Object Detection

Lecture 11 - Feb 21, 2023

Ilija Radosavovic

UC Berkeley

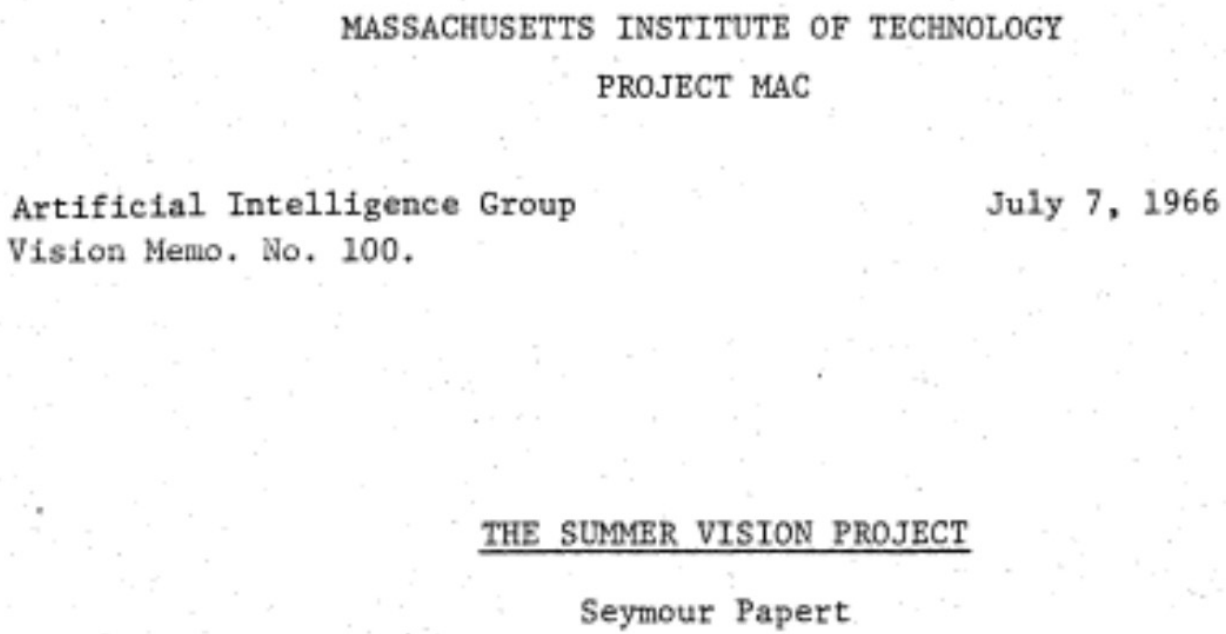
Today’s Agenda

- A brief history of object detection
- Modern object detection
- Beyond bounding boxes
- New trends

Today's Agenda

- A brief history of object detection
- Modern object detection
- Beyond bounding boxes
- New trends

The Summer Vision Project



MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence Group — July 7, 1966
Vision Memo. No. 100.

THE SUMMER VISION PROJECT

Seymour Papert

1966

Figure credit: Larry Zitnick, Justin Johnson

The Summer Vision Project

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

The final goal is OBJECT IDENTIFICATION which will actually name objects by matching them with a vocabulary of known objects.

THE SUMMER VISION PROJECT

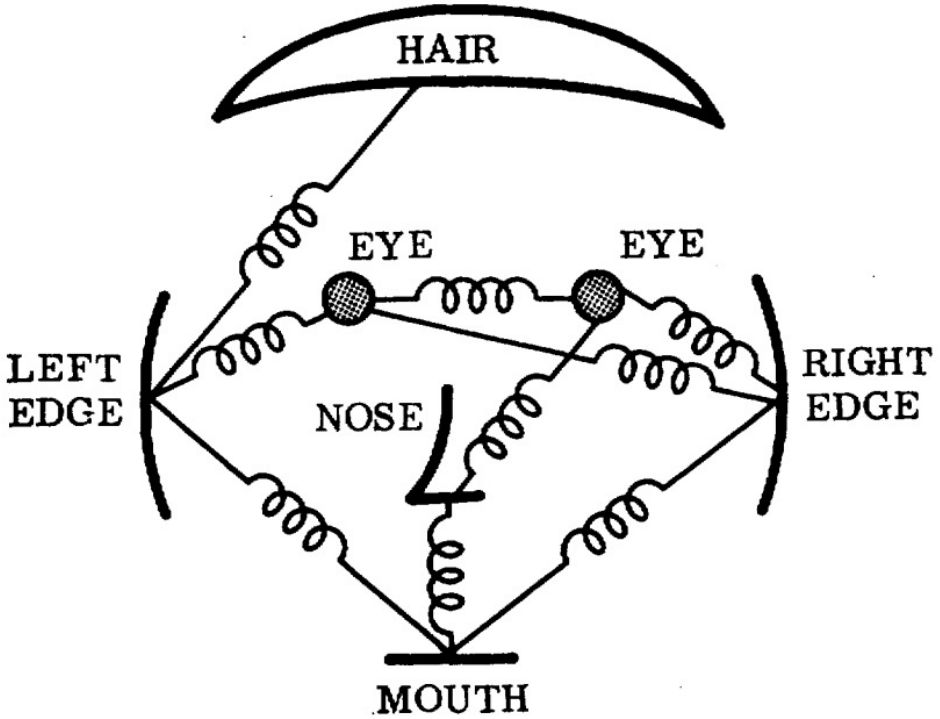
Seymour Papert

1966

Figure credit: Larry Zitnick, Justin Johnson

Pictorial Structures

[Fischler and Elschlager, 1973]

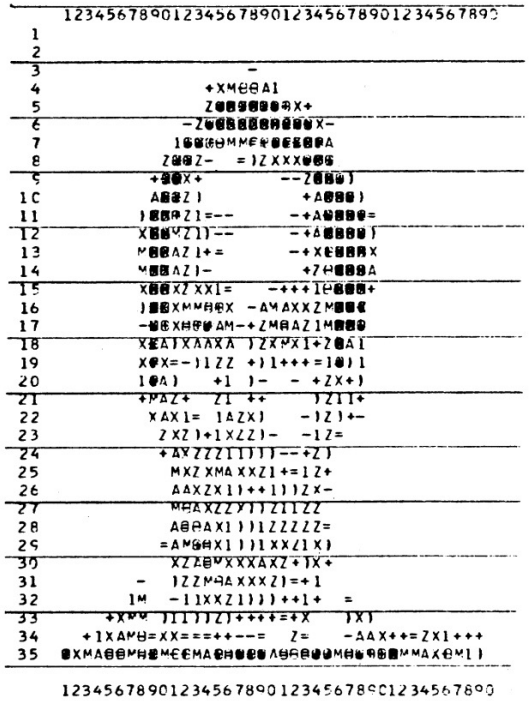


1973

Figure credit: Larry Zitnick

Pictorial Structures

[Fischler and Elschlager, 1973]



Original picture.

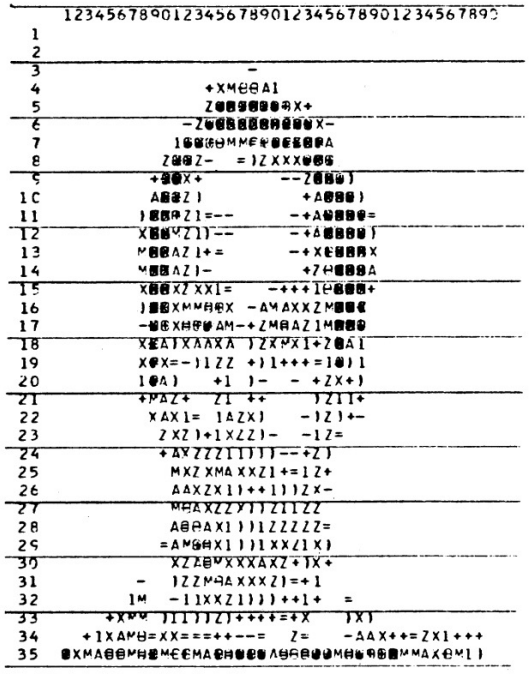
1973

Figure credit: Larry Zitnick

Pictorial Structures

[Fischler and Elschlager, 1973]

HAIR WAS LOCATED AT (6, 18)
L/EDGE WAS LOCATED AT (18, 10)
R/EDGE WAS LOCATED AT (18, 25)
L/EYE WAS LOCATED AT (17, 13)
R/EYE WAS LOCATED AT (17, 21)
NOSE WAS LOCATED AT (22, 18)
MOUTH WAS LOCATED AT (24, 17)



Original picture.

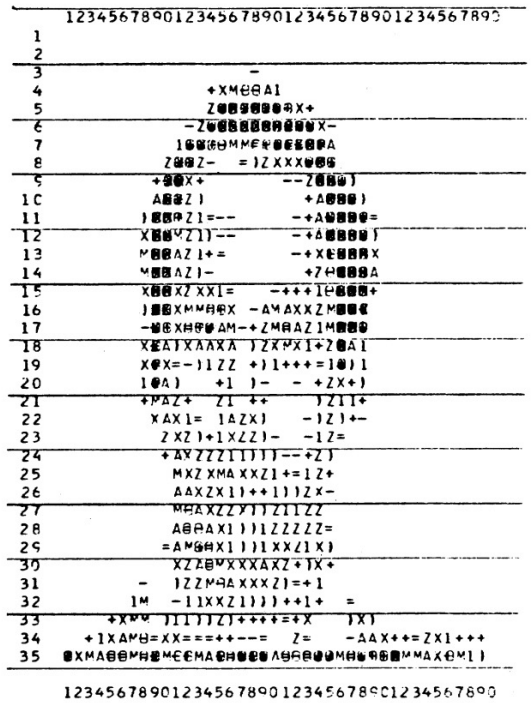
1973

Figure credit: Larry Zitnick

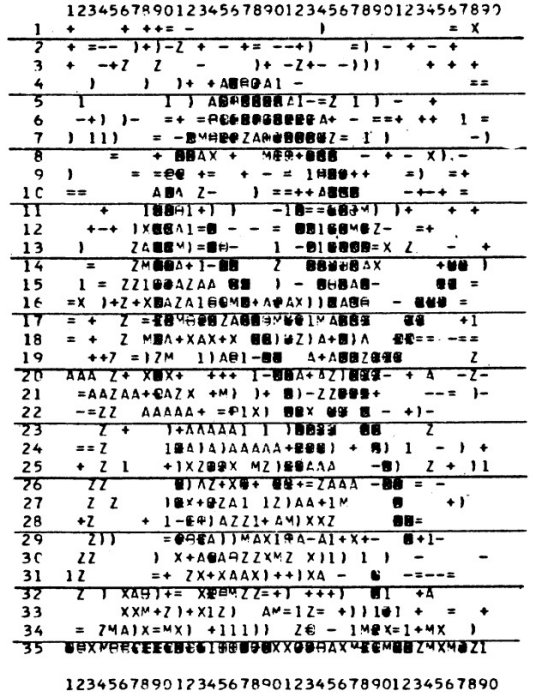
Pictorial Structures

[Fischler and Elschlager, 1973]

HAIR WAS LOCATED AT (6, 18)
L/EDGE WAS LOCATED AT (18, 10)
R/EDGE WAS LOCATED AT (18, 25)
L/EYE WAS LOCATED AT (17, 13)
R/EYE WAS LOCATED AT (17, 21)
NOSE WAS LOCATED AT (22, 18)
MOUTH WAS LOCATED AT (24, 17)



Original picture.



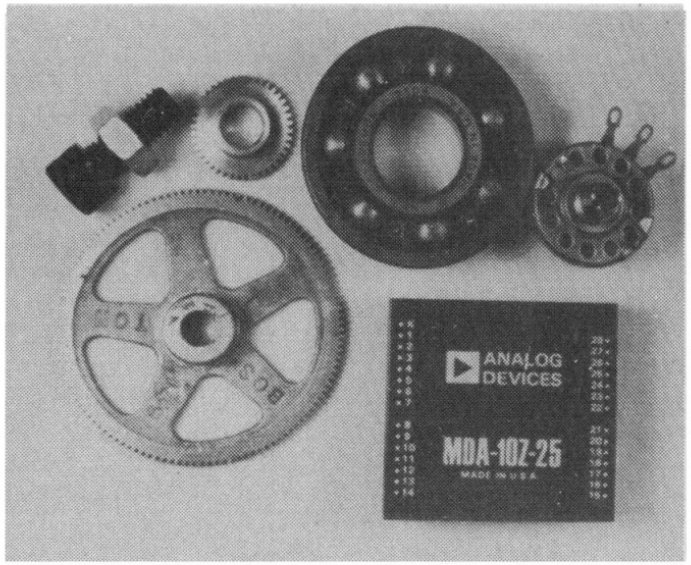
Noisy picture (sensed scene) as used in experiment.

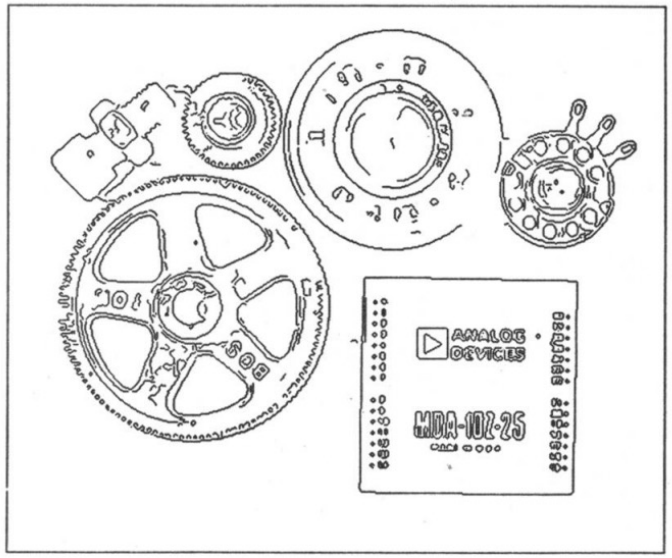
1973

Figure credit: Larry Zitnick

Edge Detection

[Canny, 1986]



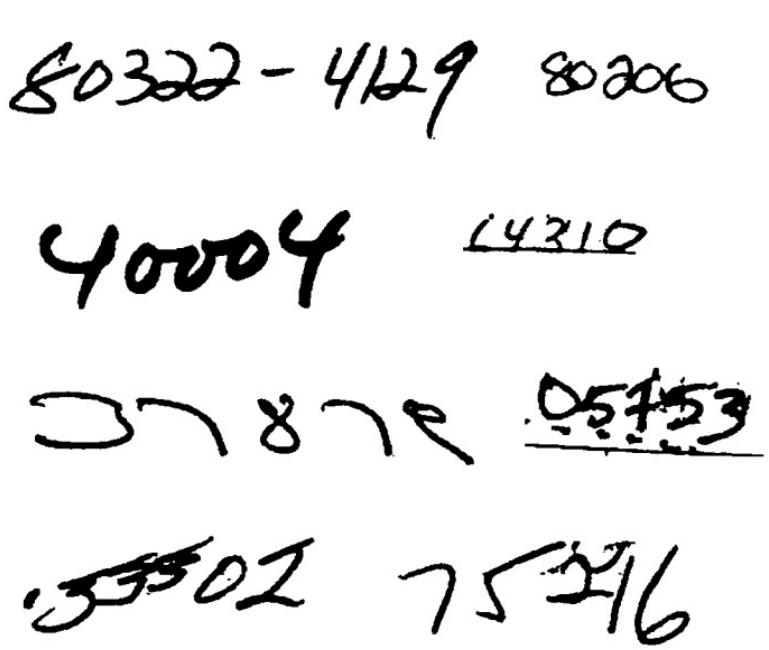


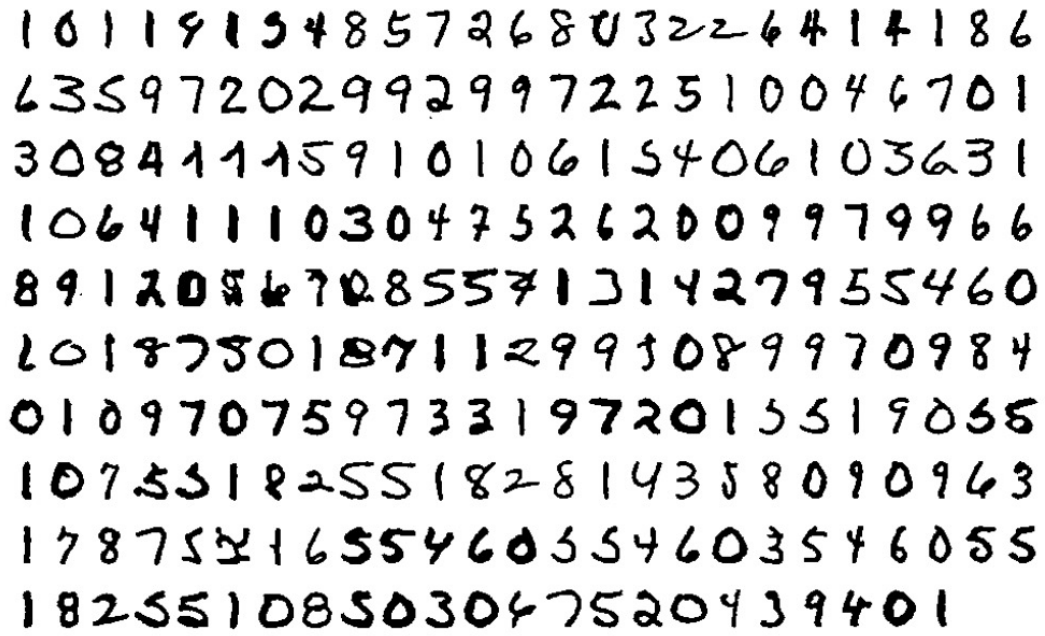
1986

Figure credit: Larry Zitnick

Digits Recognition

[LeCun et al, 1989]



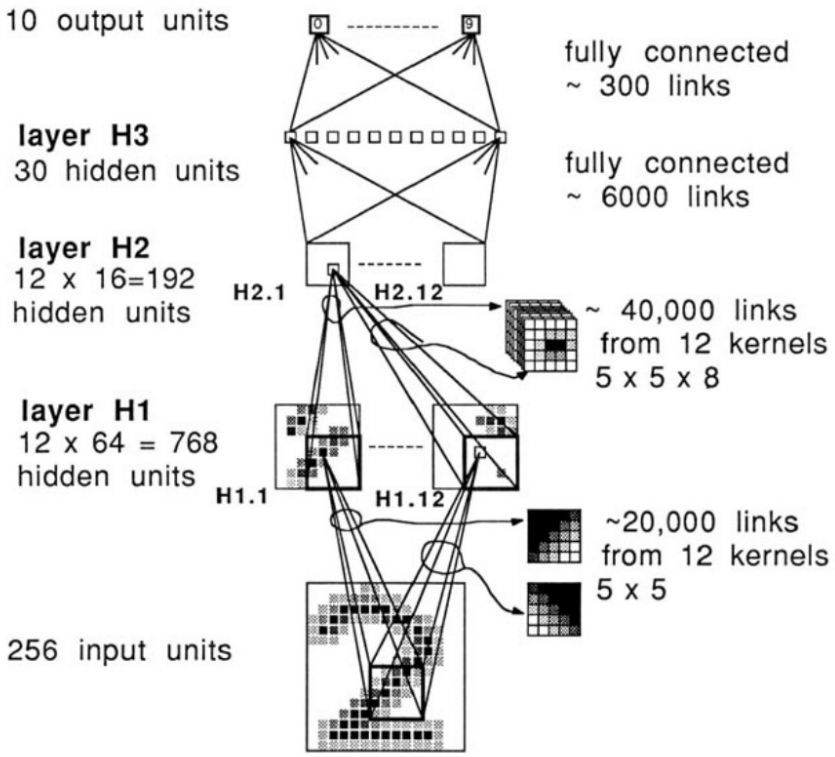


1989

Figure credit: Larry Zitnick

Digits Recognition

[LeCun et al, 1989]



1989

Figure credit: Larry Zitnick

Face Detection

[Rowley et al, 1998]

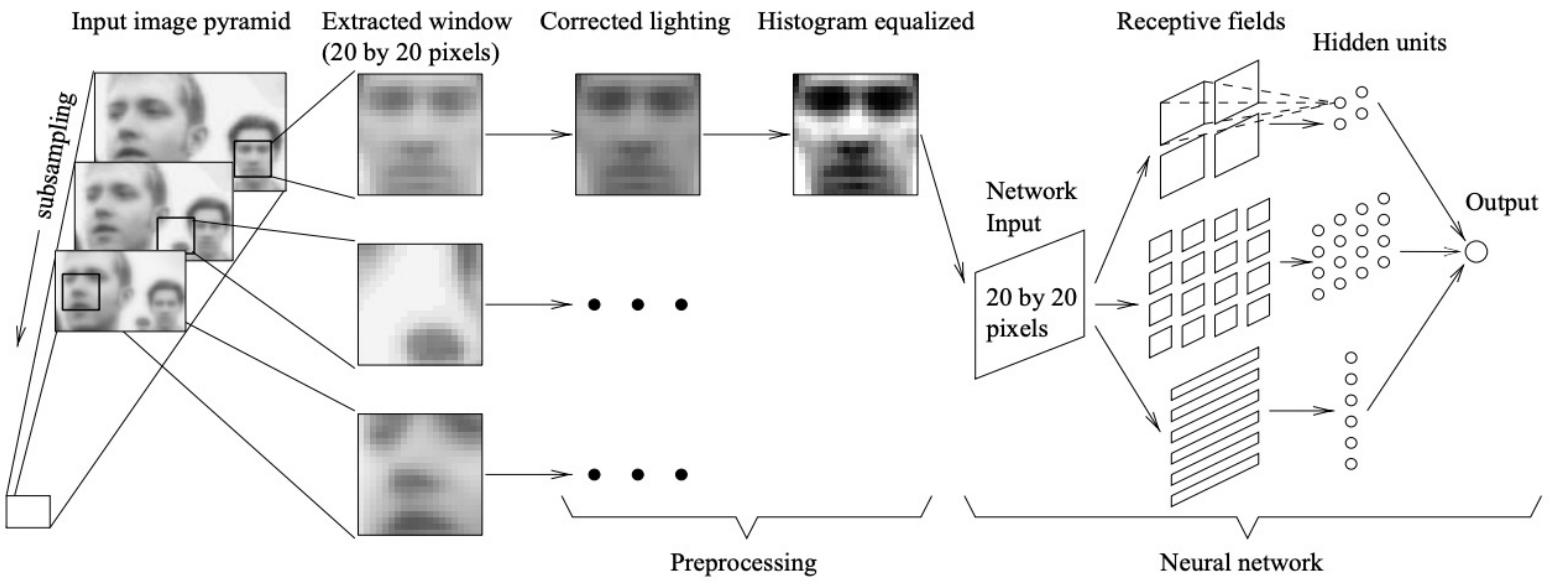
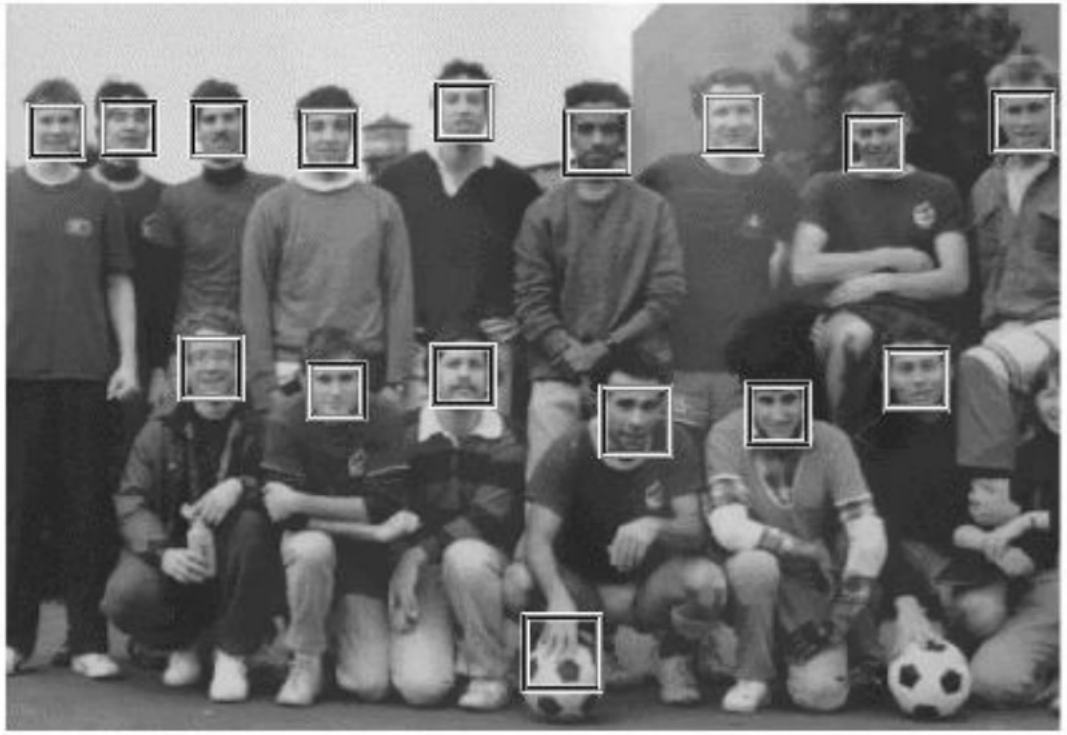


Figure credit: Larry Zitnick

Face Detection

[Viola and Jones, 2001]

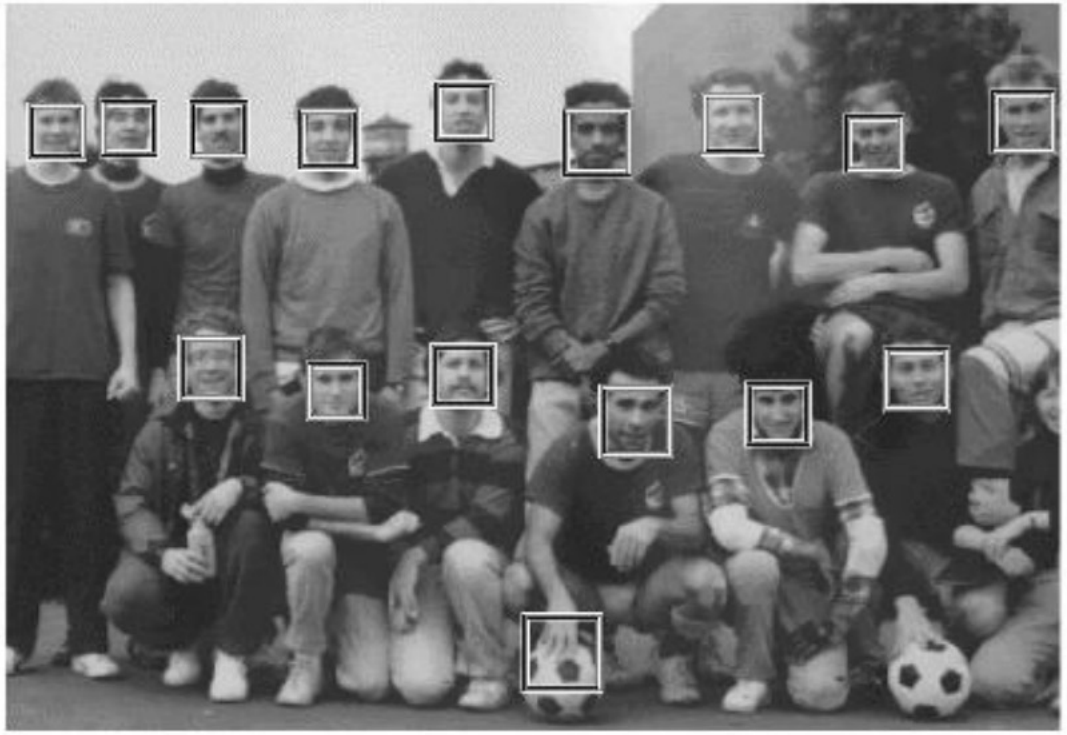


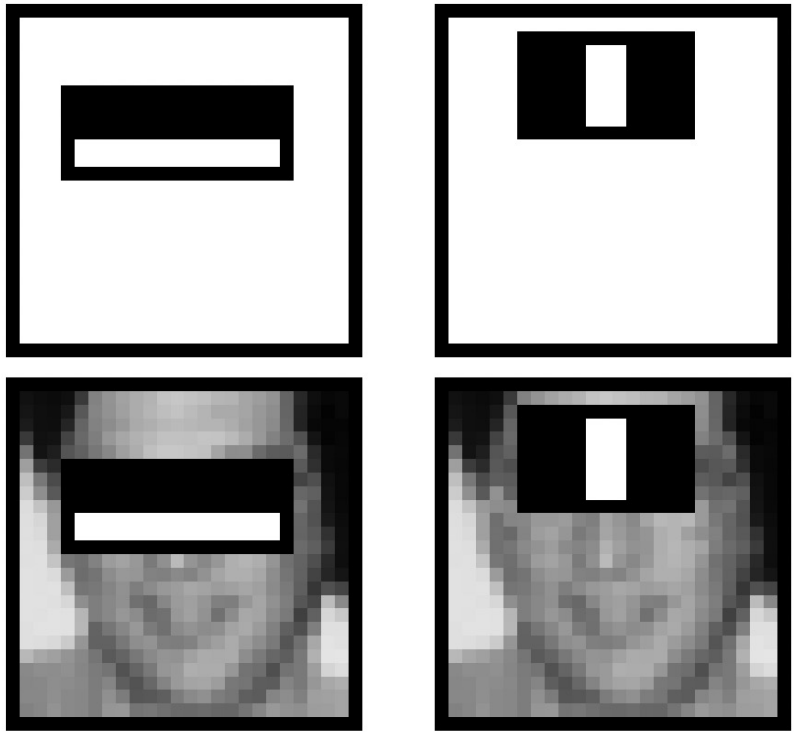
2001

Figure credit: Larry Zitnick

Face Detection

[Viola and Jones, 2001]





2001

Figure credit: Larry Zitnick

Pedestrian Detection

[Dalal and Triggs, 2005]

2005

Figure credit: Larry Zitnick

Pedestrian Detection

[Dalal and Triggs, 2005]

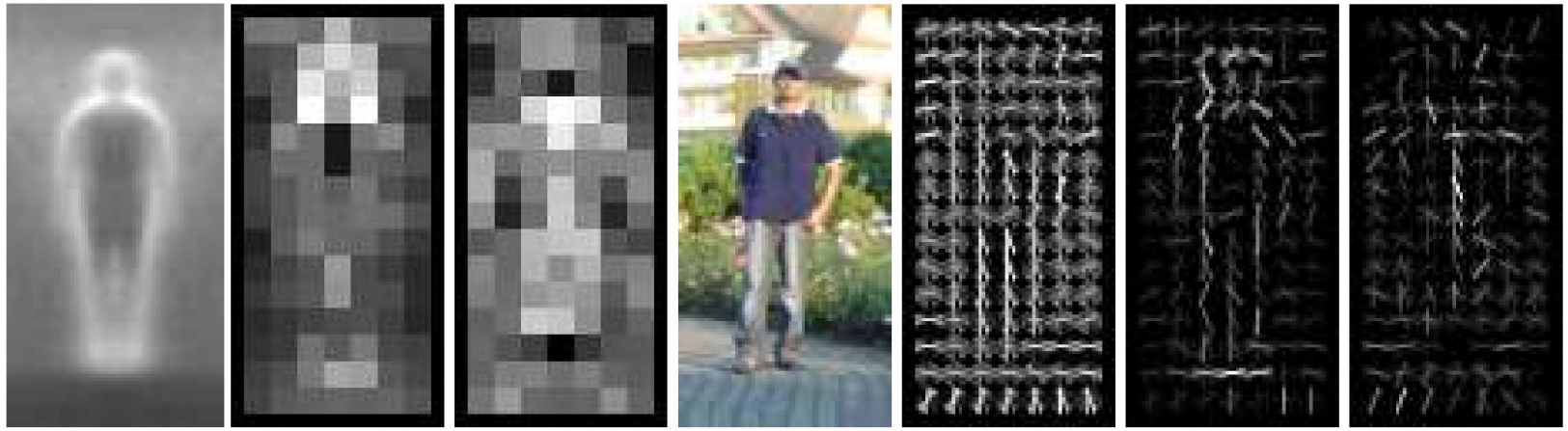
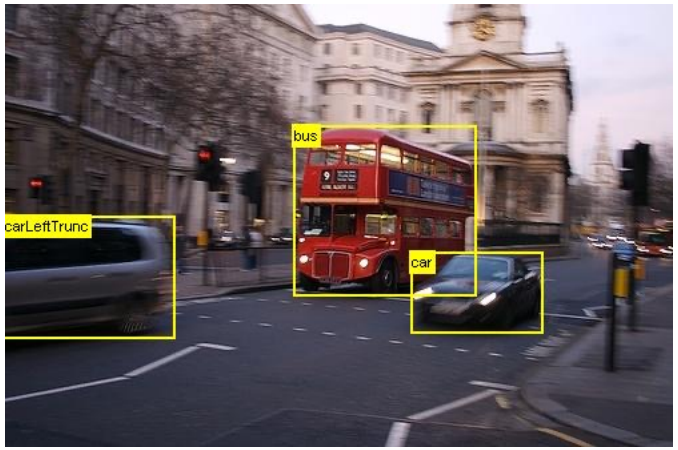
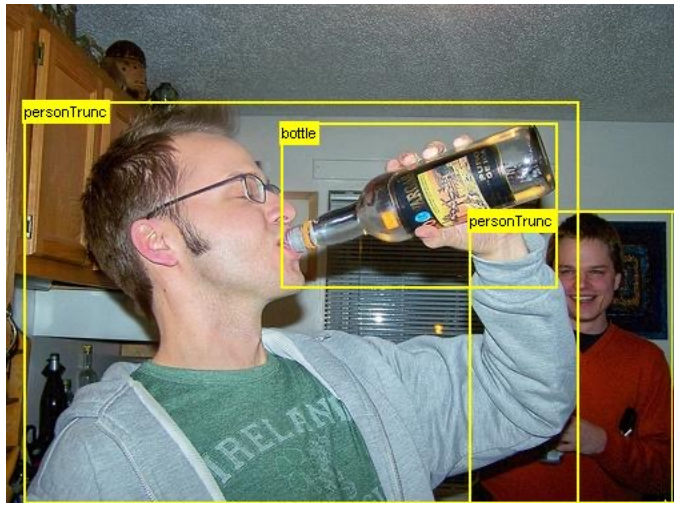


Figure credit: Larry Zitnick

PASCAL Visual Object Challenge

[Everingham et al, 2007]





2007

Figure credit: Larry Zitnick

Deformable Parts Model

[Felzenszwalb et al, 2008]

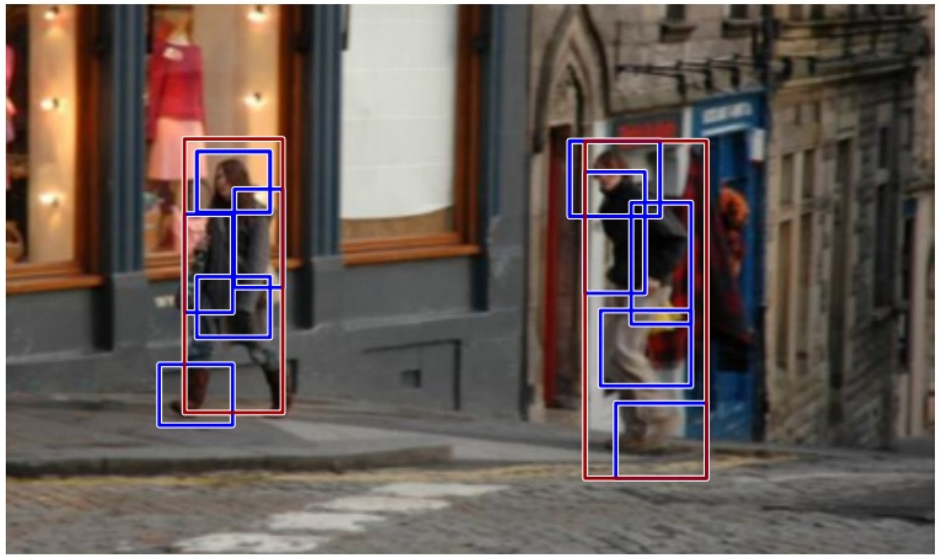
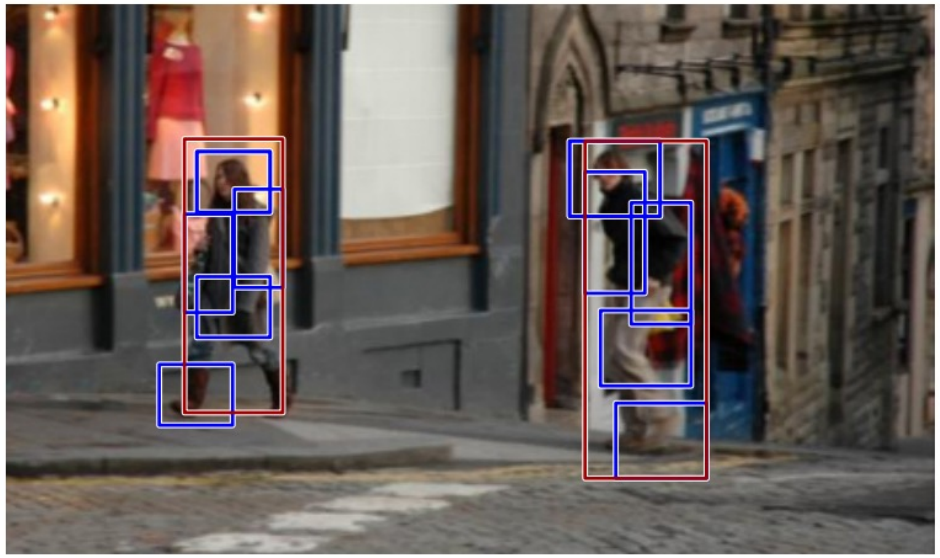


Figure credit: Larry Zitnick

Deformable Parts Model

[Felzenszwalb et al, 2008]



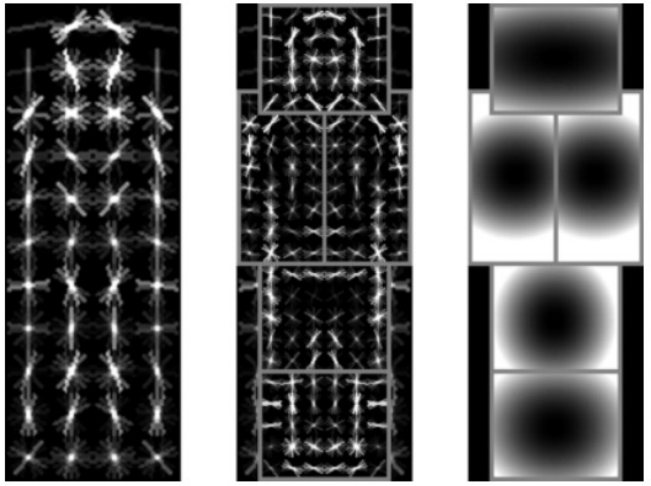


Figure credit: Larry Zitnick

ImageNet

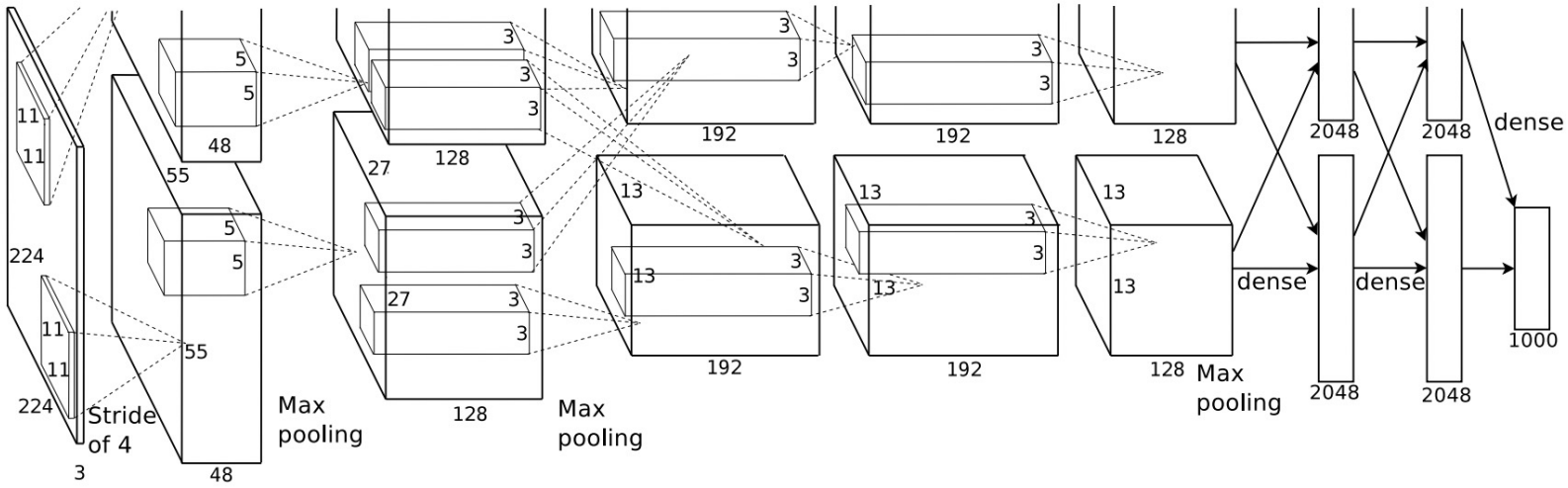
[Deng et al, 2009]

2009

Figure credit: Larry Zitnick

AlexNet

[Krizhevsky et al, 2012]

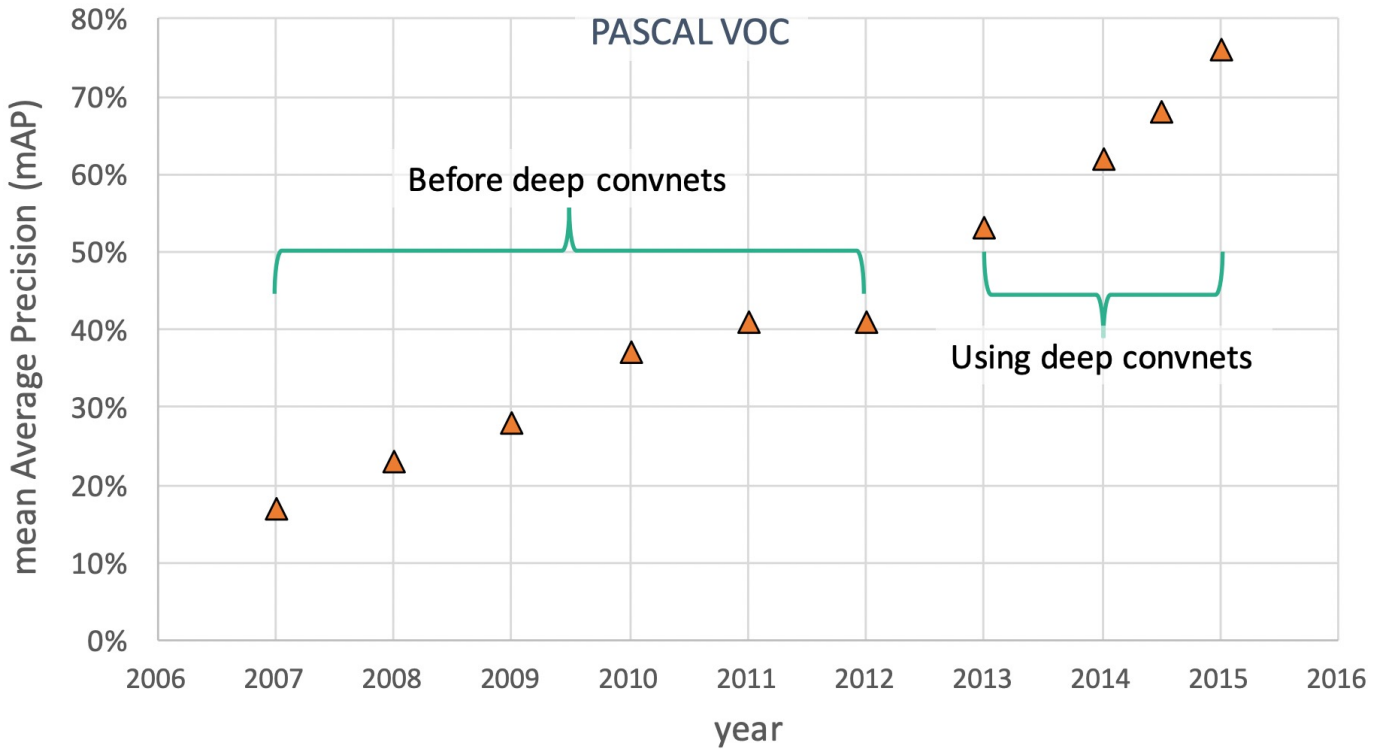


2012

Figure credit: Larry Zitnick

Object Detection Renaissance

[2013 - present]



2013

Figure credit: Ross Girshick

Today’s Agenda

- A brief history of object detection
- Modern object detection
- Beyond bounding boxes
- New trends

Definition: What objects are where?

Figure credit: Ross Girshick

Comparing Boxes

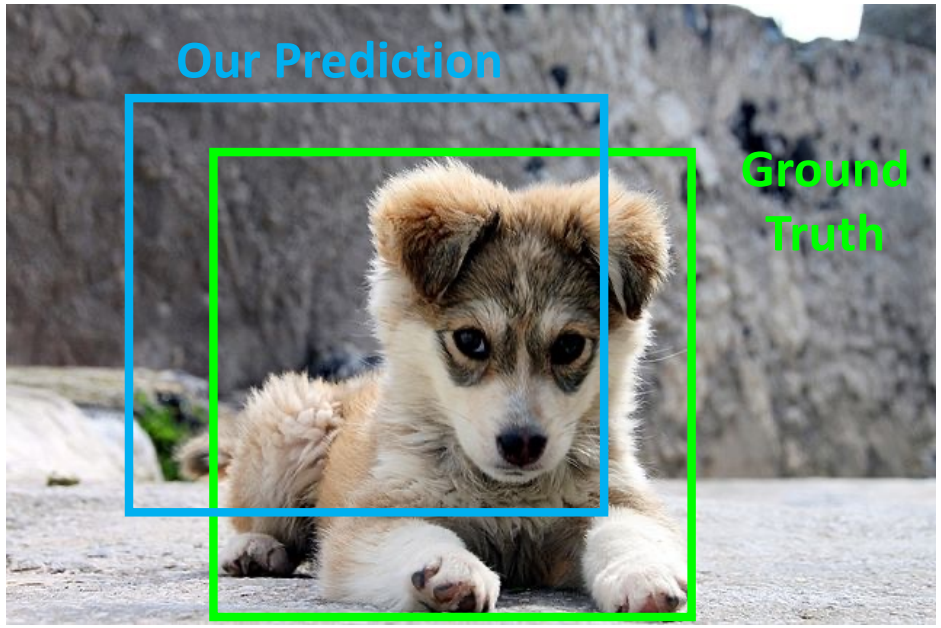
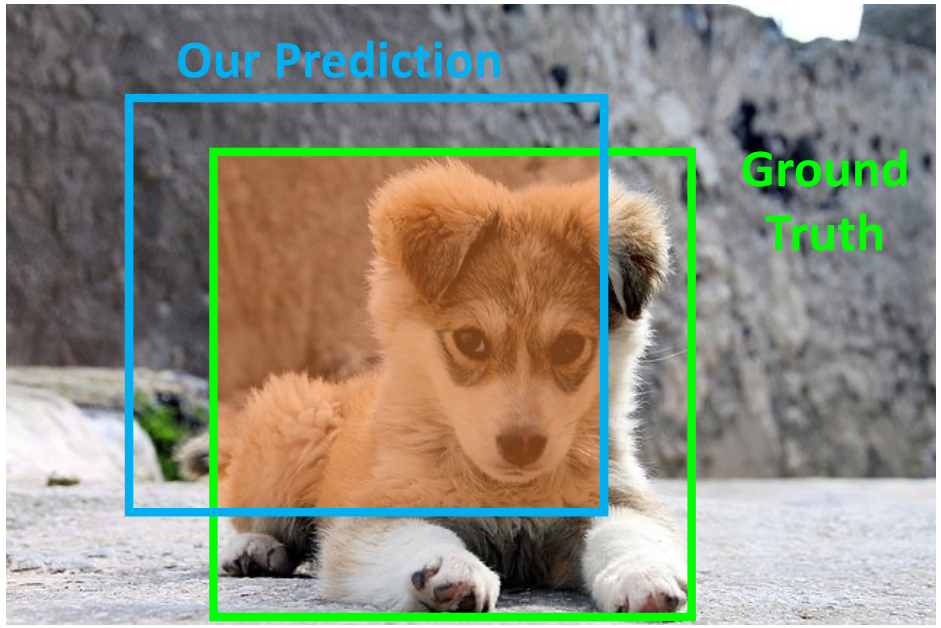
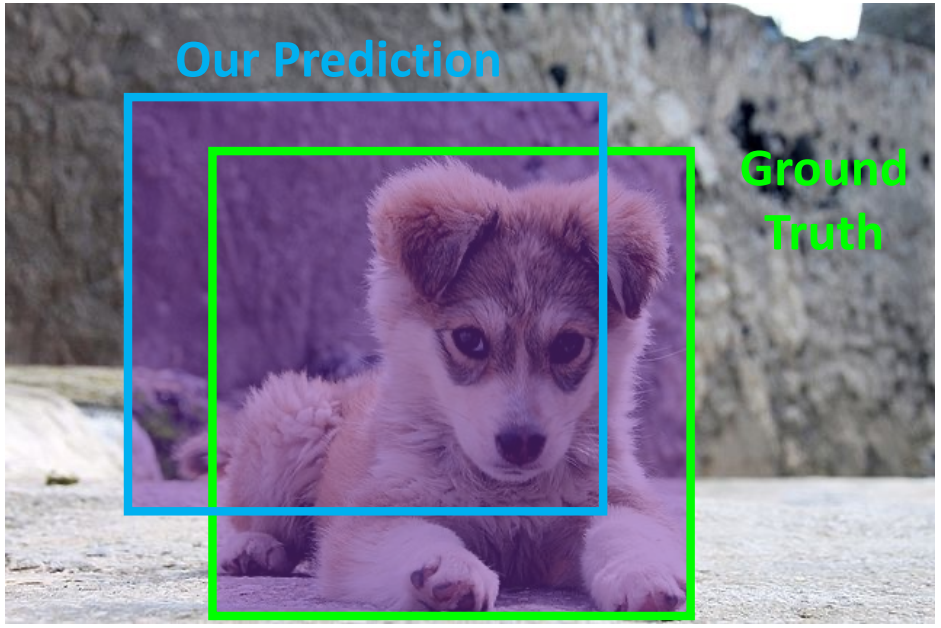


Figure credit: Justin Johnson

Comparing Boxes: Intersection over Union (IoU)

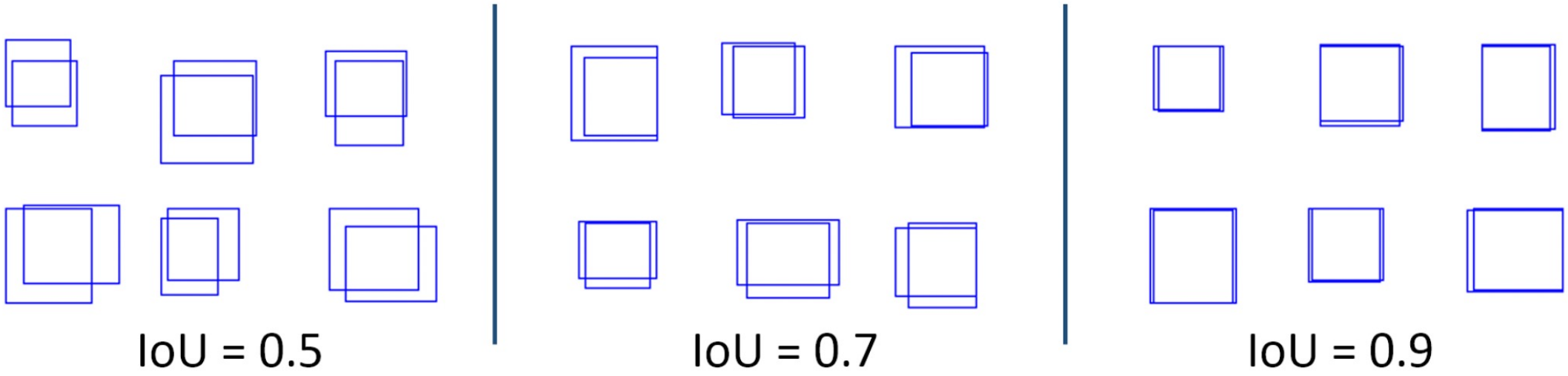




$$\text{IoU} = \frac{\textit{Area of Intersection}}{\textit{Area of Union}}$$

Figure credit: Justin Johnson

Comparing Boxes: Intersection over Union (IoU)



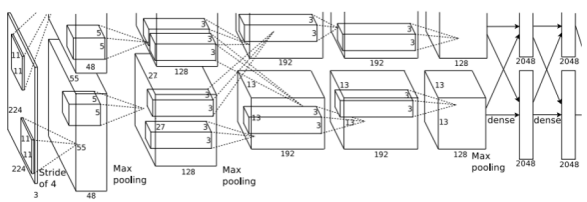
“decent”

“good”

“great”

Figure credit: Larry Zitnick, Piotr Dollár

Detecting a single object



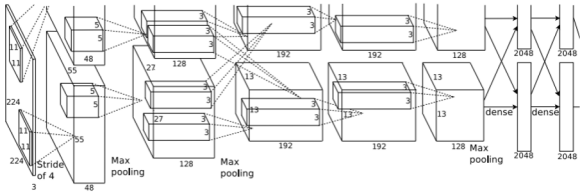
Vector:
4096

Treat localization as a regression problem!

Slide credit: Justin Johnson

Detecting a single object "What"

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

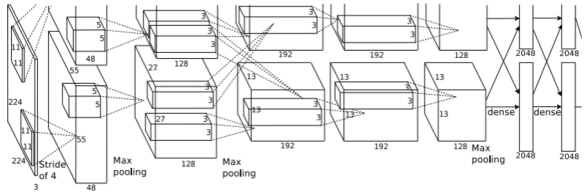
Correct label:
Cat

Treat localization as a regression problem!

Slide credit: Justin Johnson

Detecting a single object "What"

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

Treat localization as a regression problem!

Fully Connected:
4096 to 4

Box Coordinates
(x, y, w, h)

L2 Loss

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

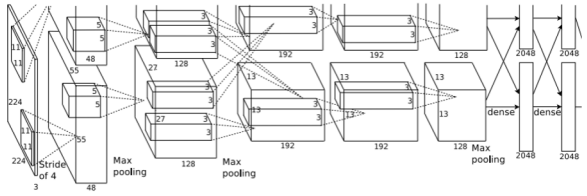
"Where"

Slide credit: Justin Johnson

Detecting a single object

"What"

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

Multitask Loss

Weighted Sum → Loss

$L = L_{cls} + \lambda L_{reg}$

Fully Connected: 4096 to 4

Box Coordinates
(x, y, w, h)

L2 Loss

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

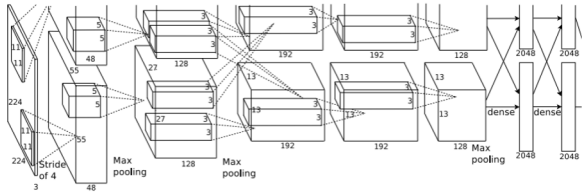
"Where"

Treat localization as a regression problem!

Slide credit: Justin Johnson

Detecting a single object

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

Multitask Loss

Weighted Sum → Loss

$L = L_{cls} + \lambda L_{reg}$

Fully Connected: 4096 to 4

Box Coordinates
(x, y, w, h)

L2 Loss

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

"What"

"Where"

Treat localization as a regression problem!

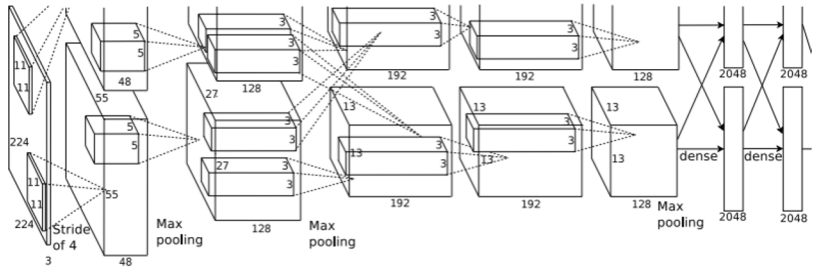
Problem:

Images can have multiple objects!

Slide credit: Justin Johnson

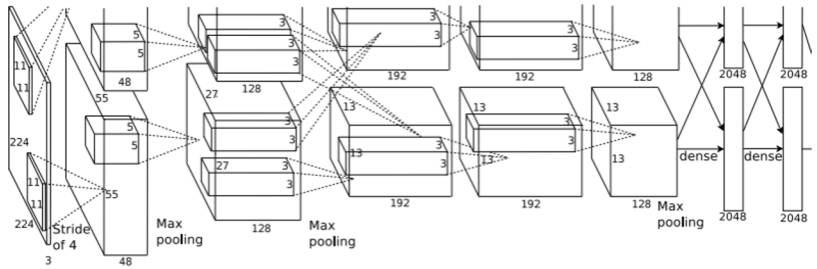
Detecting Multiple Objects

Need different numbers of outputs per image



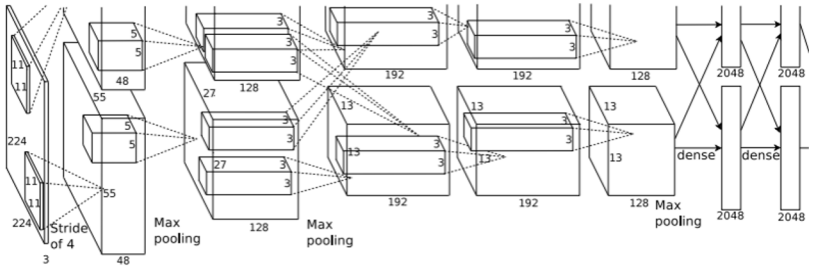
CAT: (x, y, w, h)

4 numbers



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

12 numbers



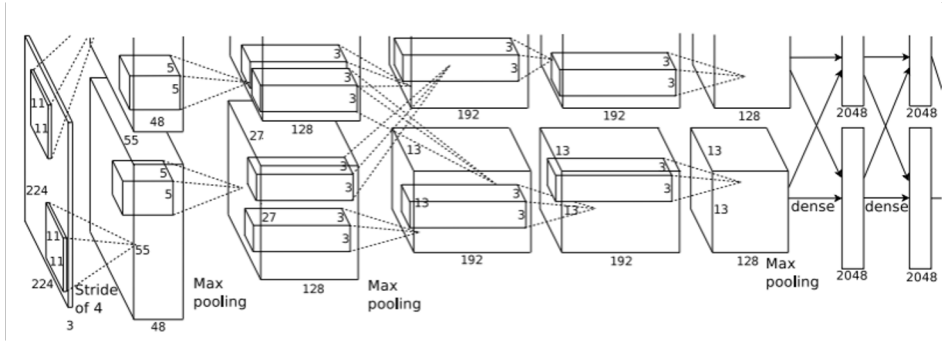
DUCK: (x, y, w, h)
DUCK: (x, y, w, h)
....

Many numbers!

Slide credit: Justin Johnson

Detecting Multiple Objects: **Sliding Window**

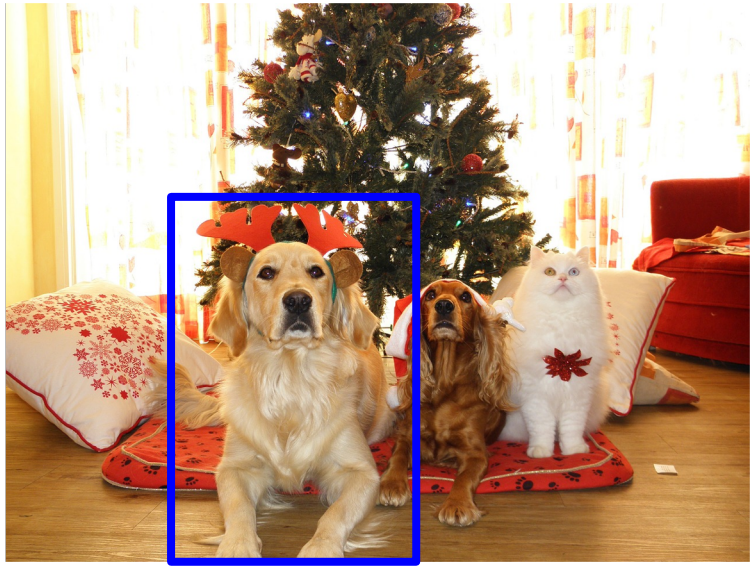
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background



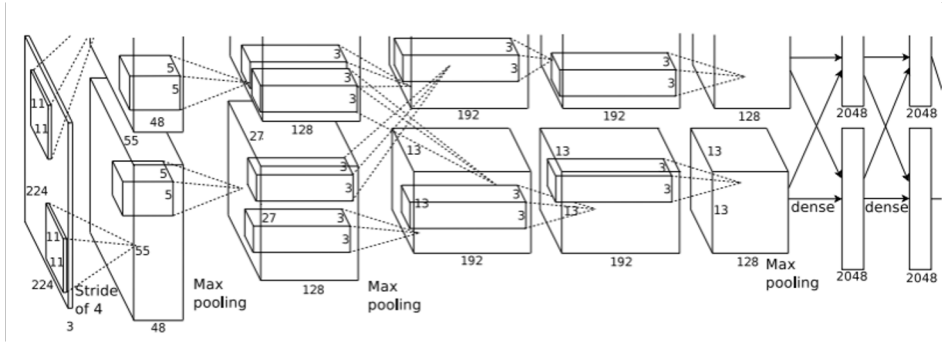
Dog? NO
Cat? NO
Background? YES

Slide credit: Justin Johnson

Detecting Multiple Objects: **Sliding Window**



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

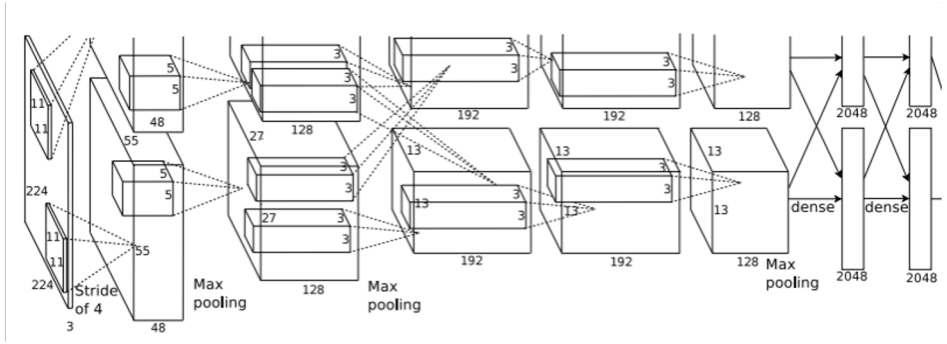


Dog? YES
Cat? NO
Background? NO

Slide credit: Justin Johnson

Detecting Multiple Objects: **Sliding Window**

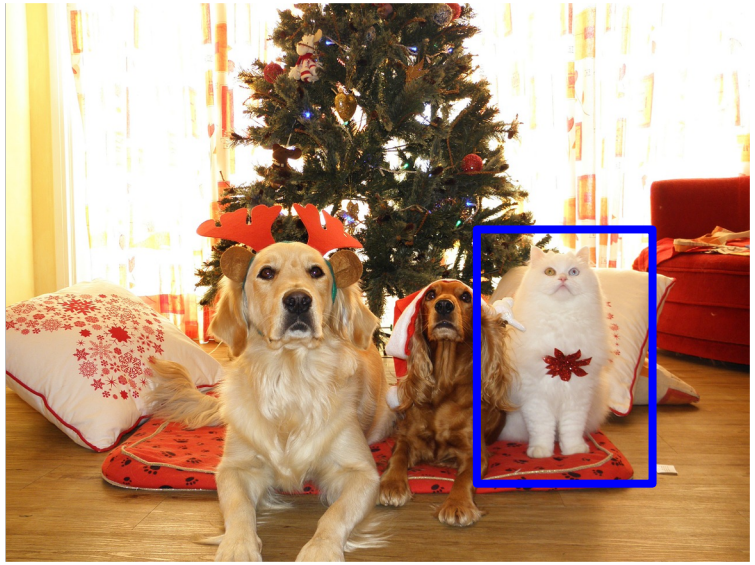
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background



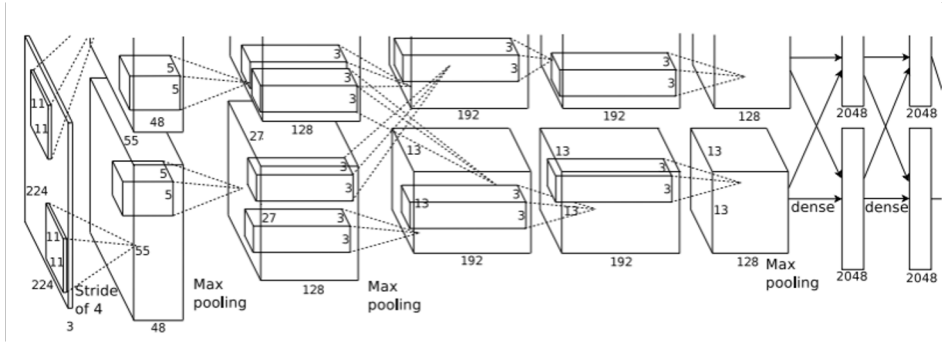
Dog? YES
Cat? NO
Background? NO

Slide credit: Justin Johnson

Detecting Multiple Objects: **Sliding Window**



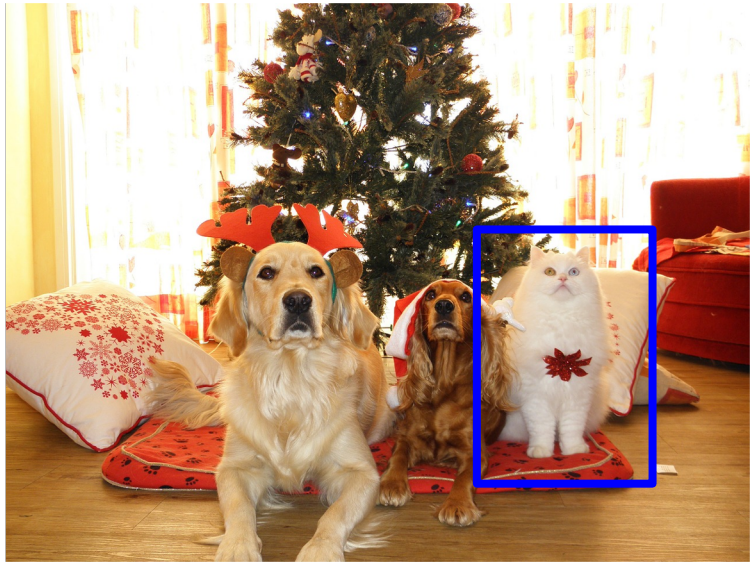
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background



Dog? NO
Cat? YES
Background? NO

Slide credit: Justin Johnson

Detecting Multiple Objects: **Sliding Window**

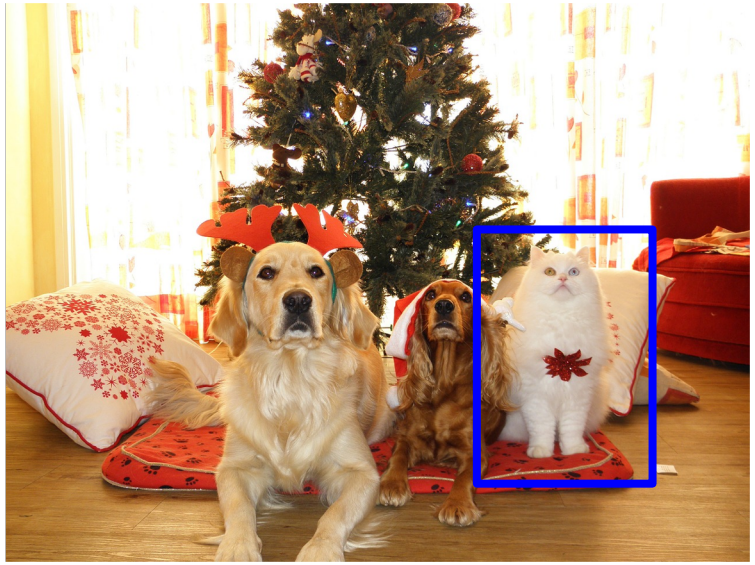


Apply a CNN to many different crops of the image, CNN classifies each crop as object or background

Question: How many possible boxes are there in an image of size H x W?

Slide credit: Justin Johnson

Detecting Multiple Objects: **Sliding Window**



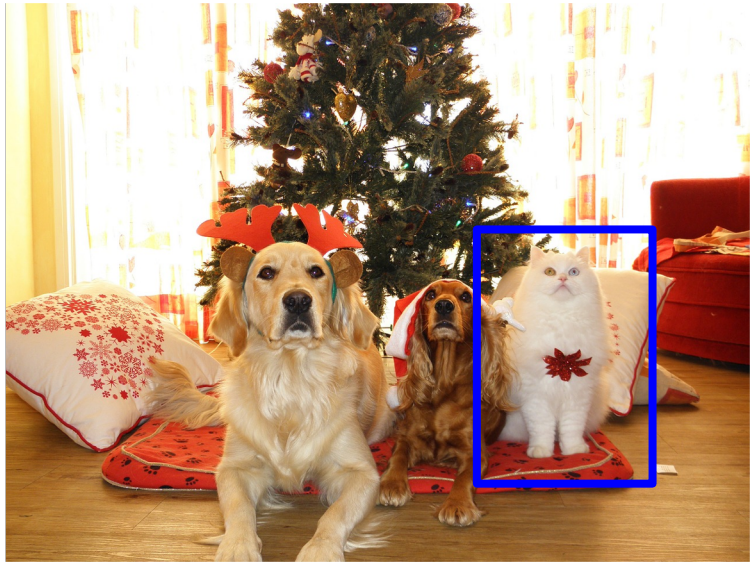
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background

Question: How many possible boxes are there in an image of size H x W?

Consider a box of size h x w:
Possible x positions: W – w + 1
Possible y positions: H – h + 1
Possible positions:
(W – w + 1) * (H – h + 1)

Slide credit: Justin Johnson

Detecting Multiple Objects: **Sliding Window**



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

Question: How many possible boxes are there in an image of size H x W?

Consider a box of size h x w:
Possible x positions: W – w + 1
Possible y positions: H – h + 1
Possible positions:
(W – w + 1) * (H – h + 1)

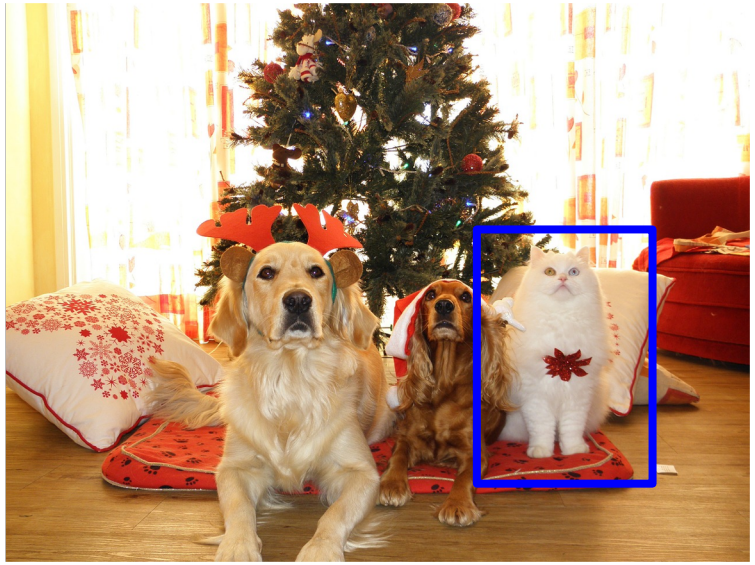
Total possible boxes:

$$\sum_{h=1}^{H}\sum_{w=1}^{W}(W-w+1)(H-h+1)$$

$$=\frac{H(H+1)}{2}\frac{W(W+1)}{2}$$

Slide credit: Justin Johnson

Detecting Multiple Objects: **Sliding Window**



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

800 x 600 image has ~58M boxes! No way we can evaluate them all

Question: How many possible boxes are there in an image of size H x W?

Consider a box of size h x w:
Possible x positions: W – w + 1
Possible y positions: H – h + 1
Possible positions:
(W – w + 1) * (H – h + 1)

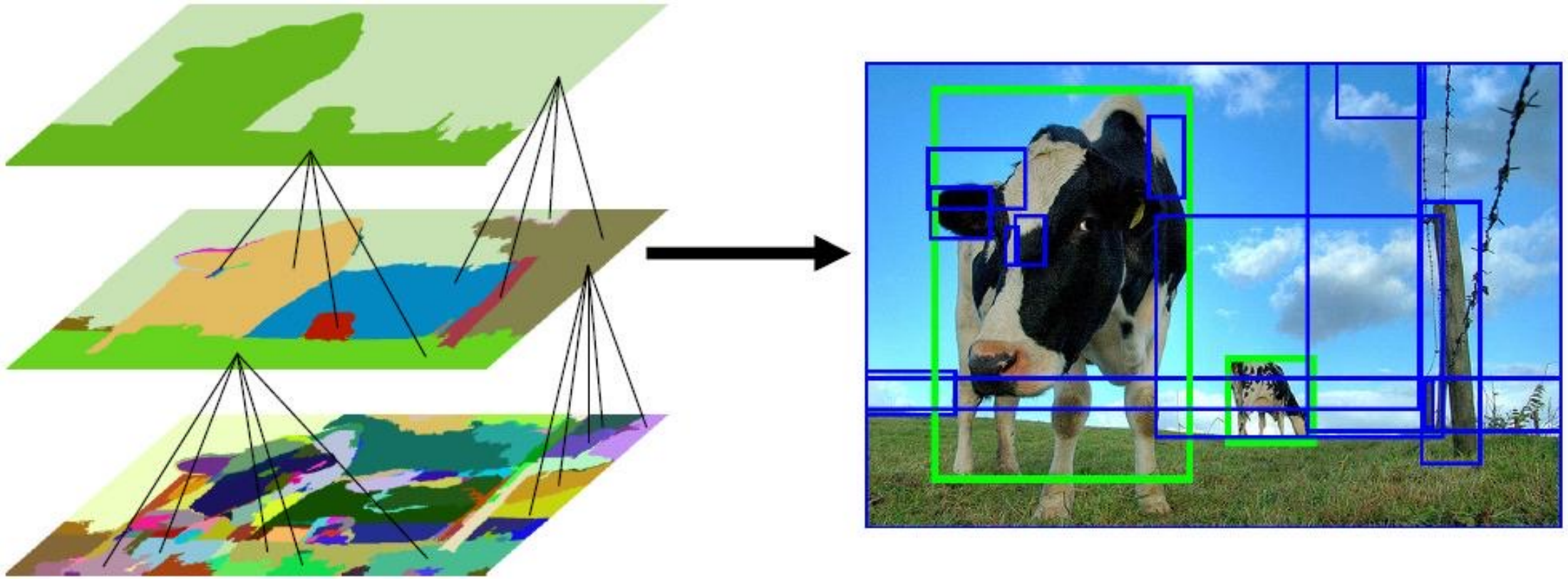
Total possible boxes:

$$\sum_{h=1}^{H}\sum_{w=1}^{W}(W-w+1)(H-h+1)$$

$$=\frac{H(H+1)}{2}\frac{W(W+1)}{2}$$

Slide credit: Justin Johnson

Idea: Object Proposals as Input for Detection



Selective Search for Object Recognition, Uijlings et al, IJCV 2013

Computer Science > Computer Vision and Pattern Recognition

[Submitted on 11 Nov 2013 (v1), last revised 22 Oct 2014 (this version, v5)]

Rich feature hierarchies for accurate object detection and semantic segmentation

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik

Object detection performance, as measured on the canonical PASCAL VOC dataset, has plateaued in the last few years. The best-performing methods are complex ensemble systems that typically combine multiple low-level image features with high-level context. In this paper, we propose a simple and scalable detection algorithm that improves mean average precision (mAP) by more than 30% relative to the previous best result on VOC 2012---achieving a mAP of 53.3%. Our approach combines two key insights: (1) one can apply high-capacity convolutional neural networks (CNNs) to bottom-up region proposals in order to localize and segment objects and (2) when labeled training data is scarce, supervised pre-training for an auxiliary task, followed by domain-specific fine-tuning, yields a significant performance boost. Since we combine region proposals with CNNs, we call our method R-CNN: Regions with CNN features. We also compare R-CNN to OverFeat, a recently proposed sliding-window detector based on a similar CNN architecture. We find that R-CNN outperforms OverFeat by a large margin on the 200-class ILSVRC2013 detection dataset. Source code for the complete system is available at this http URL.

Comments: Extended version of our CVPR 2014 paper; latest update (v5) includes results using deeper networks (see Appendix G. Changelog)
Subjects: **Computer Vision and Pattern Recognition (cs.CV)**
Cite as: arXiv:1311.2524 **[cs.CV]**
(or arXiv:1311.2524v5 **[cs.CV]** for this version)
https://doi.org/10.48550/arXiv.1311.2524

Submission history

From: Ross Girshick [view email]
[v1] Mon, 11 Nov 2013 18:43:49 UTC (3,704 KB)
[v2] Tue, 15 Apr 2014 01:44:31 UTC (16,729 KB)
[v3] Wed, 7 May 2014 17:09:23 UTC (6,644 KB)
[v4] Mon, 9 Jun 2014 22:07:33 UTC (6,644 KB)
[v5] Wed, 22 Oct 2014 17:23:20 UTC (6,660 KB)

R-CNN: Region-Based CNN

Input image

Slide credit: Ross Girshick, Justin Johnson

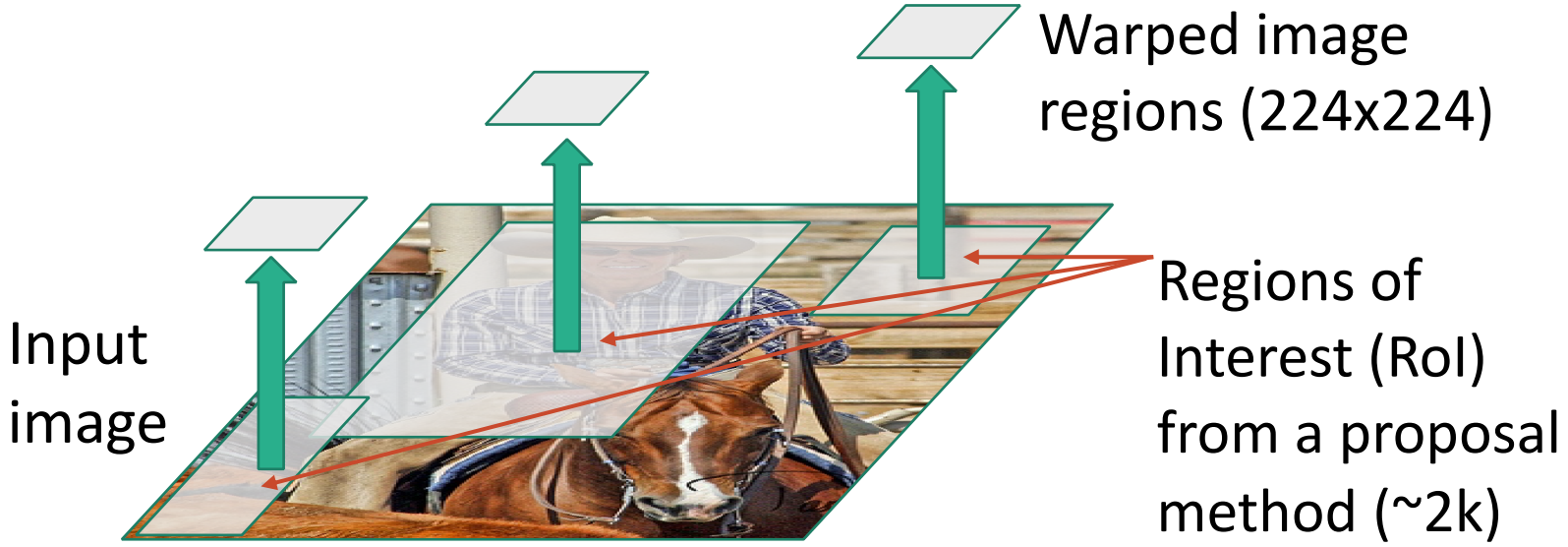
R-CNN: Region-Based CNN

Input image

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

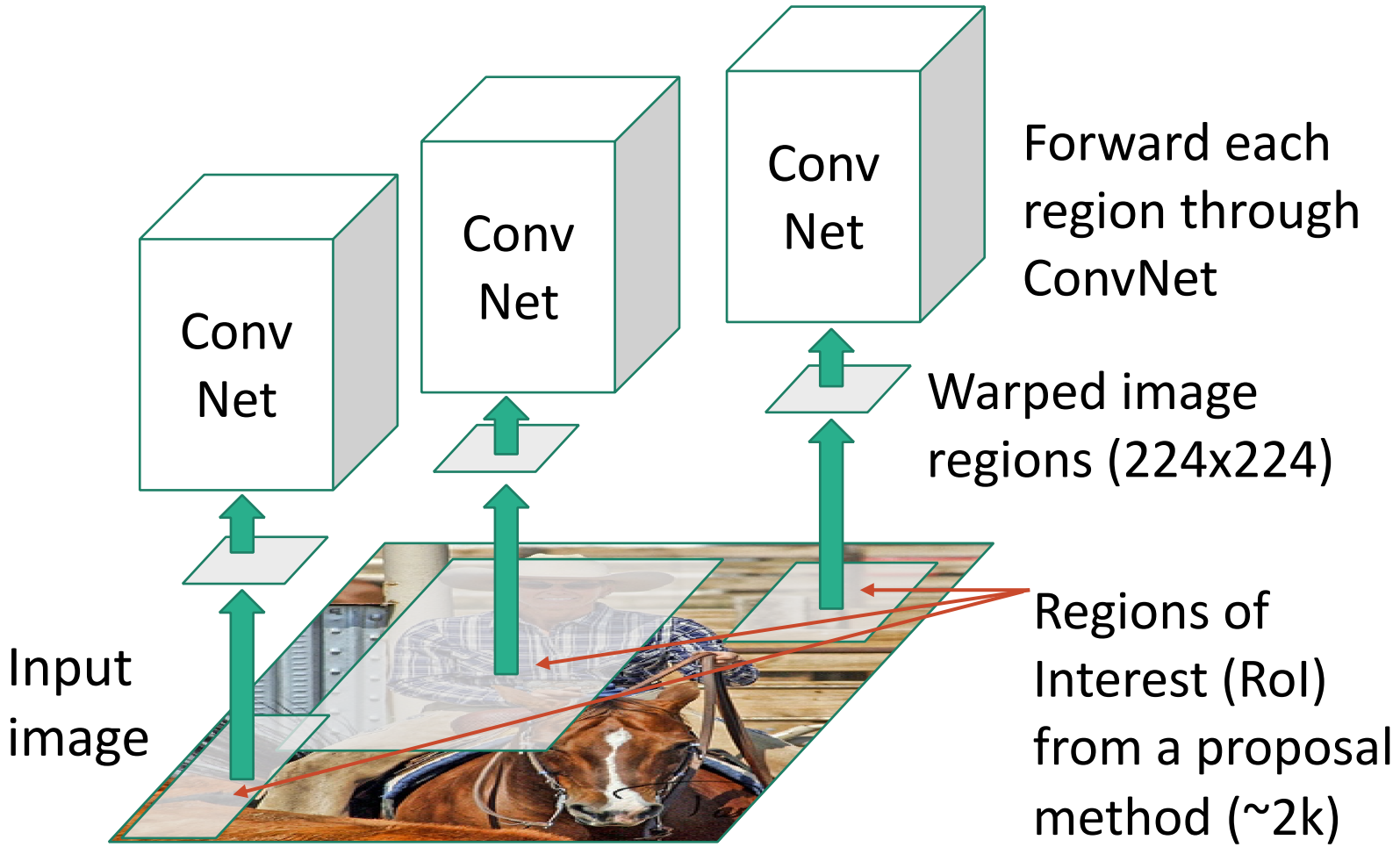
Slide credit: Ross Girshick, Justin Johnson

R-CNN: Region-Based CNN



Slide credit: Ross Girshick, Justin Johnson

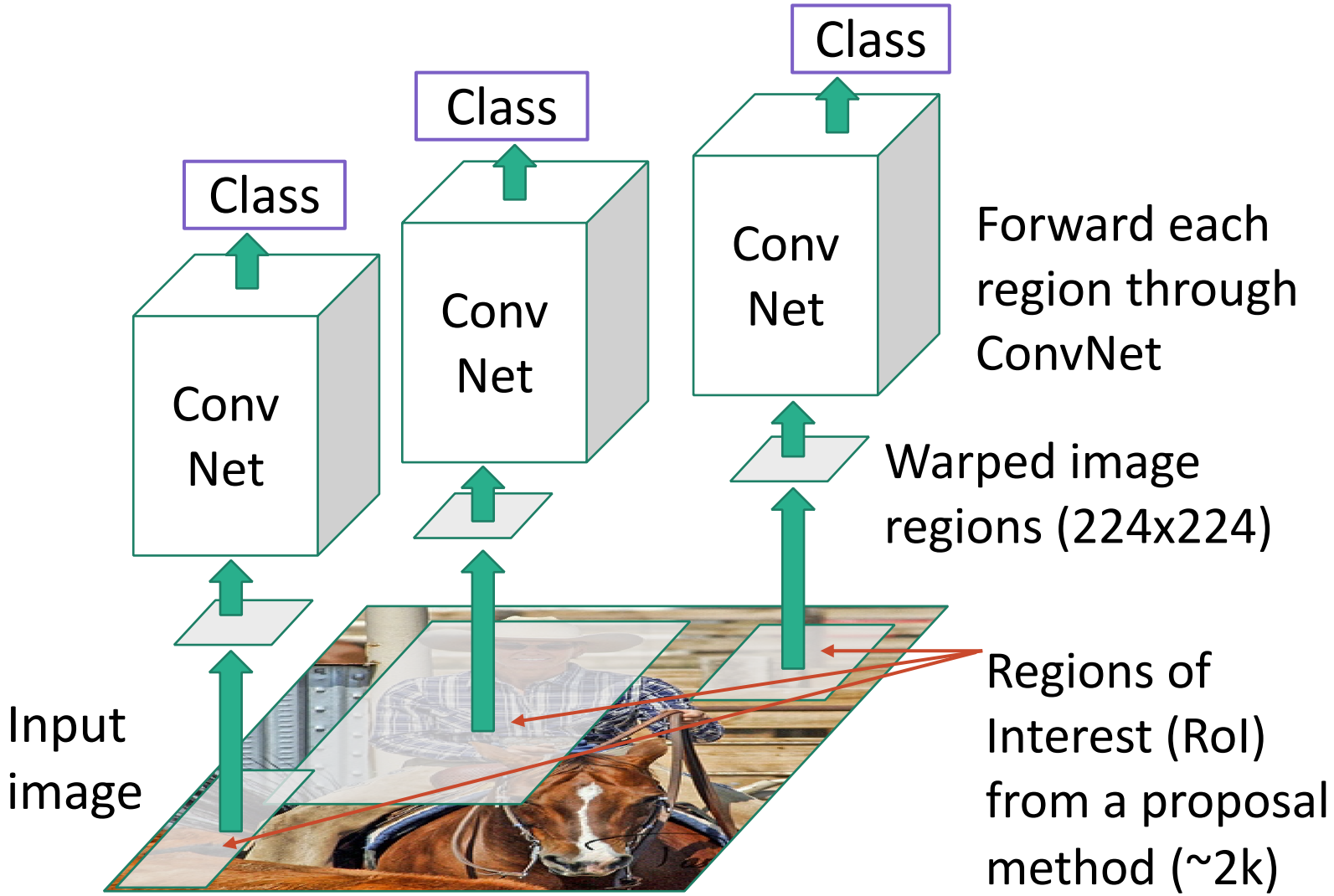
R-CNN: Region-Based CNN



Slide credit: Ross Girshick, Justin Johnson

R-CNN: Region-Based CNN

Classify each region

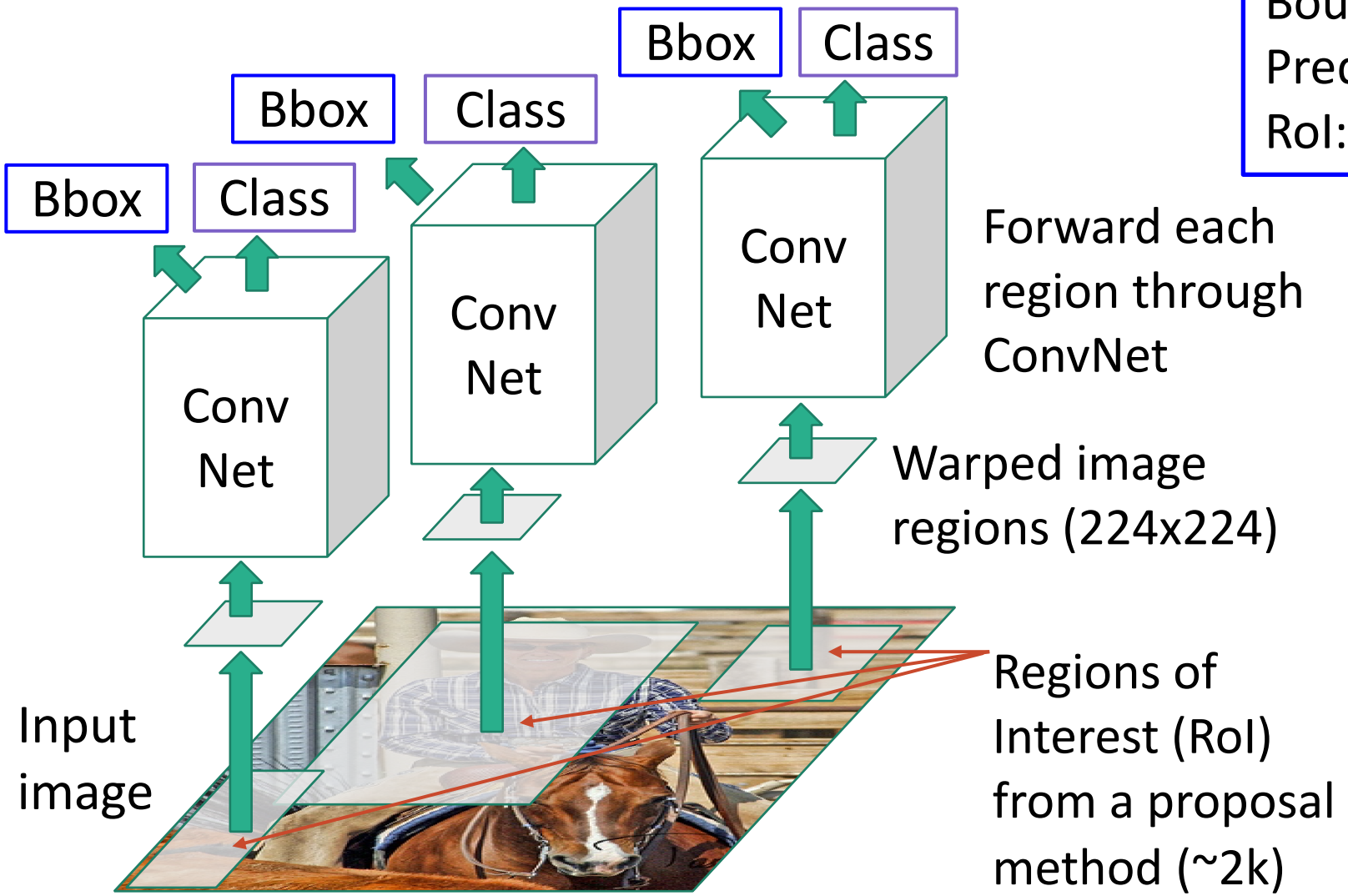


Slide credit: Ross Girshick, Justin Johnson

R-CNN: Region-Based CNN

Classify each region

Bounding box regression:
Predict "transform" to correct the RoI: 4 numbers (t_x, t_y, t_h, t_w)

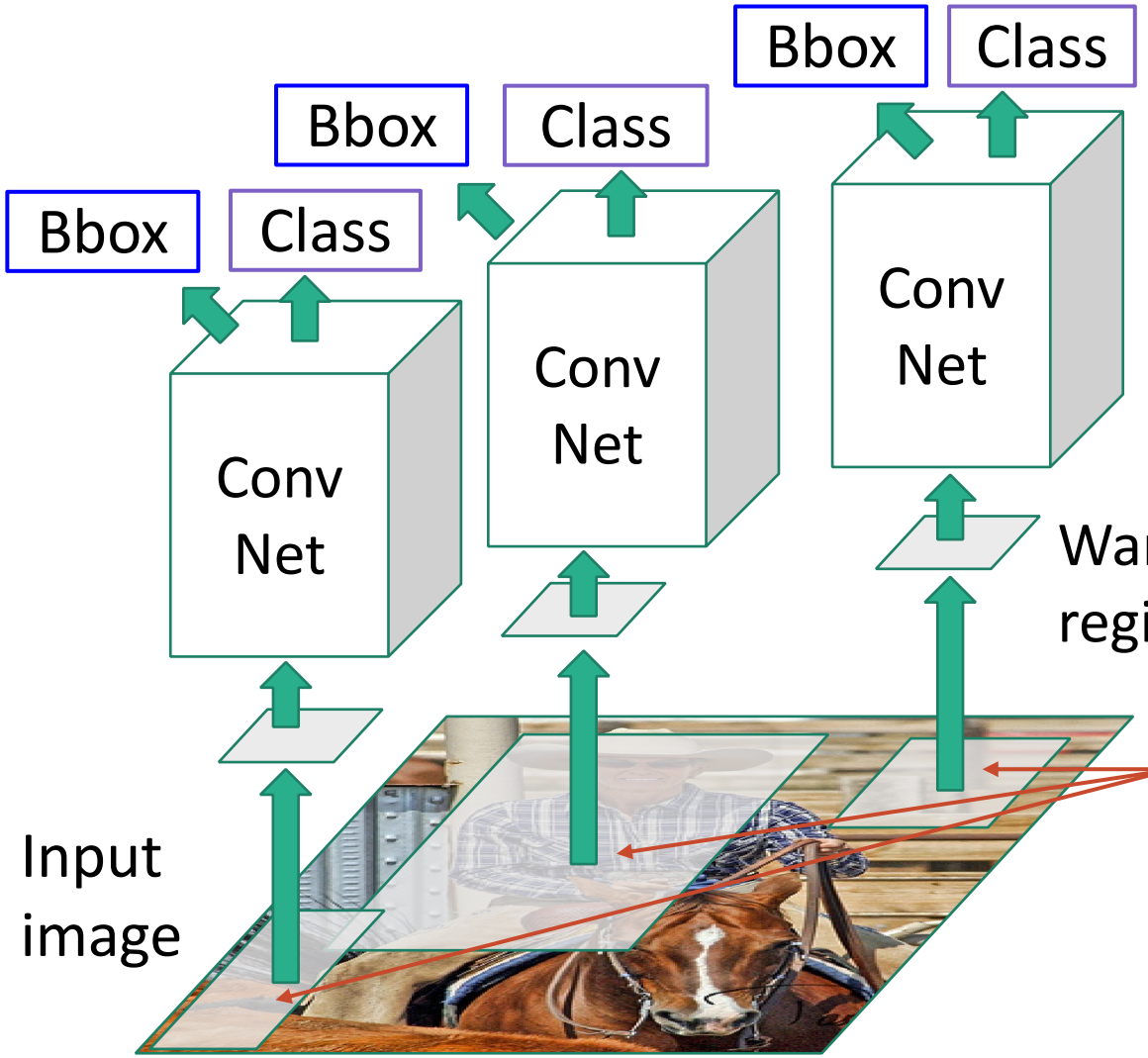


Slide credit: Ross Girshick, Justin Johnson

R-CNN: Region-Based CNN

Classify each region

Bounding box regression:
Predict "transform" to correct the RoI: 4 numbers (t_x, t_y, t_h, t_w)



Forward each region through ConvNet

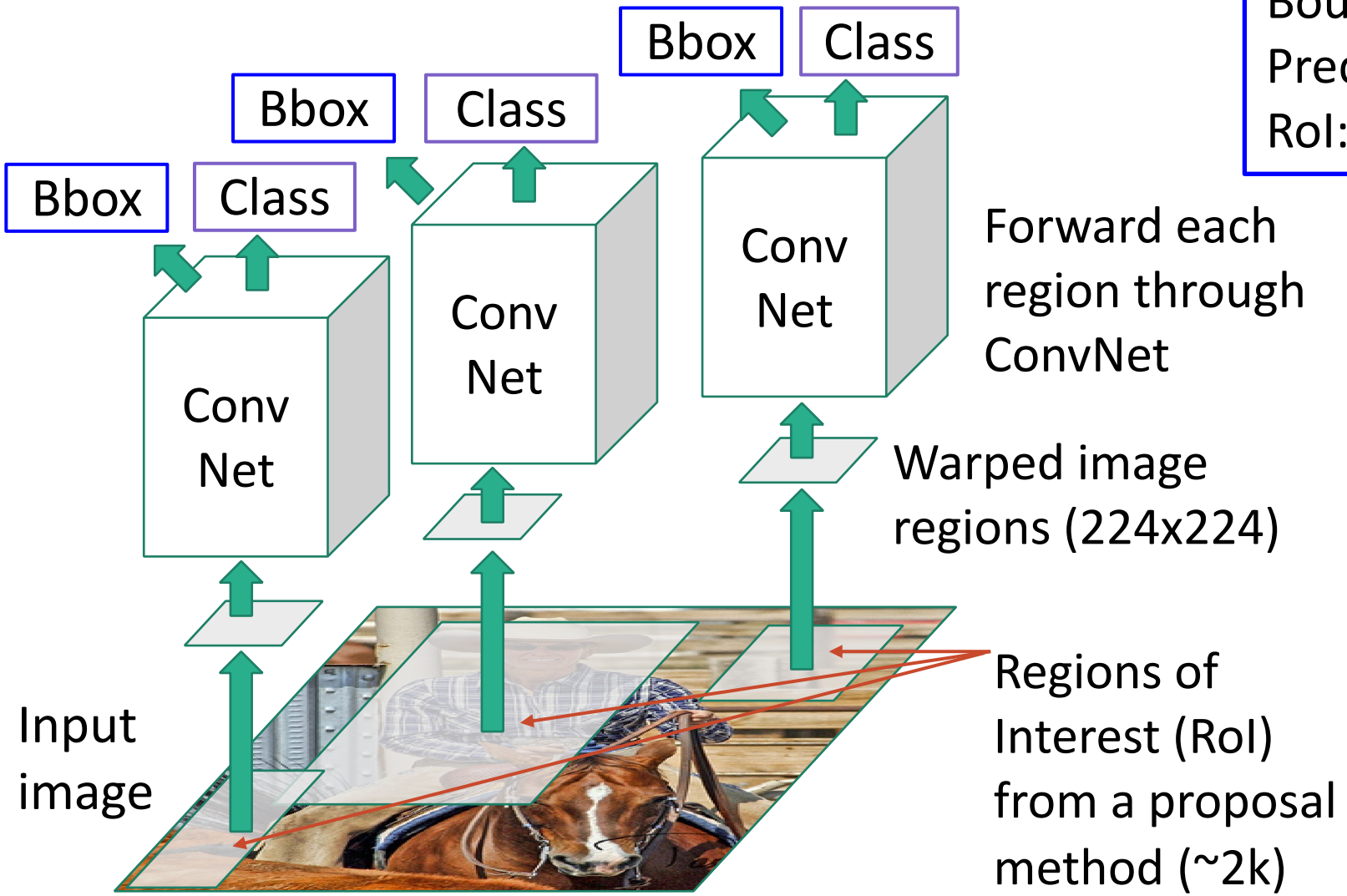
Warped image regions (224x224)

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

Problem: slow! Requires 2000 forward passes per image

Slide credit: Ross Girshick, Justin Johnson

R-CNN: Region-Based CNN



Classify each region

Bounding box regression:
Predict "transform" to correct the RoI: 4 numbers (t_x, t_y, t_h, t_w)

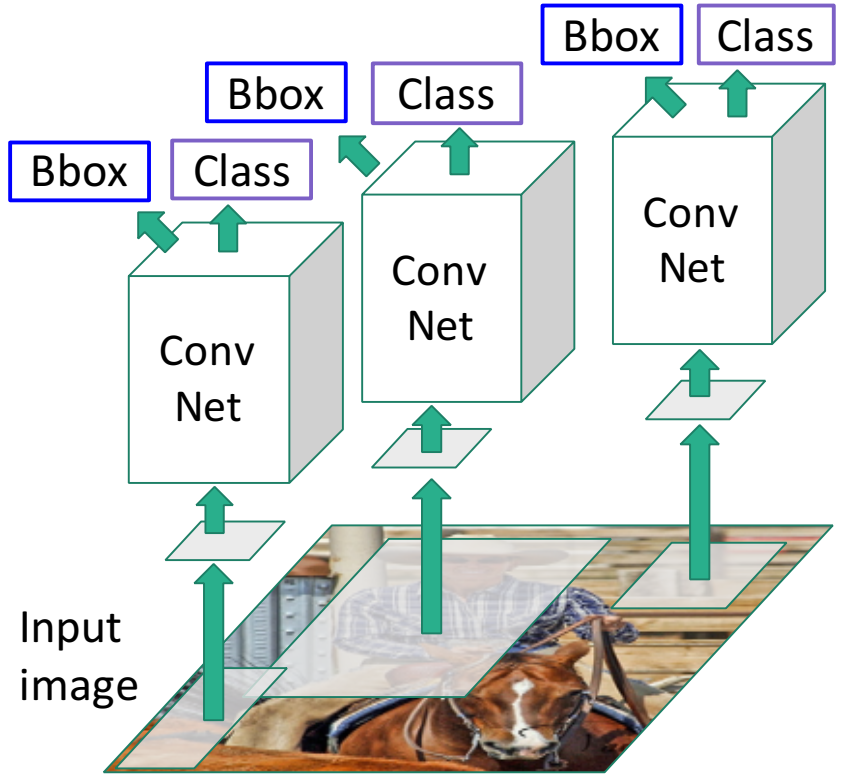
Problem: slow! Requires 2000 forward passes per image

Idea: Overlapping proposals. Repeated computation. Can we avoid this?

Slide credit: Ross Girshick, Justin Johnson

“Slow” R-CNN

Process each region independently



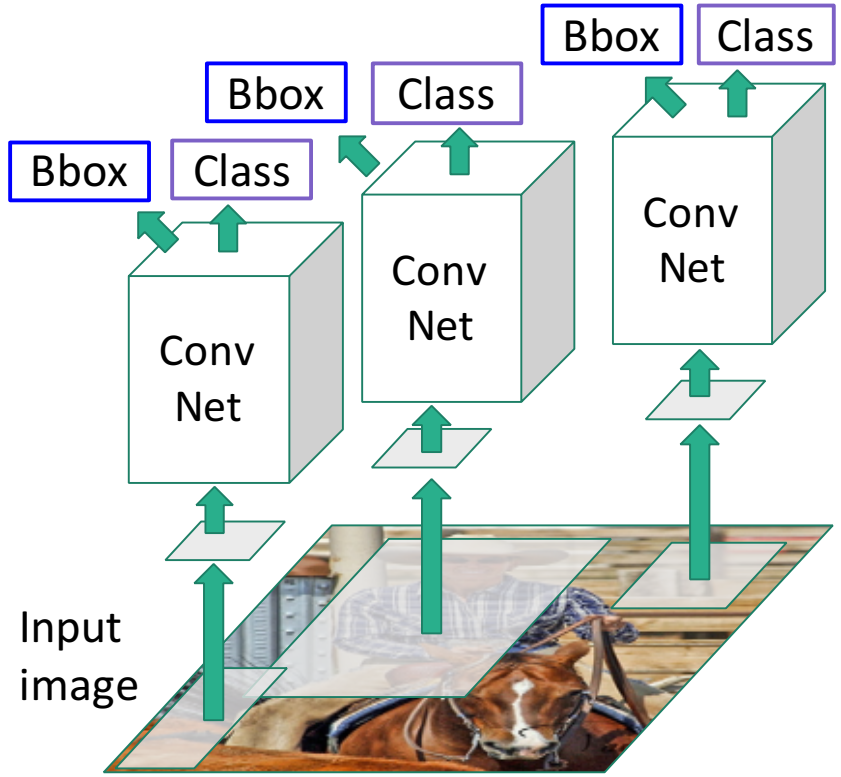
Slide credit: Ross Girshick, Justin Johnson

Fast R-CNN

Input image

"Slow" R-CNN

Process each region independently

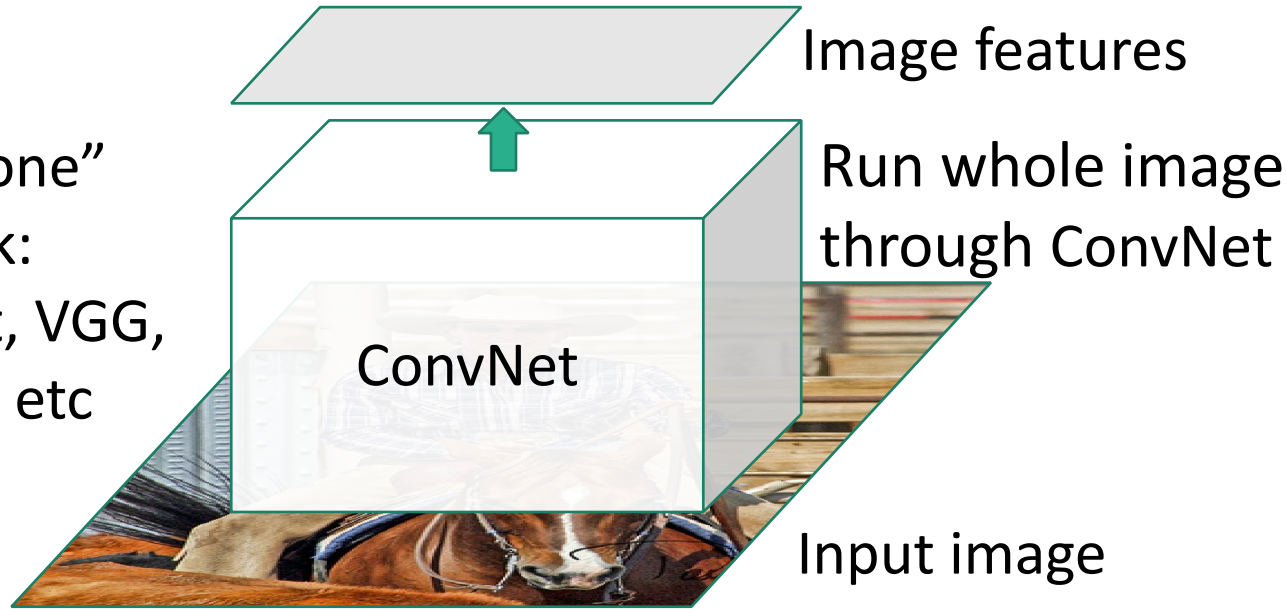


Fast R-CNN, Girshick, ICCV 2015

Slide credit: Ross Girshick, Justin Johnson

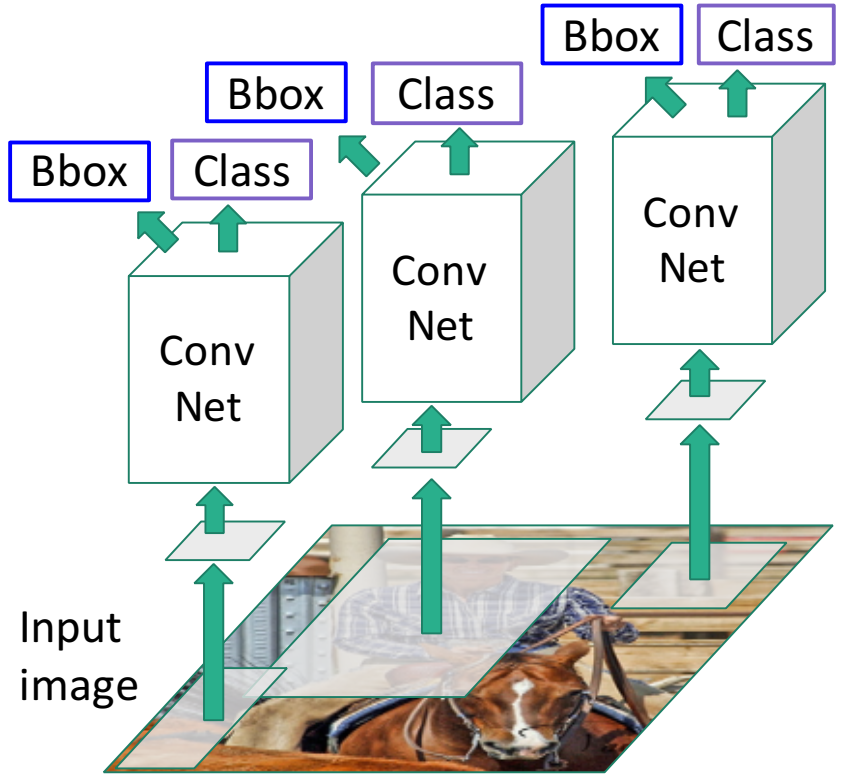
Fast R-CNN

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



"Slow" R-CNN

Process each region independently



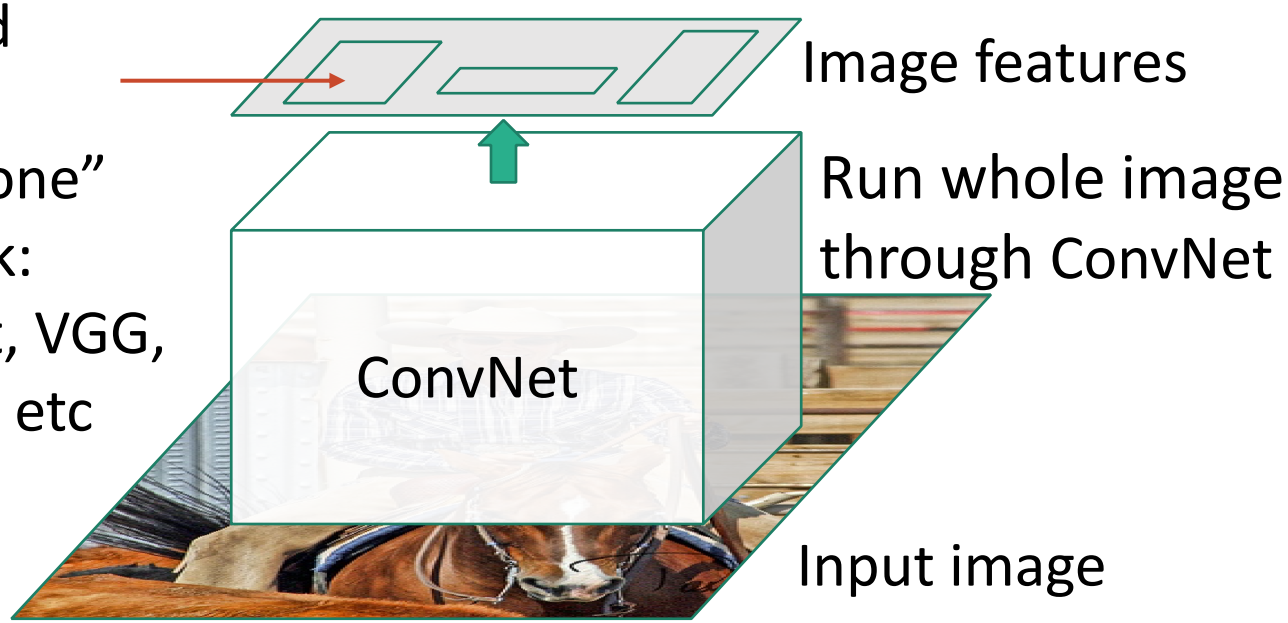
Fast R-CNN, Girshick, ICCV 2015

Slide credit: Ross Girshick, Justin Johnson

Fast R-CNN

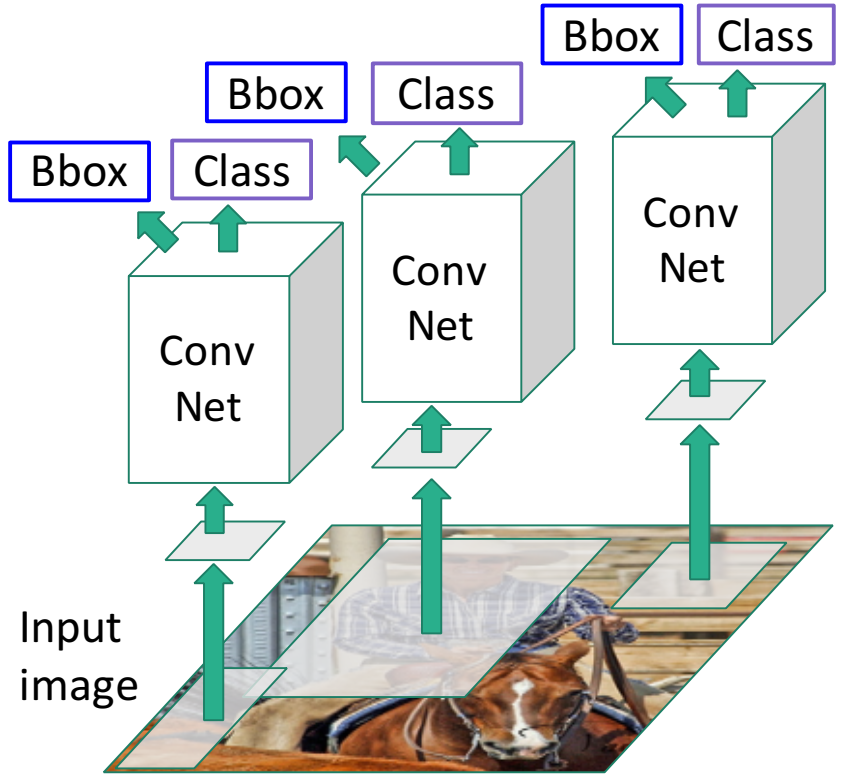
Regions of Interest (RoIs) from a proposal method

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



"Slow" R-CNN

Process each region independently



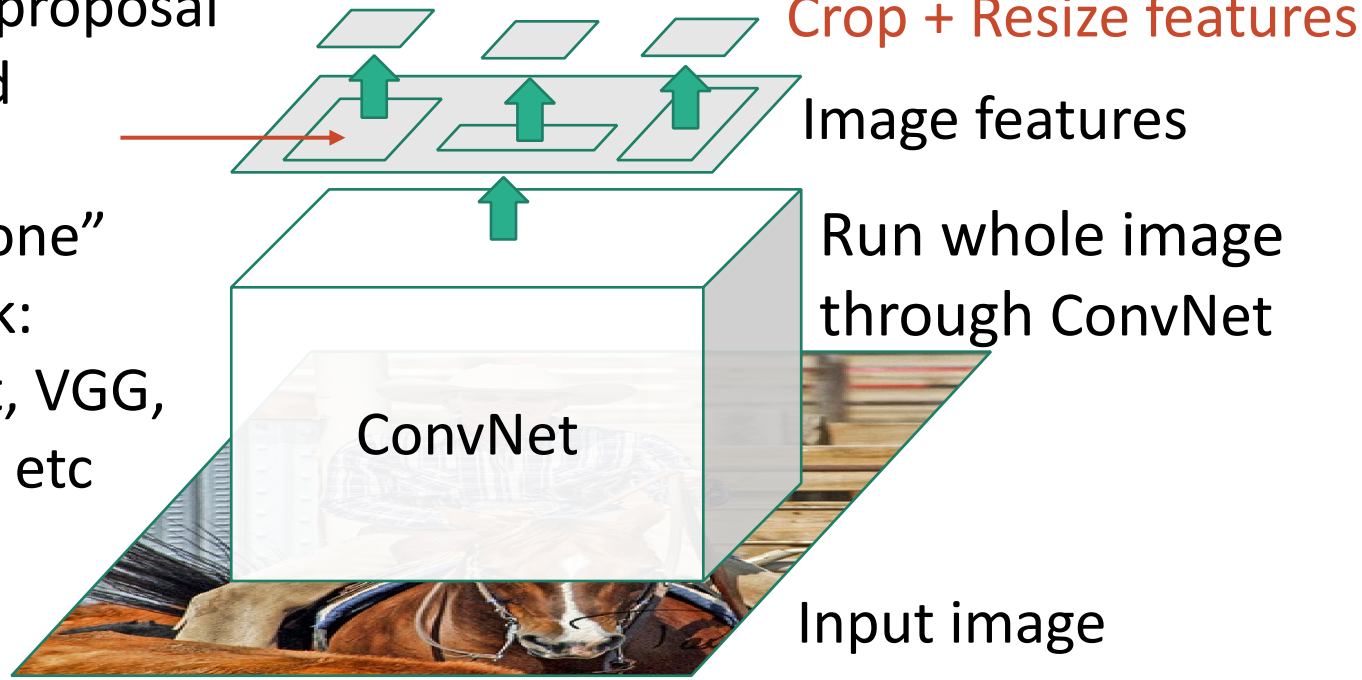
Fast R-CNN, Girshick, ICCV 2015

Slide credit: Ross Girshick, Justin Johnson

Fast R-CNN

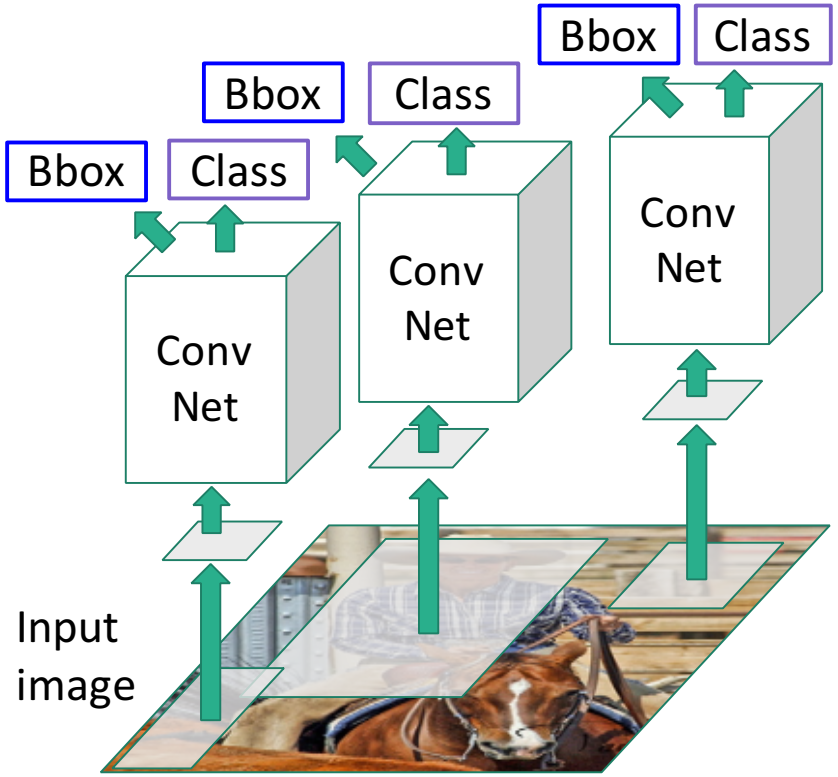
Regions of Interest (RoIs) from a proposal method

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



"Slow" R-CNN

Process each region independently



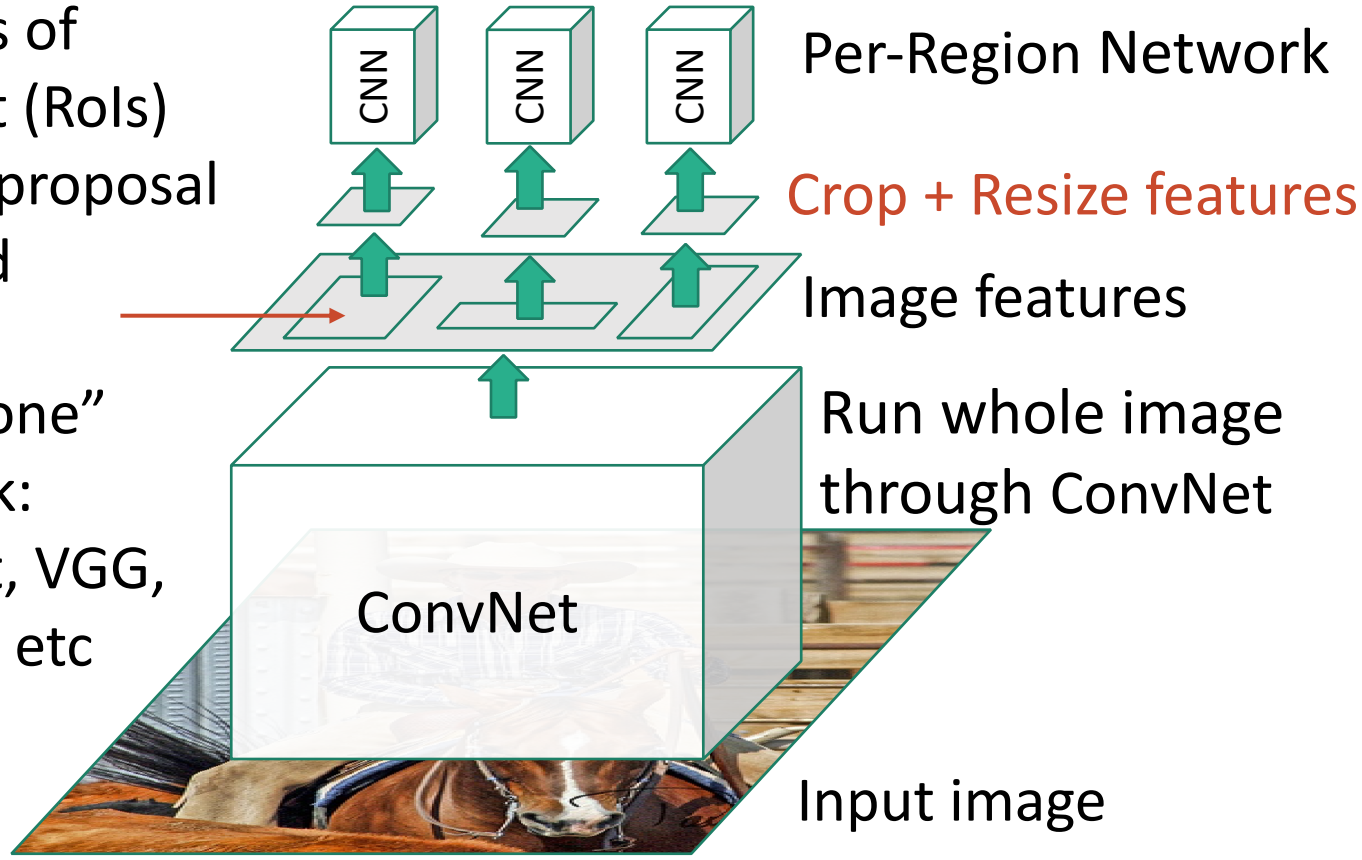
Fast R-CNN, Girshick, ICCV 2015

Slide credit: Ross Girshick, Justin Johnson

Fast R-CNN

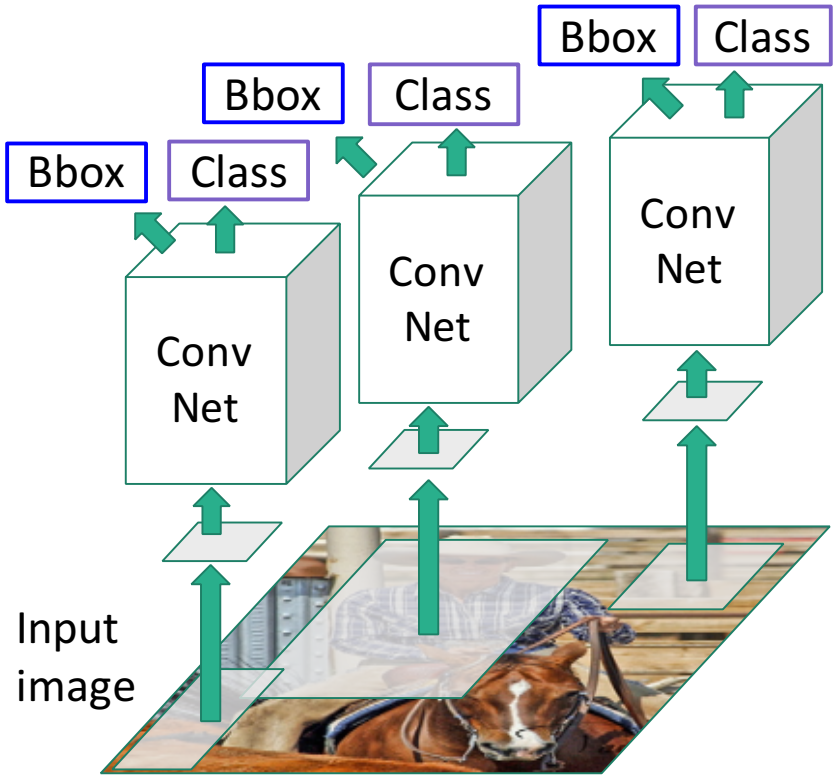
Regions of Interest (RoIs) from a proposal method

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



"Slow" R-CNN

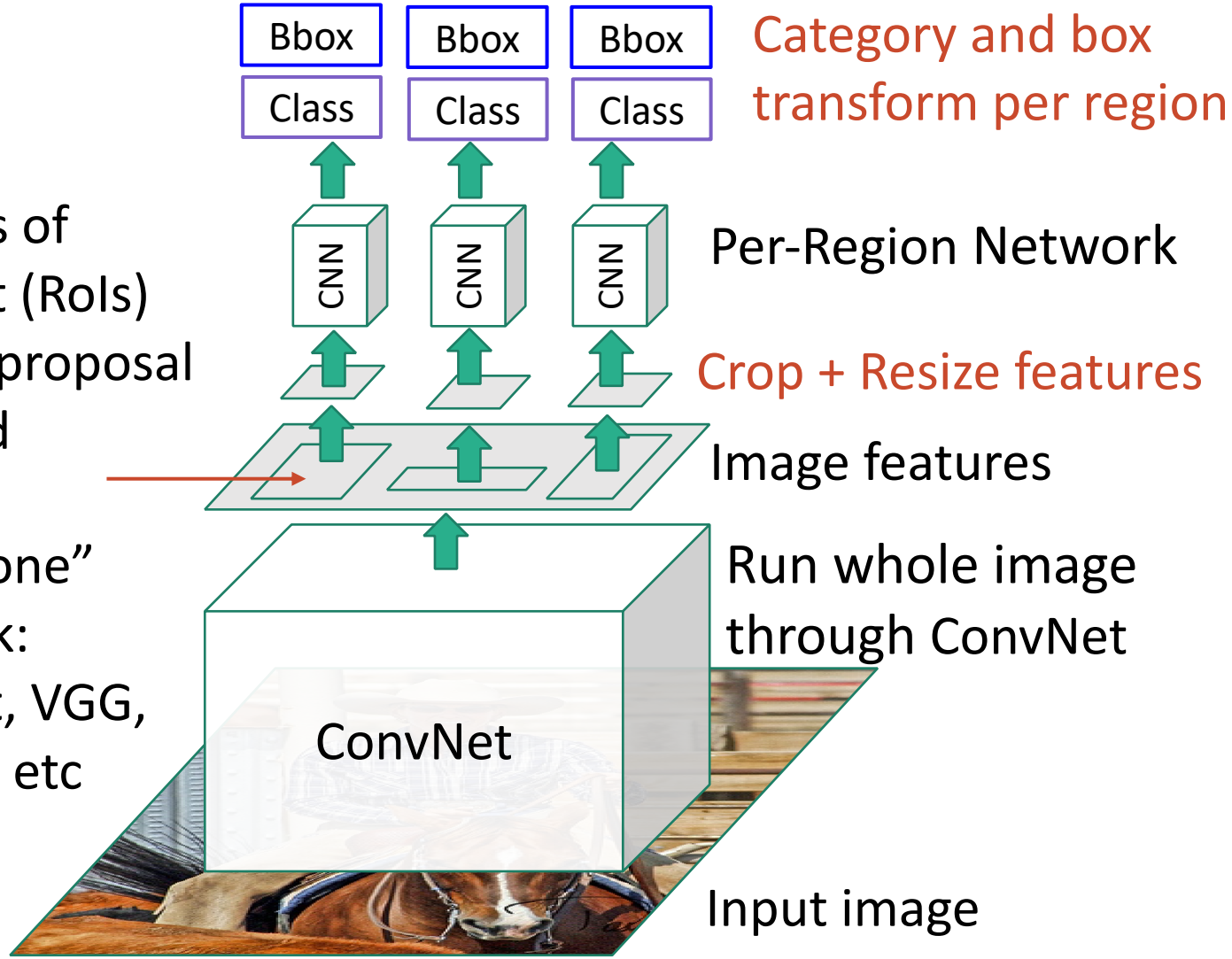
Process each region independently



Fast R-CNN, Girshick, ICCV 2015

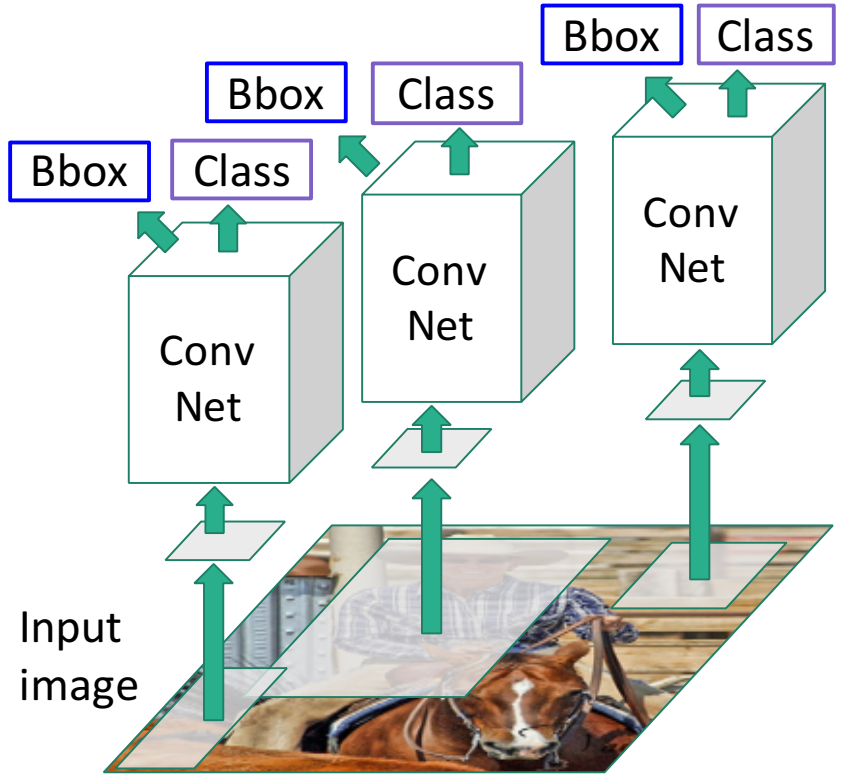
Slide credit: Ross Girshick, Justin Johnson

Fast R-CNN



"Slow" R-CNN

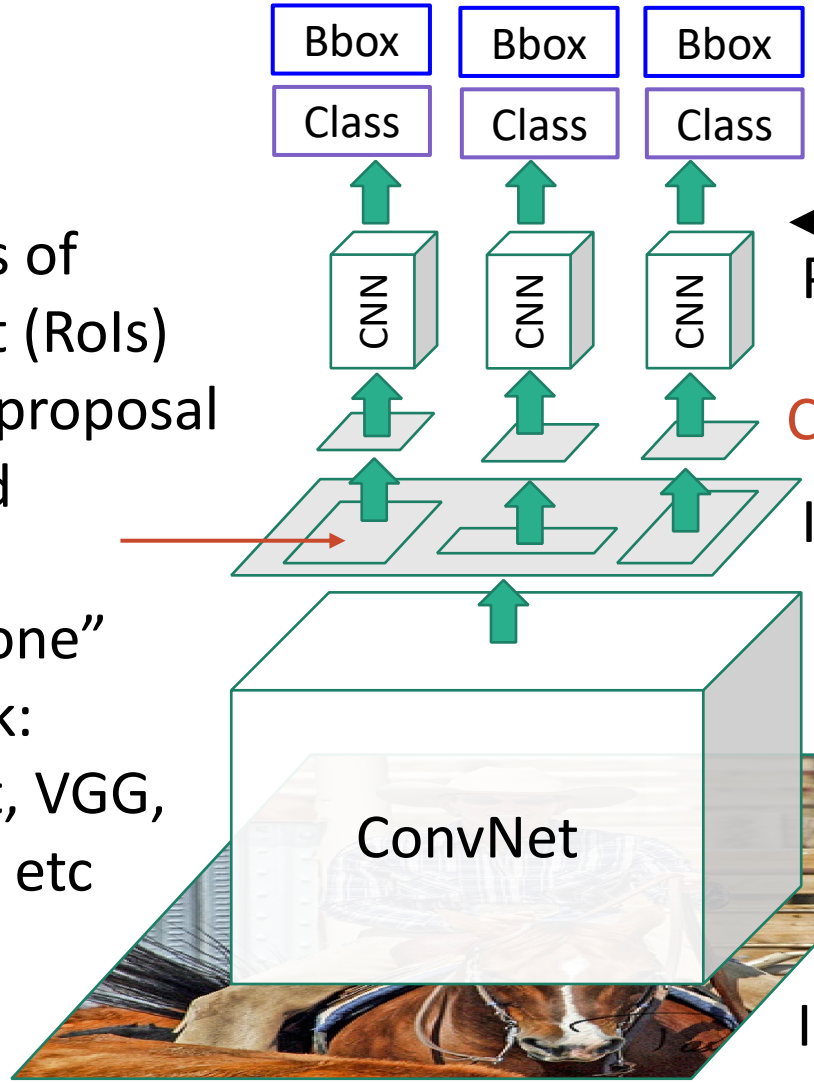
Process each region independently



Fast R-CNN, Girshick, ICCV 2015

Slide credit: Ross Girshick, Justin Johnson

Fast R-CNN



Category and box transform per region

Per-Region network is relatively lightweight

Per-Region Network

Regions of Interest (RoIs) from a proposal method

Crop + Resize features

Image features

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

Run whole image through ConvNet

ConvNet

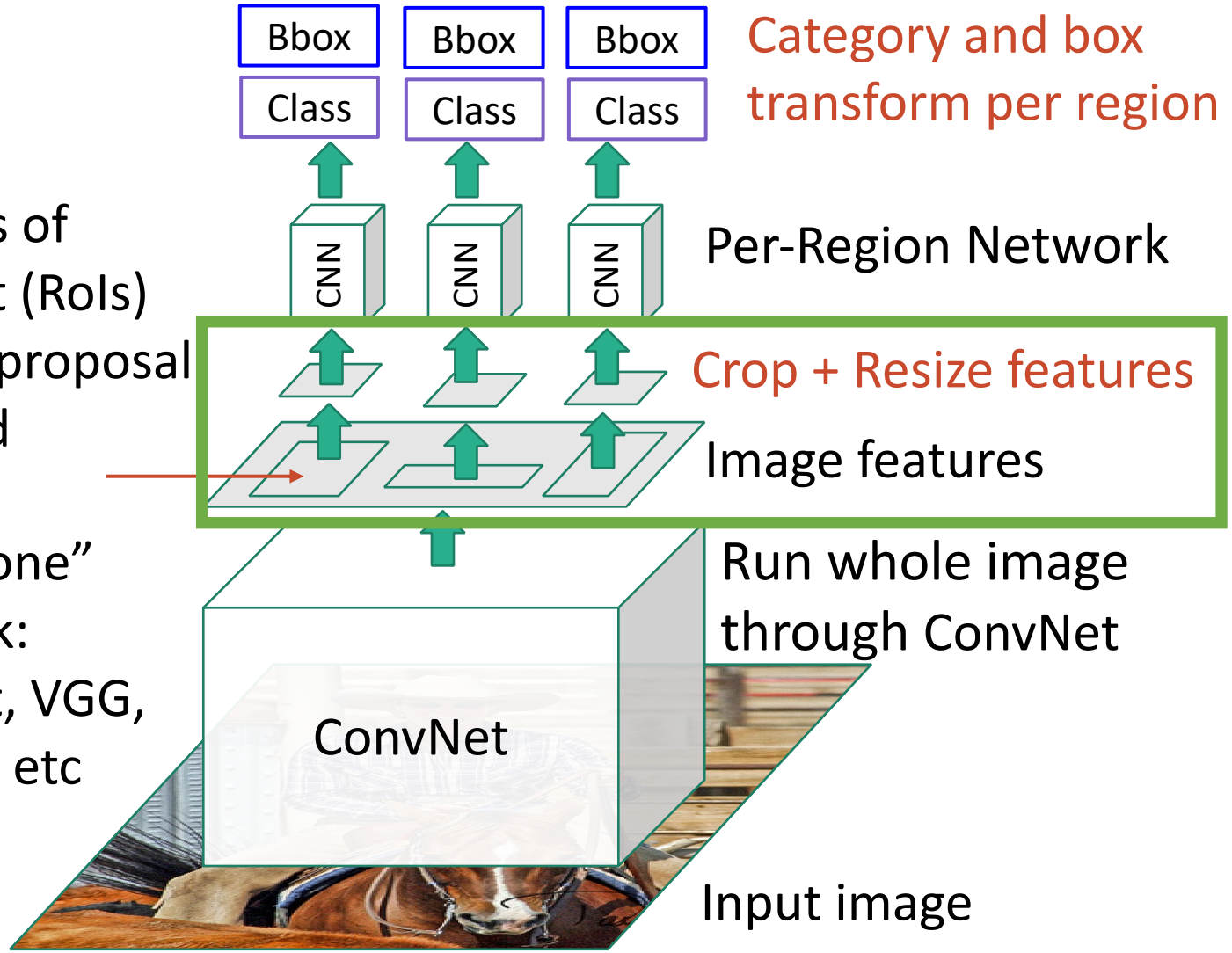
Most of the computation happens in backbone network; this saves work for overlapping region proposals

Input image

Fast R-CNN, Girshick, ICCV 2015

Slide credit: Ross Girshick, Justin Johnson

Fast R-CNN



Regions of Interest (RoIs) from a proposal method

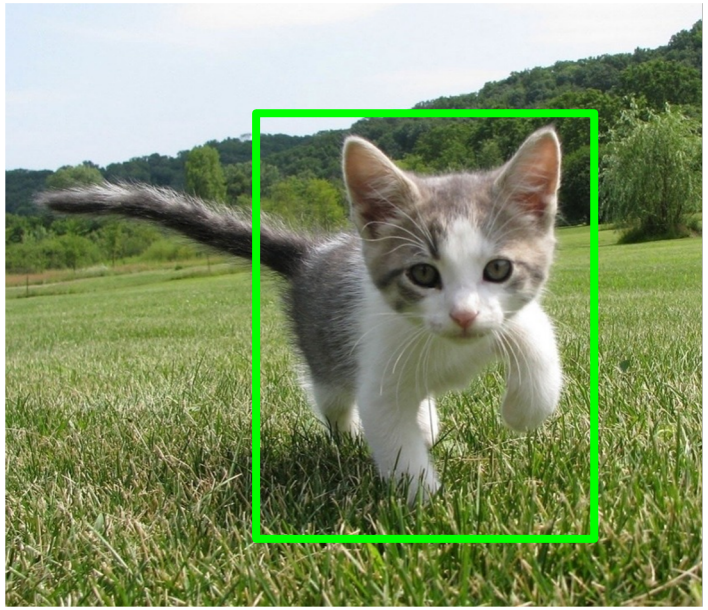
"Backbone" network: AlexNet, VGG, ResNet, etc

How to crop features?

Fast R-CNN, Girshick, ICCV 2015

Slide credit: Ross Girshick, Justin Johnson

Cropping Features: RoI Pool

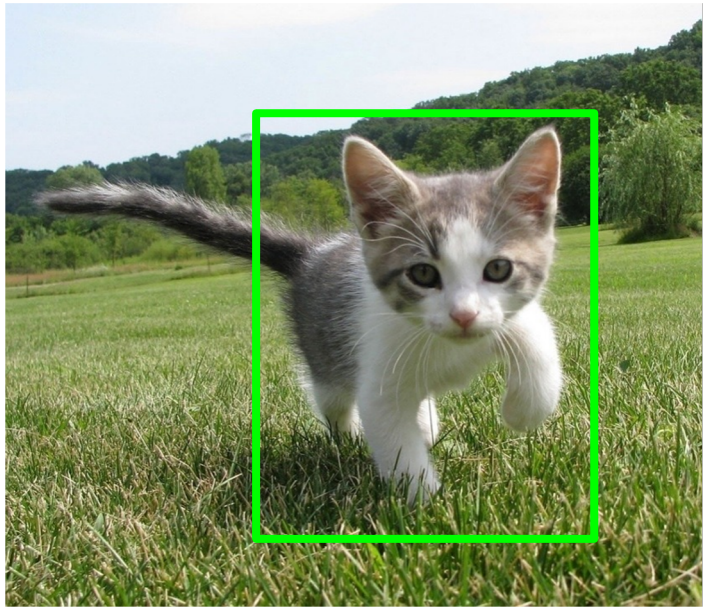


Input Image
(e.g. 3 x 640 x 480)

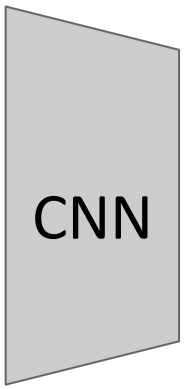
Fast R-CNN, Girshick, ICCV 2015

Slide credit: Justin Johnson

Cropping Features: RoI Pool



Input Image
(e.g. 3 x 640 x 480)



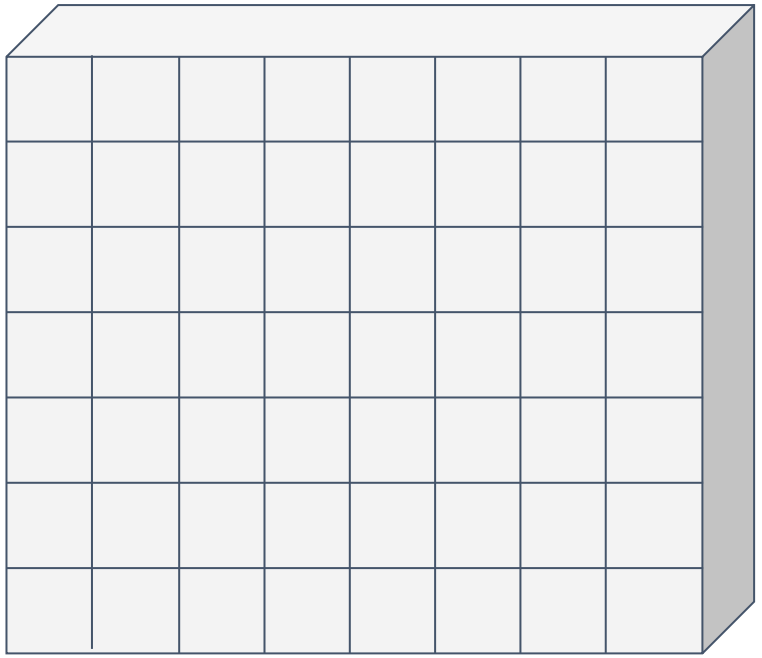


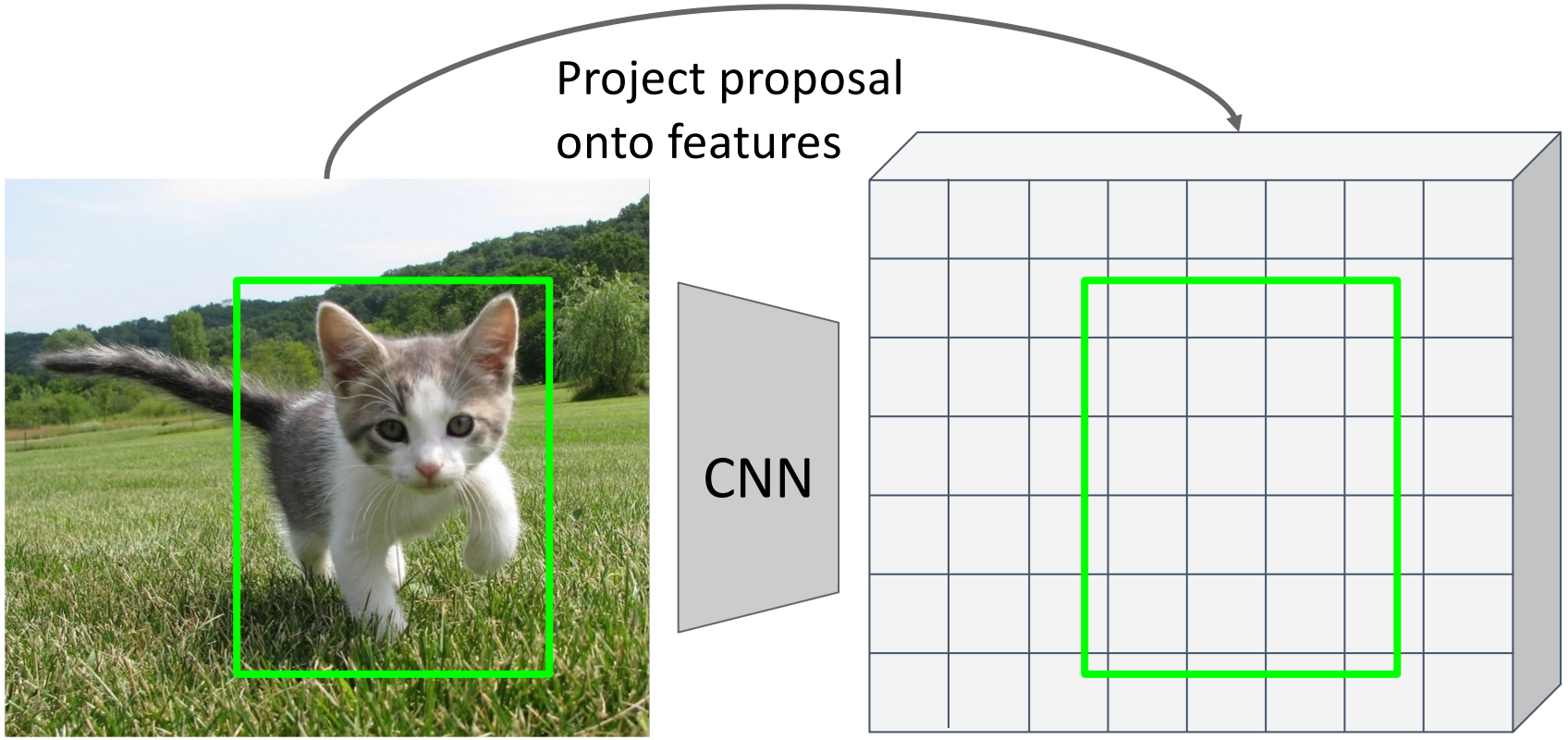
Image features
(e.g. 512 x 20 x 15)

Want features for the box of a fixed size (2x2 in this example, 7x7 or 14x14 in practice)

Fast R-CNN, Girshick, ICCV 2015

Slide credit: Justin Johnson

Cropping Features: RoI Pool

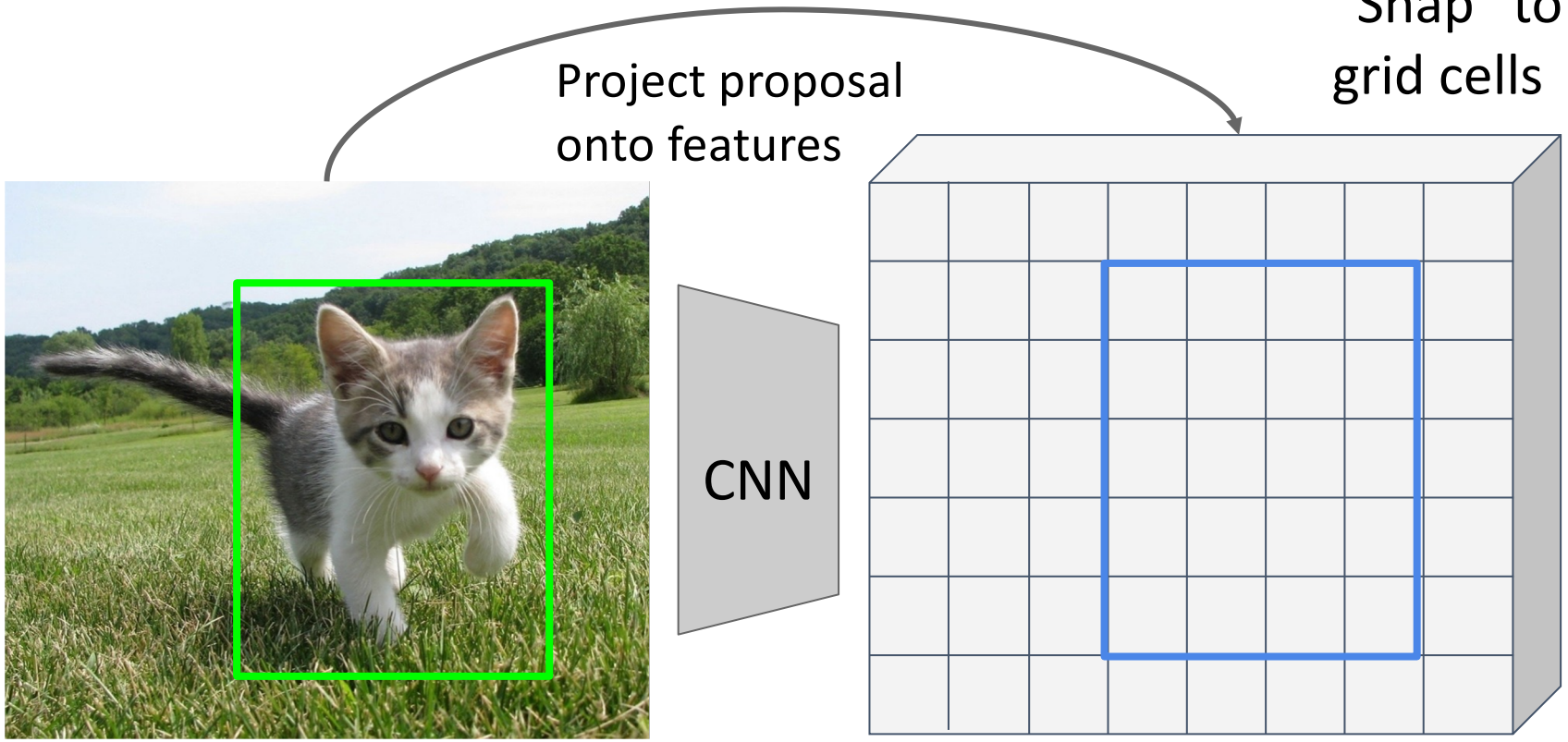


Want features for the box of a fixed size (2x2 in this example, 7x7 or 14x14 in practice)

Fast R-CNN, Girshick, ICCV 2015

Slide credit: Justin Johnson

Cropping Features: RoI Pool

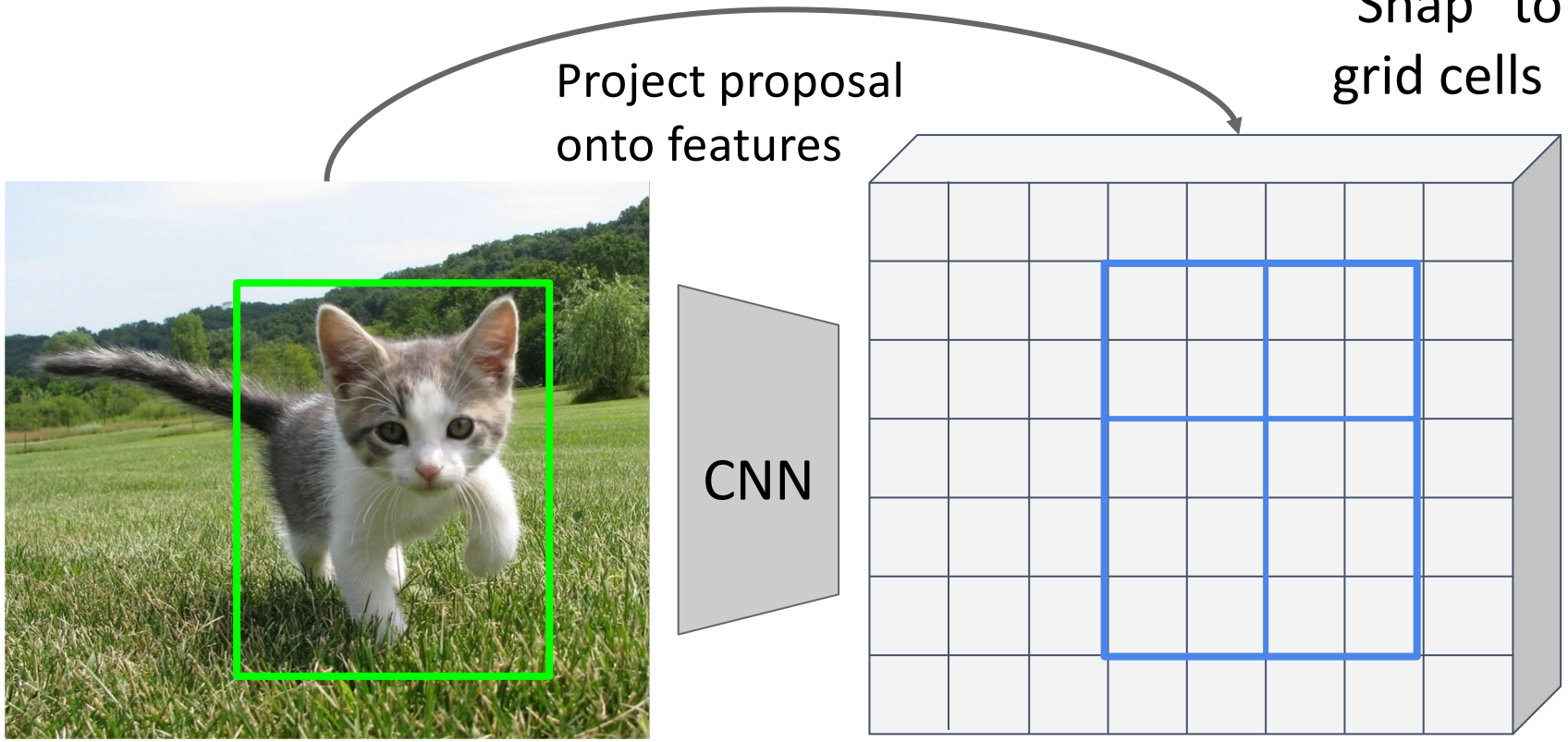


Want features for the box of a fixed size (2x2 in this example, 7x7 or 14x14 in practice)

Fast R-CNN, Girshick, ICCV 2015

Slide credit: Justin Johnson

Cropping Features: RoI Pool

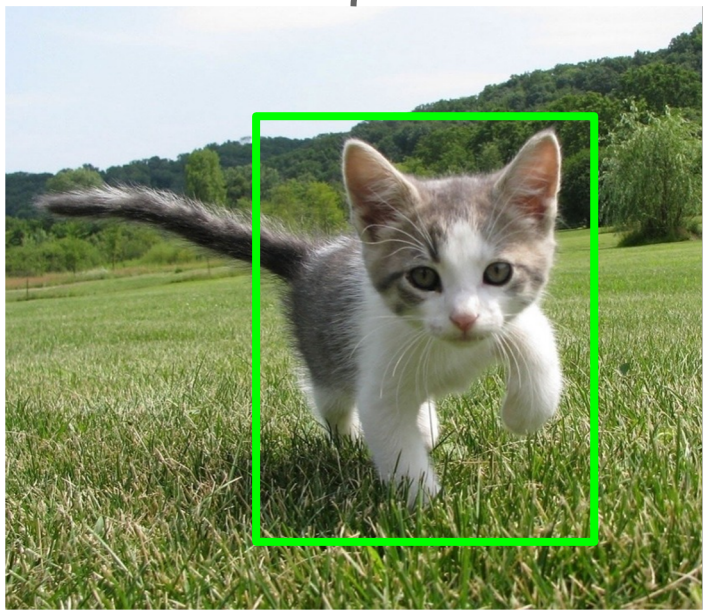


Want features for the box of a fixed size (2x2 in this example, 7x7 or 14x14 in practice)

Fast R-CNN, Girshick, ICCV 2015

Slide credit: Justin Johnson

Cropping Features: RoI Pool



Input Image
(e.g. 3 x 640 x 480)

Project proposal onto features

CNN

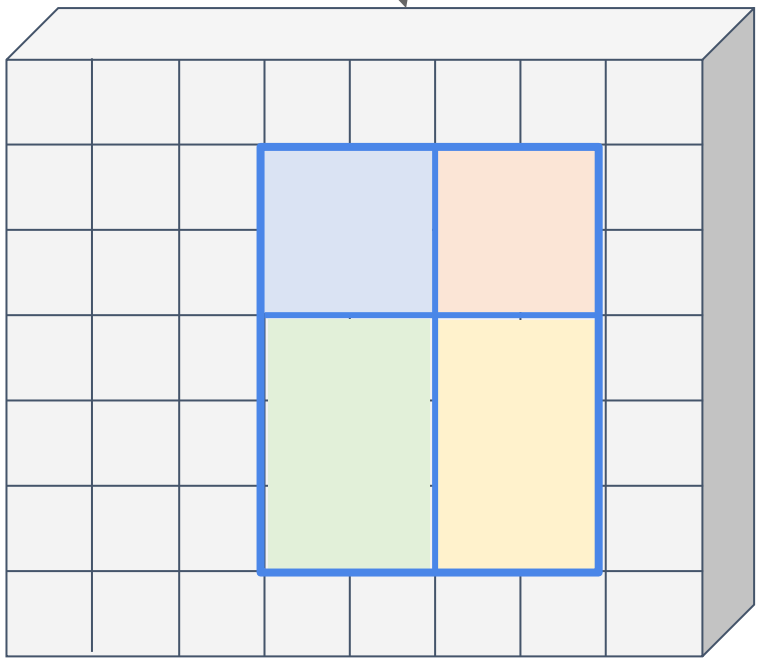
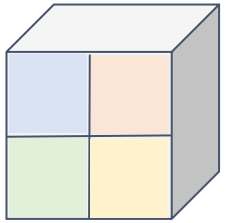


Image features
(e.g. 512 x 20 x 15)

"Snap" to grid cells

Divide into 2x2 grid of (roughly) equal subregions

Max-pool within each subregion



Region features
(here 512 x 2 x 2;
In practice 512x7x7)

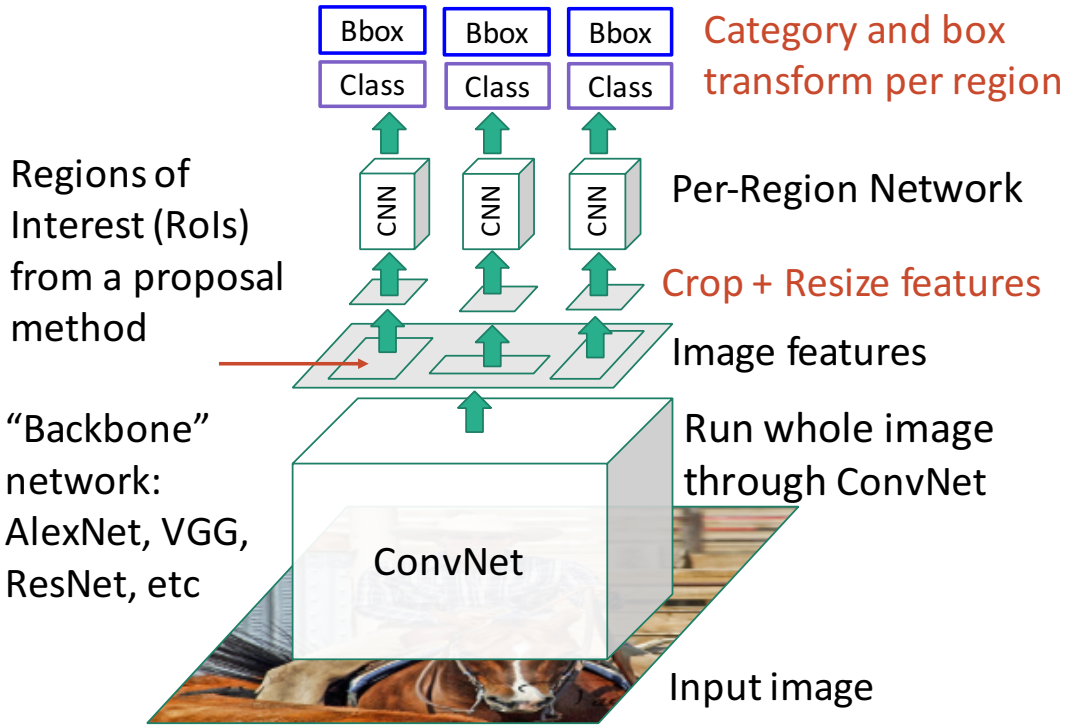
Region features always the same size even if input regions have different sizes!

Fast R-CNN, Girshick, ICCV 2015

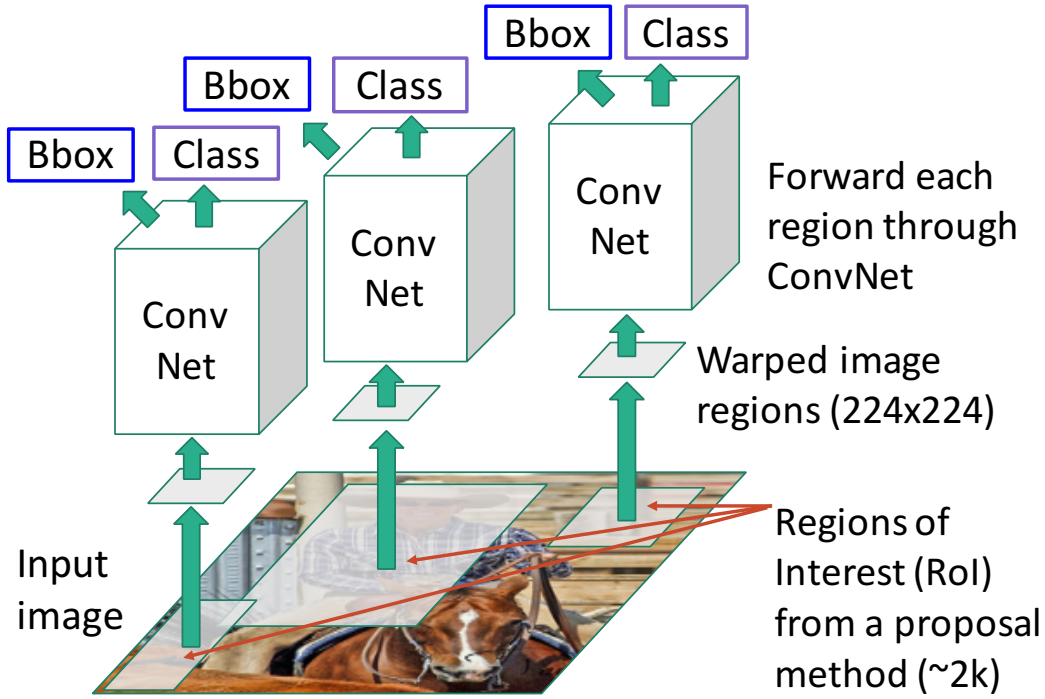
Slide credit: Justin Johnson

Fast R-CNN vs “Slow” R-CNN

Fast R-CNN: Apply differentiable cropping to shared image features

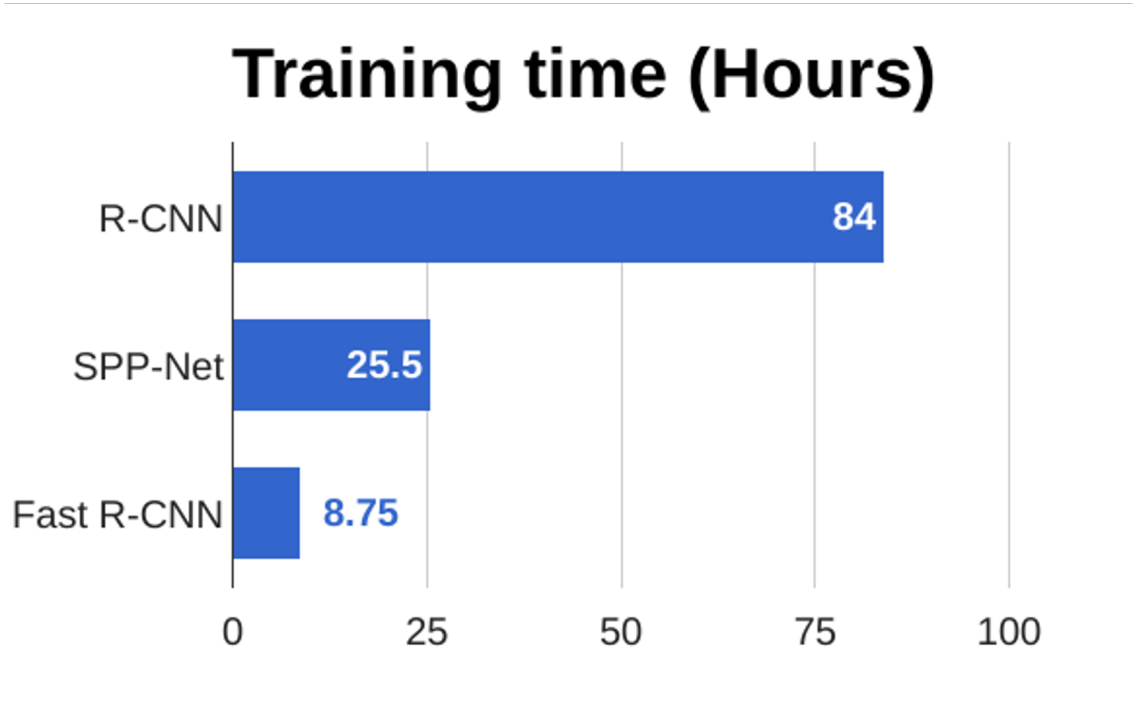


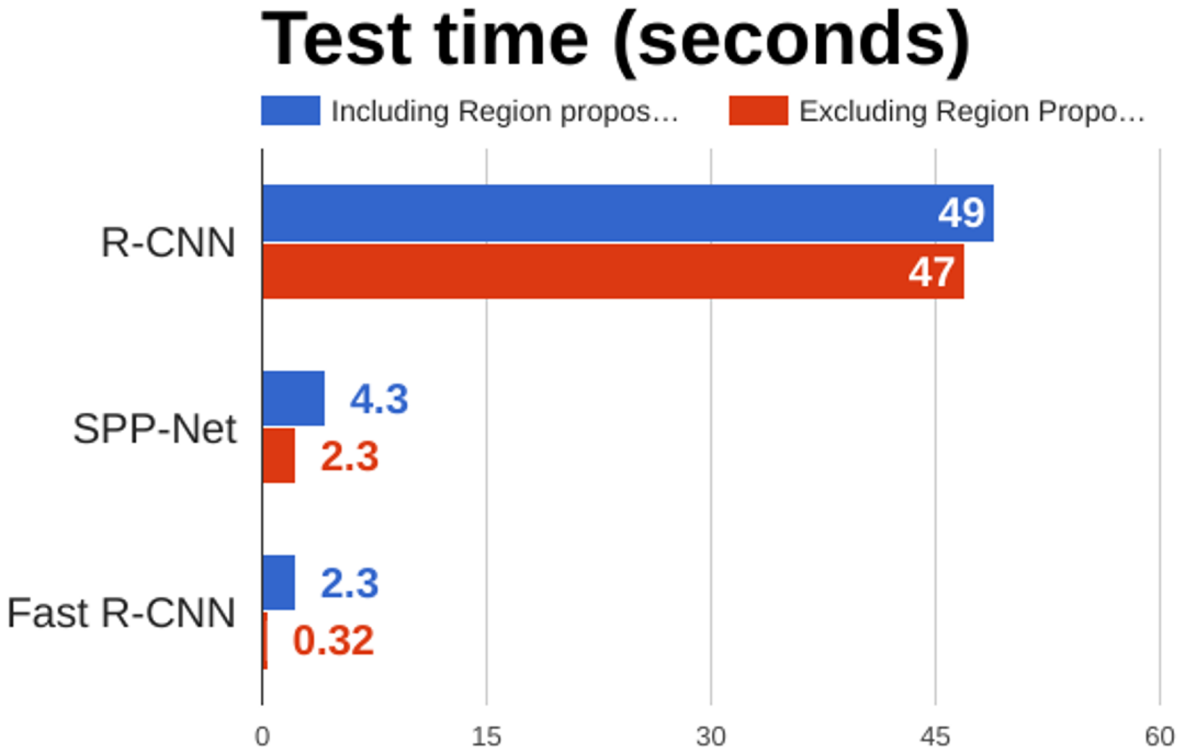
“Slow” R-CNN: Apply differentiable cropping to shared image features



Slide credit: Justin Johnson

Fast R-CNN vs “Slow” R-CNN

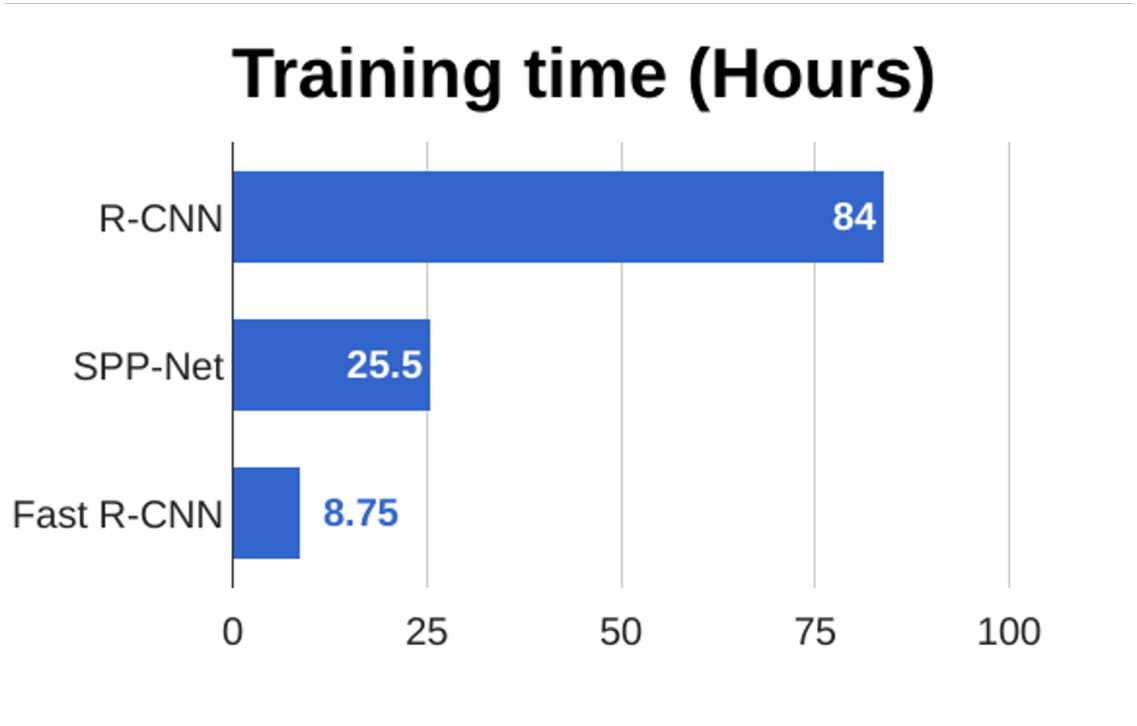


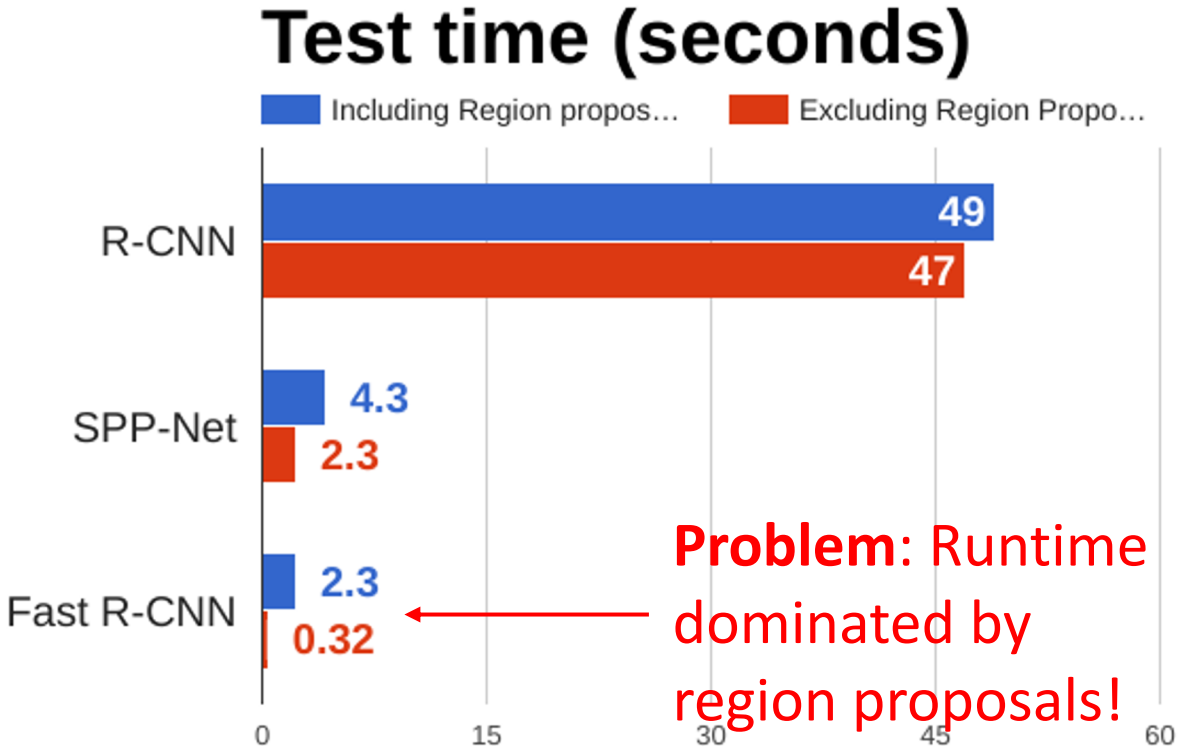


Fast R-CNN, Girshick, ICCV 2015

Slide credit: Justin Johnson

Fast R-CNN vs “Slow” R-CNN

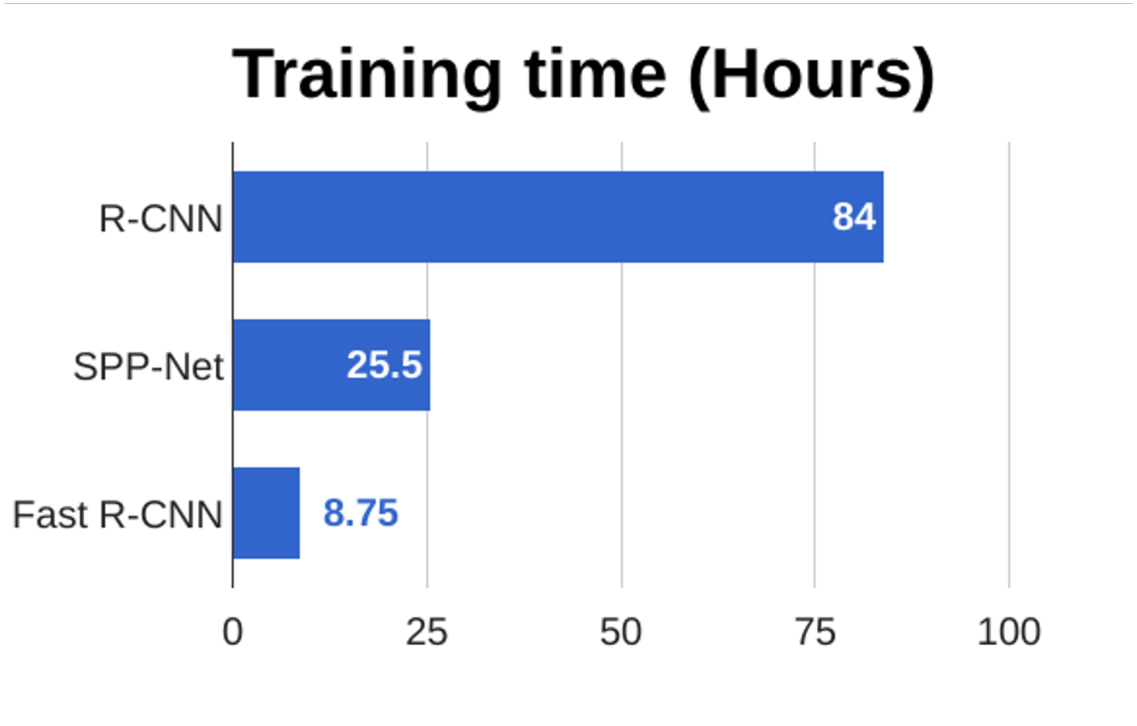


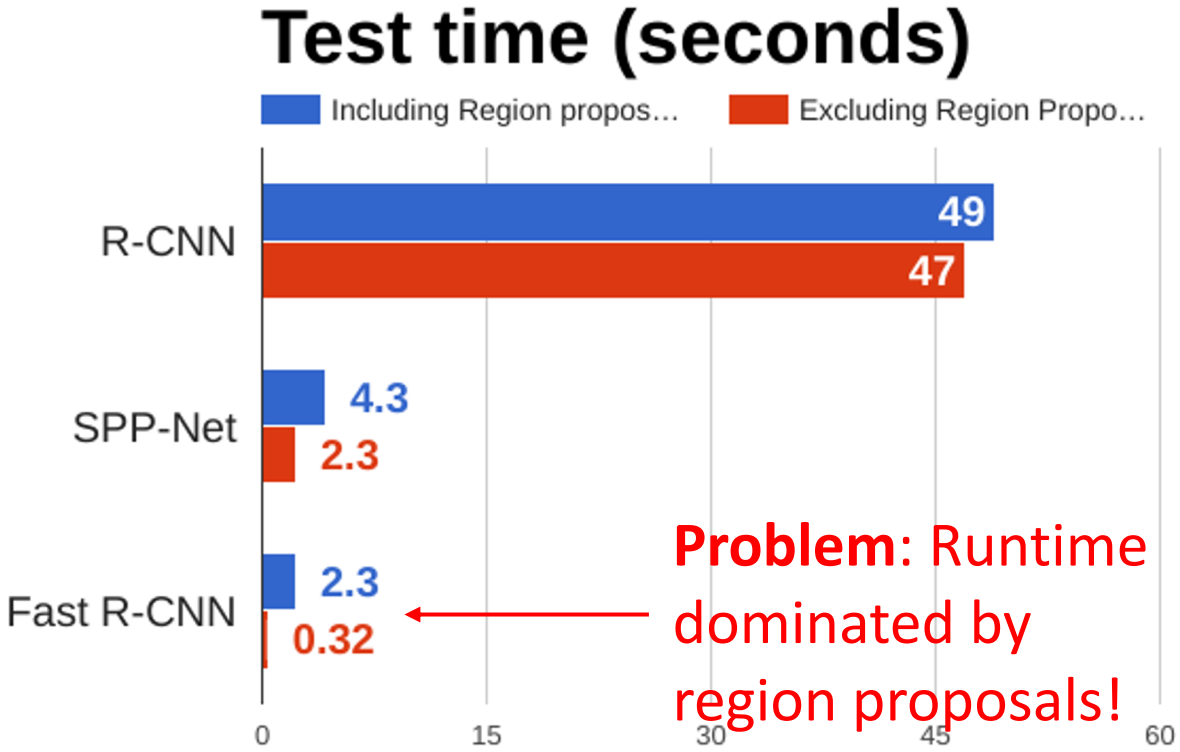


Fast R-CNN, Girshick, ICCV 2015

Slide credit: Justin Johnson

Fast R-CNN vs “Slow” R-CNN





Problem: Runtime dominated by region proposals!

Recall: Region proposals computed by heuristic ”Selective Search” algorithm on CPU -- let’s learn them with a CNN instead!

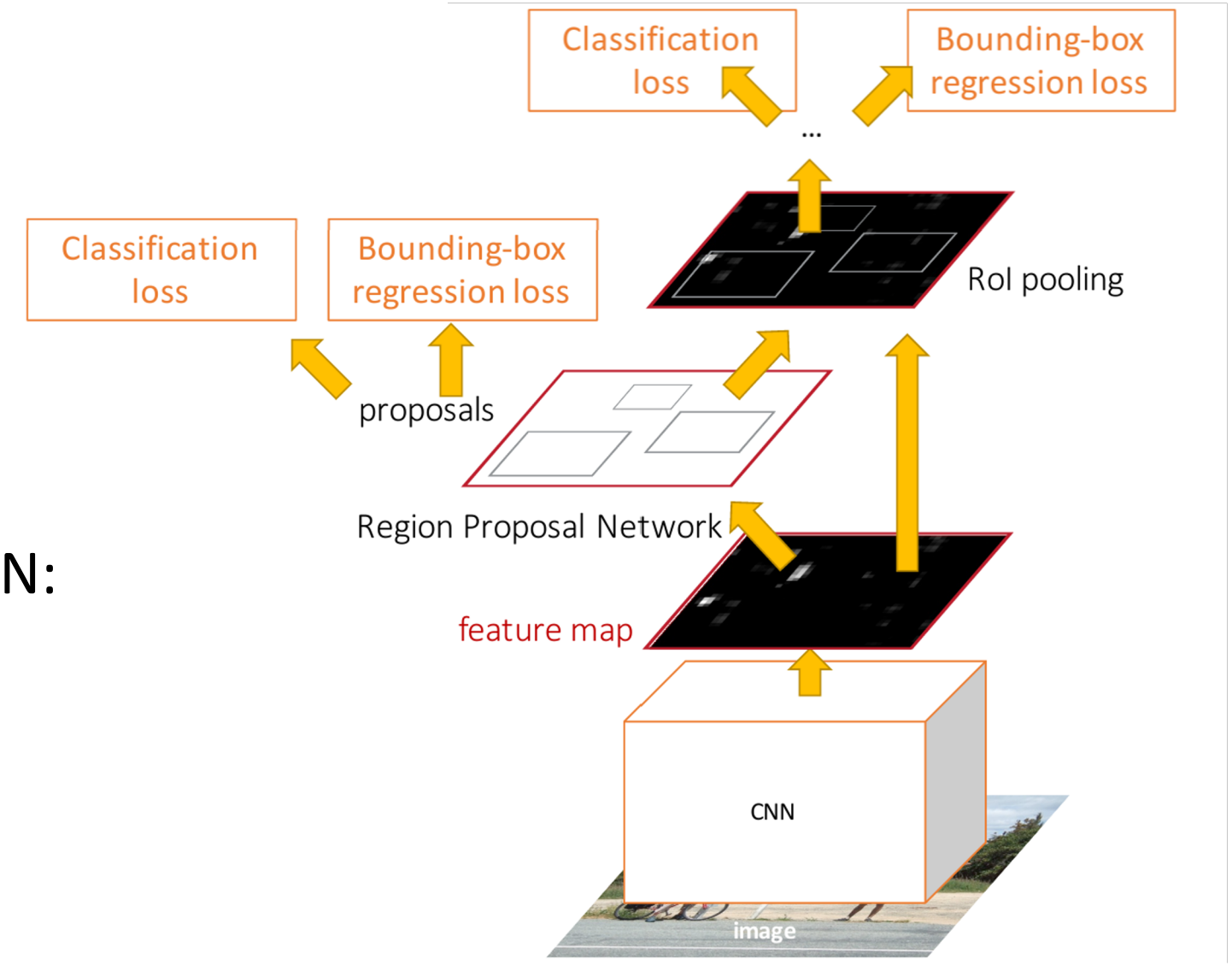
Fast R-CNN, Girshick, ICCV 2015

Slide credit: Justin Johnson

Faster R-CNN: Learnable Region 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



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

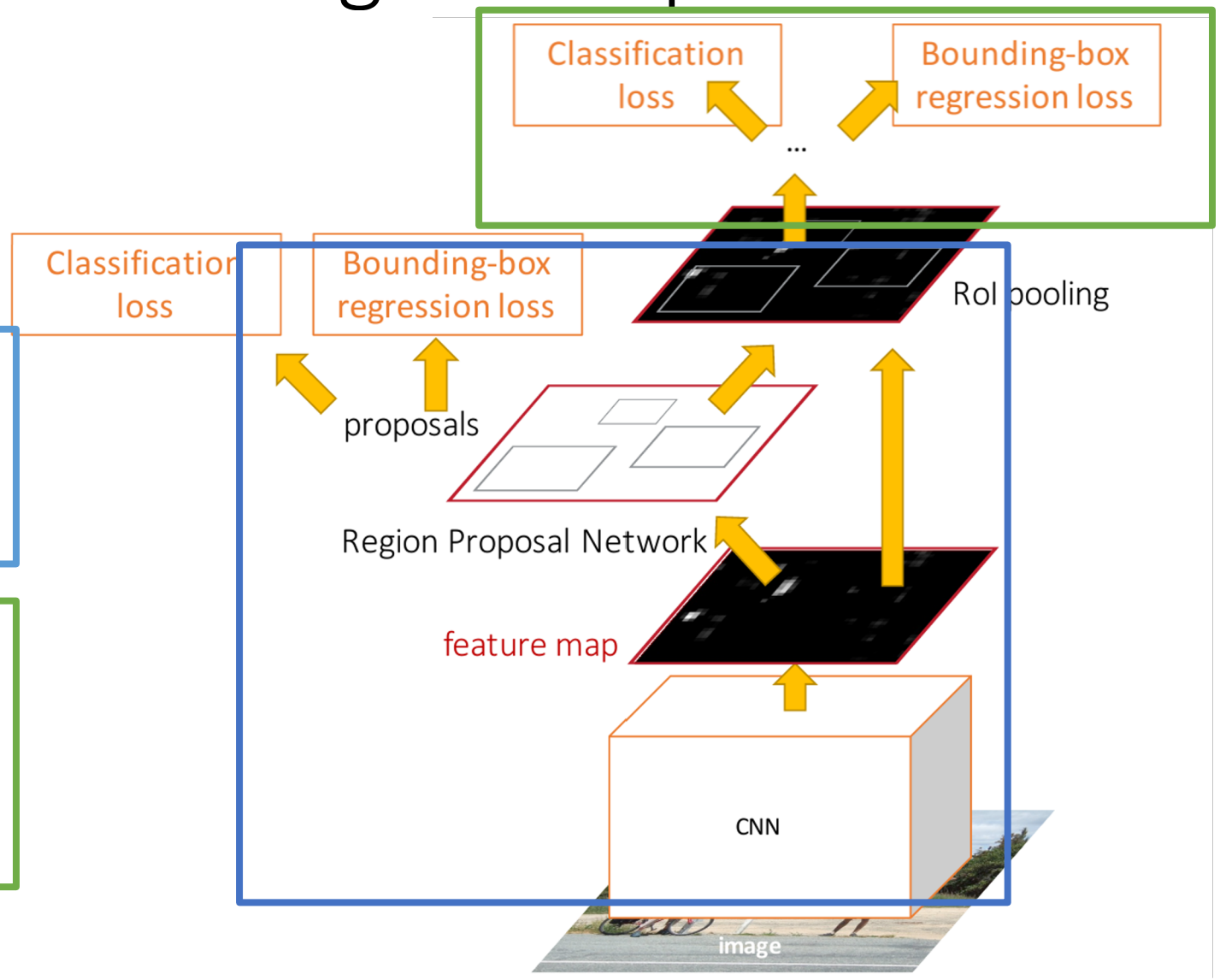
Slide credit: Justin Johnson

Faster R-CNN: Learnable Region Proposals

Faster R-CNN is a
Two-stage object detector

First stage: Run once per image
- Backbone network
- Region proposal network

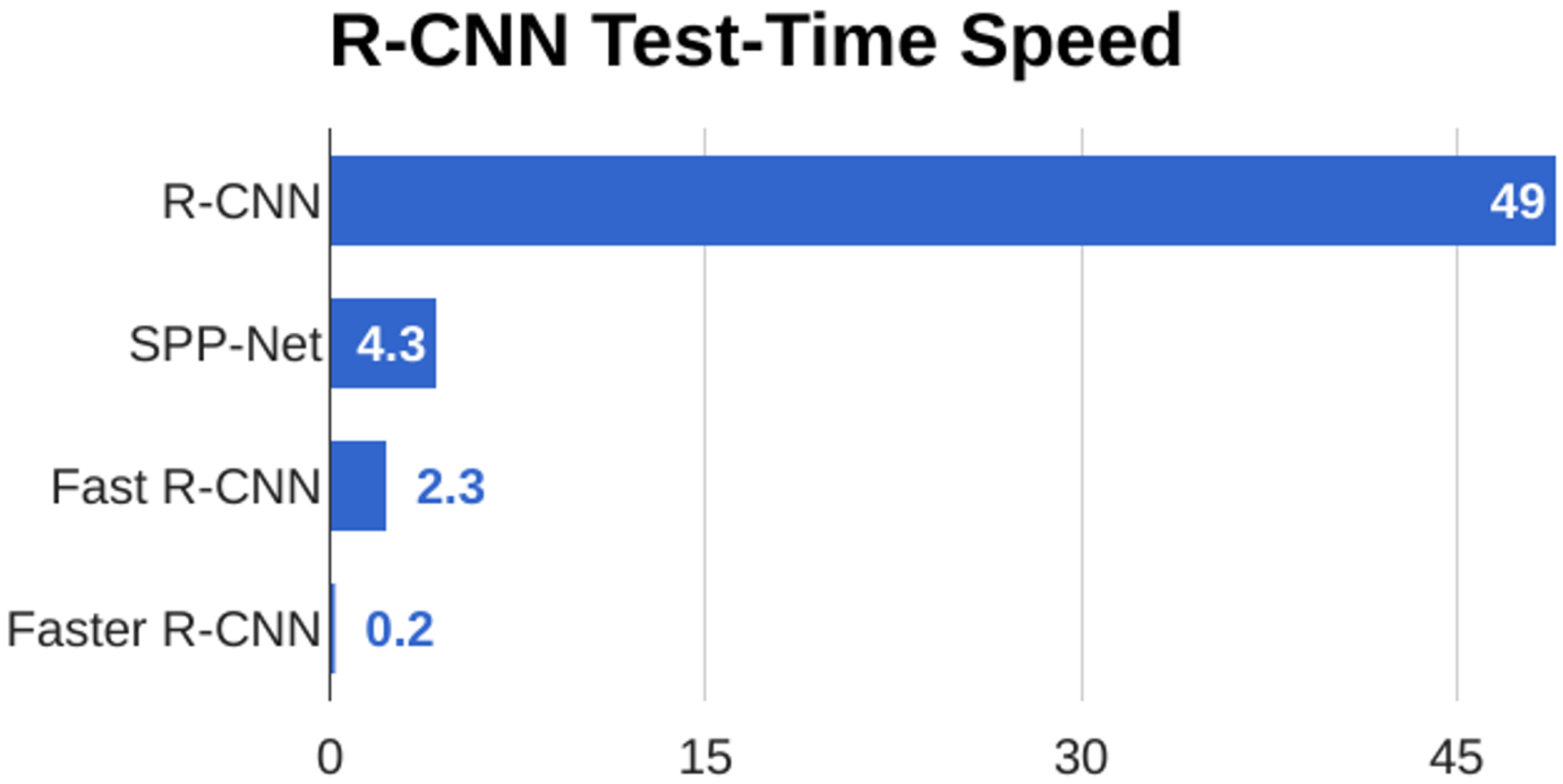
Second stage: Run once per region
- Crop features: RoI pool / align
- Predict object class
- Prediction bbox offset



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

Slide credit: Justin Johnson

Faster R-CNN: Learnable Region Proposals



Slide credit: Justin Johnson

Summary

- Broadly two families of approaches:
  1. Sliding Window
  2. Region-based

- R-CNN family
  - R-CNN [Girshick et al, 2014]
  - SPP-Net [He et al, 2014]
  - Fast R-CNN [Girshick et al 2015]
  - Faster R-CNN [Ren et al, 2015]

Is Object Detection Solved?

Today’s Agenda

- A brief history of object detection
- Modern object detection
- Beyond bounding boxes
- New trends

What’s Wrong with Boxes?

Figure credit: Piotr Dollár

What’s Wrong with Boxes?

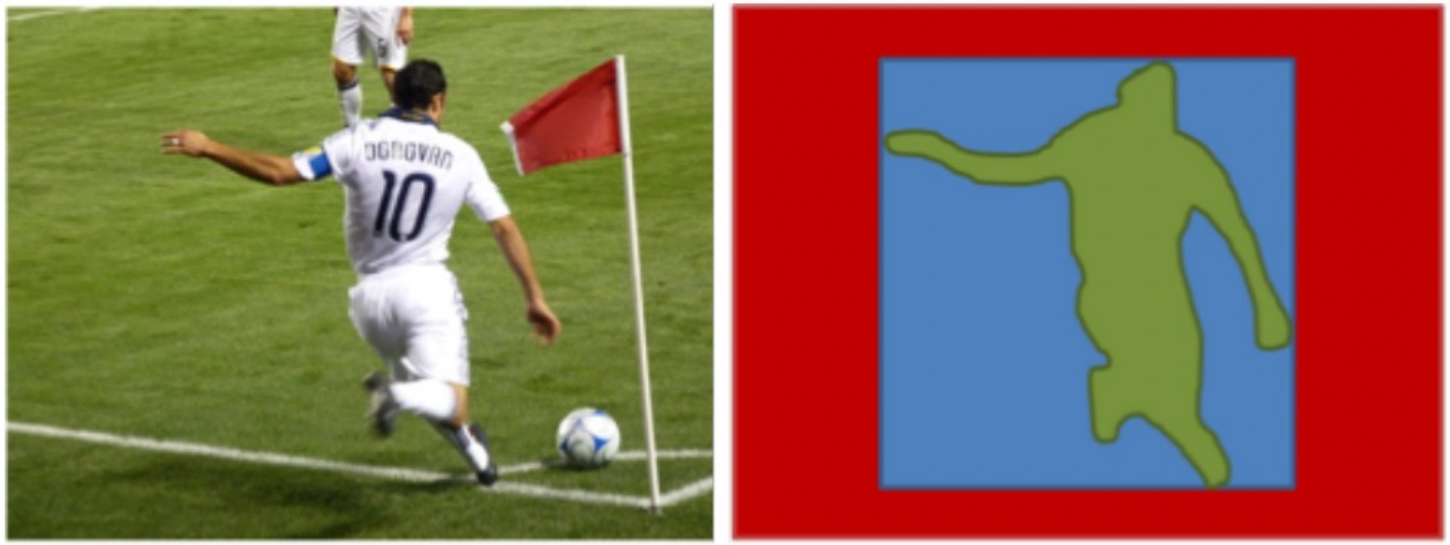
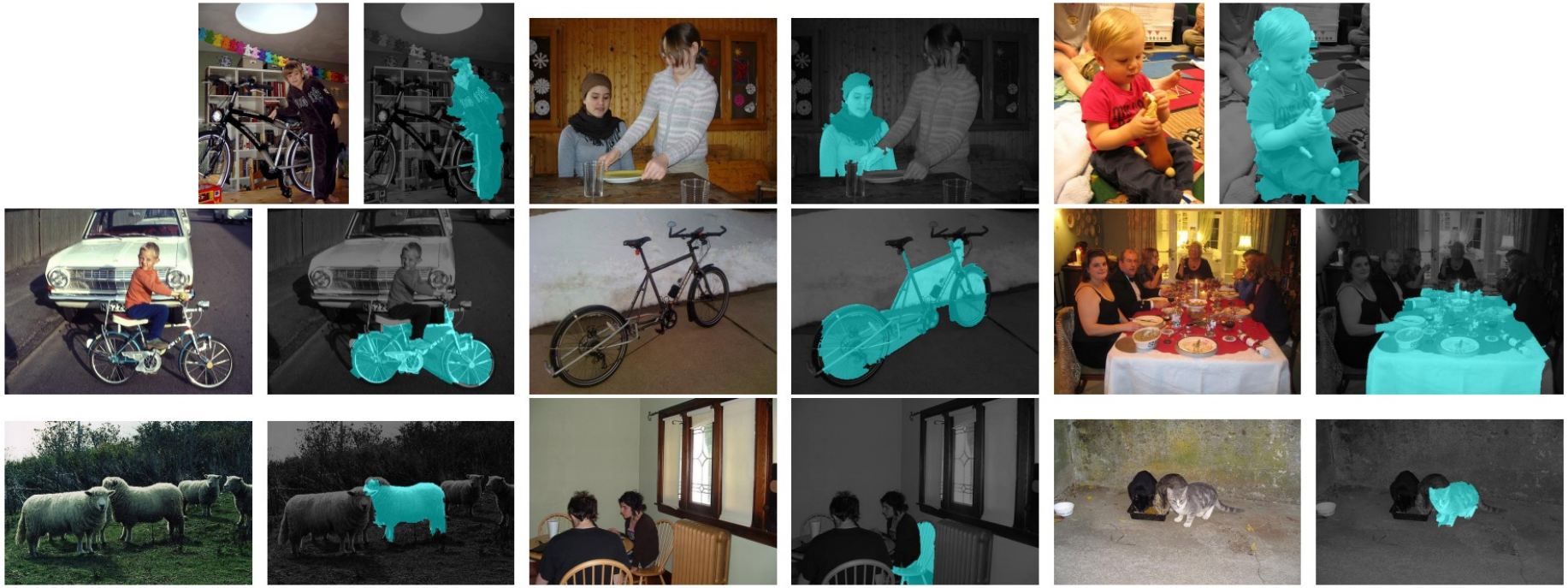


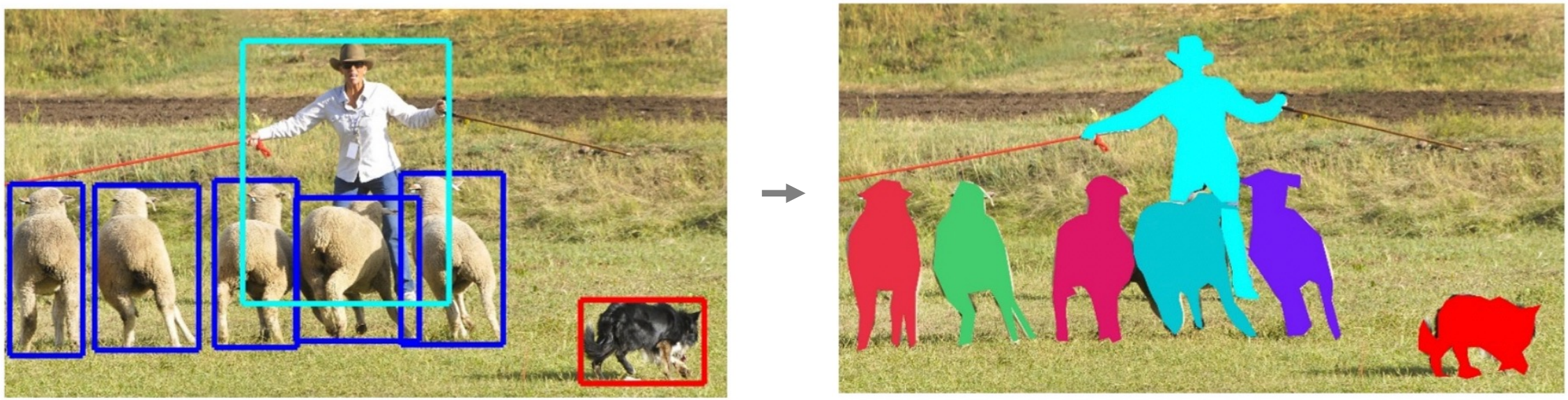
Figure credit: Piotr Dollár

Simultaneous Detection and Segmentation



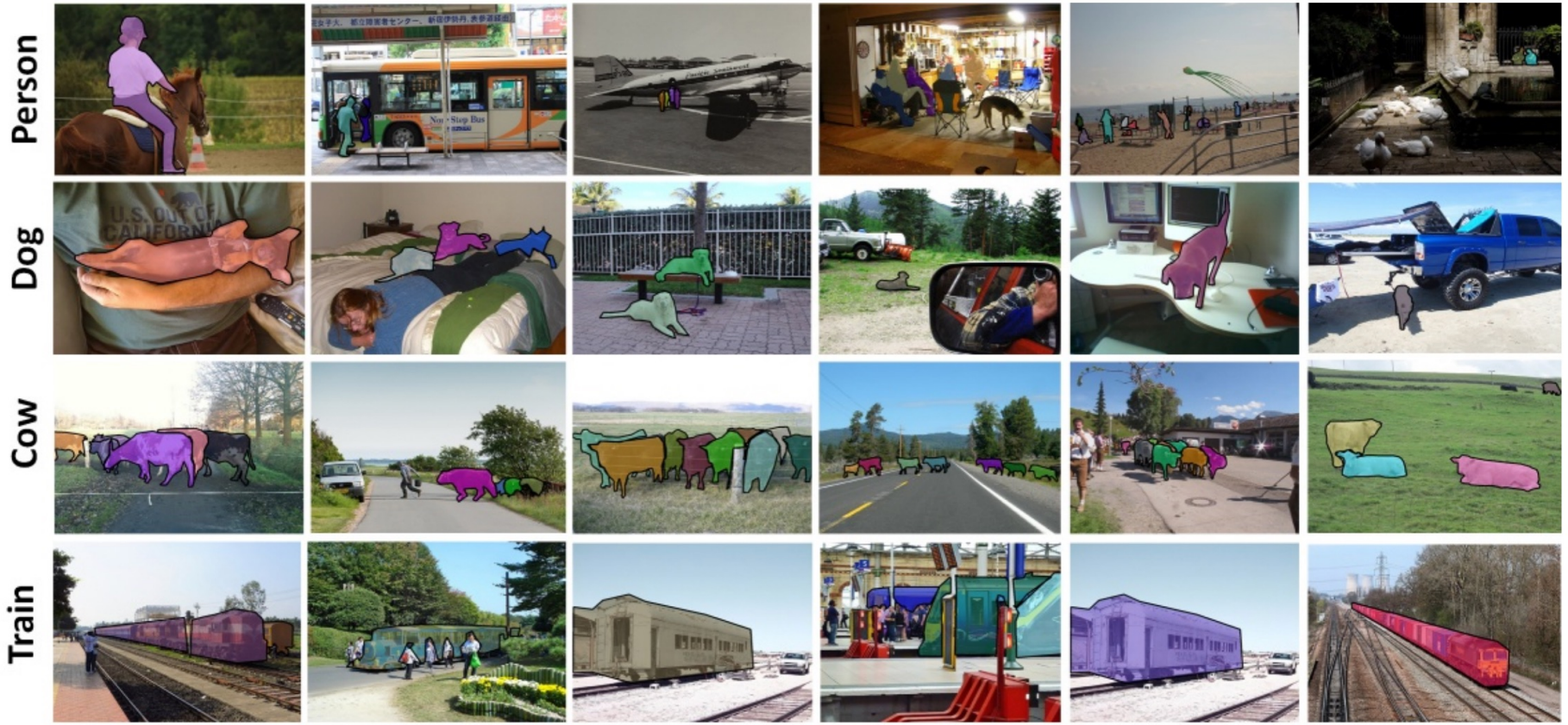
Simultaneous Detection and Segmentation, Hariharan et al, ECCV 2014

COCO Instance Segmentation



Microsoft COCO: Common Objects in Context, Lin et al, ECCV 2016

COCO Instance Segmentation



Microsoft COCO: Common Objects in Context, Lin et al, ECCV 2016

Computer Science > Computer Vision and Pattern Recognition

[Submitted on 20 Mar 2017 (v1), last revised 24 Jan 2018 (this version, v3)]

Mask R-CNN

Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick

We present a conceptually simple, flexible, and general framework for object instance segmentation. Our approach efficiently detects objects in an image while simultaneously generating a high-quality segmentation mask for each instance. The method, called Mask R-CNN, extends Faster R-CNN by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition. Mask R-CNN is simple to train and adds only a small overhead to Faster R-CNN, running at 5 fps. Moreover, Mask R-CNN is easy to generalize to other tasks, e.g., allowing us to estimate human poses in the same framework. We show top results in all three tracks of the COCO suite of challenges, including instance segmentation, bounding-box object detection, and person keypoint detection. Without bells and whistles, Mask R-CNN outperforms all existing, single-model entries on every task, including the COCO 2016 challenge winners. We hope our simple and effective approach will serve as a solid baseline and help ease future research in instance-level recognition. Code has been made available at: this https URL

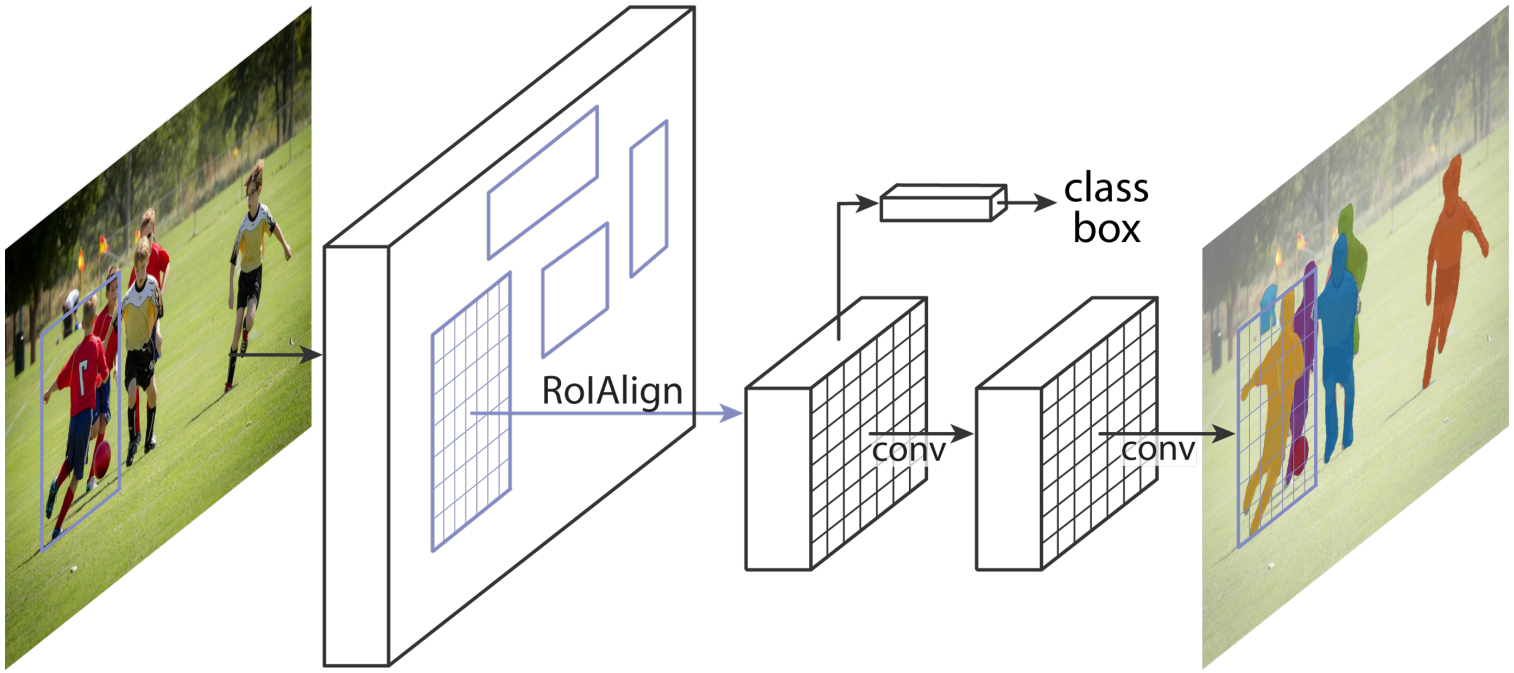
Comments:	open source; appendix on more results
Subjects:	**Computer Vision and Pattern Recognition (cs.CV)**
Cite as:	arXiv:1703.06870 **[cs.CV]**
	(or arXiv:1703.06870v3 **[cs.CV]** for this version)
	https://doi.org/10.48550/arXiv.1703.06870

Submission history

From: Kaiming He [view email]
[v1] Mon, 20 Mar 2017 17:53:38 UTC (6,270 KB)
[v2] Wed, 5 Apr 2017 20:14:55 UTC (7,041 KB)
[v3] Wed, 24 Jan 2018 07:54:08 UTC (7,061 KB)

Instance Segmentation

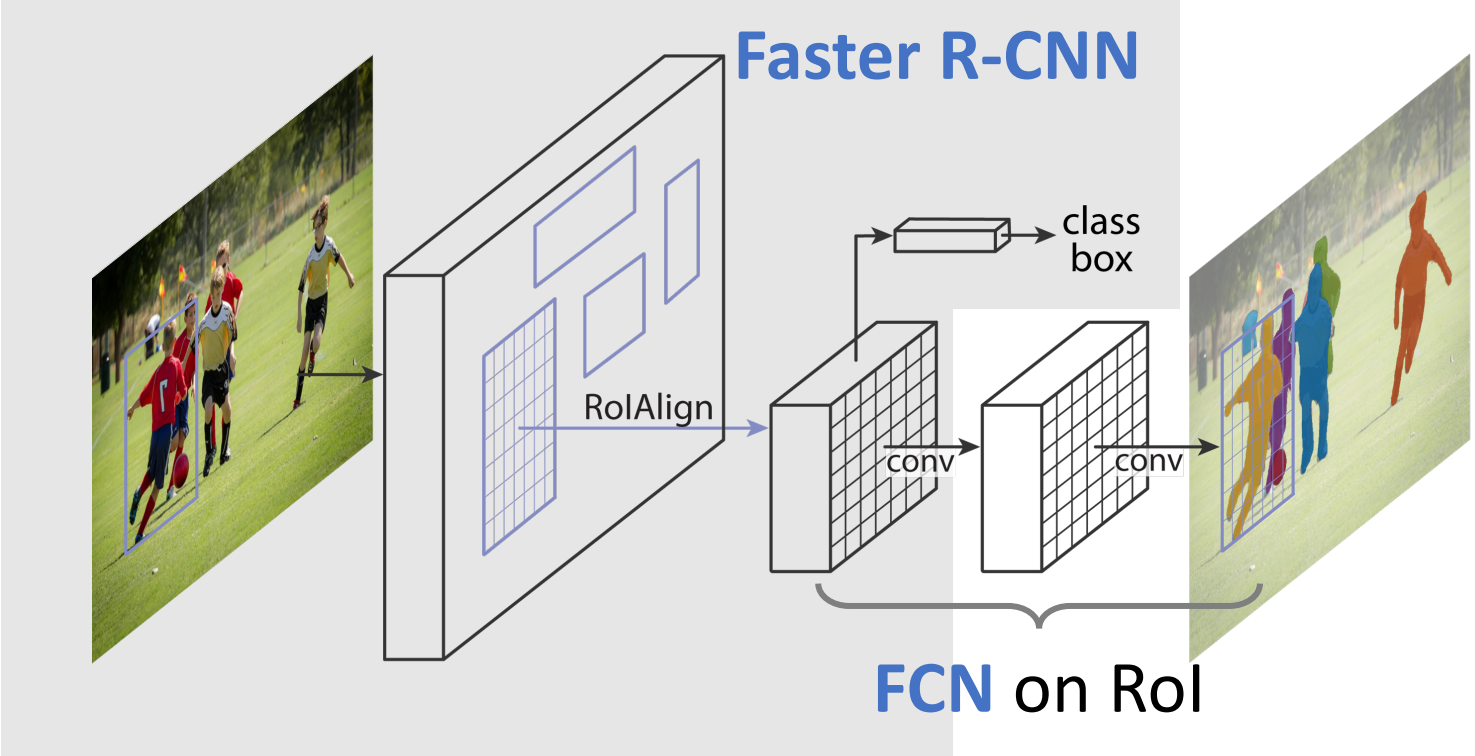
- **Goals** of Mask R-CNN
  - ✓Good speed
  - ✓Good accuracy
  - ✓Intuitive
  - ✓Easy to use



Slide credit: Kaiming He

What is Mask R-CNN

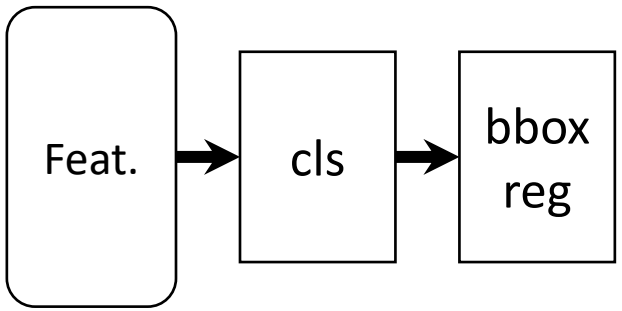
- Mask R-CNN = **Faster R-CNN** with **FCN** on each RoI



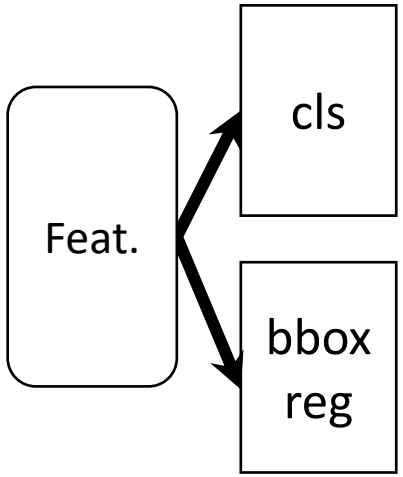
Slide credit: Kaiming He

What is Mask R-CNN: **Parallel** Heads

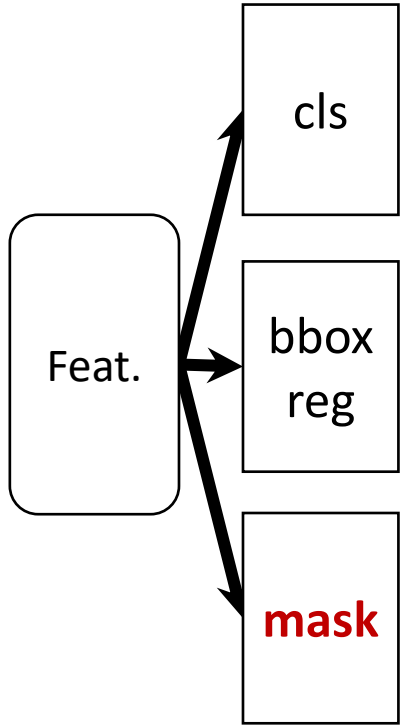
- Easy, fast to implement and train



(slow) R-CNN



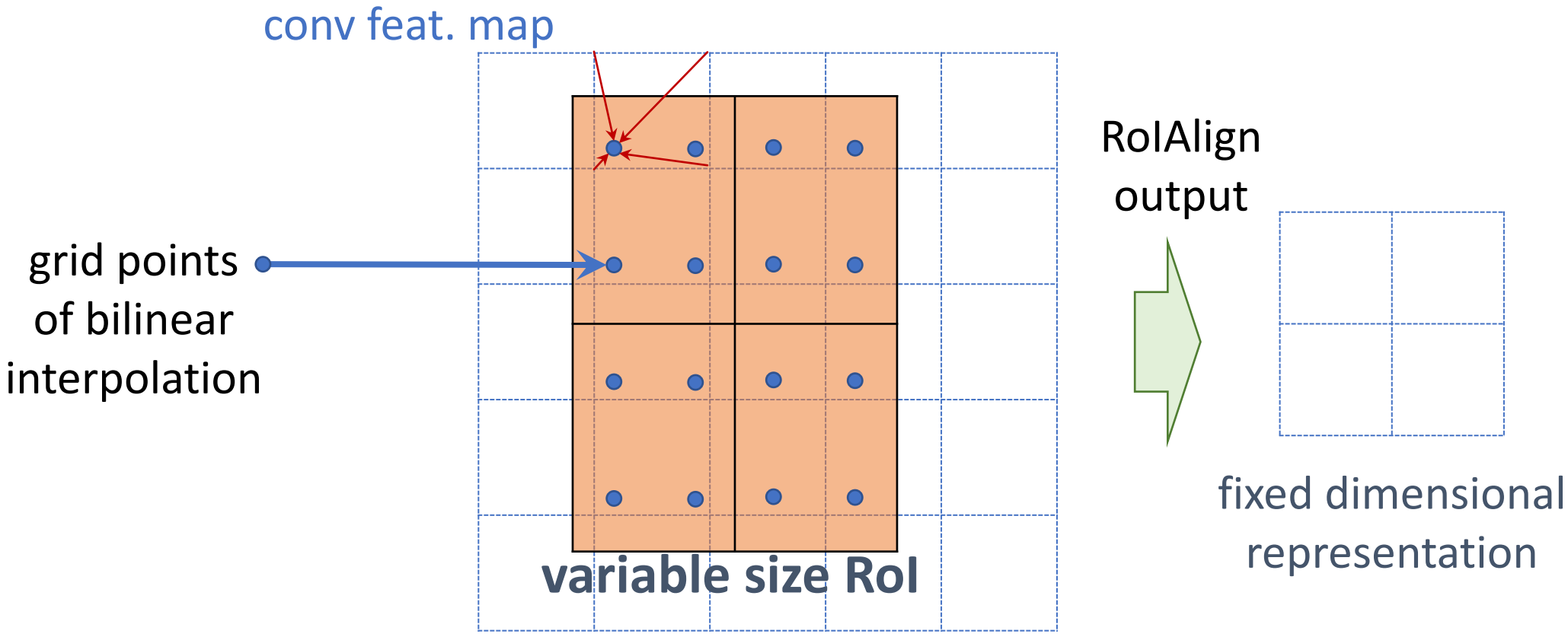
Fast/er R-CNN



Mask R-CNN

Slide credit: Kaiming He

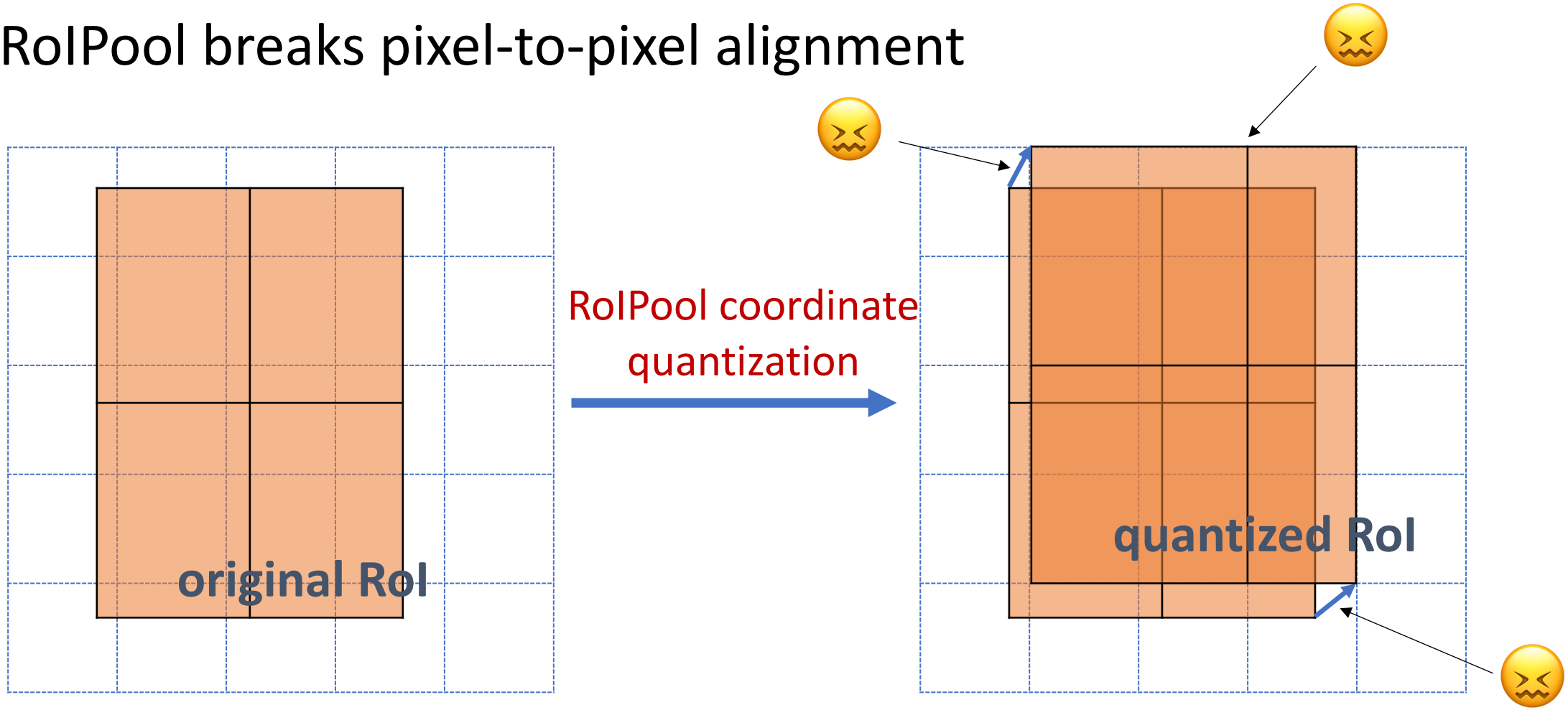
What is Mask R-CNN: **RoIAlign**



Slide credit: Kaiming He

RoIAlign vs. RoIPool

- RoIPool breaks pixel-to-pixel alignment



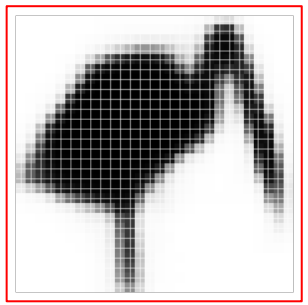
Slide credit: Kaiming He

What is Mask R-CNN: **FCN** Mask Head

- Pixel-to-pixel aligned

RoI

28x28 soft prediction



Resized soft prediction

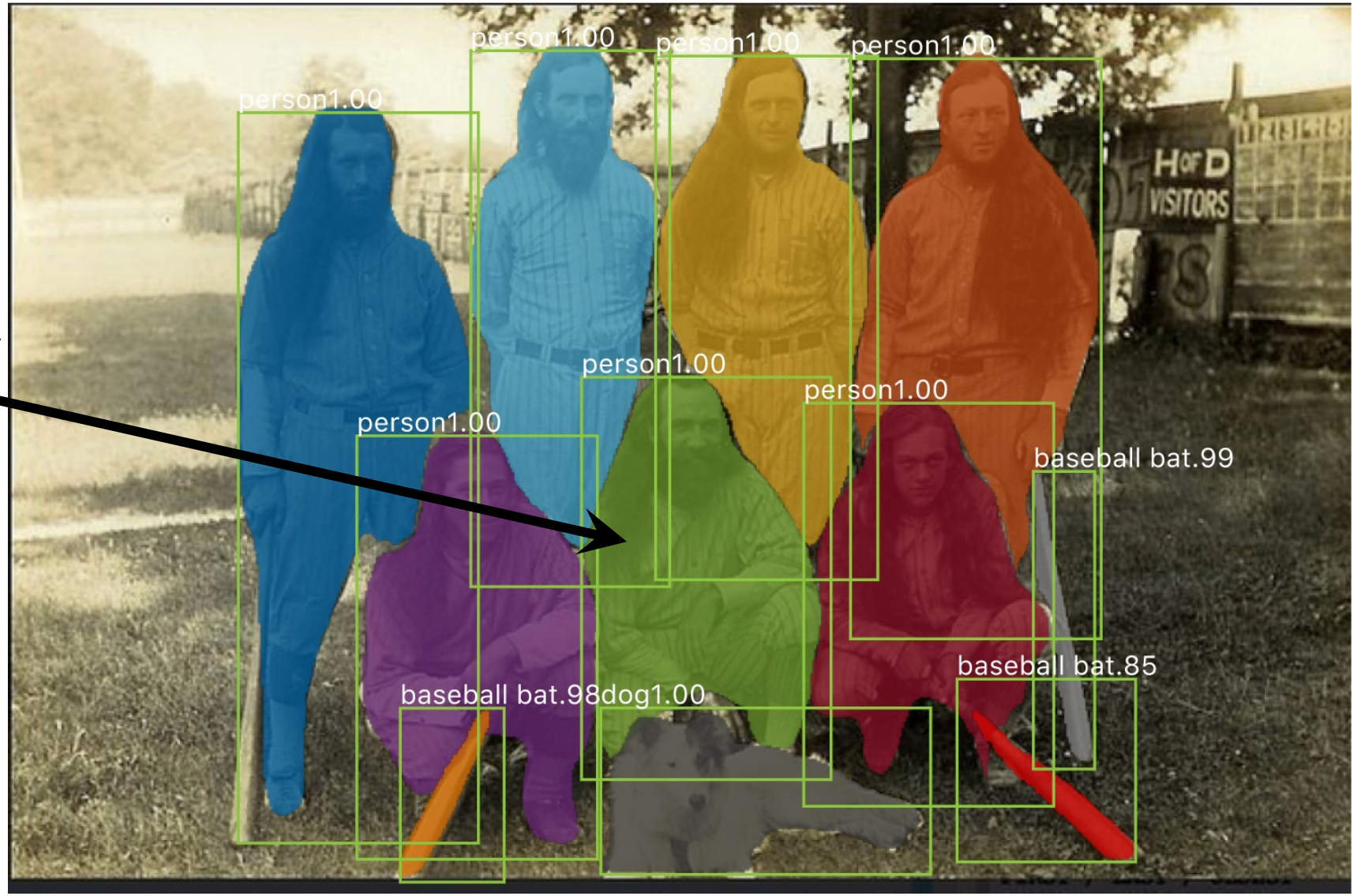
Final mask

Slide credit: Kaiming He

Mask R-CNN results on COCO

Slide credit: Kaiming He

object surrounded by same-category objects



Mask R-CNN results on COCO

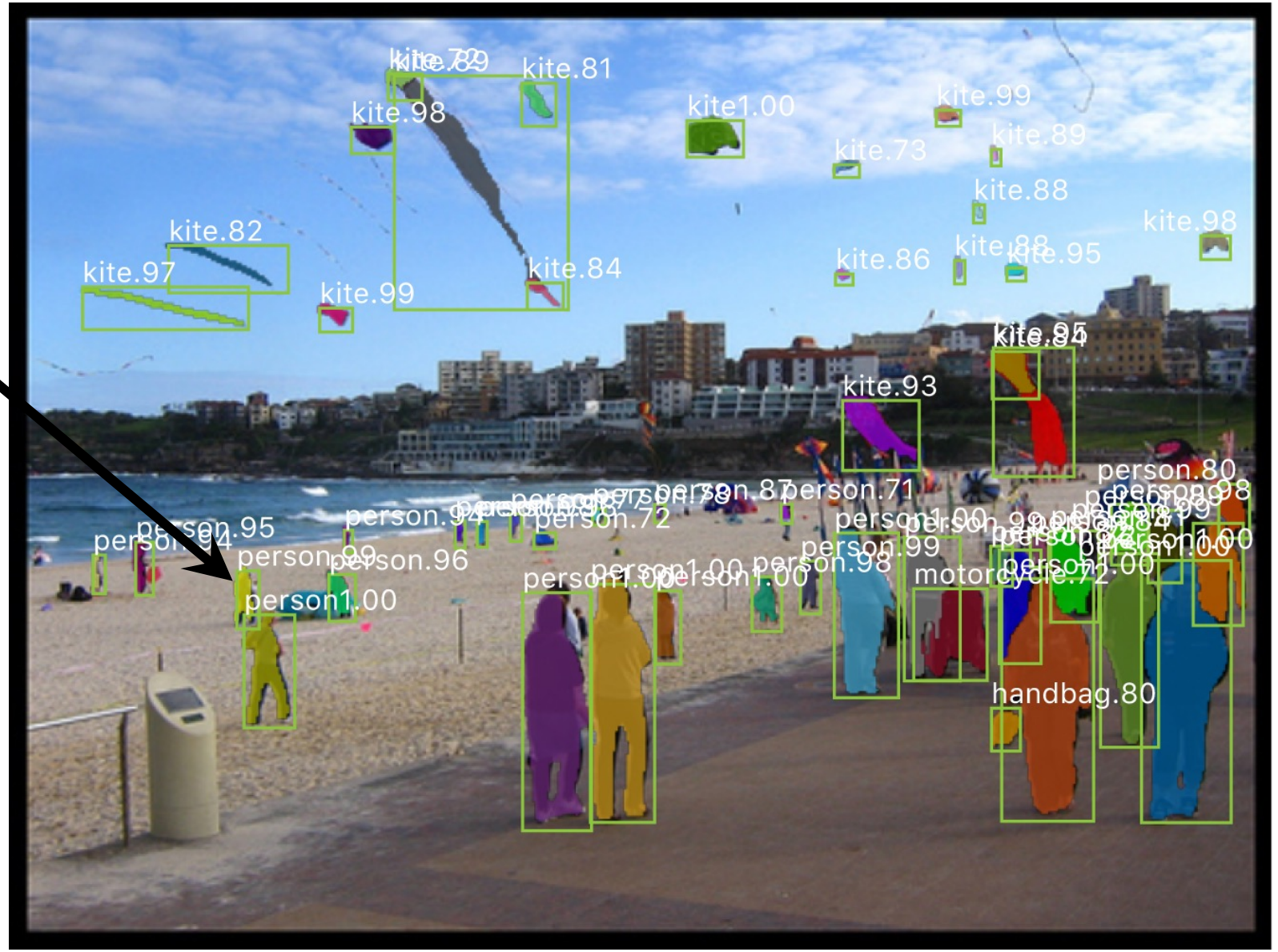
Slide credit: Kaiming He

disconnected object

Mask R-CNN results on COCO

Slide credit: Kaiming He

small objects



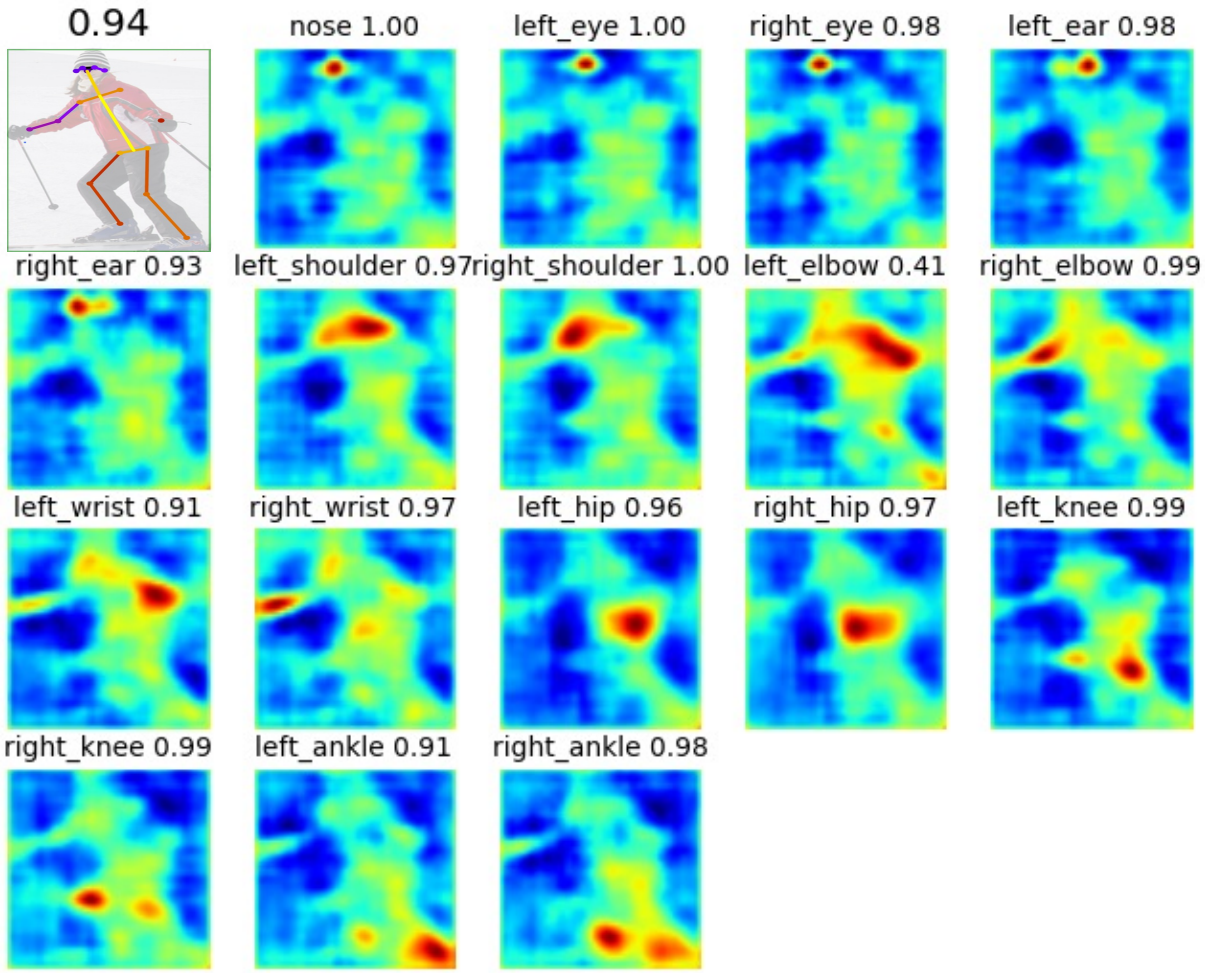
Mask R-CNN results on COCO

Slide credit: Kaiming He

Extension: for Human Keypoint Detection

- 1 keypoint = 1-hot mask
- Human pose = 17 masks

- Softmax over spatial locations

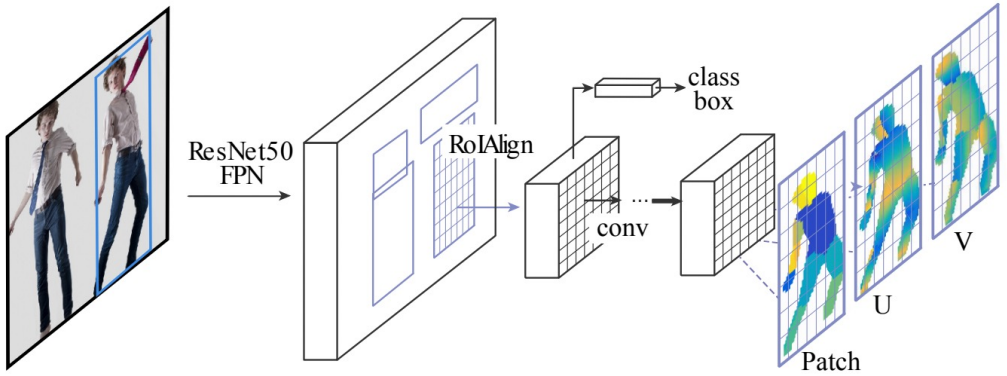


Slide credit: Kaiming He

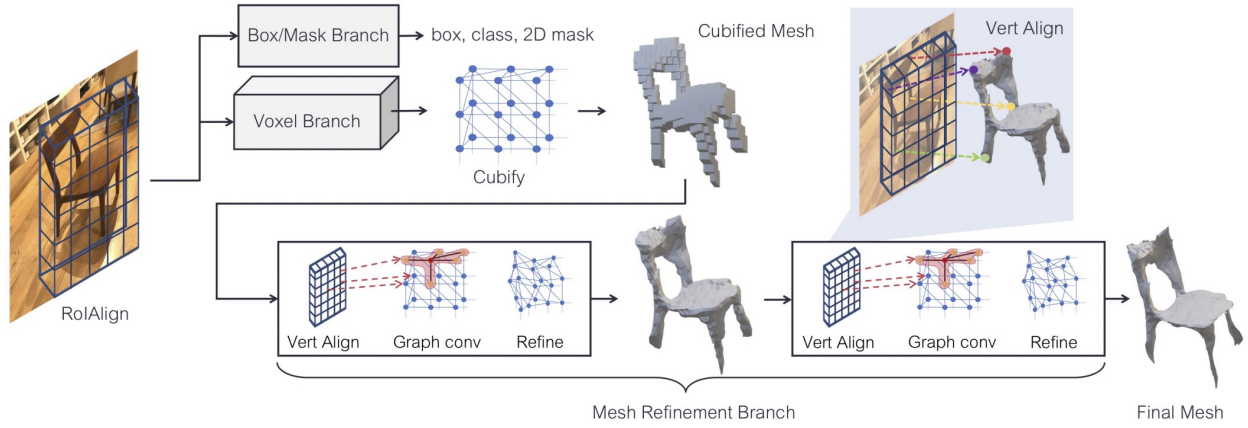
Mask R-CNN results for human pose on COCO

Slide credit: Kaiming He

Generalized R-CNN: Beyond Masks



DensePose R-CNN



Mesh R-CNN

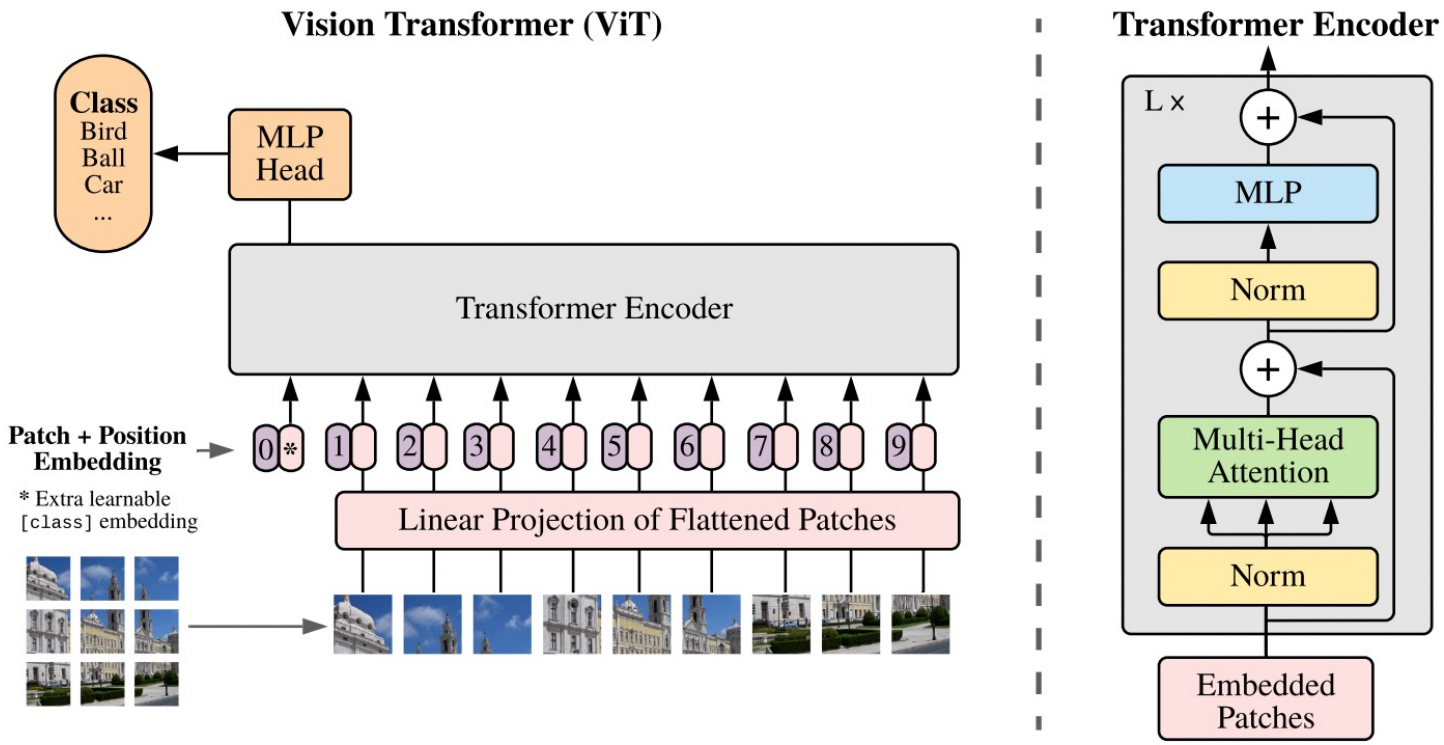
Mesh R-CNN, Gkioxari et al, ICCV 2019

DensePose: Dense Human Pose Estimation in the Wild, Güler et al, CVPR 2018

Today's Agenda

- A brief history of object detection
- Modern object detection
- Beyond bounding boxes
- New trends

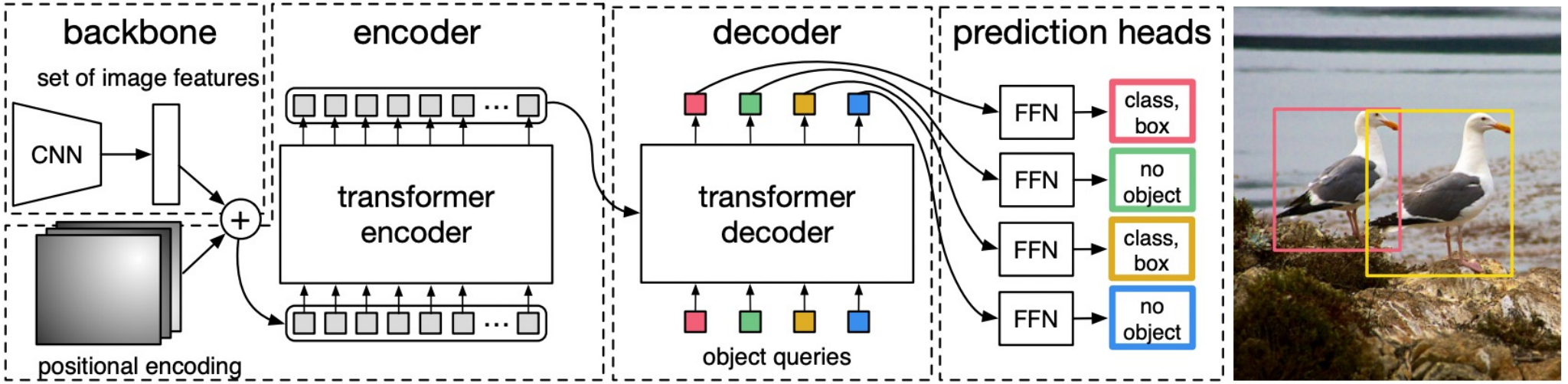
Transformer Architecture



Attention is All you Need, Vaswani et al, NIPS 2017

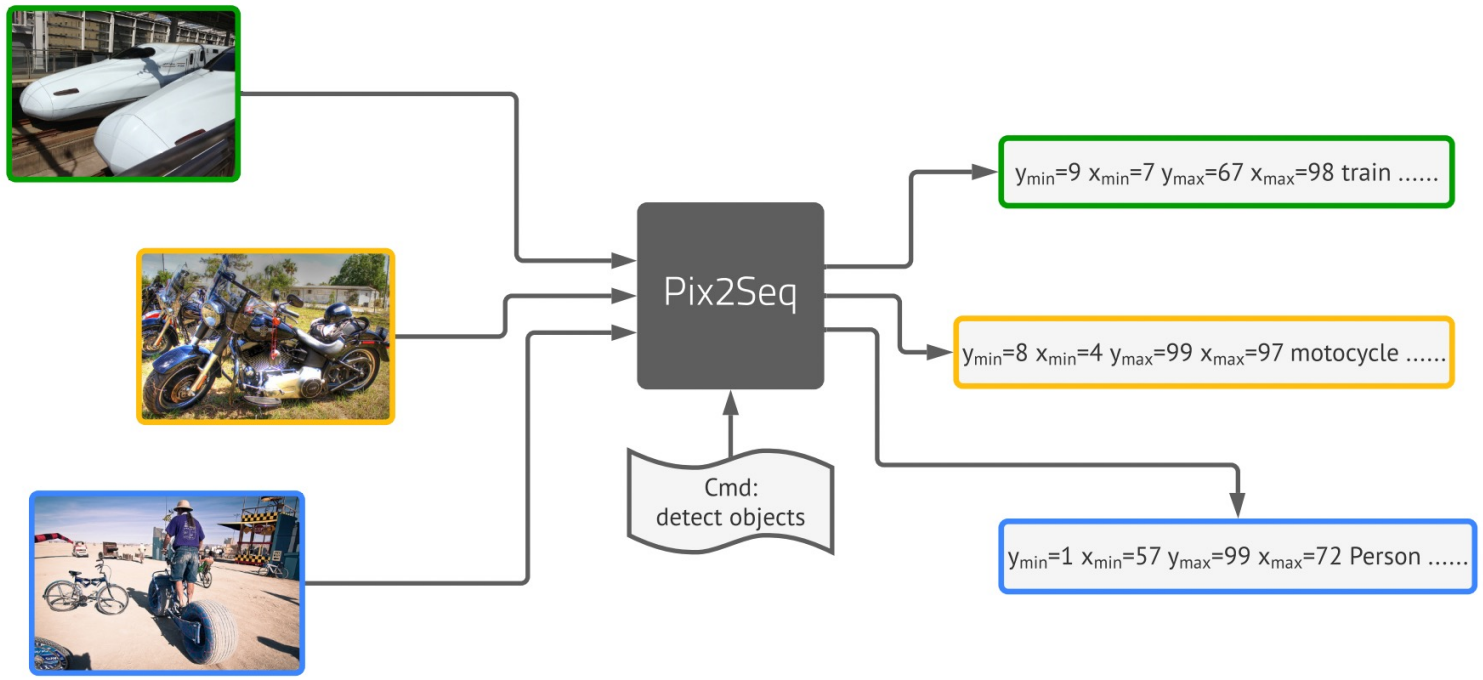
An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, Dosovitskiy et al, ICLR 2021

Detection Transformer



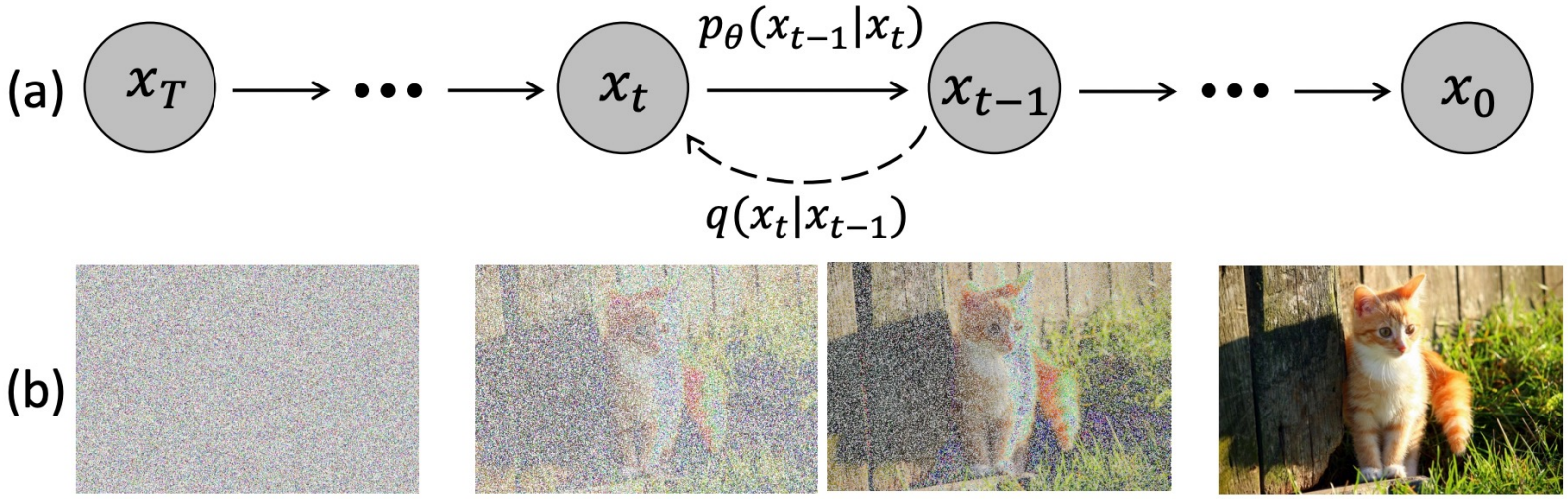
End-to-End Object Detection with Transformers, Carion et al, ECCV 2020

Object Detection as Language Modeling



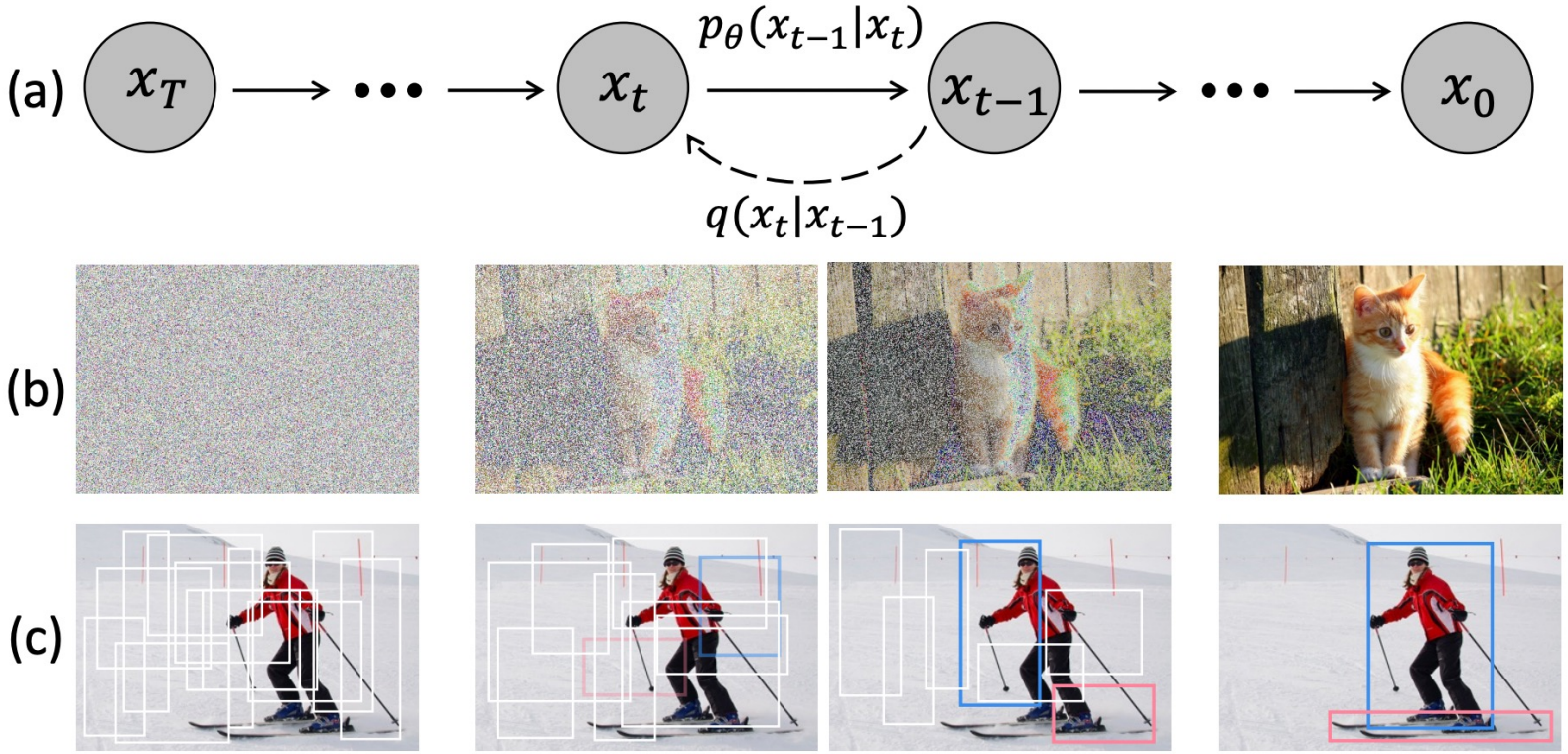
Pix2Seq: A Language Modeling Framework for Object Detection, Chen et al, 2022

Object Detection with Diffusion Models



DiffusionDet: Diffusion Model for Object Detection, Shen et al, 2022

Object Detection with Diffusion Models



DiffusionDet: Diffusion Model for Object Detection, Shen et al, 2022

Today’s Agenda

- A brief history of object detection
- Modern object detection
- Instance segmentation
- New trends

References

- Kaiming He's Talk:
  - https://www.youtube.com/watch?v=g7z4mkfRjI4
- Ross Girshick's Tutorials:
  - https://www.youtube.com/watch?v=m60uJVIE4Ys
  - https://www.youtube.com/watch?v=her4_rzx09o
- Larry Zitnick's Talks:
  - https://www.youtube.com/watch?v=fbFYdzatOMg
  - https://www.youtube.com/watch?v=a1JKttFTG3M
- Justin Johnson's Lectures:
  - https://web.eecs.umich.edu/~justincj/slides/eecs498/WI2022/598_WI2022_lecture13.pdf
  - https://web.eecs.umich.edu/~justincj/slides/eecs498/WI2022/598_WI2022_lecture14.pdf