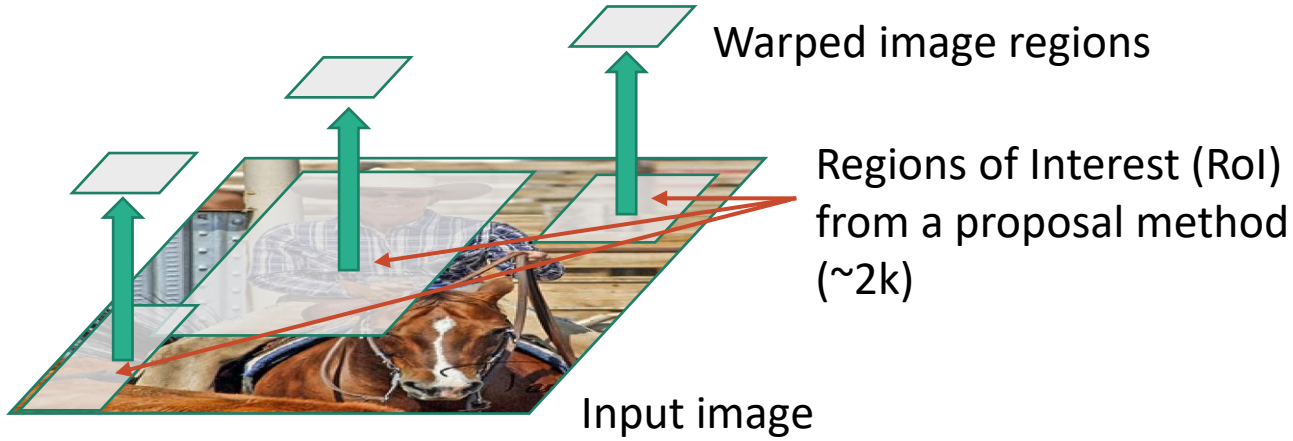
R-CNN

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

Input image

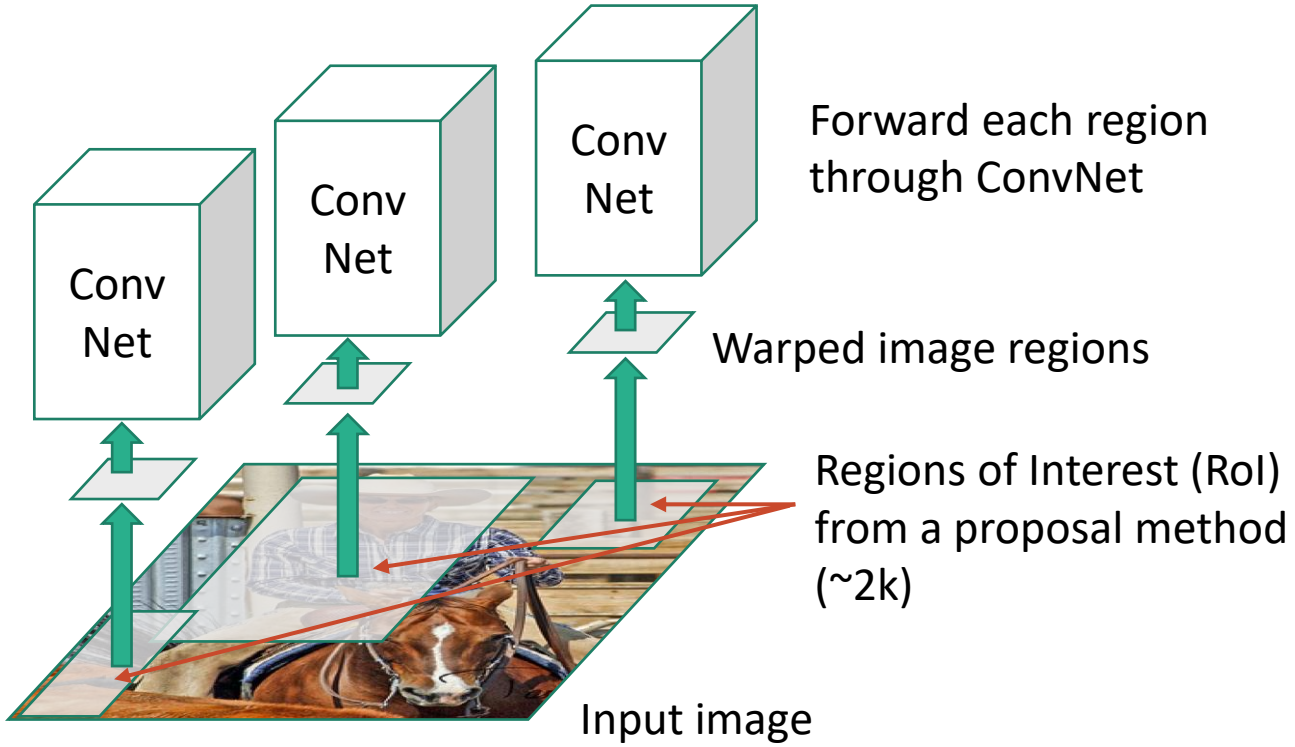
Girshick et al, "Rich feature hierarchies for accurate object detection and semantic s egmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

R-CNN



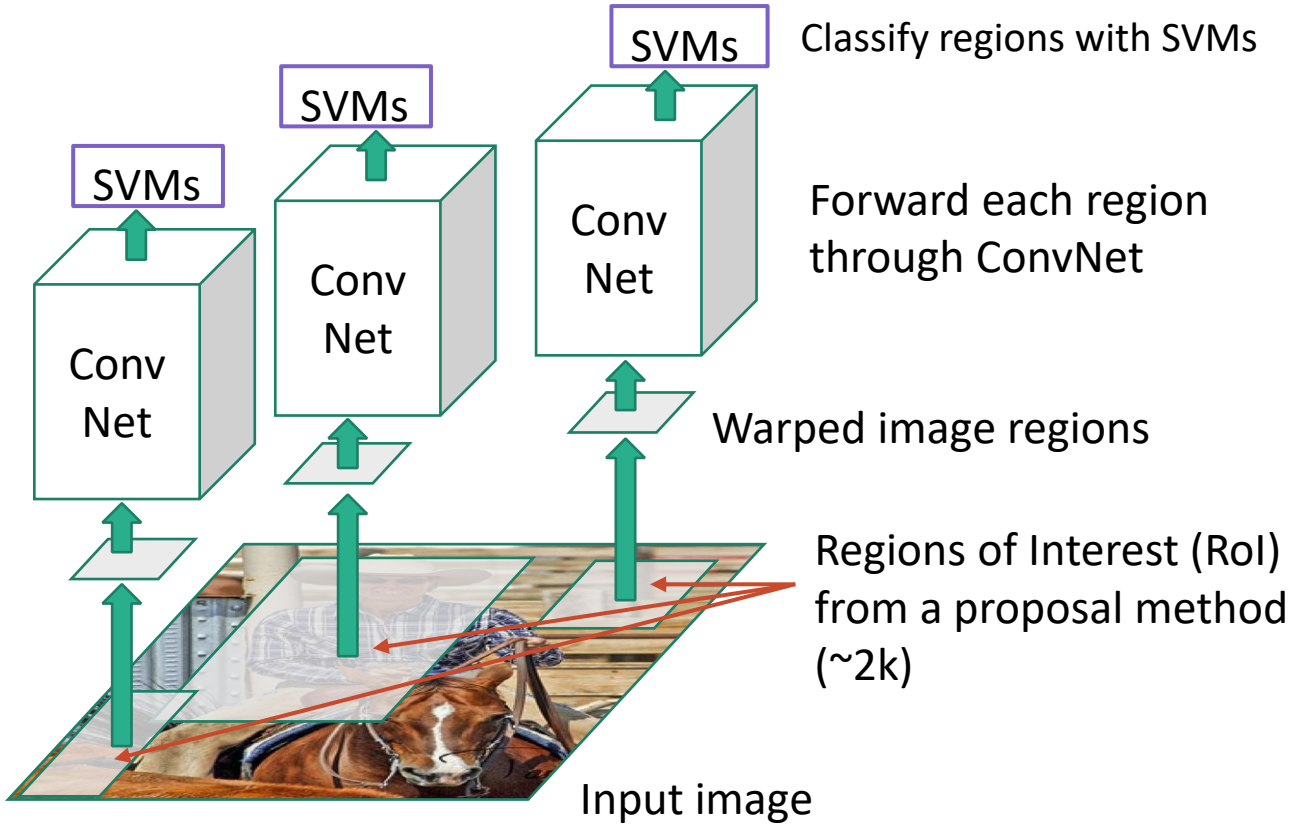
Girshick et al, "Rich feature hierarchies for accurate object detection and semantic s egmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

R-CNN



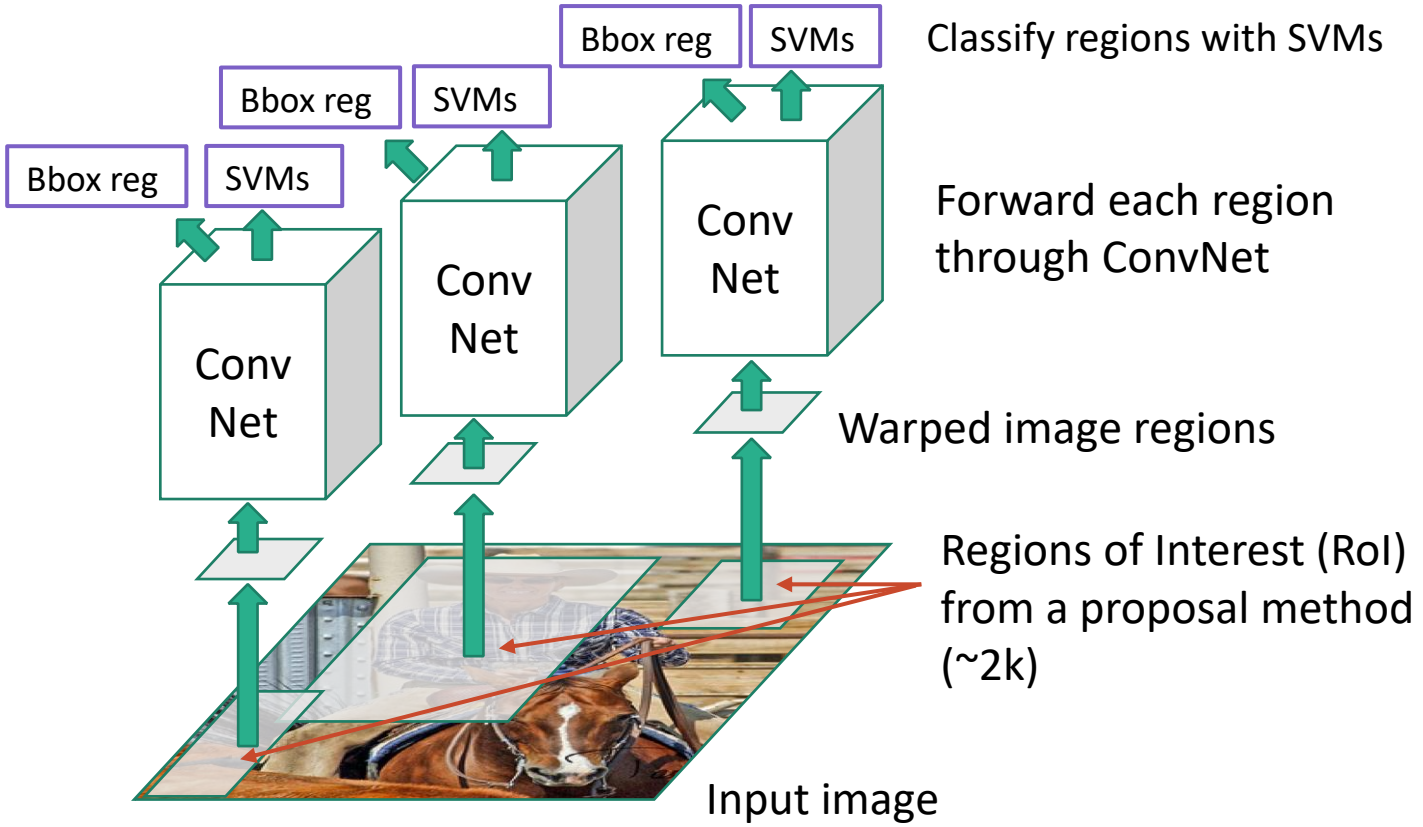
Girshick et al, "Rich feature hierarchies for accurate object detection and semantic s egmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

R-CNN



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

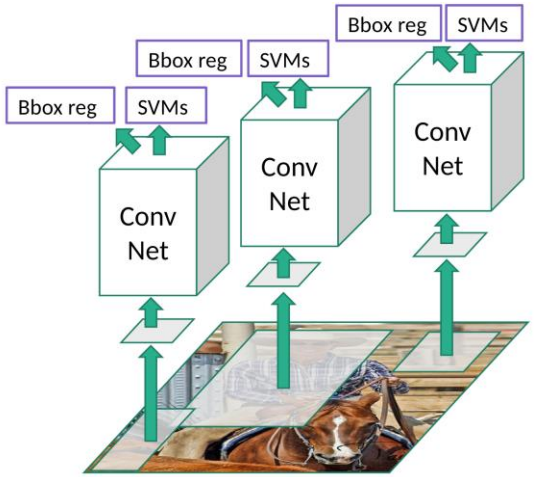
R-CNN



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

R-CNN

- Problems
  - Ad hoc training objectives
    - Fine-tune network with softmax classifier
      - log loss
    - Train post-hoc linear SVMs
      - hinge loss
    - Train post-hoc bounding-box regressions
      - least squares
  - Training is slow (84h), takes a lot of disk space
  - Inference (detection) is slow
    - 47s / image with VGG16 [Simonyan & Zisserman. ICLR15]
    - Fixed by SPP-net [He et al. ECCV14]



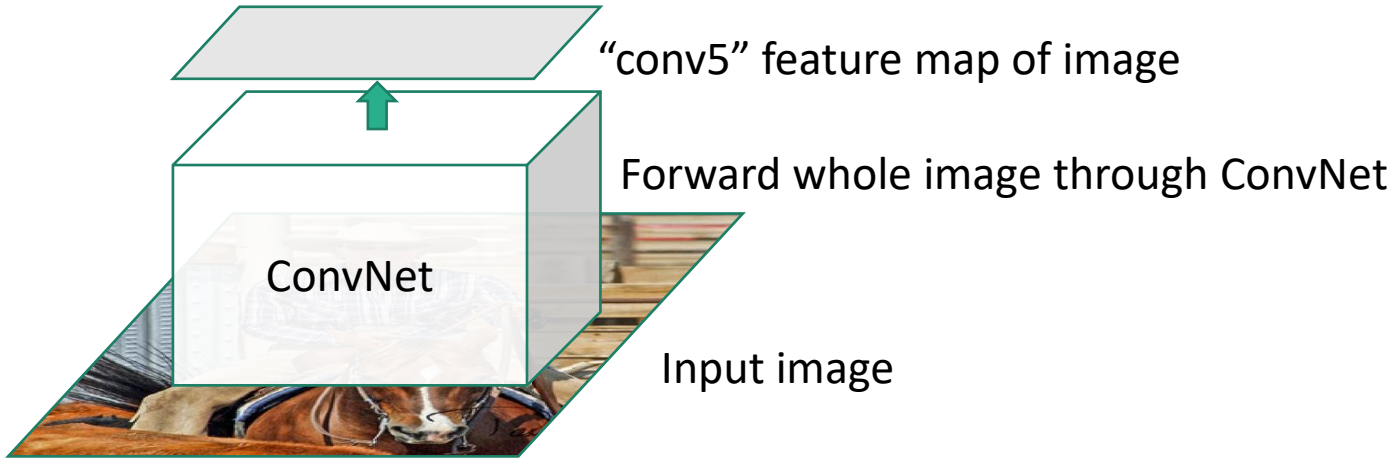
Girshick et al, "Rich feature hierarchies for accurate object detection and semantic s egmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Fast R-CNN

Input image

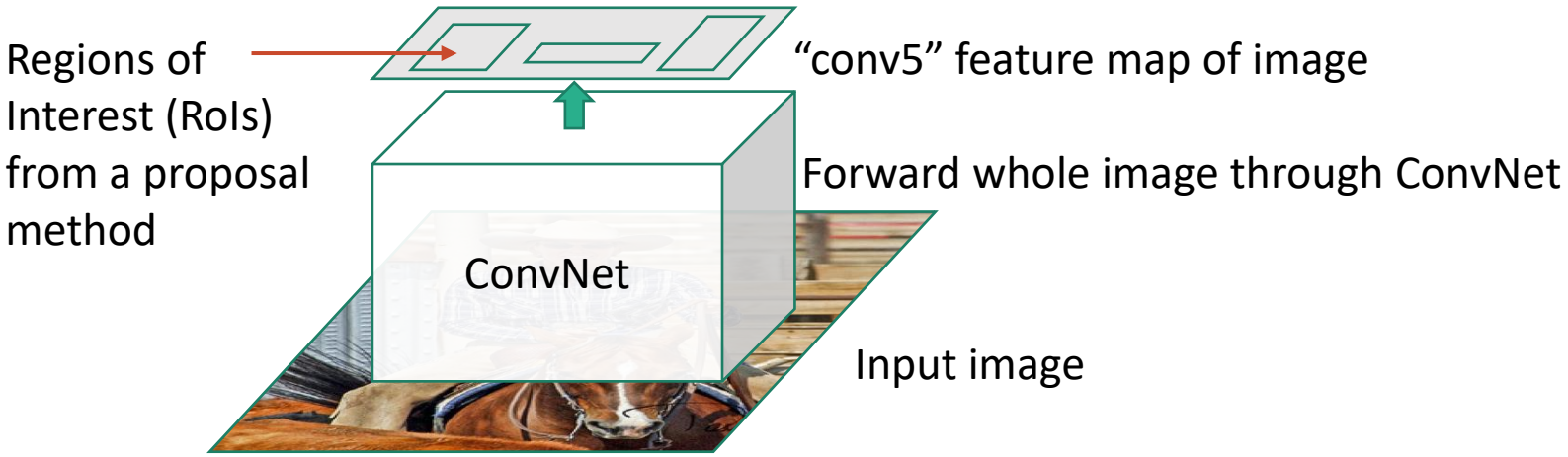
Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Fast R-CNN



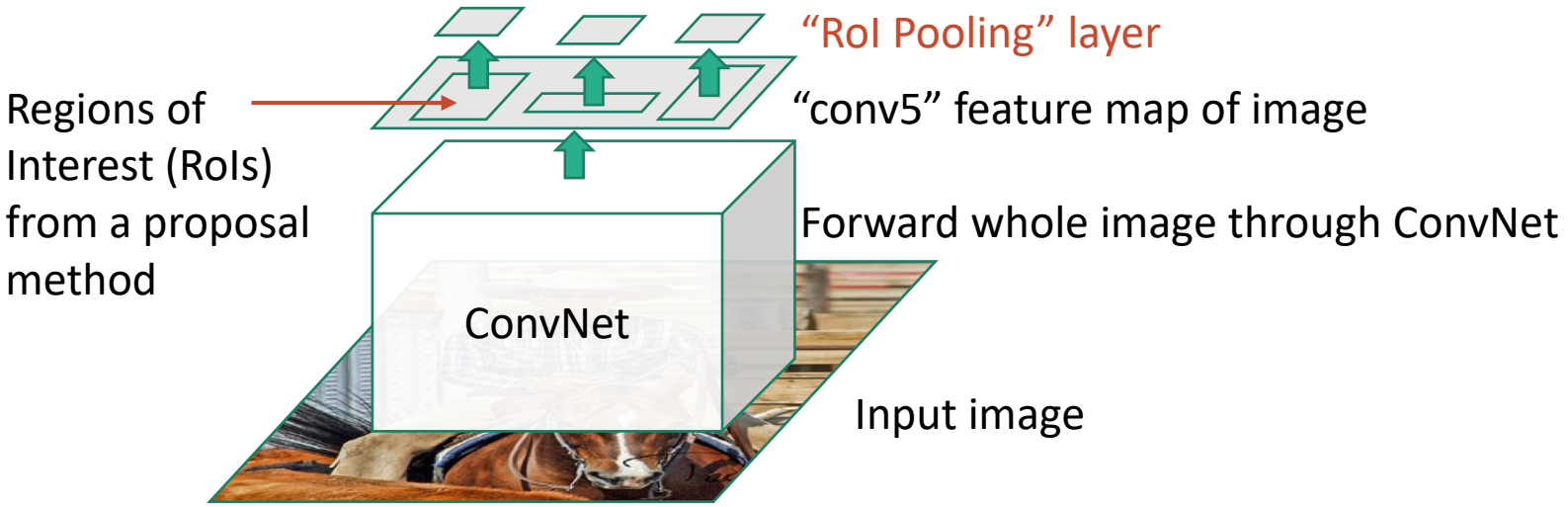
Girshick, “Fast R-CNN”, ICCV 2015.

Fast R-CNN



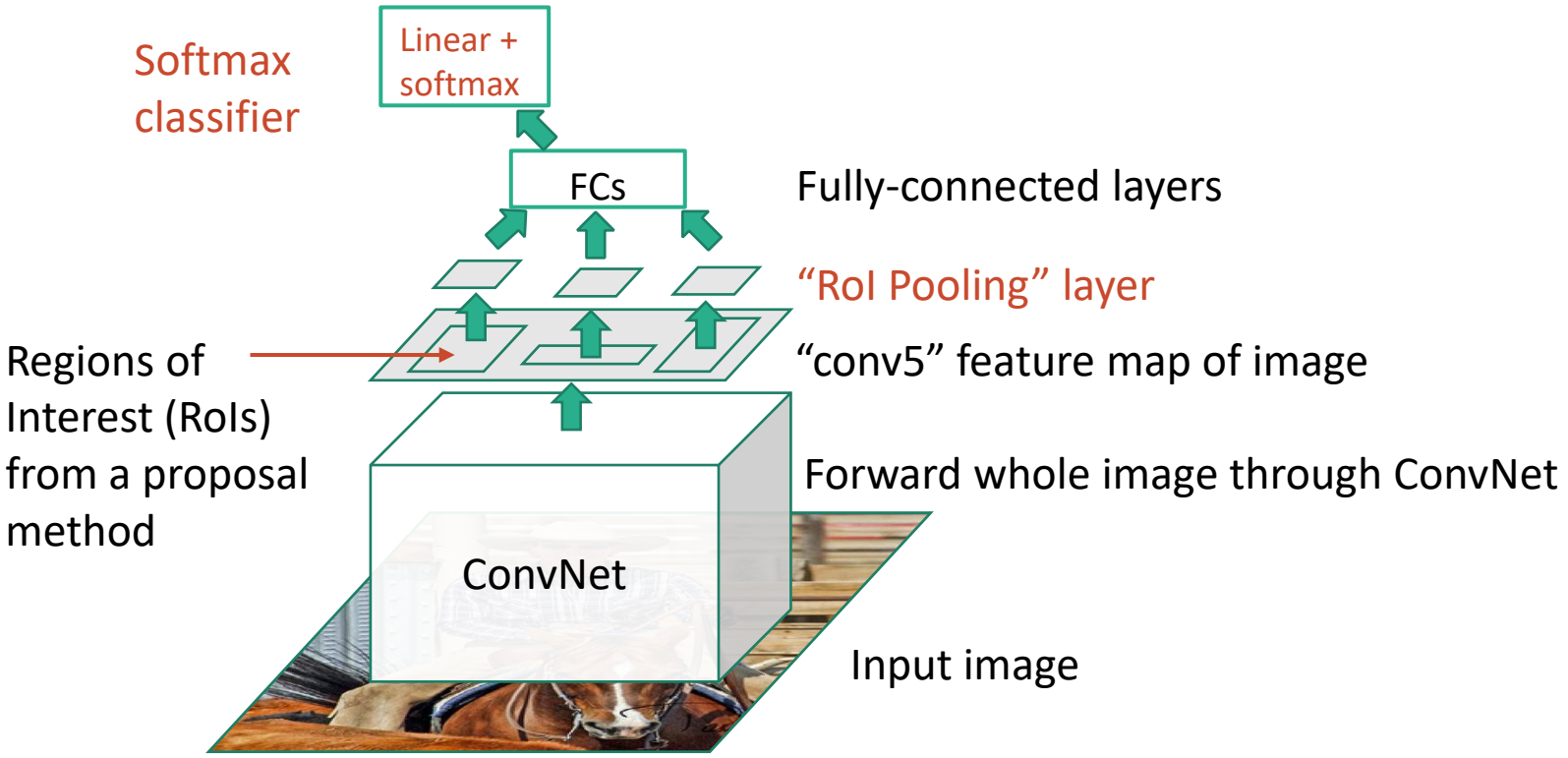
Girshick, "Fast R-CNN", ICCV 2015.

Fast R-CNN



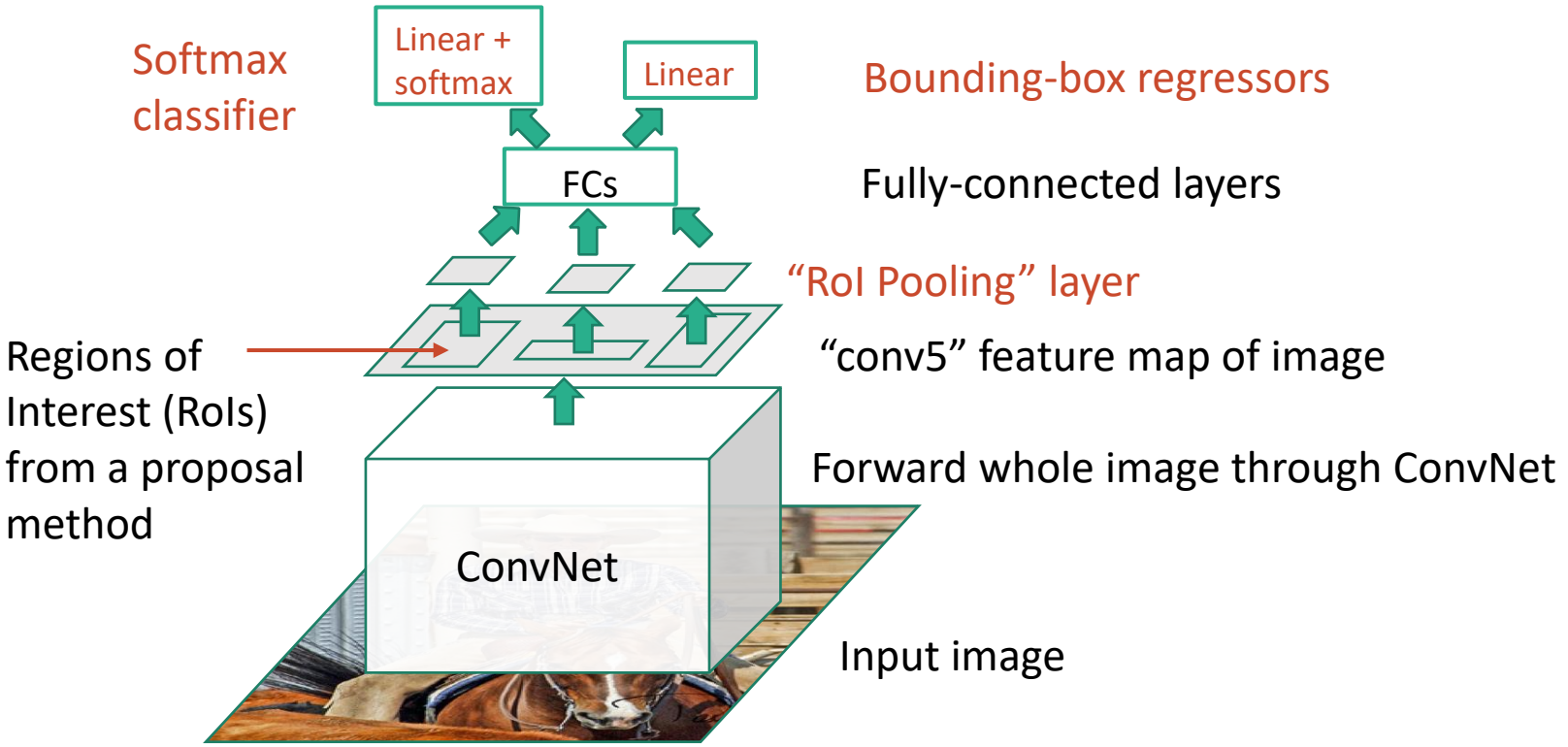
Girshick, "Fast R-CNN", ICCV 2015.

Fast R-CNN



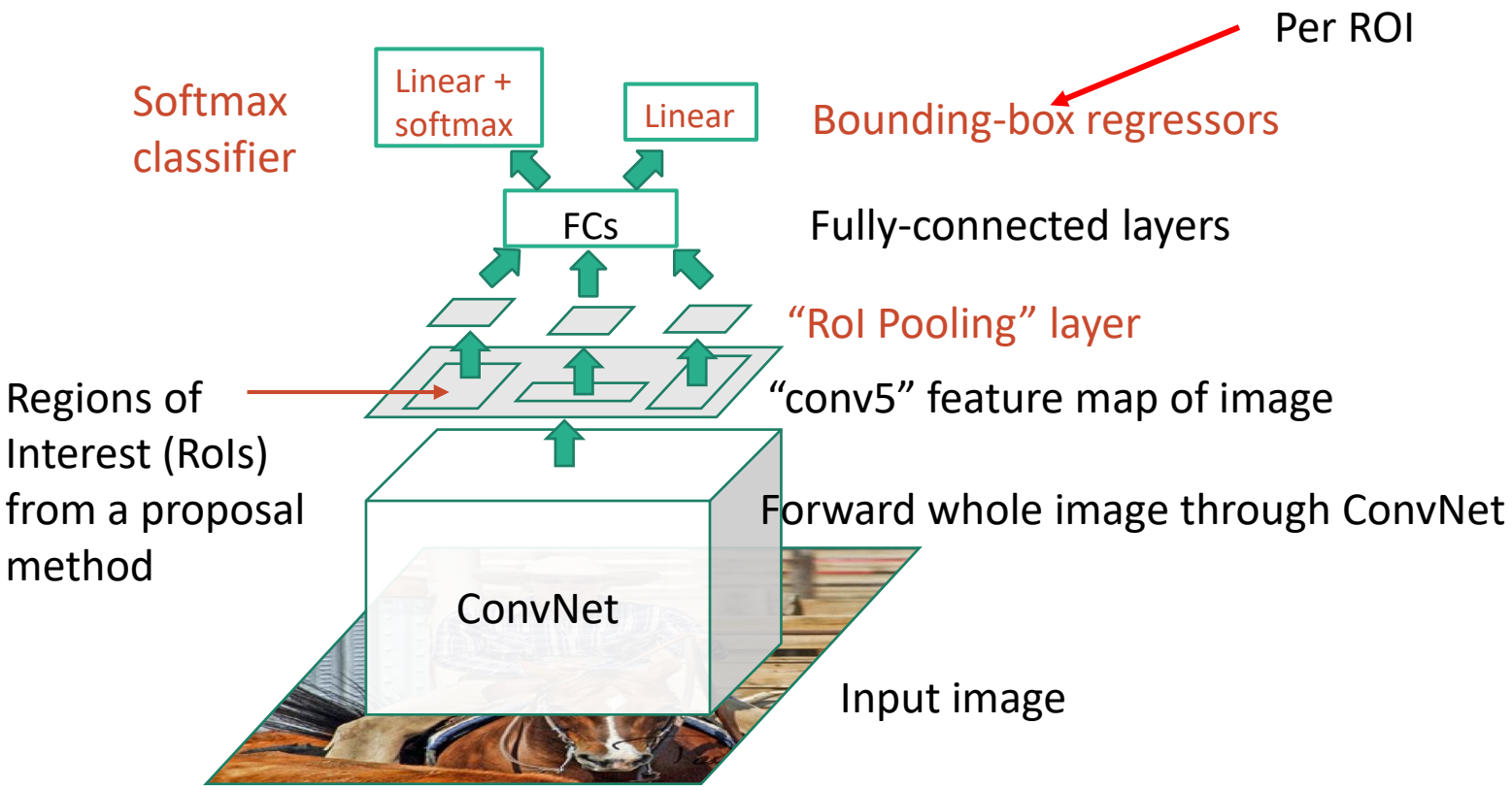
Girshick, "Fast R-CNN", ICCV 2015.

Fast R-CNN



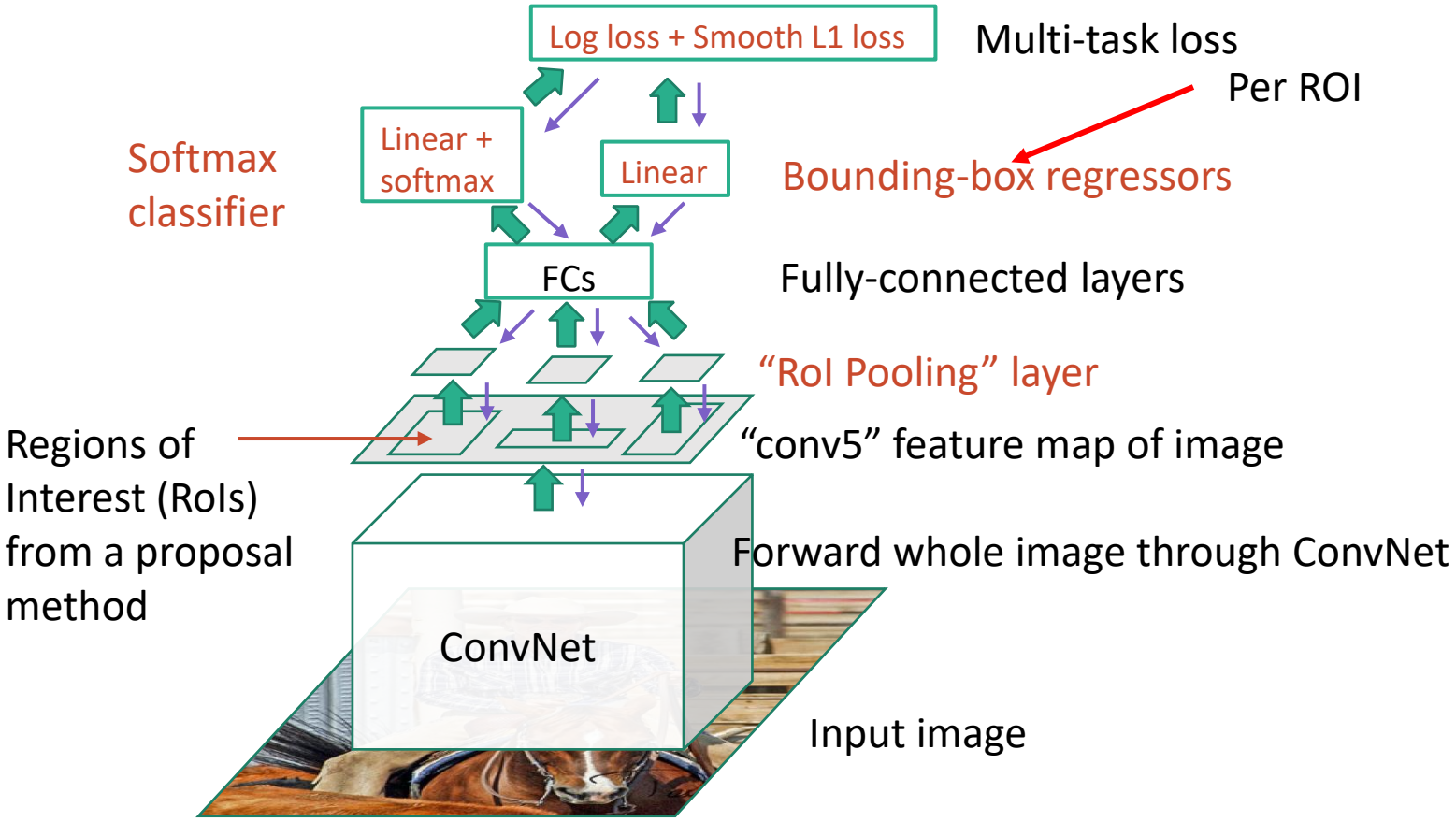
Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015; source. Reproduced with permission.

Fast R-CNN



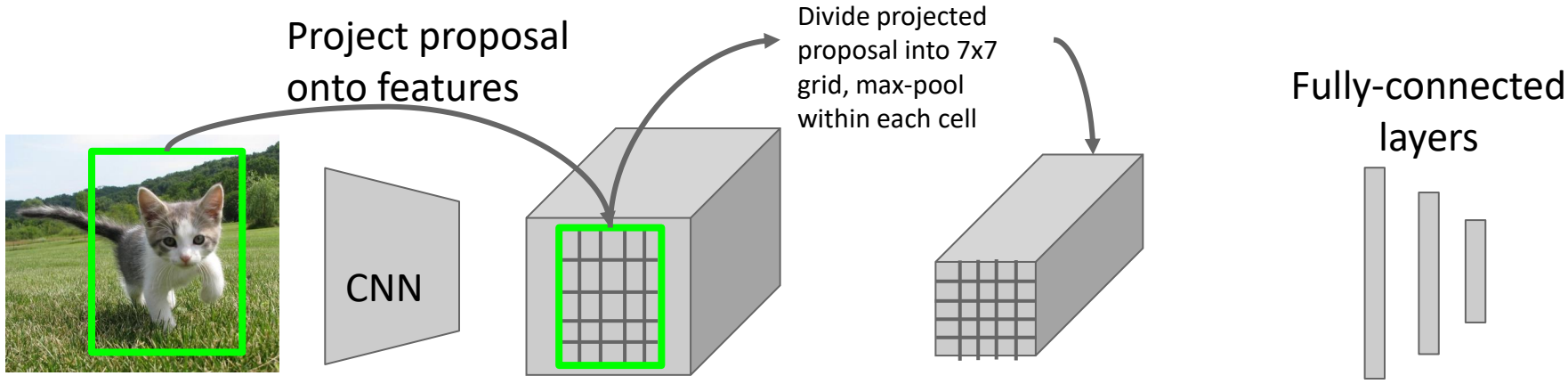
Girshick, "Fast R-CNN", ICCV 2015.

Fast R-CNN (Training)



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

Fast R-CNN: RoI Pooling



Hi-res input image:
3 x 640 x 480
with region proposal

Hi-res conv features:
512 x 20 x 15;

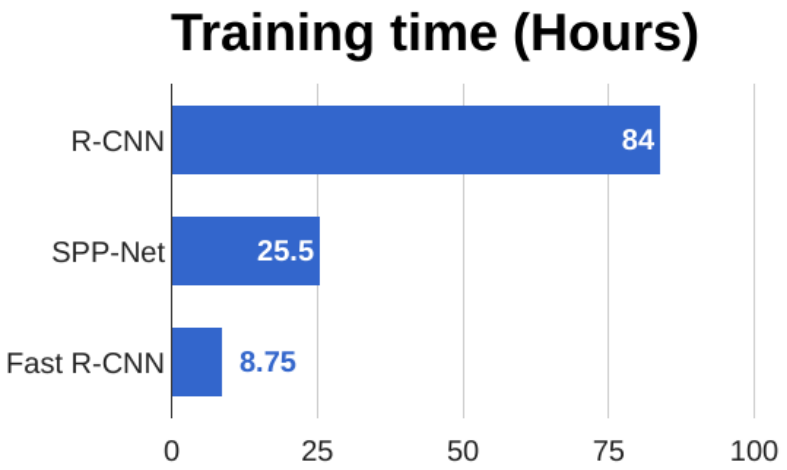
Projected region proposal is e.g.
512 x 18 x 8
(varies per proposal)

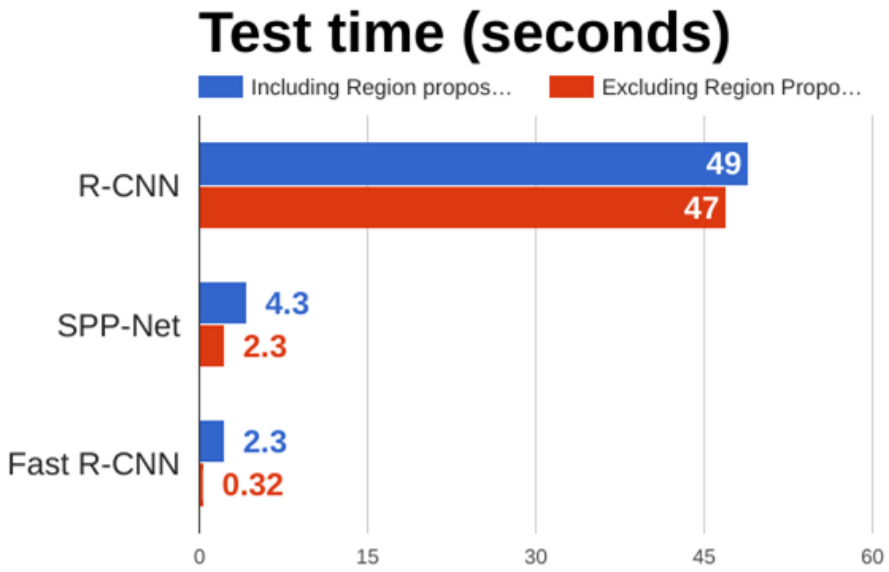
RoI conv features:
512 x 7 x 7
for region proposal

Fully-connected layers expect low-res conv features:
512 x 7 x 7

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

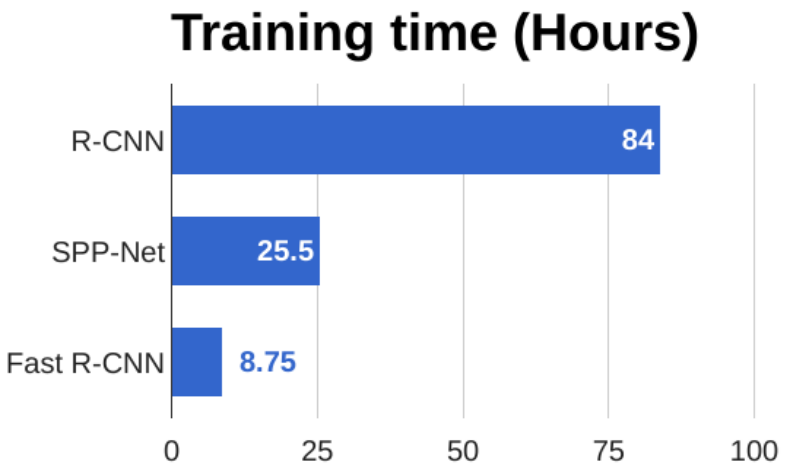
R-CNN vs SPP vs Fast R-CNN

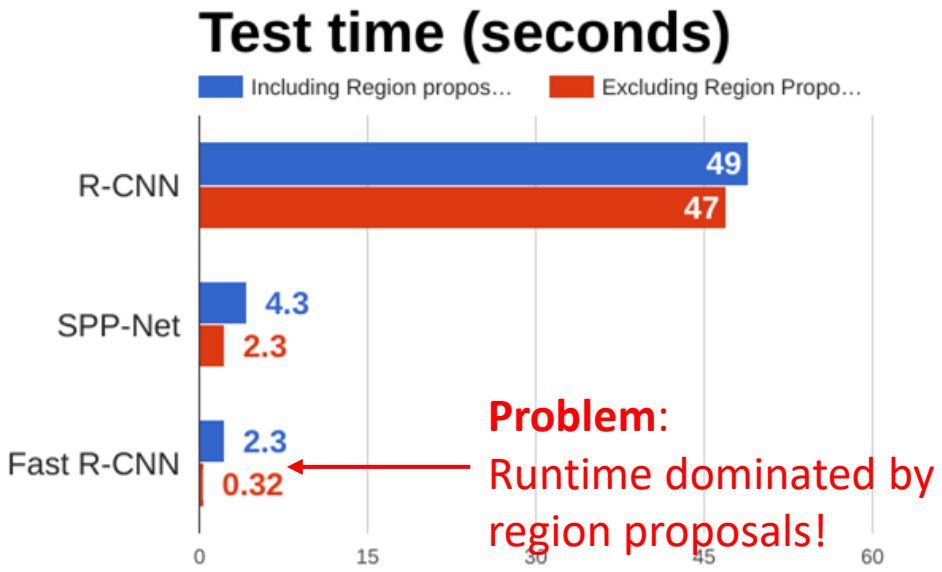




Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
He et al, "Spatial pyramid pooling in deep convolutional networks for visual recognition", ECCV 2014
Girshick, "Fast R-CNN", ICCV 2015

R-CNN vs SPP vs Fast R-CNN





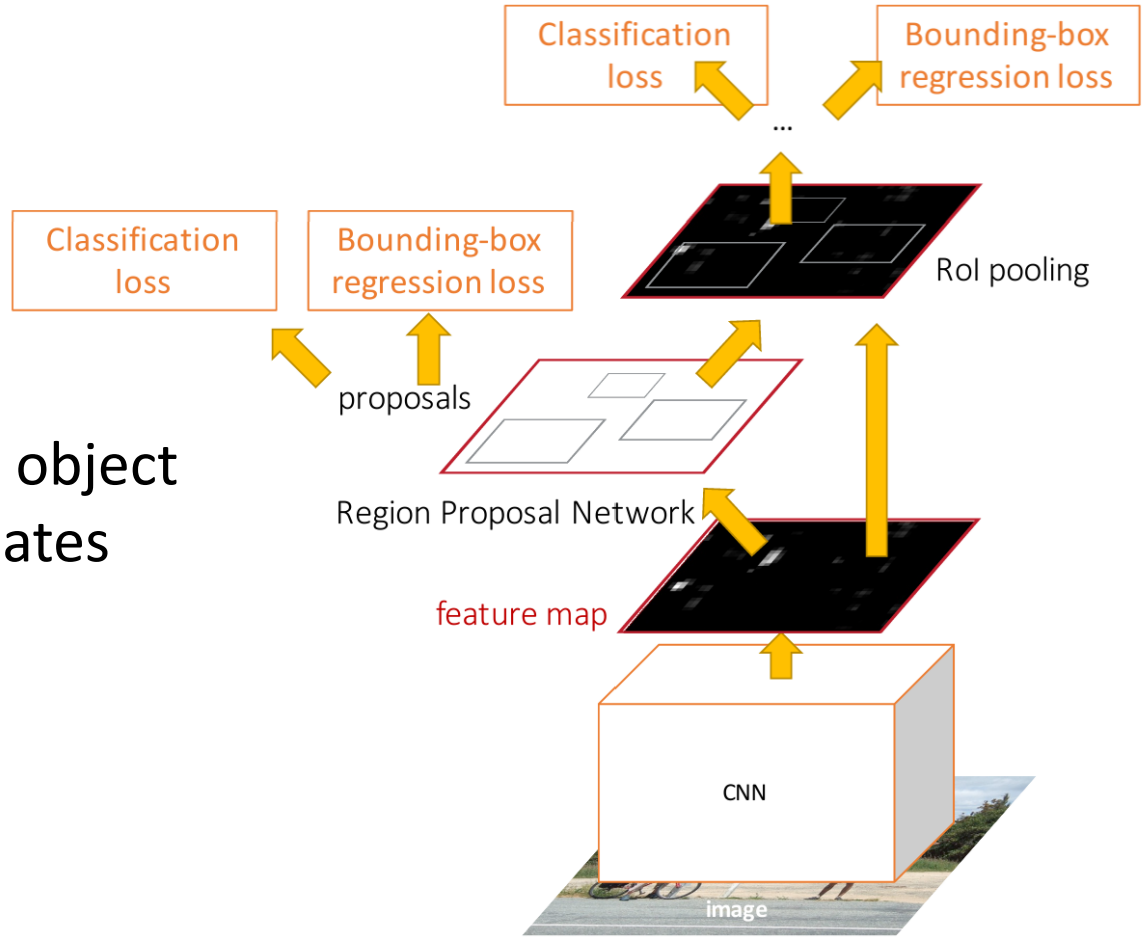
Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
He et al, "Spatial pyramid pooling in deep convolutional networks for visual recognition", ECCV 2014
Girshick, "Fast R-CNN", ICCV 2015

Faster R-CNN

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

Jointly train with 4 losses:

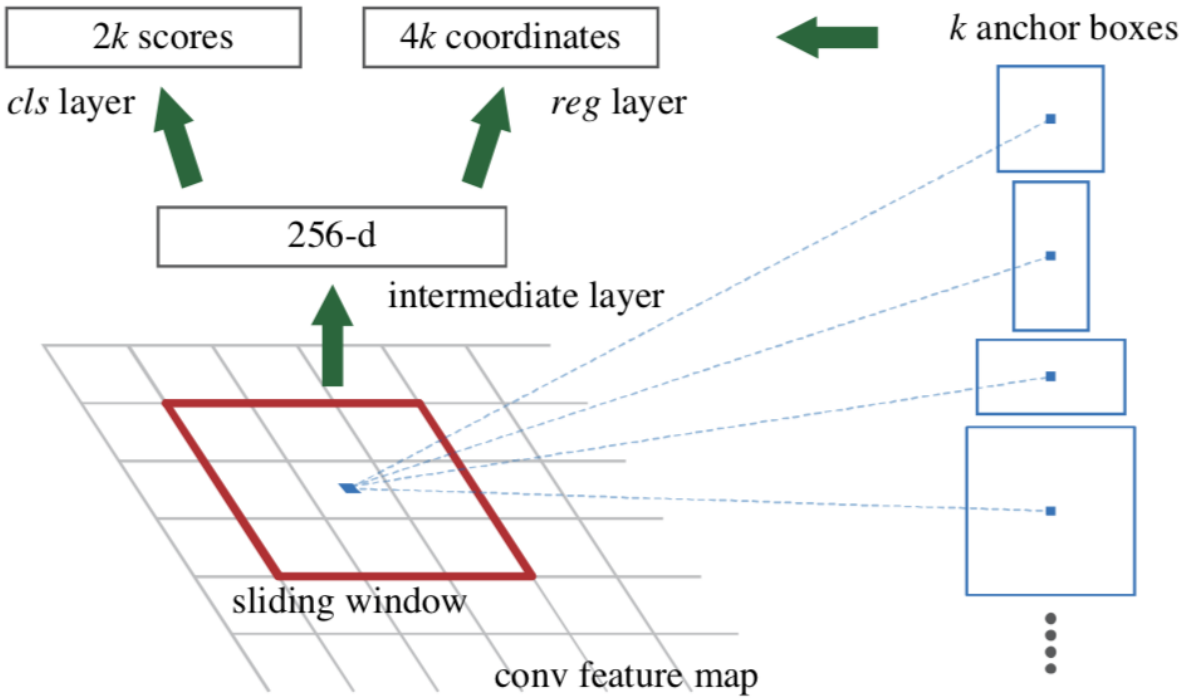
1. RPN classify object / not object
2. RPN regress box coordinates
3. Final classification score (object classes)
4. Final box coordinates



Ren et al, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NIPS 2015
Figure copyright 2015, Ross Girshick; reproduced with permission

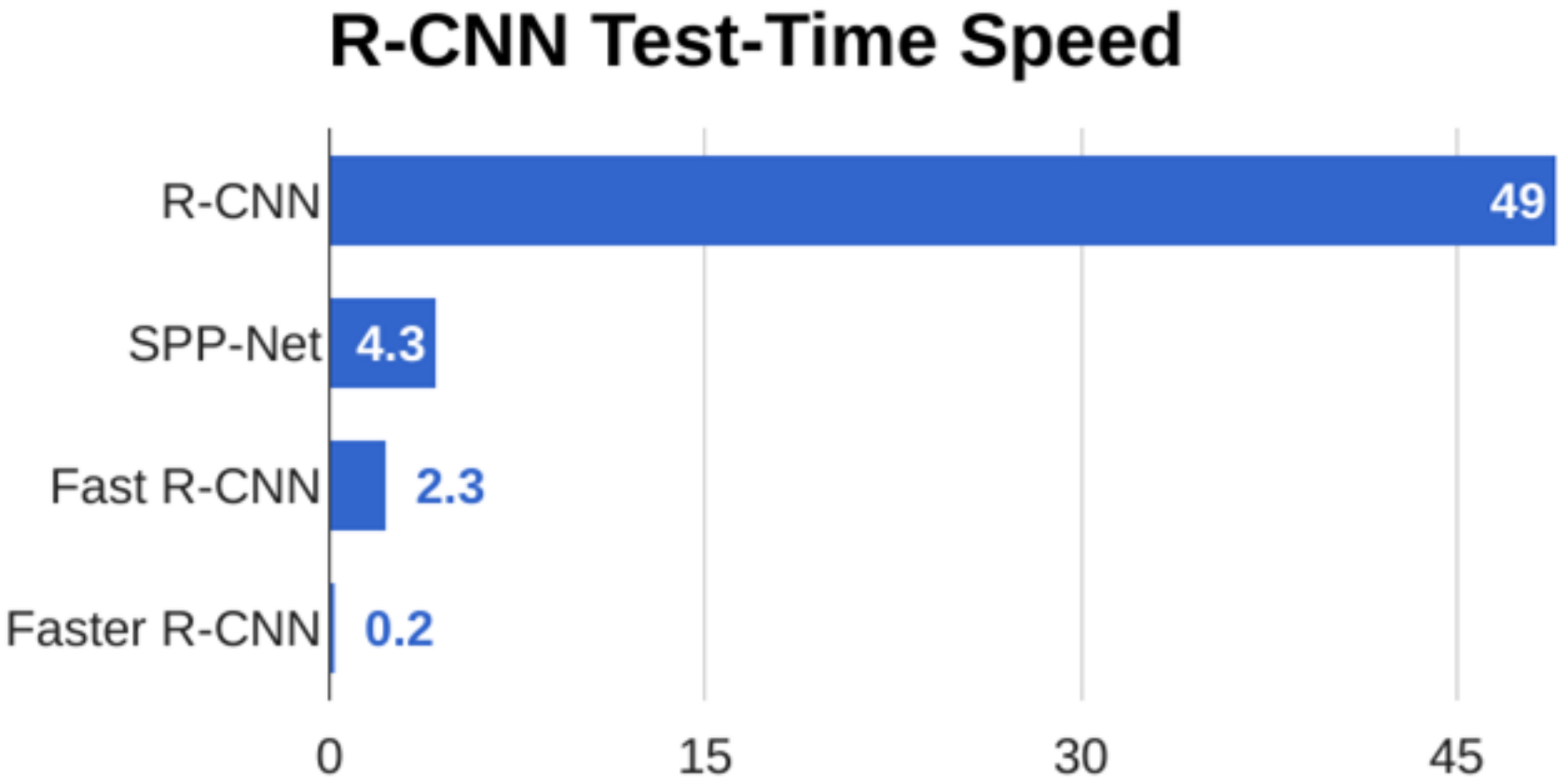
Faster R-CNN

Region Proposal Network (RPN)



Ren et al, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NIPS 2015
Figure copyright 2015, Ross Girshick; reproduced for educational purposes.

Faster R-CNN



Today's Lecture

- **What's Object Detection?**
- **R-CNN / Fast RCNN / Faster RCNN**
- SSD / YOLO