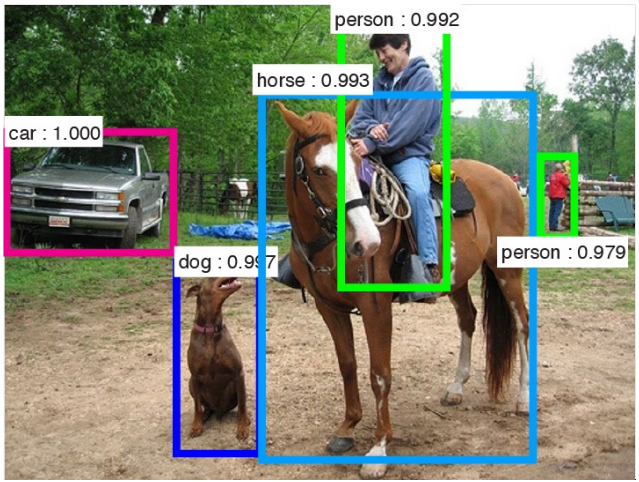
Recognition

Convolutional neural networks and sequence models for segmentation object detection, and classification



CSC420
David Lindell
University of Toronto
cs.toronto.edu/~lindell/teaching/420
Slide credit: Babak Taati ←Ahmed Ashraf ←Sanja Fidler; Bill Freeman, Phillip Isola, Tianhong Li, Justin Johnson

TCIG
Toronto Computational Imaging Group

Course Overview

Outline

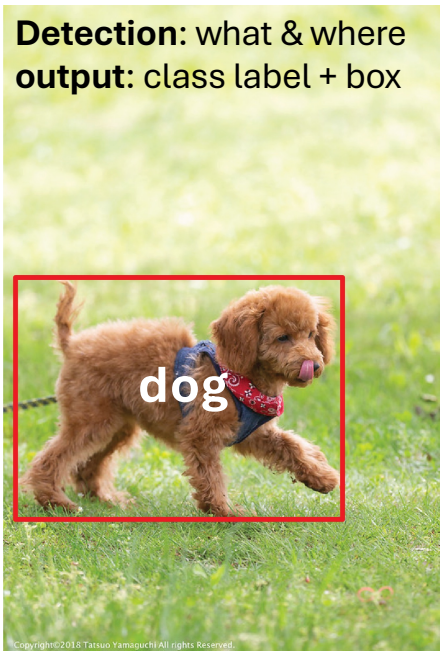
- motivation / overview of recognition tasks
- CNNs
  - Classification
  - Segmentation
  - Object Detection

Outline

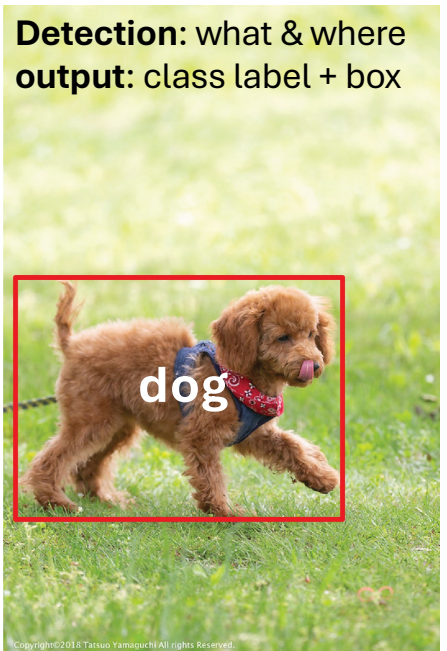
- motivation / overview of recognition tasks
- CNNs
  - Classification
  - Segmentation
  - Object Detection

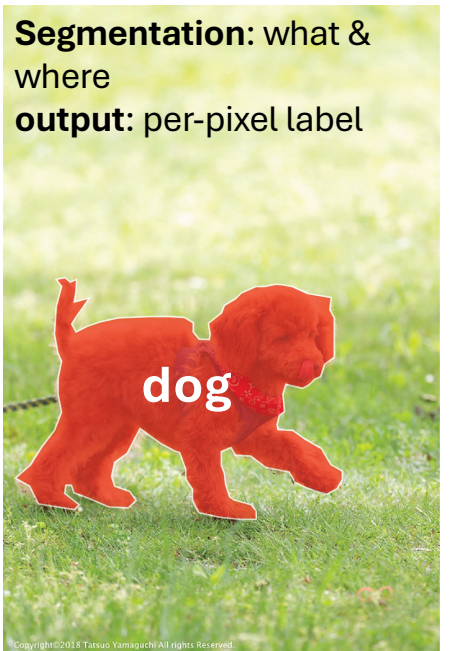
Visual Recognition Tasks

Visual Recognition Tasks



Visual Recognition Tasks

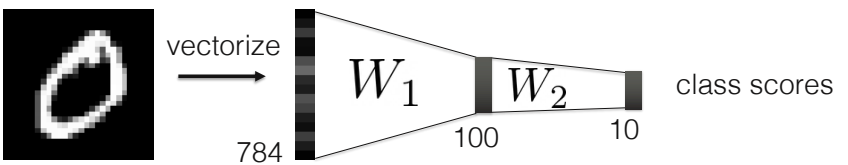




Outline

- motivation / overview of recognition tasks
- CNNs
  - Classification
  - Segmentation
  - Object Detection

Drawbacks of fully-connected networks



- spatial structure is destroyed
- fully-connected weights do not scale

Convolutional Neural Networks

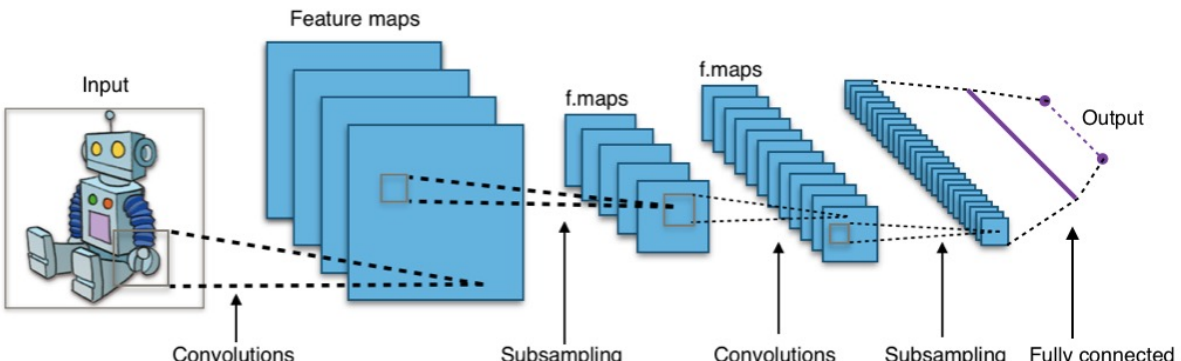


Image: CC Aphex34

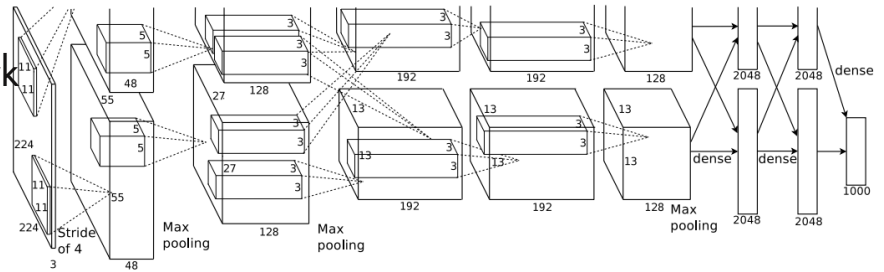
- Exploit spatial structure
- Scale to large inputs with fewer parameters
- Remarkable performance for processing visual data

AlexNet & surge in popularity

2010: ImageNet Large Scale Visual Recognition Challenge

- 14 million labeled images

First convolutional network for image classification



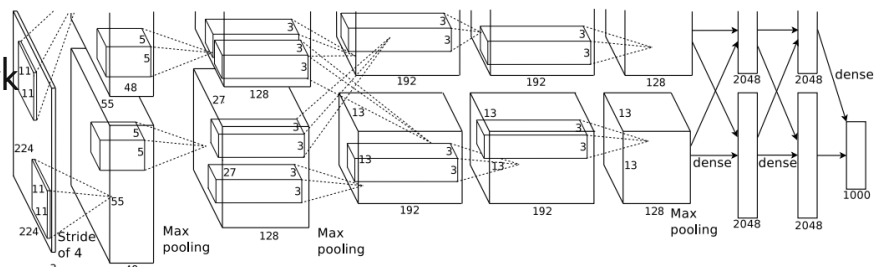
AlexNet [Krizhevsky '12]

AlexNet & surge in popularity

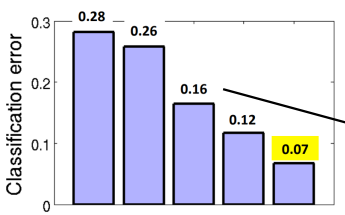
2010: ImageNet Large Scale Visual Recognition Challenge

- 14 million labeled images

First convolutional network for image classification



AlexNet [Krizhevsky '12]



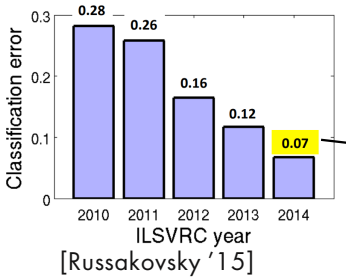
[Russakovsky '15]

CNNs

AlexNet & surge in popularity

2010: ImageNet Large Scale Visual Recognition Challenge

- 14 million labeled images



[Russakovsky '15]

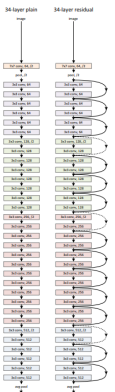
Deep networks

VGG

[Simonyan '14]

GoogLeNet

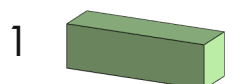
[Szegedy '14]



ResNet

[He '15]

Fully-Connected Layer

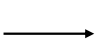


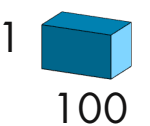
Vectorized Image

Wx

weights 100 x 3072

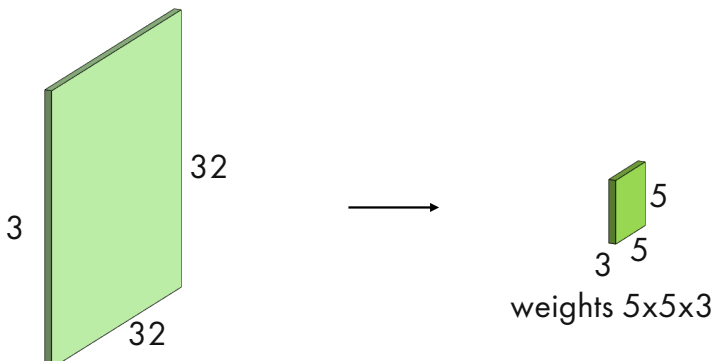
Weight Matrix





Output Activation

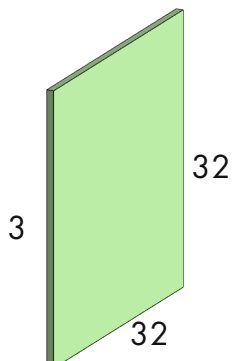
Convolutional Layer

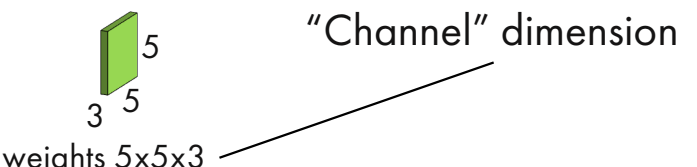


Input Image

Filter

Convolutional Layer

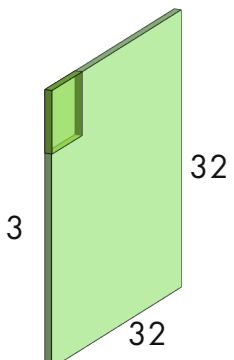




Input Image

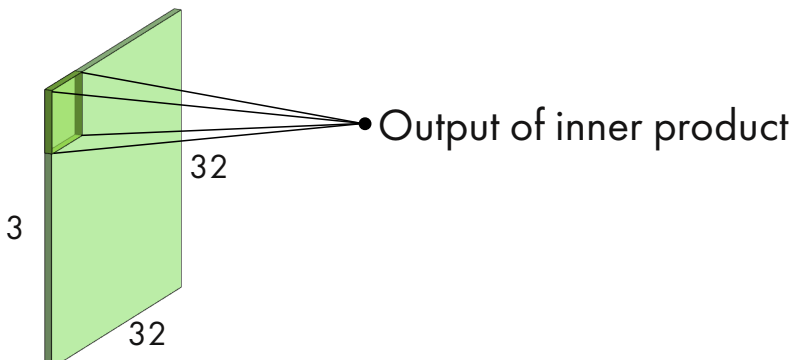
Filter

Convolutional Layer



Input Image

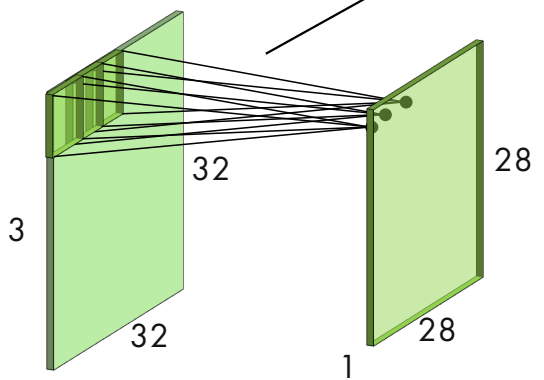
Convolutional Layer



Input Image

Convolutional Layer

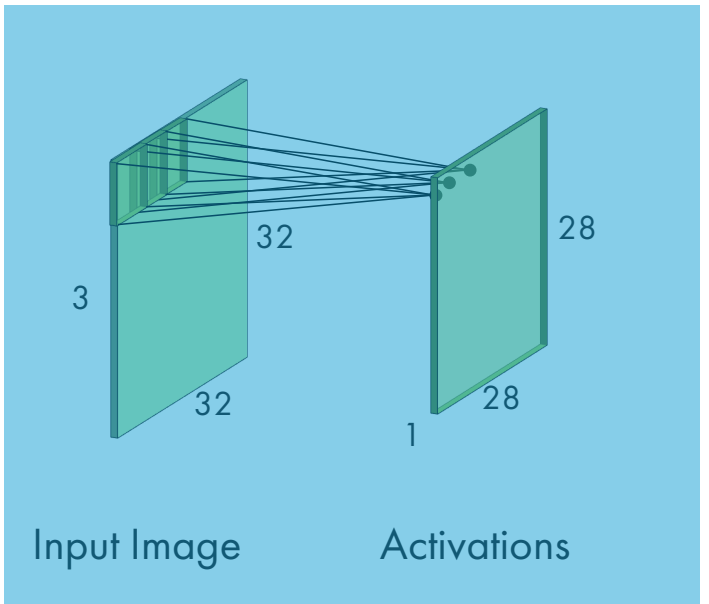
Convolution = sliding window + inner product

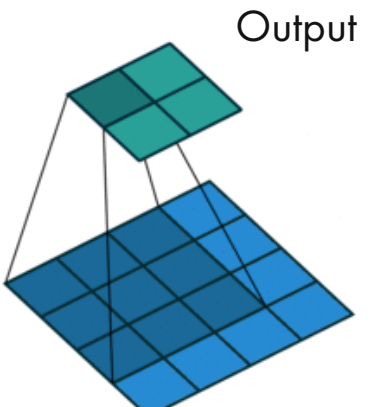


Input Image

Activations

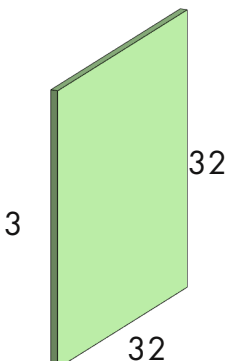
Convolutional Layer



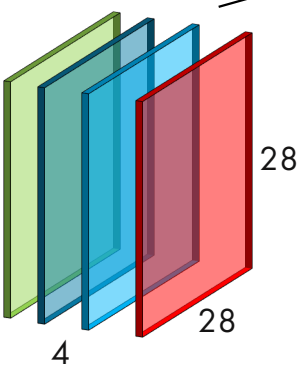


https://github.com/vdumoulin/conv_arithmetic

Convolutional Layer



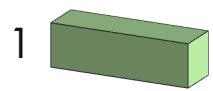
Input Image



Activations

Multiple output channels
using multiple filters

Fully-Connected Layer

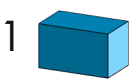


Vectorized Image

$$Wx$$

weights 100 x 3072

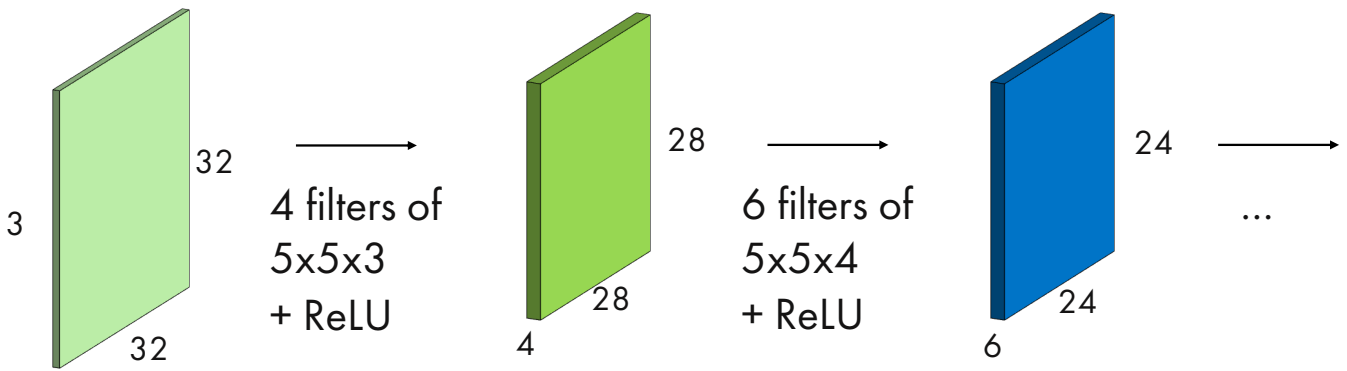
Weight Matrix



Output Activation

Special case of convolutional layer when filter size = input size!

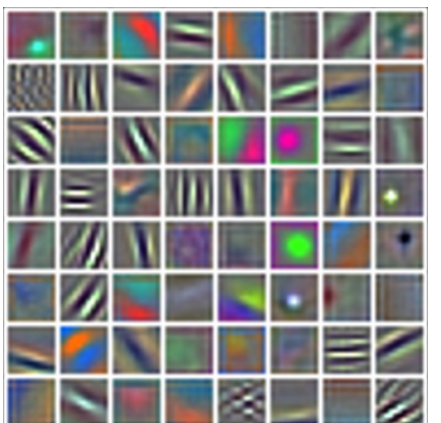
Convolutional Neural Network



Case Study: AlexNet

Input Image

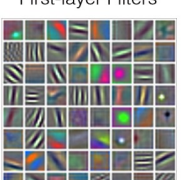
First-layer Filters

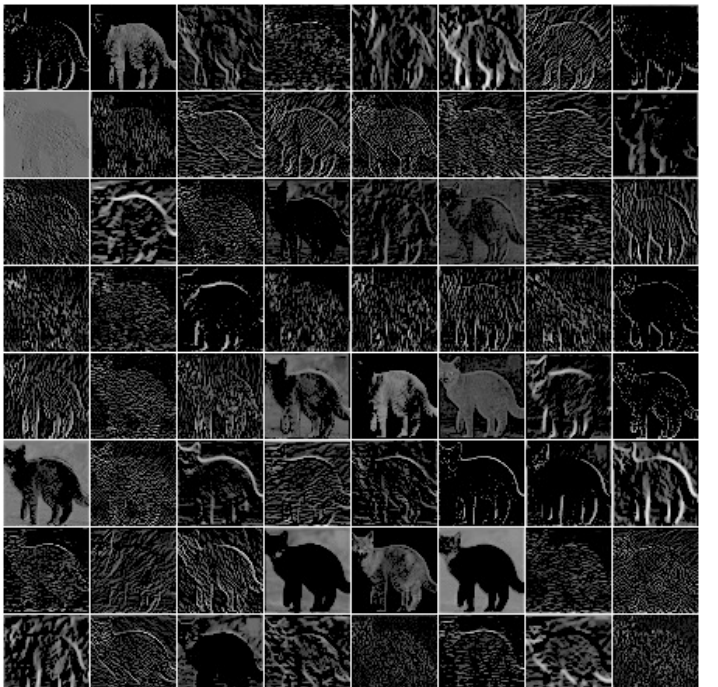


Case Study: AlexNet

Input Image

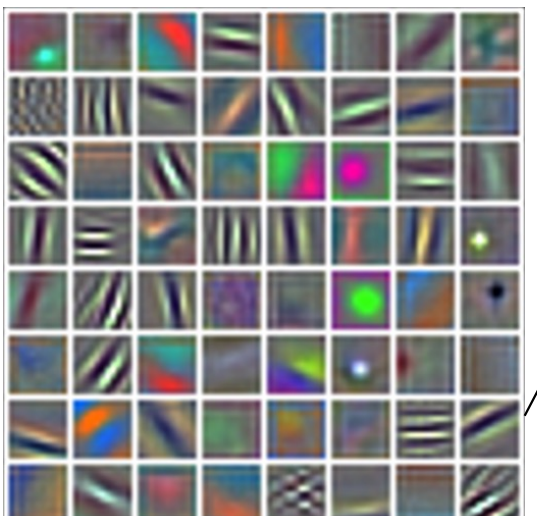
First-layer Filters





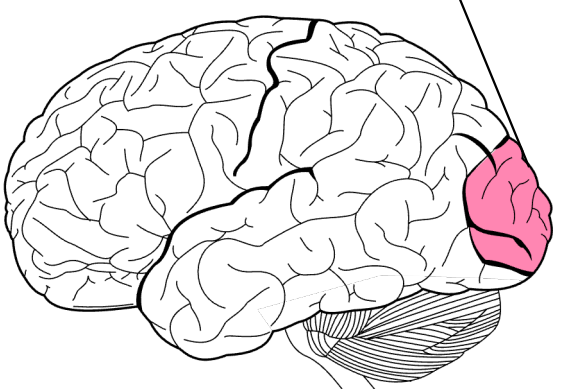
Activations

Case Study: AlexNet

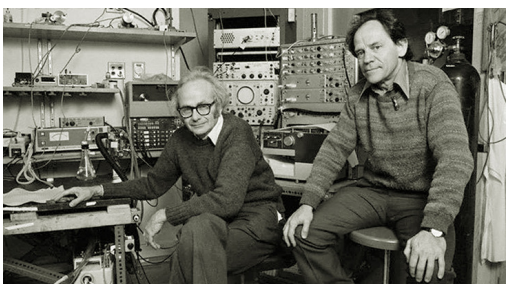


First-layer Filters

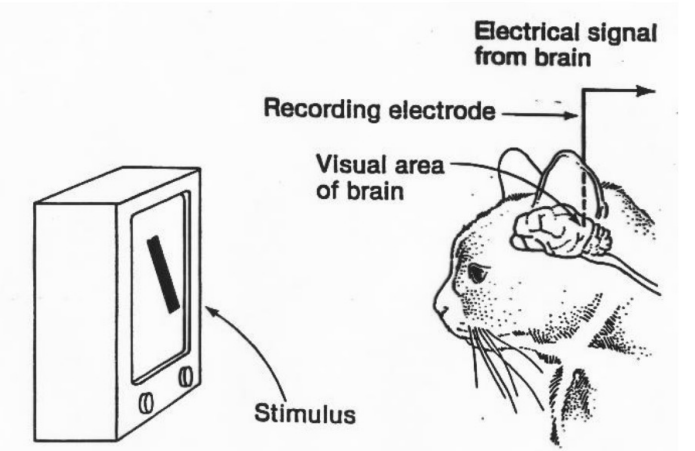
Similar to simple cells in visual cortex!
- Edge detectors



Case Study: AlexNet



[Hubel & Wiesel 1959]



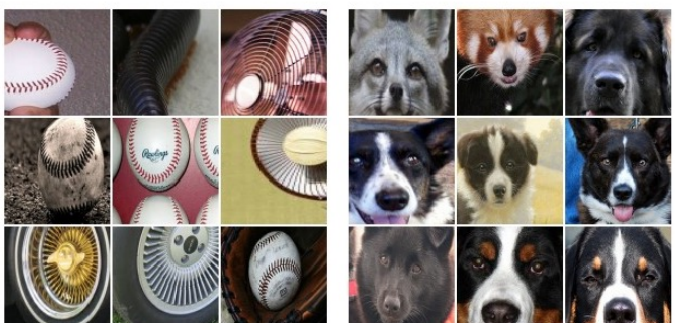
Simple cells in visual cortex detect edges, complex cells compose earlier responses

CNN higher layer filters

Dataset examples that maximize neuron outputs

[Olah ‘17]

CNN higher layer filters



Dataset examples that maximize neuron outputs

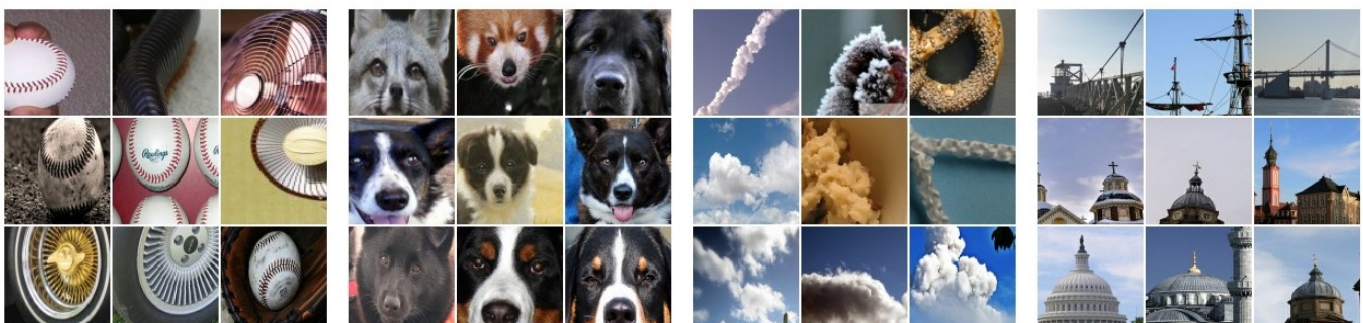
[Olah ‘17]

CNN higher layer filters

Dataset examples that maximize neuron outputs

[Olah ‘17]

CNN higher layer filters



Dataset examples that maximize neuron outputs

[Olah ‘17]

CNN Building Blocks

Design choices:

- filter size
- number of filters
- padding
- stride

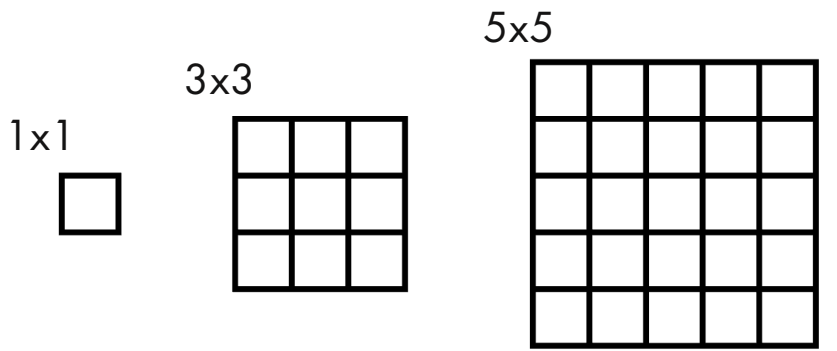
Layer types:

- pooling
- transpose convolutions
- upsampling layers*
- batch normalization*
- softmax layers*

*no time to cover all of these layers! Check out PyTorch docs for details...
e.g., https://pytorch.org/docs/stable/generated/torch.nn.BatchNorm2d.html

CNN Building Blocks

Filter size



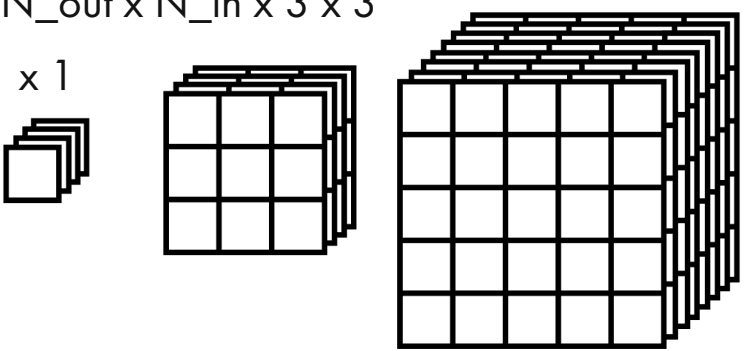
CNN Building Blocks

Number of channels

N_out x N_in x 1 x 1

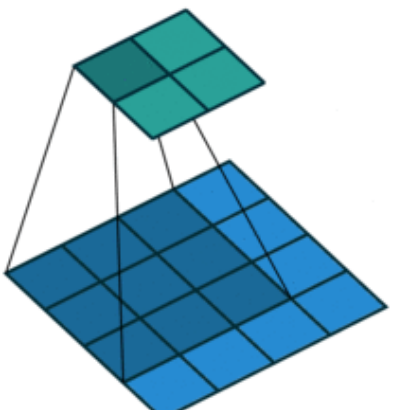
N_out x N_in x 3 x 3

N_out x N_in x 5 x 5

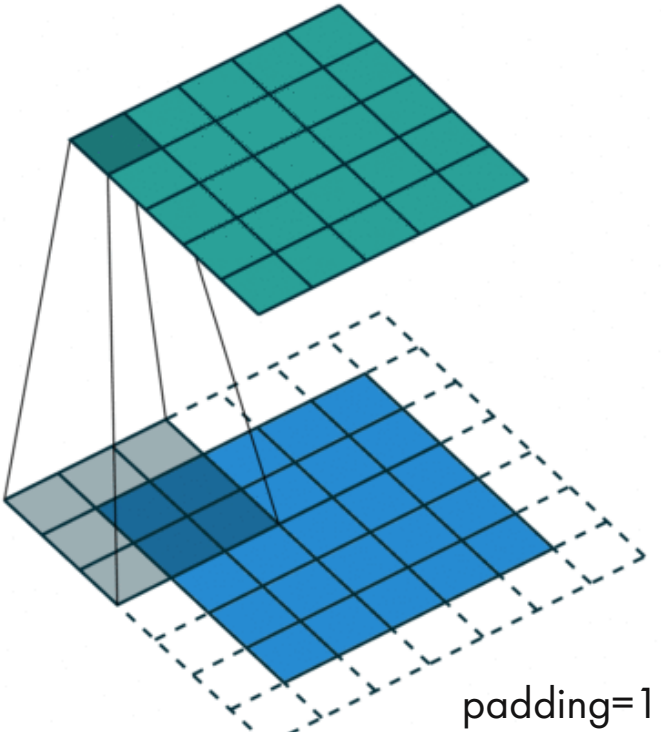


CNN Building Blocks

padding



No padding

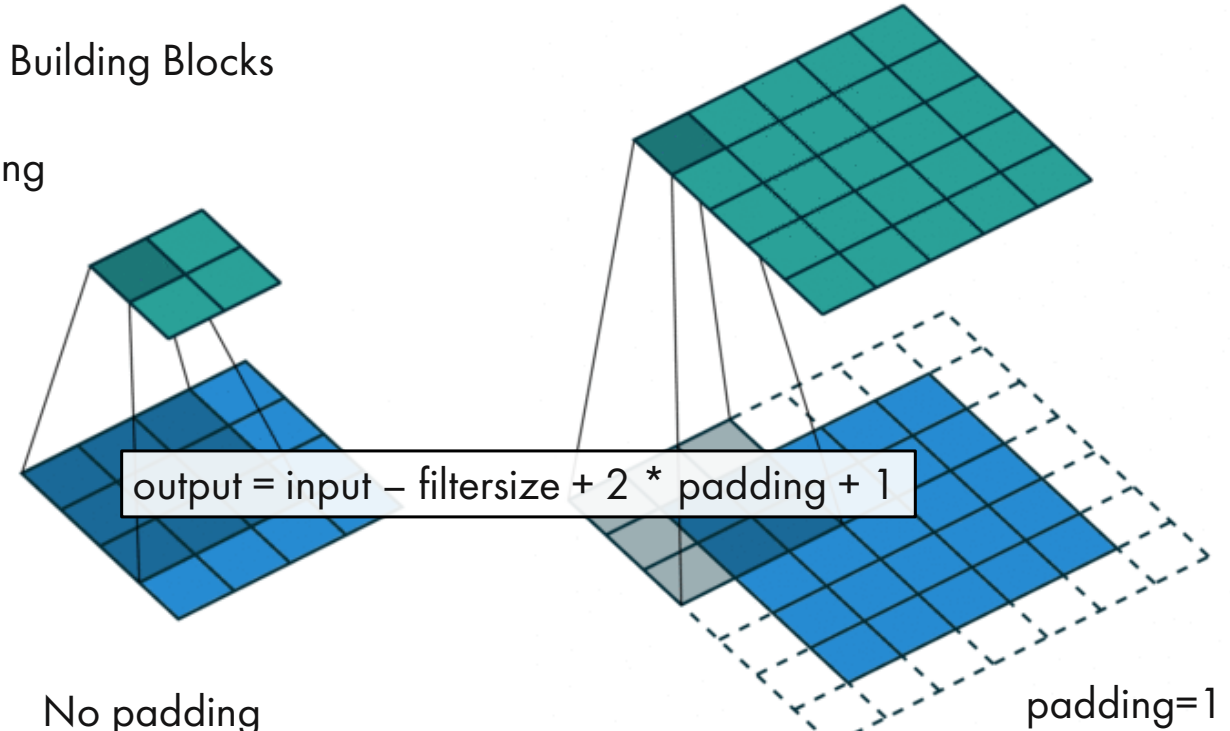


padding=1

https://github.com/vdumoulin/conv_arithmetic

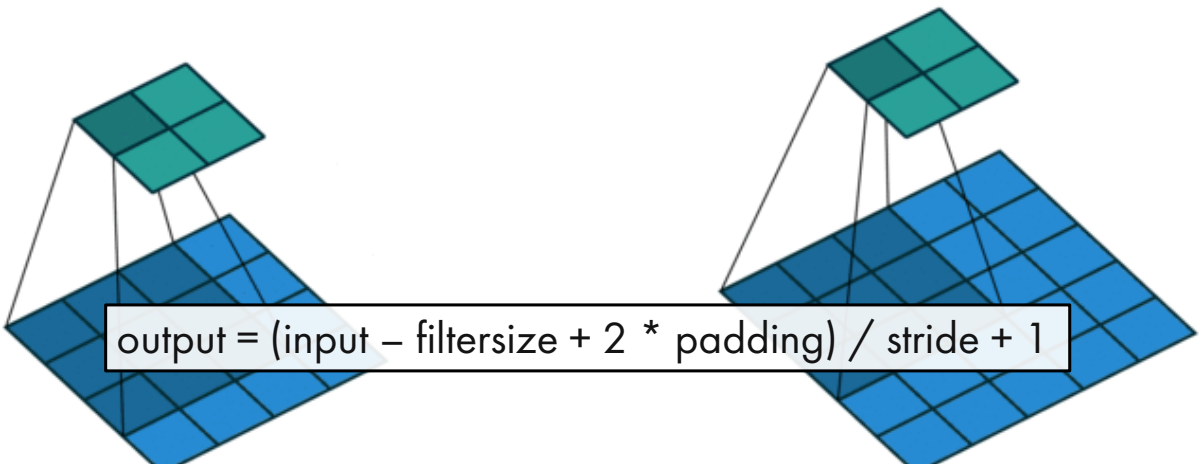
CNN Building Blocks

padding



CNN Building Blocks

stride

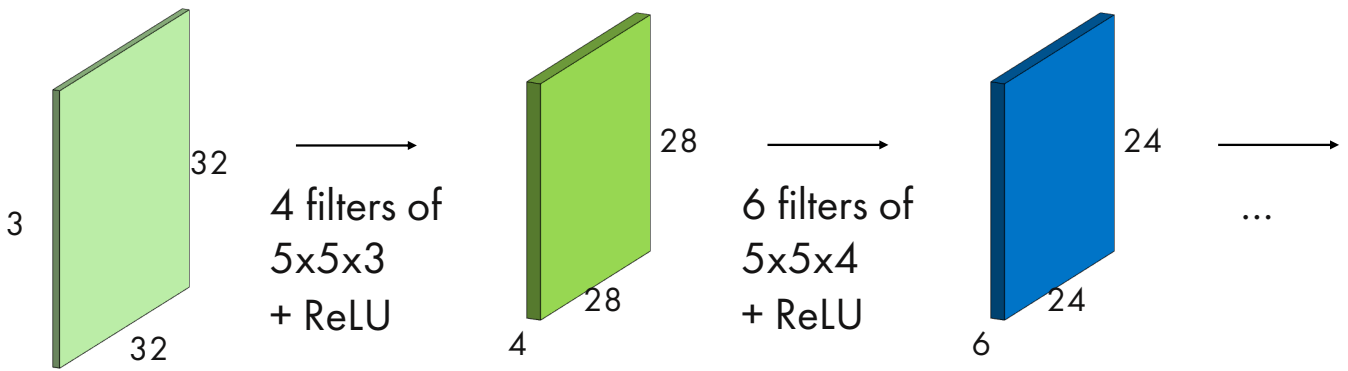


stride = 1

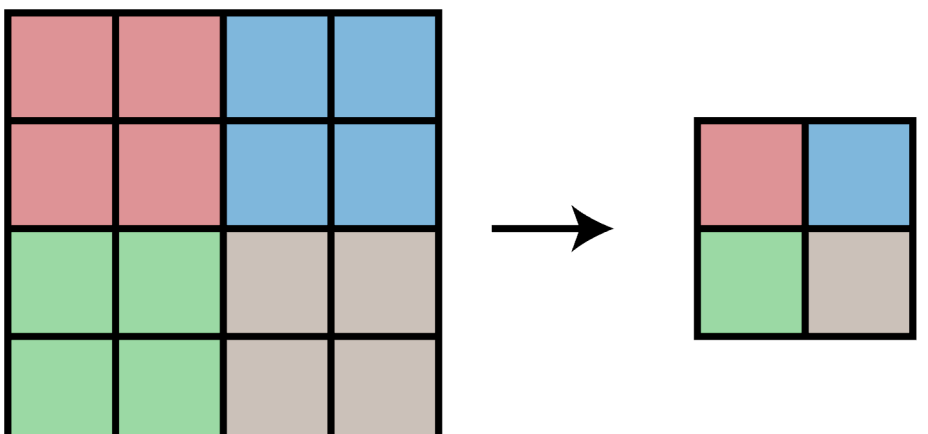
stride = 2

https://github.com/vdumoulin/conv_arithmetic

Convolutional Neural Network



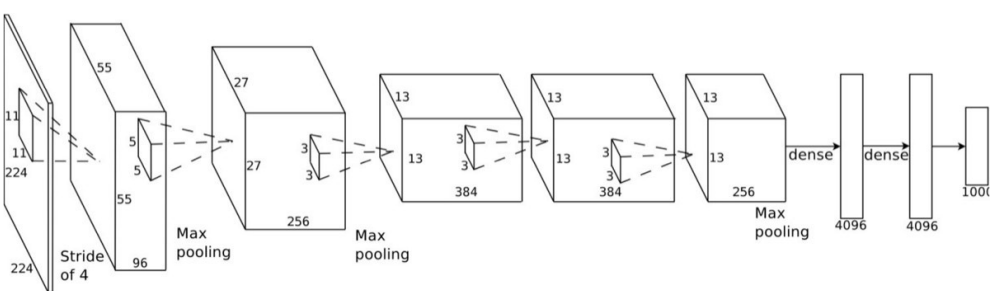
Layer types: Pooling



e.g., max pool size=2, stride=2

Convolutional Neural Networks (CNN)

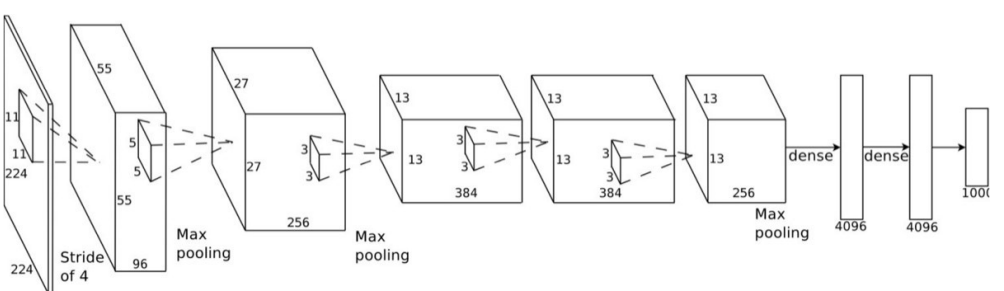
- AlexNet (from UofT!): A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, NeurIPS 2012. This network won the Imagenet Challenge of 2012, and revolutionized computer vision.



[Pic adopted from: A. Krizhevsky]

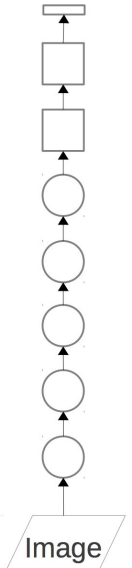
Convolutional Neural Networks (CNN)

- AlexNet (from UofT!): A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, NeurIPS 2012. This network won the Imagenet Challenge of 2012, and revolutionized computer vision.
- How many parameters (weights) does this network have?



[Pic adopted from: A. Krizhevsky]

Convolutional Neural Networks (CNN)



- Trained with stochastic gradient descent on two NVIDIA GPUs for about a week
- 650,000 neurons
- 60,000,000 parameters
- 630,000,000 connections
- **Final feature layer:** 4096-dimensional

Convolutional layer: convolves its input with a bank of 3D filters, then applies point-wise non-linearity

Fully-connected layer: applies linear filters to its input, then applies point-wise non-linearity

Figure: From http://www.image-net.org/challenges/LSVRC/2012/supervision.pdf

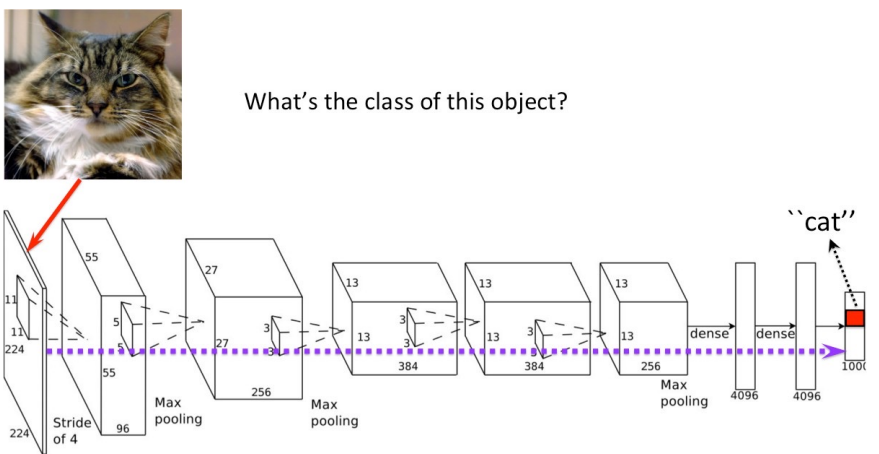
[Pic adopted from: A. Krizhevsky]

Outline

- motivation / overview of recognition tasks
- CNNs
  - Classification
  - Segmentation
  - Object Detection

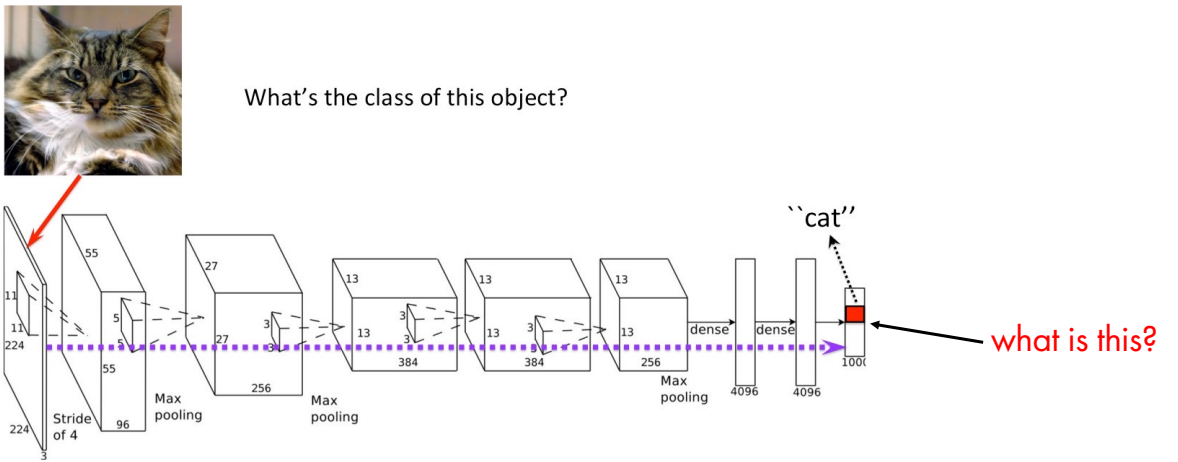
Classification

- Once trained we can do classification. Just feed in an image or a crop of the image, run through the network, and read out the class with the highest probability in the last (classification) layer.



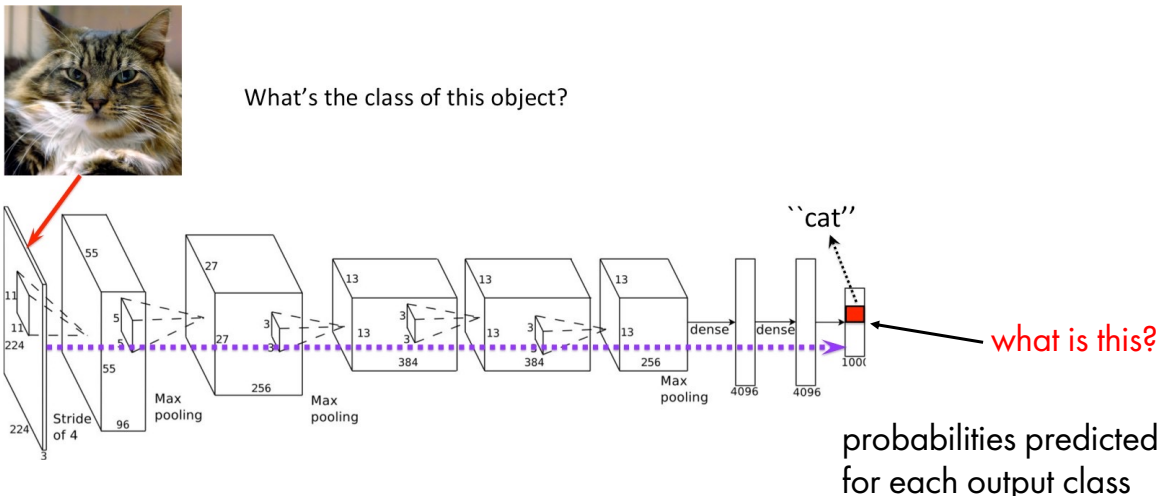
Classification

- Once trained we can do classification. Just feed in an image or a crop of the image, run through the network, and read out the class with the highest probability in the last (classification) layer.



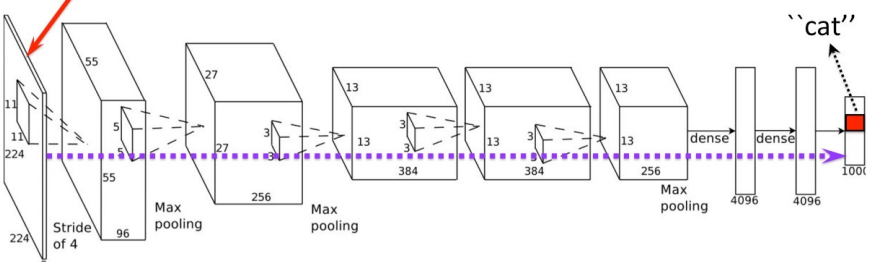
Classification

- Once trained we can do classification. Just feed in an image or a crop of the image, run through the network, and read out the class with the highest probability in the last (classification) layer.



Classification

What's the class of this object?

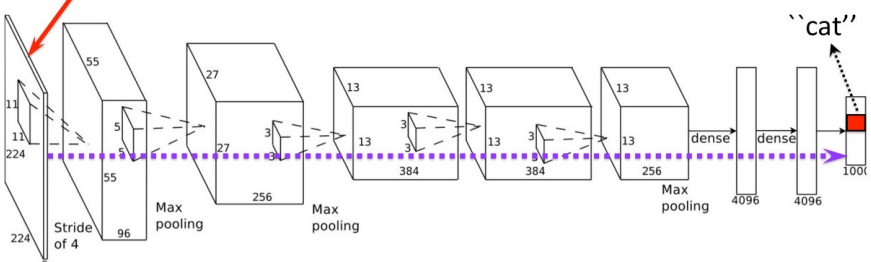


loss function

$$L_i = -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right)$$

Classification

What's the class of this object?



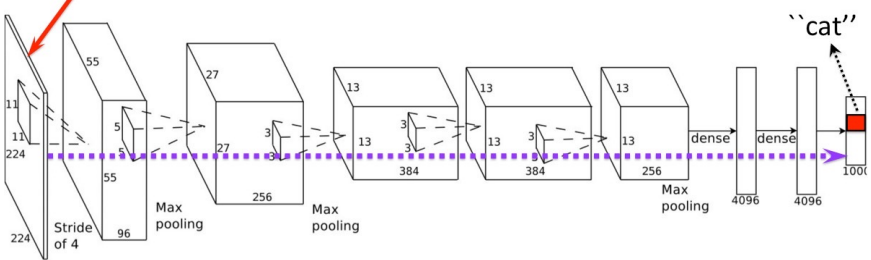
loss function

$$L_i = -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right)$$

softmax function (squashes the values to be probability distribution)

Classification

What's the class of this object?



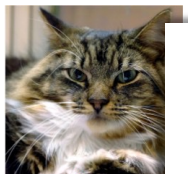
loss function

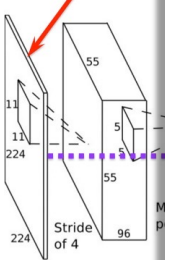
output for true class "slot"

$$L_i = -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right)$$

softmax function (squashes the values to be probability distribution)

Classification





loss

Output layer

$$\begin{bmatrix} 1.3 \\ 5.1 \\ 2.2 \\ 0.7 \\ 1.1 \end{bmatrix}$$

Softmax activation function

$$\frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Probabilities

$$\begin{bmatrix} 0.02 \\ 0.90 \\ 0.05 \\ 0.01 \\ 0.02 \end{bmatrix}$$

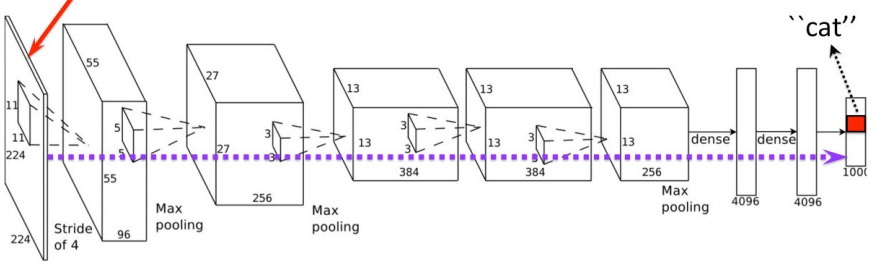
$$L_i = -l$$

$$\left(\sum_j e^{I_j}\right)$$

softmax function (squashes the values to be probability distribution)

Classification

What's the class of this object?



loss function

$$L_i = -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right)$$

output for true class "slot"

output for other classes

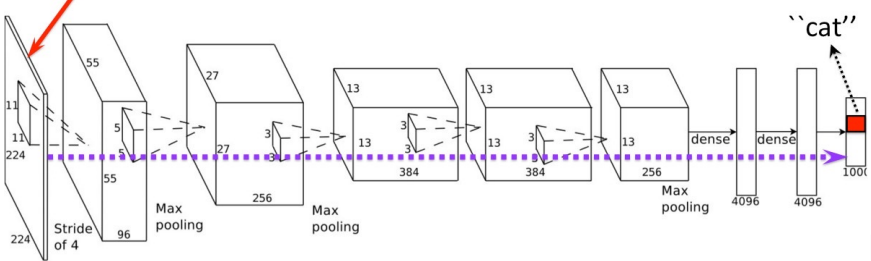
softmax function (squashes the values to be probability distribution)

cross entropy!

$$H(p,q) = -\sum_x p(x)\log q(x)$$

Classification

What's the class of this object?



what is p(x)?

loss function

$$L_i = -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right)$$

output for true class "slot"

output for other classes

softmax function (squashes the values to be probability distribution)

cross entropy!

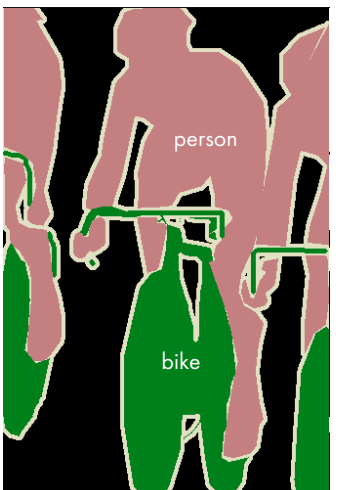
$$H(p,q) = -\sum_x p(x)\log q(x)$$

Outline

- motivation / overview of recognition tasks
- CNNs
  - Classification
  - Segmentation
  - Object Detection

Semantic Segmentation: Problem Definition

- **Task**: Label every pixel
  - Don't differentiate instances

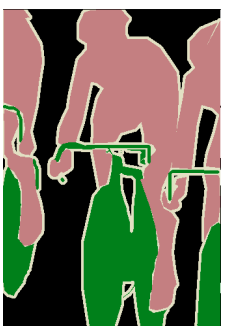


Semantic Segmentation: **Evaluation**

- Intersection over Union (IoU), averaged over classes

$$\text{IoU} = \frac{\text{intersection}}{\text{union}}$$

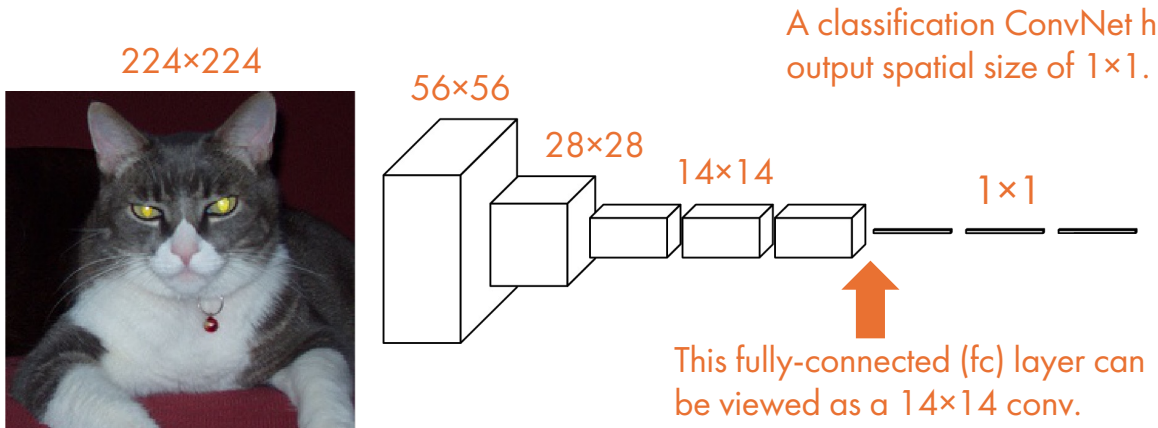
prediction



ground truth

Fully Convolutional Networks (FCN)

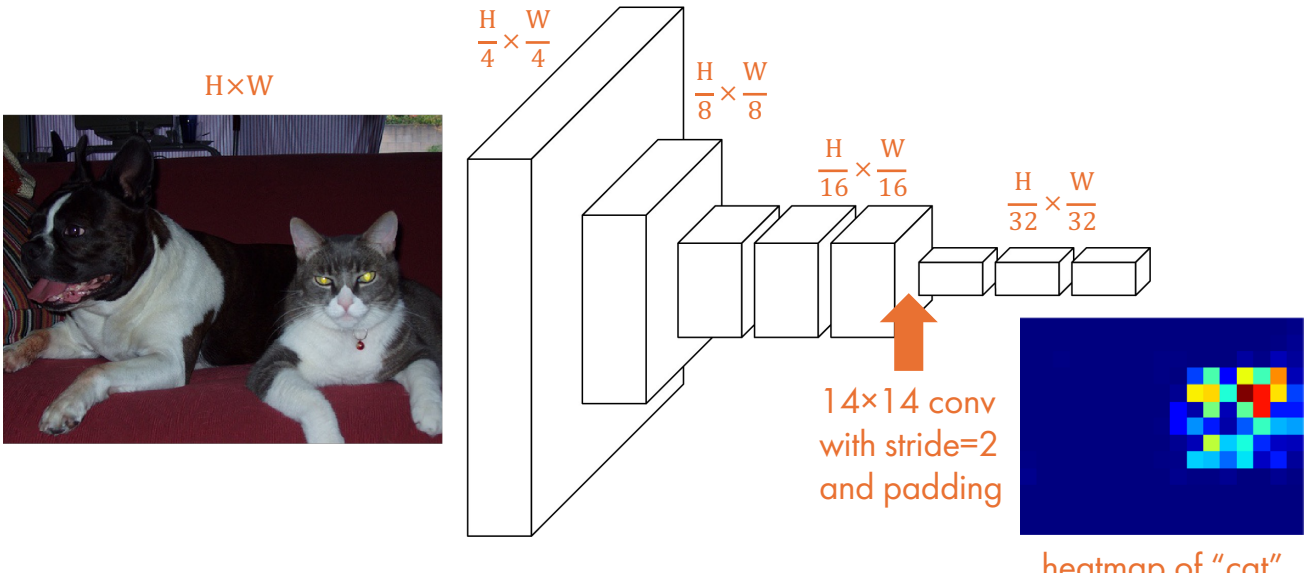
- Repurpose a classification ConvNet for semantic segmentation



Jonathan Long, Evan Shelhamer, Trevor Darrell, "Fully Convolutional Networks for Semantic Segmentation", CVPR 2015

Fully Convolutional Networks (FCN)

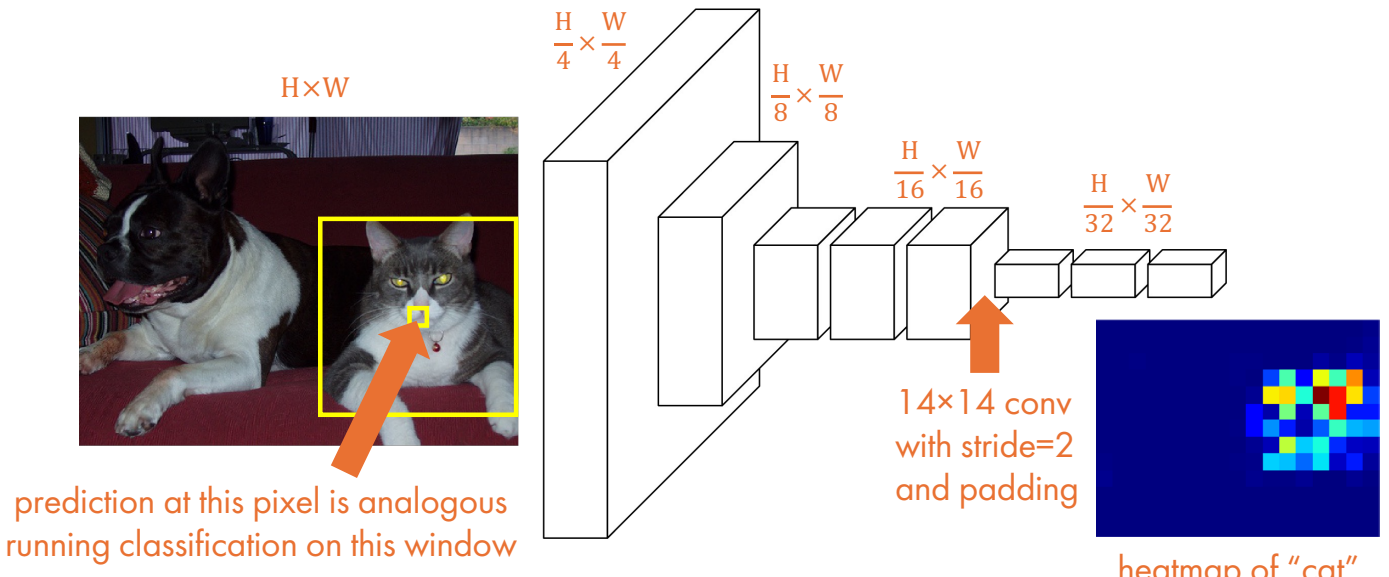
- Repurpose a classification ConvNet for semantic segmentation



Jonathan Long, Evan Shelhamer, Trevor Darrell, "Fully Convolutional Networks for Semantic Segmentation", CVPR 2015

Fully Convolutional Networks (FCN)

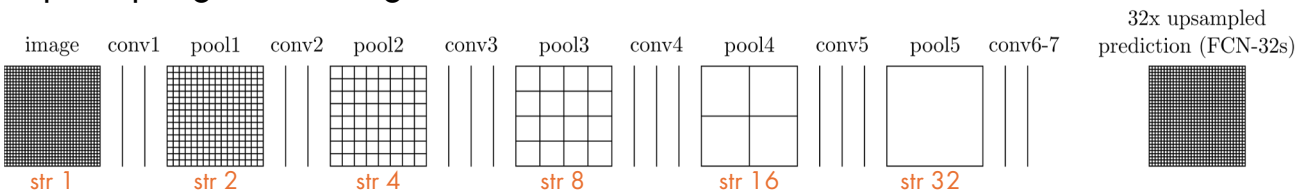
- Repurpose a classification ConvNet for semantic segmentation



Jonathan Long, Evan Shelhamer, Trevor Darrell, "Fully Convolutional Networks for Semantic Segmentation", CVPR 2015

Fully Convolutional Networks (FCN)

- Upsampling: combining “where” and “what”

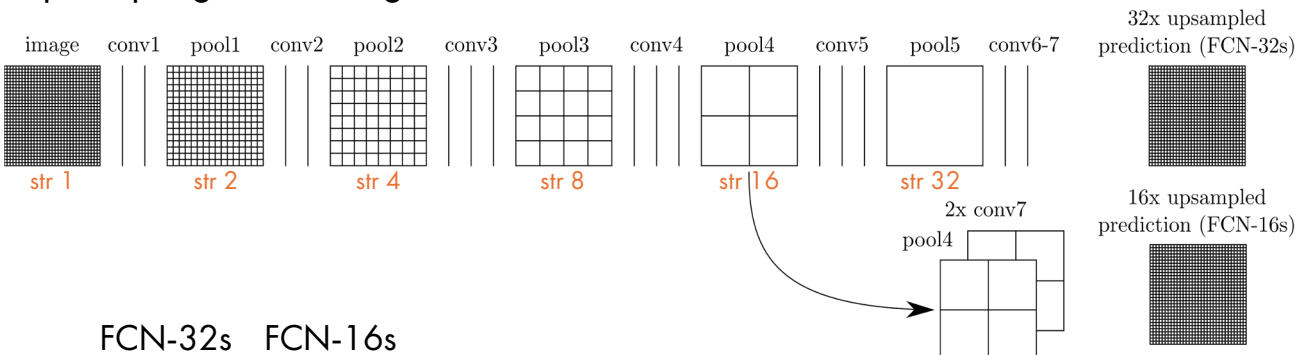


FCN-32s

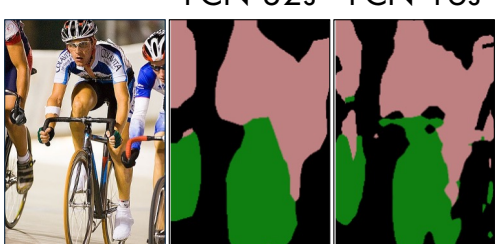
Jonathan Long, Evan Shelhamer, Trevor Darrell, “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015

Fully Convolutional Networks (FCN)

- Upsampling: combining “where” and “what”



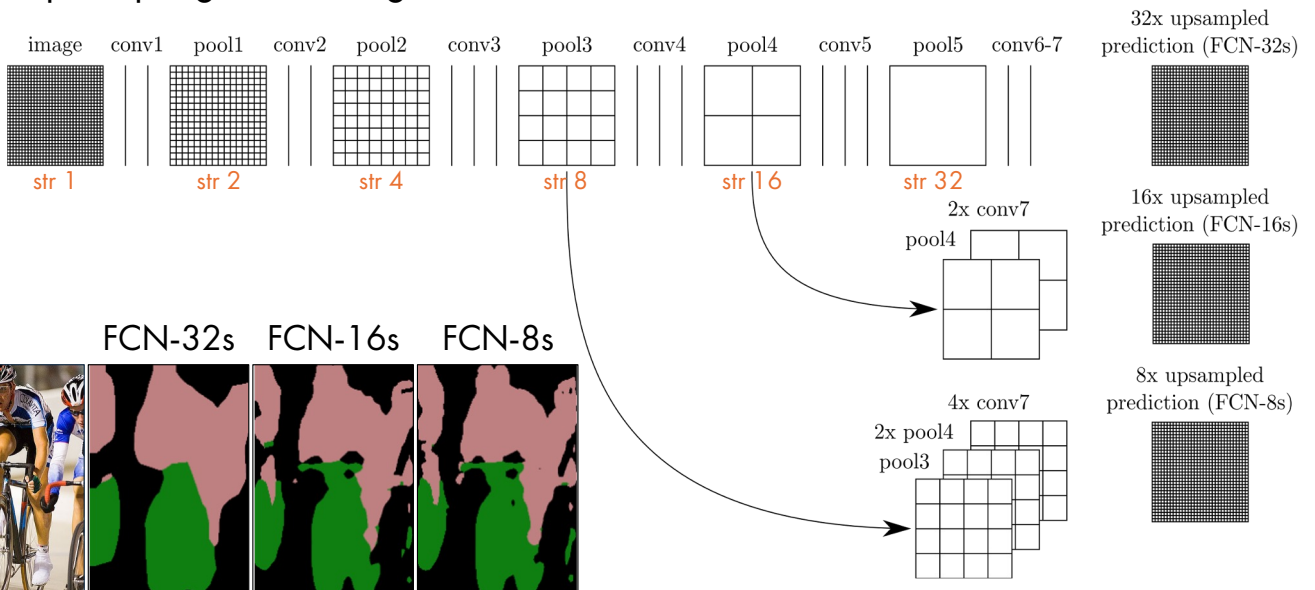
FCN-32s FCN-16s



Jonathan Long, Evan Shelhamer, Trevor Darrell, “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015

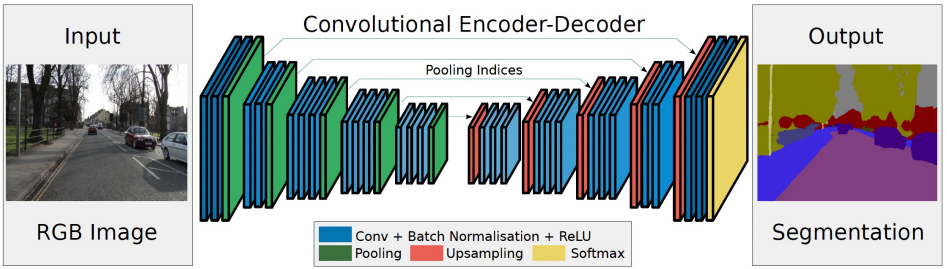
Fully Convolutional Networks (FCN)

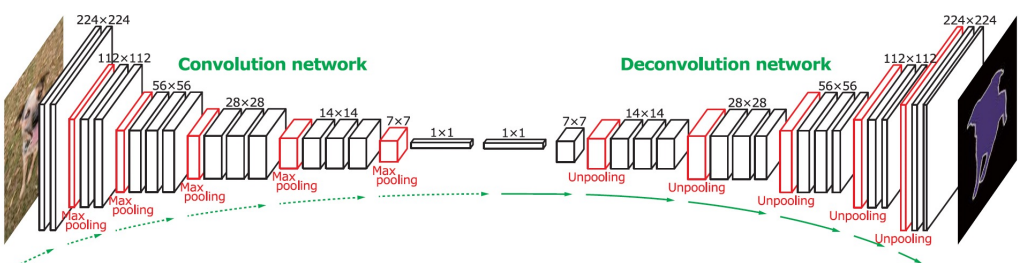
- Upsampling: combining “where” and “what”



Jonathan Long, Evan Shelhamer, Trevor Darrell, “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015

Again, an autoencoder!





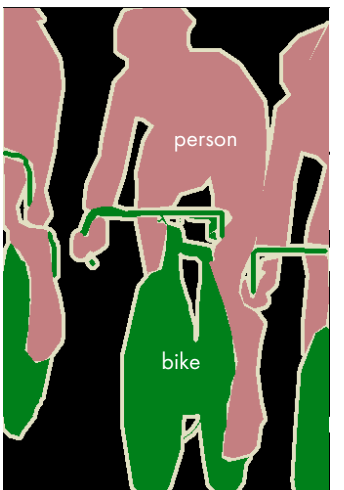
Hyeonwoo Noh, Seunghoon Hong, Bohyung Han, "Learning Deconvolution Network for Semantic Segmentation", ICCV 2015
Vijay Badrinarayanan, Alex Kendall, Roberto Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation", arXiv 2015, PAMI 2017

Semantic Segmentation

- Predict per-pixel class labels given an image
- Network structure: fully convolutional network (a CNN autoencoder)
- Repurpose a pre-trained image classification network for semantic segmentation
- Limitation: cannot differentiate instances

Semantic Segmentation Don't Differentiate Instances

- **Task**: Label every pixel
  - Don't differentiate instances

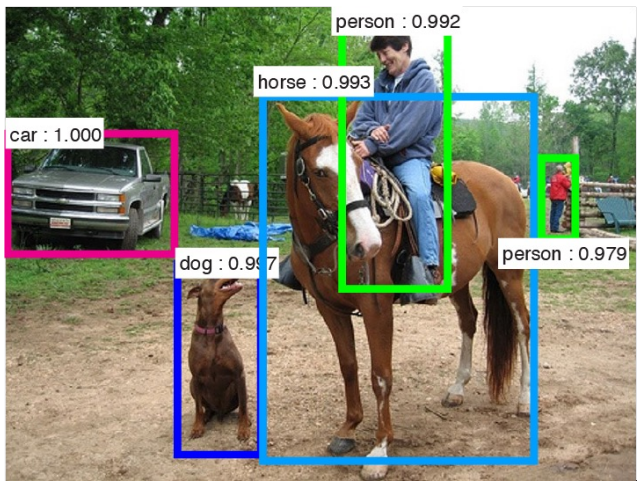


Outline

- motivation / overview of recognition tasks
- CNNs
  - Classification
  - Segmentation
  - Object Detection

Object Detection: Problem Definition

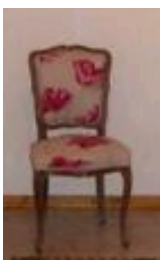
- **Task**: identify, locate, and recognize objects
- **Identify**:
  - Are there objects presented?
- **Locate**:
  - Where is each object?
  - (x, y, h, w)
- **Recognize**:
  - What is each object?
  - class label



Object Recognition

- Object recognition is the task of **identifying** and **localizing** objects within images or video.

This is a chair

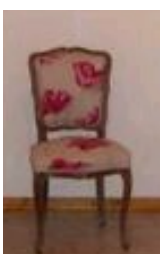


Find the chair in this image

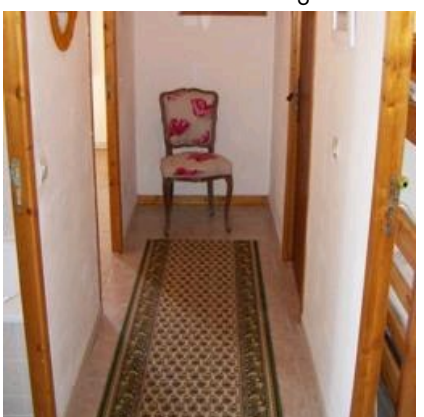
Object Recognition

- Object recognition is the task of **identifying** and **localizing** objects within images or video.

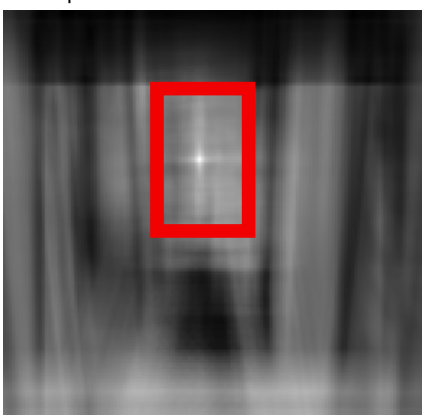
This is a chair



Find the chair in this image



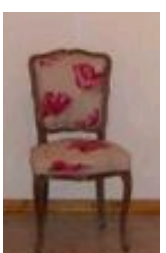
Output of normalized correlation



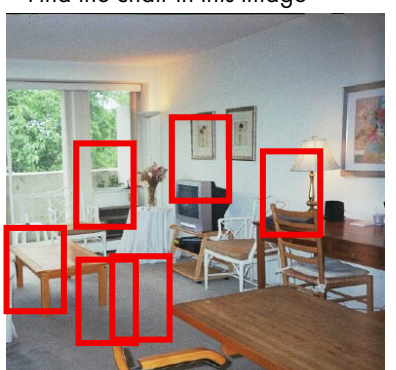
Object Recognition

- Object recognition is the task of **identifying** and **localizing** objects within images or video.

This is a chair

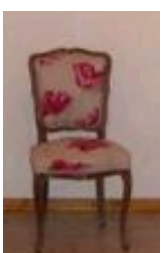


Find the chair in this image



Object Recognition

This is a chair

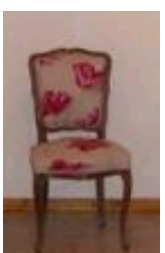


Find the chair in this image

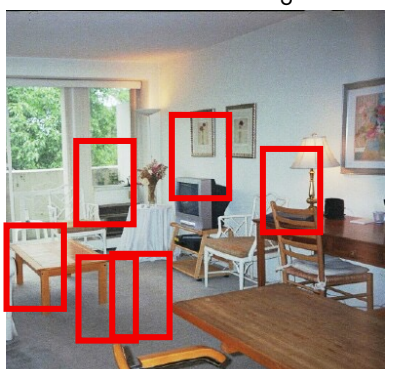
A "popular method is that of template matching, by point to point correlation of a model pattern with the image pattern. These techniques are inadequate for three-dimensional scene analysis for many reasons, such as occlusion, changes in viewing angle, and articulation of parts." Nivatia & Binford, 1977.

Object Recognition

This is a chair



Find the chair in this image



A "popular method is that of template matching, by point to point correlation of a model pattern with the image pattern. These techniques are inadequate for three-dimensional scene analysis for many reasons, such as occlusion, changes in viewing angle, and articulation of parts." Nivatia & Binford, 1977.

Object Detection: Evaluation

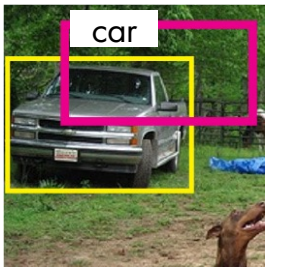
correct
(true positive)

Object Detection: Evaluation

correct
(true positive)

missed
(false negative)

Object Detection: Evaluation

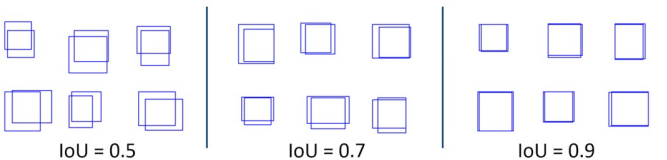


correct
(true positive)

missed
(false negative)

mislocalized
(false positive)

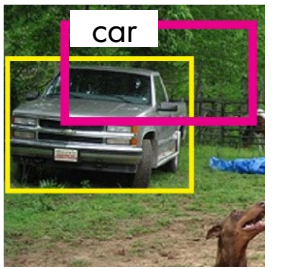
define by an IoU threshold



Object Detection: Evaluation

correct
(true positive)

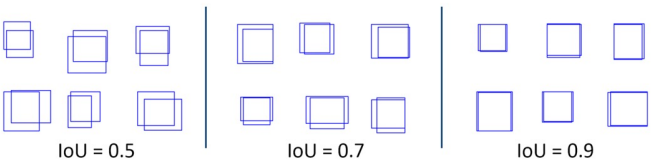
missed
(false negative)



mislocalized
(false positive)

misrecognized
(false positive)

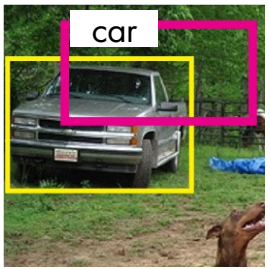
define by an IoU threshold



Object Detection: Evaluation

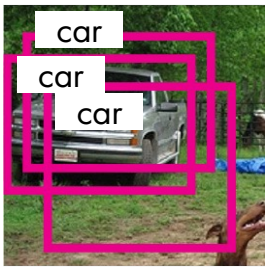
correct
(true positive)

missed
(false negative)



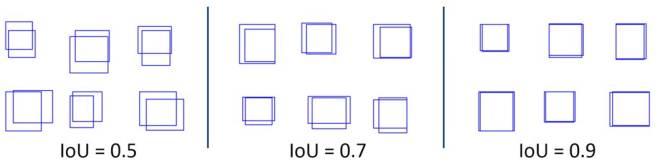
mislocalized
(false positive)

misrecognized
(false positive)



duplicated
(false positive)

define by an IoU threshold



Object Detection: Evaluation

- Precision: What % of detected objects are correct?

$$\text{Precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}}$$

- Recall: What % of objects are detected?

$$\text{Recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}}$$

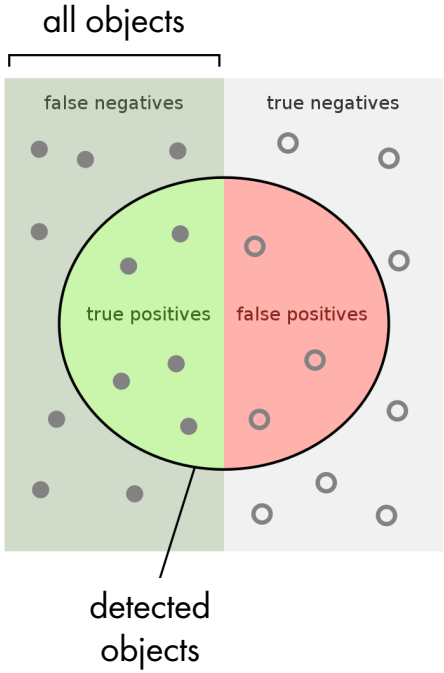
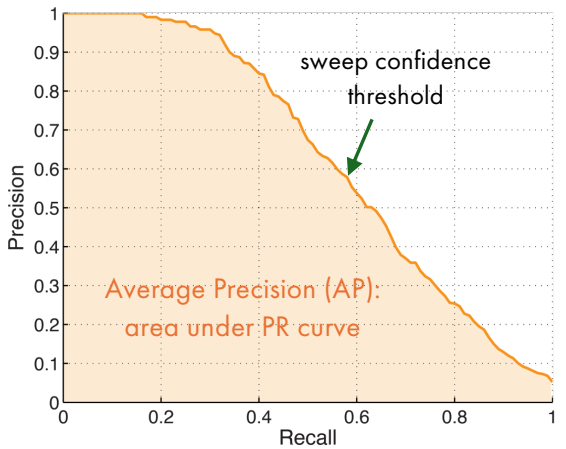


Figure adapted from: https://en.wikipedia.org/wiki/Precision_and_recall

Object Detection: Evaluation

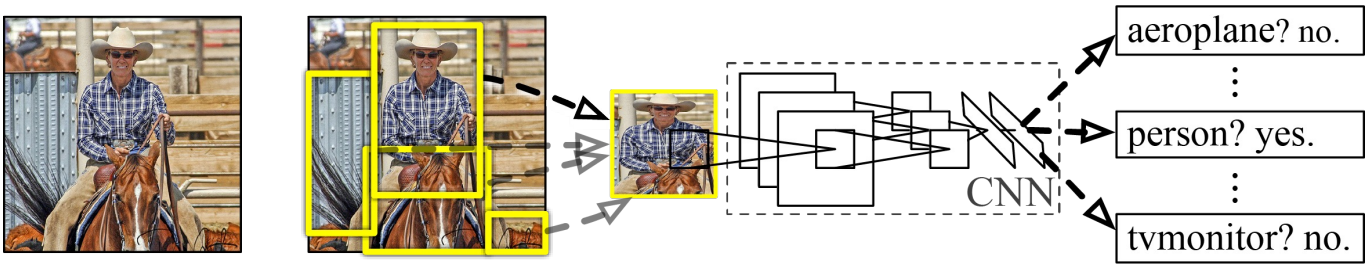
- Precision-Recall Tradeoff



threshold = 0.4: Recall ↑, Precision ↓

threshold = 0.7: Recall ↓, Precision ↑

Region-Based CNN (R-CNN)

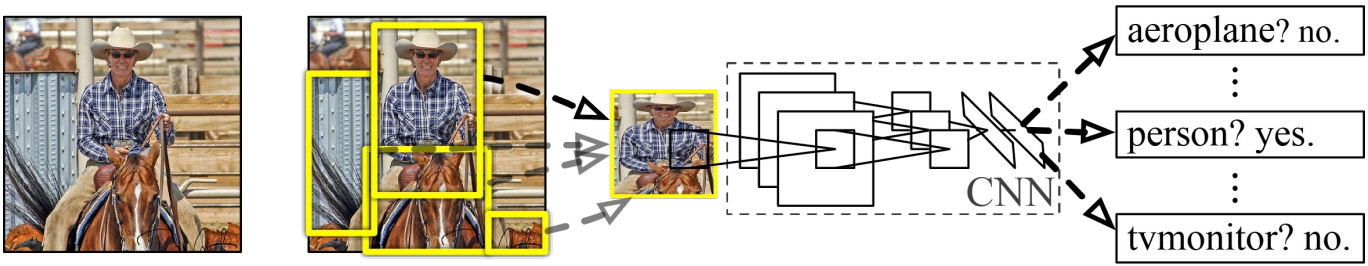


- THE deep learning revolution in the vision community.

the ILSVRC workshop at ECCV 2012.[1] The central issue of debate can be distilled as: **To what extent do the CNN *classification* results on ImageNet generalize to object *detection* results on the PASCAL VOC Challenge?** In this paper, we study object detection using features computed by a large convolutional neural network, answering this important scientific question.[2]

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

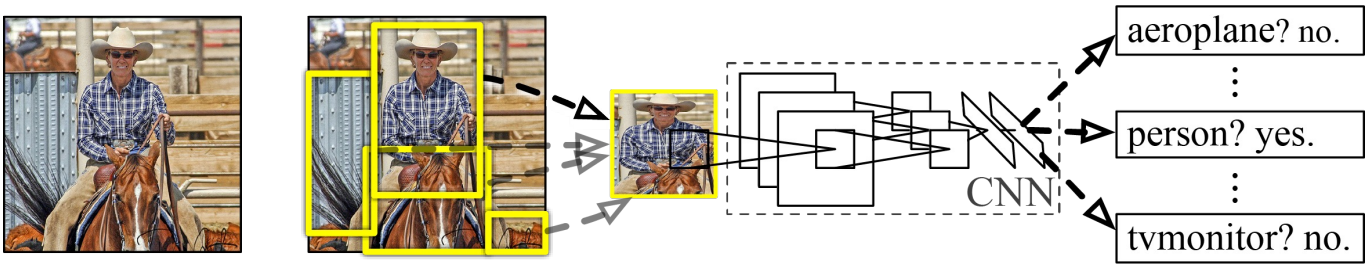
Region-Based CNN (R-CNN)



- THE deep learning revolution in the vision community.
- R-CNN: Regions with CNN features
- 40% relative improvement over classical methods
- Deep Learning made work and transferable beyond image classification

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

Region-Based CNN (R-CNN)



input

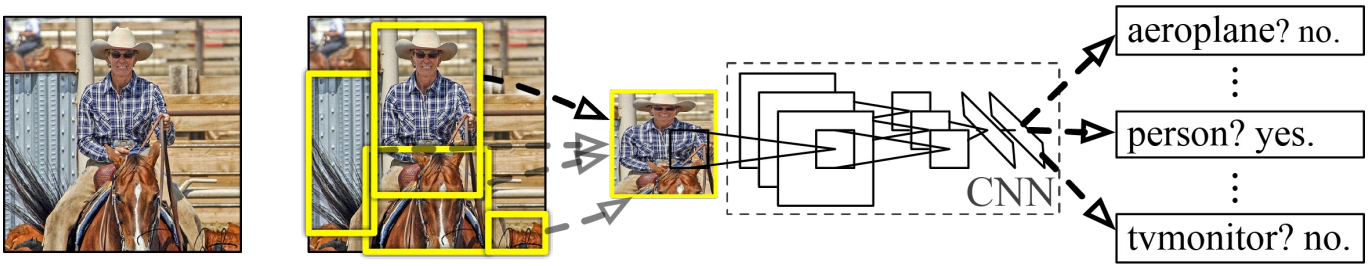
extract
"region proposals"

compute CNN features
for each region

classify regions
& refine boxes

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

Region-Based CNN (R-CNN)

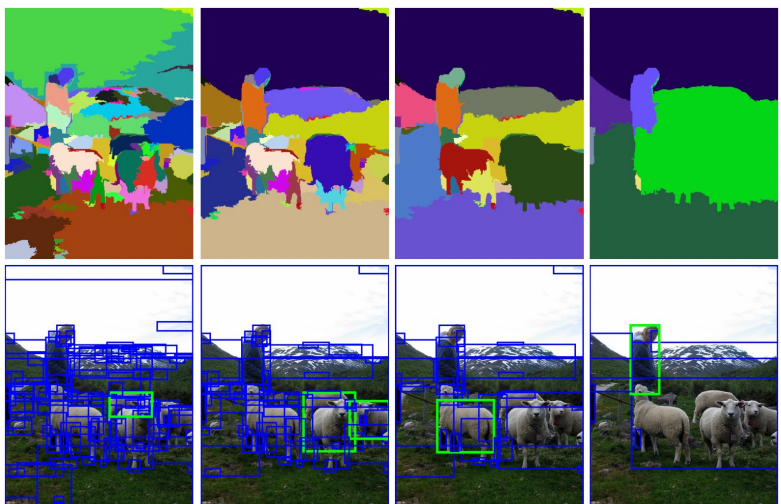


extract
"region proposals"

- Reduce the search space ("selective search")
- ~2000 regions / image
- Recall-driven
  - a proposal may not be an object
  - an object is highly likely in the proposals

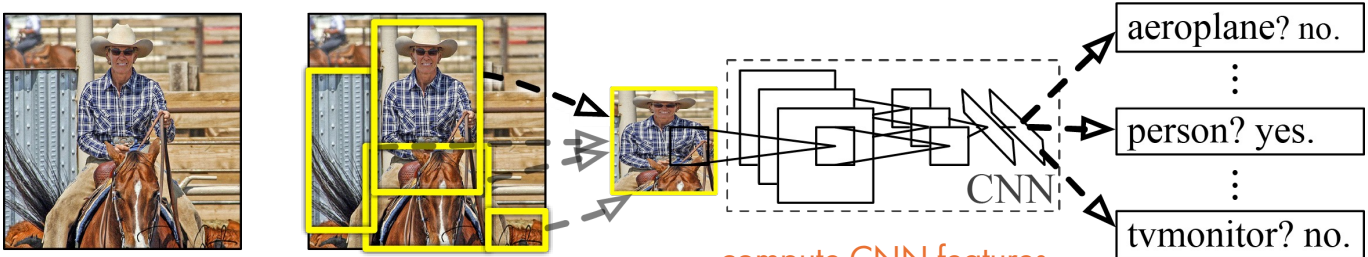
Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

e.g.: Selective Search for Region Proposal



J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders. “Selective search for object recognition”. IJCV 2013

Region-Based CNN (R-CNN)

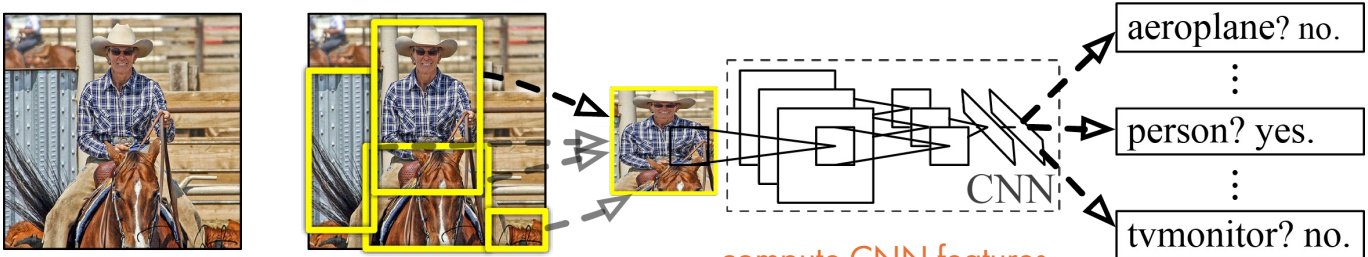


compute CNN features
for each region

one proposal

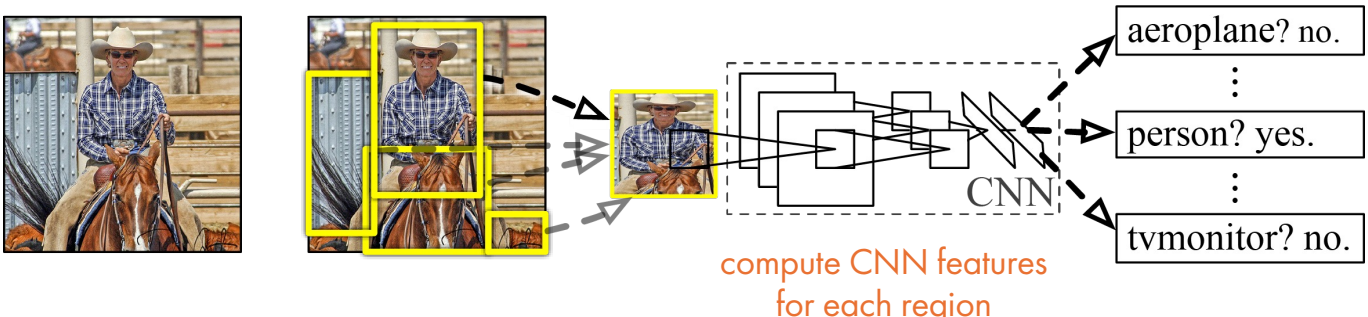
Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

Region-Based CNN (R-CNN)



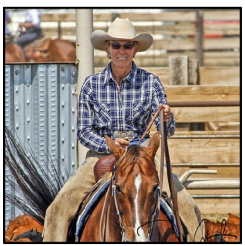
Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

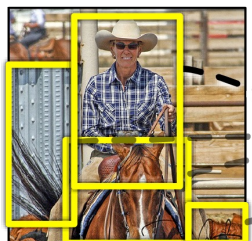
Region-Based CNN (R-CNN)

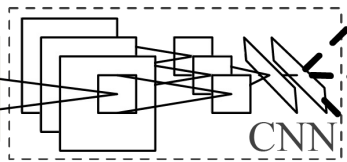


Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

Region-Based CNN (R-CNN)







aeroplane? no.

person? yes.

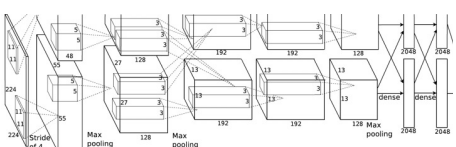
tvmonitor? no.

compute CNN features
for each region

one proposal

crop

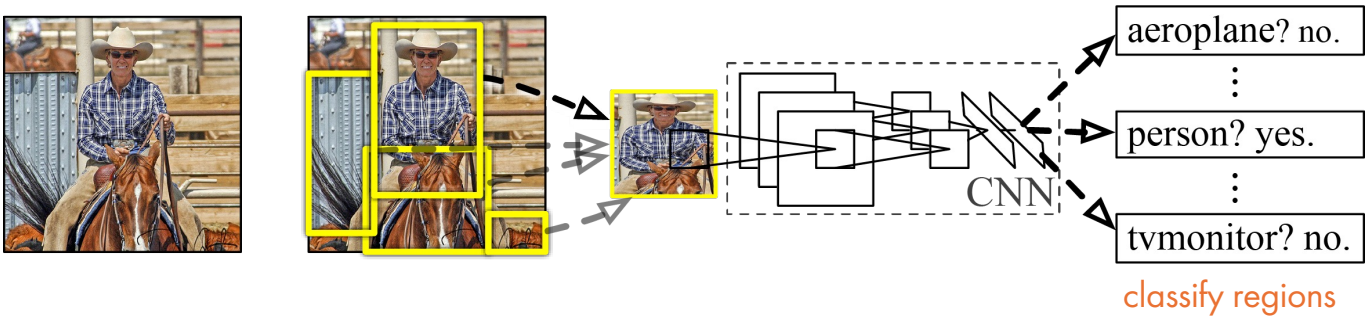
warp
(e.g., 224×224)



apply a classification ConvNet
(e.g., AlexNet)

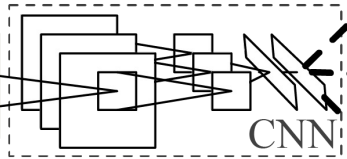
Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

Region-Based CNN (R-CNN)



Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

Region-Based CNN (R-CNN)

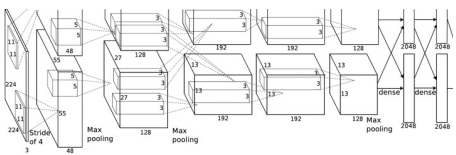


aeroplane? no.

person? yes.

tvmonitor? no.

classify regions



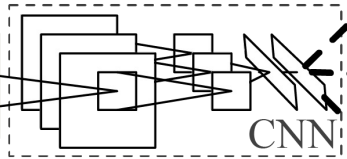
background: 0.0

person: 0.9

horse: 0.1

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

Region-Based CNN (R-CNN)

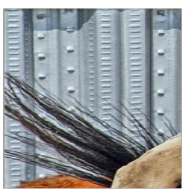


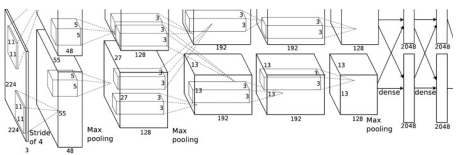
aeroplane? no.

person? yes.

tvmonitor? no.

classify regions





Detection is first of all a foreground/background classification problem.

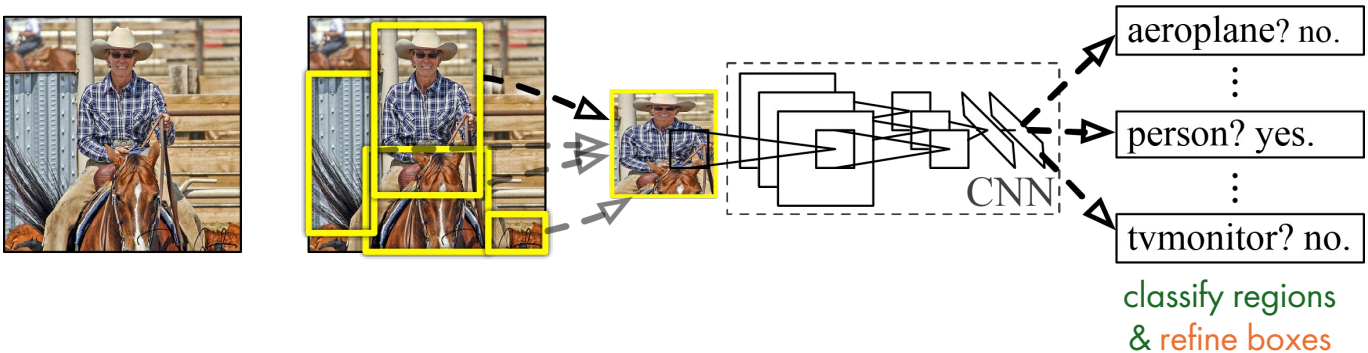
background: 0.9

person: 0.0

horse: 0.1

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

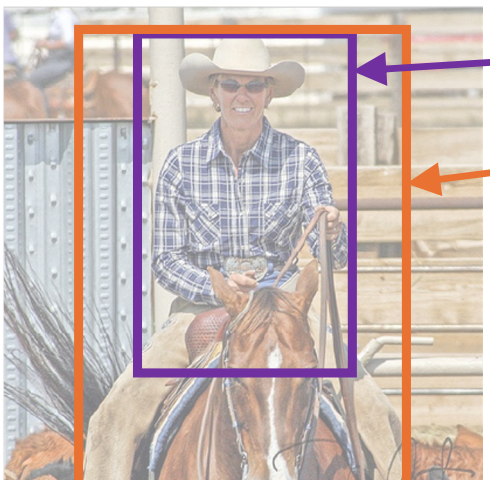
Region-Based CNN (R-CNN)



Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

Bounding-box Regression

- Geometric prediction from deep neural networks



reference box: (x, y, w, h)

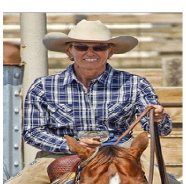
target box: (x', y', w', h')

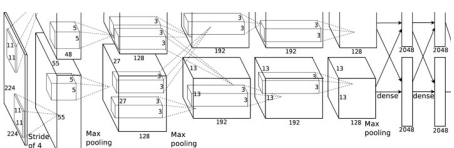
$$\Delta x = (x' - x)/w$$
$$\Delta y = (y' - y)/h$$
$$\Delta \log w = \log w' - \log w$$
$$\Delta \log h = \log h' - \log h$$

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

Bounding-box Regression

- Geometric prediction from deep neural networks
- one network, multiple tasks





classification

bbox regression

$\Delta x, \Delta y, \Delta \log w, \Delta \log h$

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

R-CNN

Slides adapted from: Ross Girshick, ICCV 2015

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation”, CVPR 2014

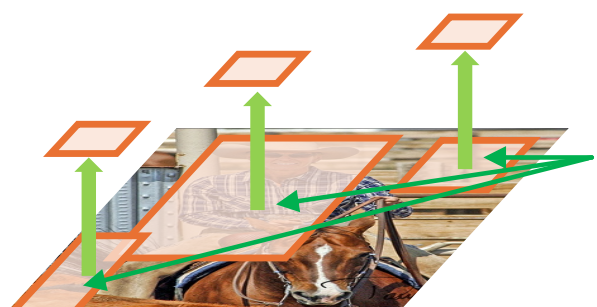
R-CNN

- Propose ~2000 Regions of Interest (RoI)

Slides adapted from: Ross Girshick, ICCV 2015

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

R-CNN

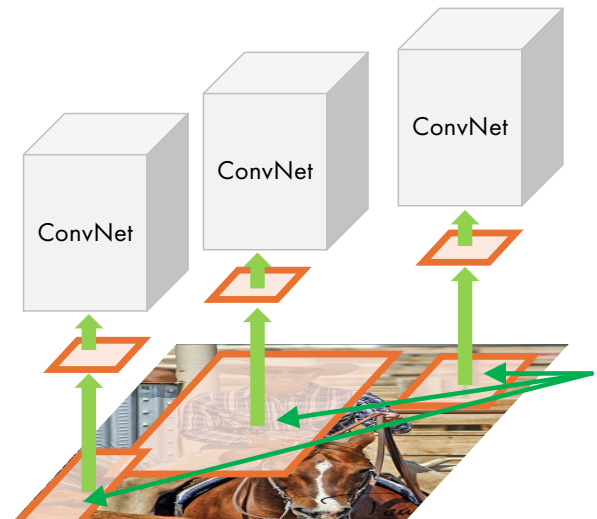


- Warp regions
- Propose ~2000 Regions of Interest (RoI)

Slides adapted from: Ross Girshick, ICCV 2015

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

R-CNN

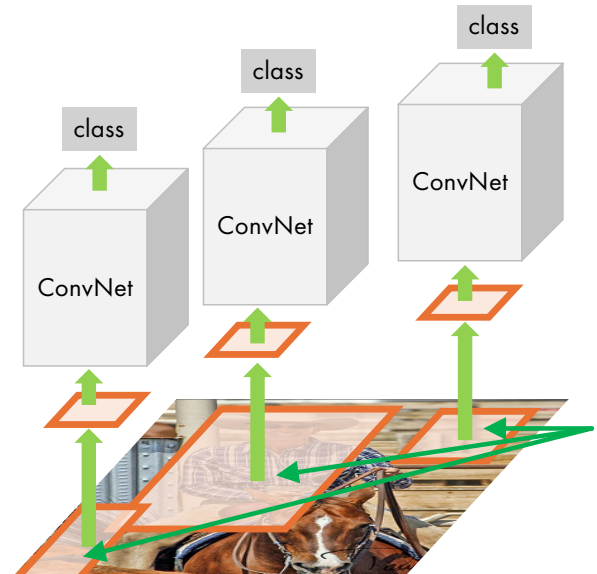


- Apply ConvNets to regions
- Warp regions
- Propose ~2000 Regions of Interest (RoI)

Slides adapted from: Ross Girshick, ICCV 2015

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

R-CNN

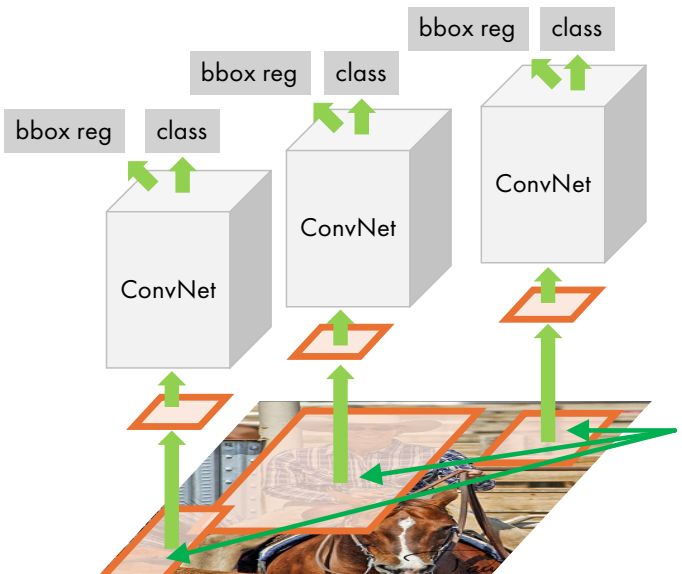


- Classify regions
- Apply ConvNets to regions
- Warp regions
- Propose ~2000 Regions of Interest (RoI)

Slides adapted from: Ross Girshick, ICCV 2015

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

R-CNN

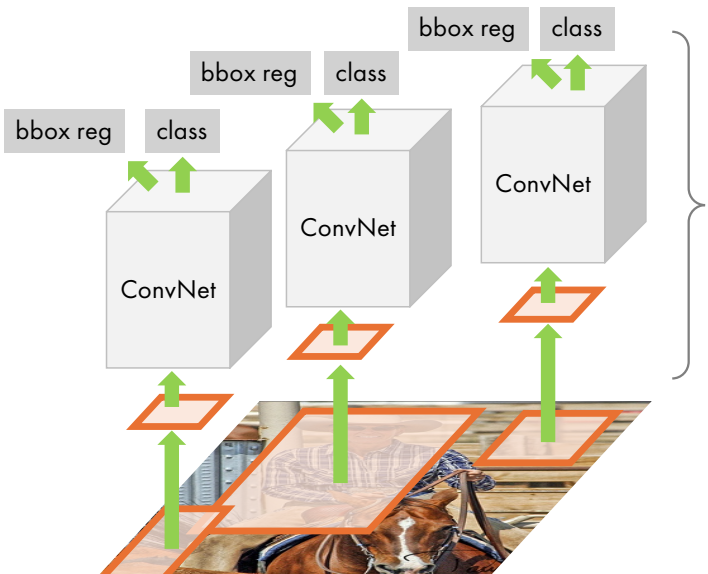


- Regress bounding-boxes
- Classify regions
- Apply ConvNets to regions
- Warp regions
- Propose ~2000 Regions of Interest (RoI)

Slides adapted from: Ross Girshick, ICCV 2015

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

R-CNN is "Slow" R-CNN (quote [Ross Girshick, talk at ICCV 2015])



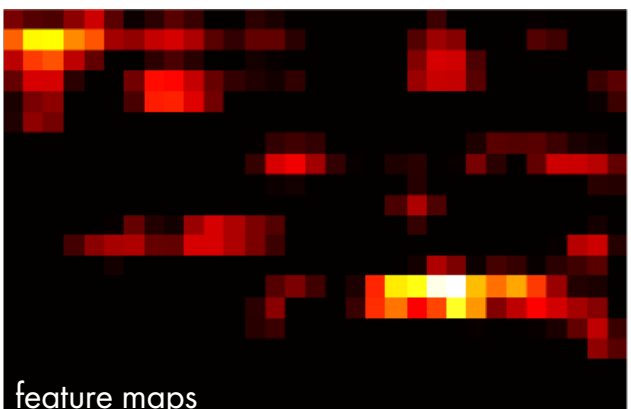
What's wrong w/ Slow R-CNN?
Many (~2000) ConvNets applied on regions!

Slides adapted from: Ross Girshick, ICCV 2015

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", CVPR 2014

Towards Fast R-CNN (SPPnet)

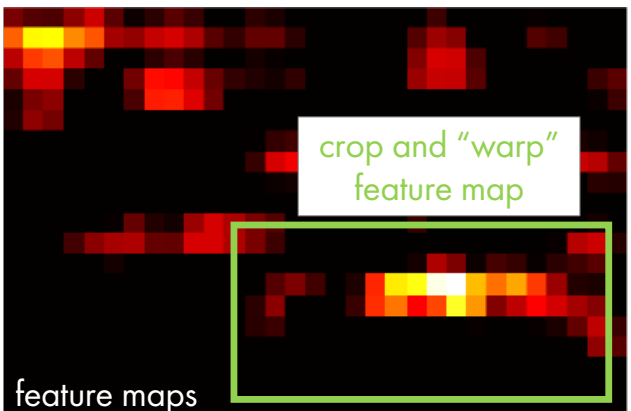
- ConvNets are Fully Convolutional!



Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition", ECCV 2014

Towards Fast R-CNN (SPPnet)

- ConvNets are Fully Convolutional!



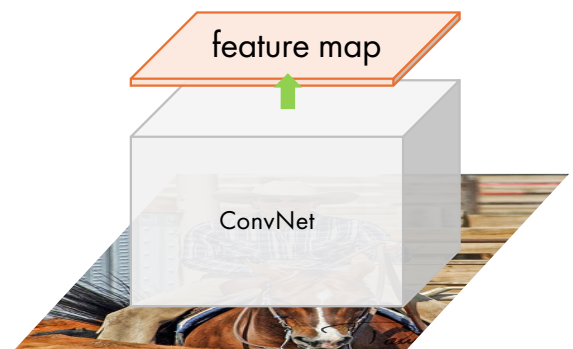
Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition", ECCV 2014

Towards Fast R-CNN (SPPnet)

Slides adapted from: Ross Girshick, ICCV 2015

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, “Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition”, ECCV 2014

Towards Fast R-CNN (SPPnet)

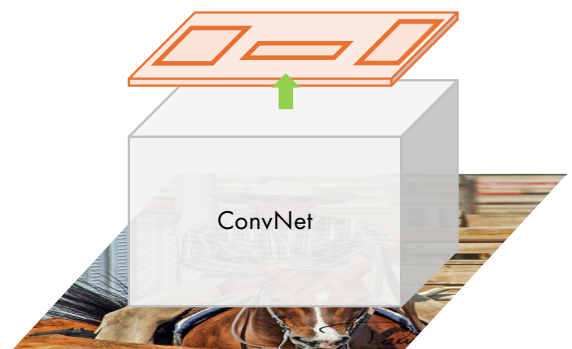


- Apply ConvNet to whole image

Slides adapted from: Ross Girshick, ICCV 2015

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition", ECCV 2014

Towards Fast R-CNN (SPPnet)

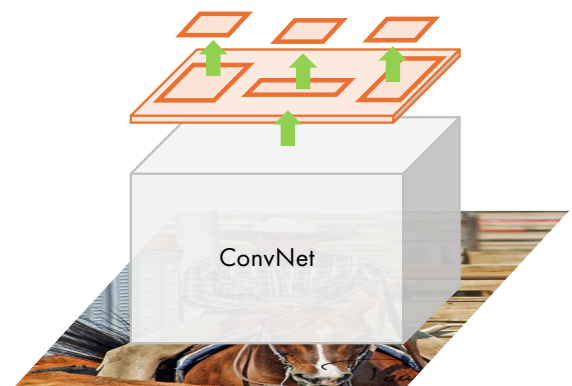


- Map proposed RoIs to feature maps
- Apply ConvNet to whole image

Slides adapted from: Ross Girshick, ICCV 2015

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition", ECCV 2014

Towards Fast R-CNN (SPPnet)



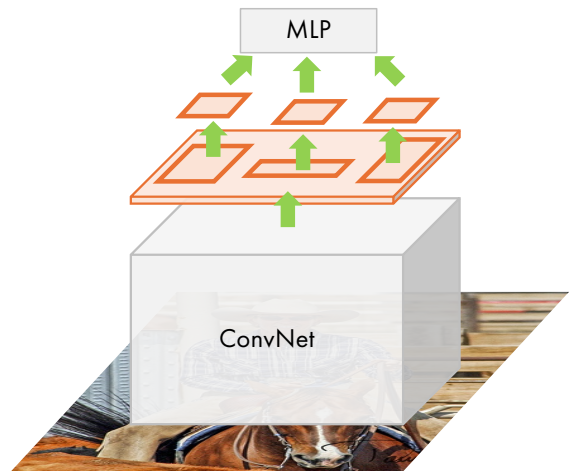
- Crop and pool features
- Map proposed RoIs to feature maps
- Apply ConvNet to whole image

*In original SPPnet, pooling is done with a fancier variant (spatial pyramid pooling)

Slides adapted from: Ross Girshick, ICCV 2015

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition", ECCV 2014

Towards Fast R-CNN (SPPnet)

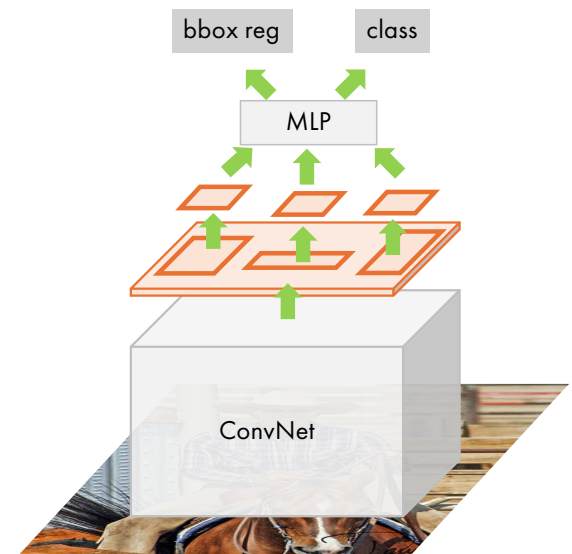


- Apply MLP on regions
- Crop and pool features
- Map proposed RoIs to feature maps
- Apply ConvNet to whole image

Slides adapted from: Ross Girshick, ICCV 2015

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition", ECCV 2014

Towards Fast R-CNN (SPPnet)



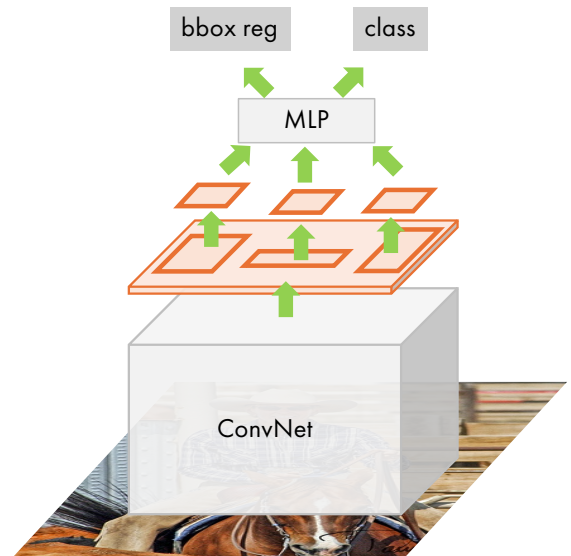
- Classify and regress boxes
- Apply MLP on regions
- Crop and pool features
- Map proposed RoIs to feature maps
- Apply ConvNet to whole image

Slides adapted from: Ross Girshick, ICCV 2015

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition", ECCV 2014

Towards Fast R-CNN (SPPnet)

- What's good with "fully convolutional" ConvNet?



many but small MLPs on regions

one "fully conv" ConvNet

- Classify and regress boxes
- Apply MLP on regions
- Crop and pool features
- Map proposed RoIs to feature maps
- Apply ConvNet to whole image

Slides adapted from: Ross Girshick, ICCV 2015

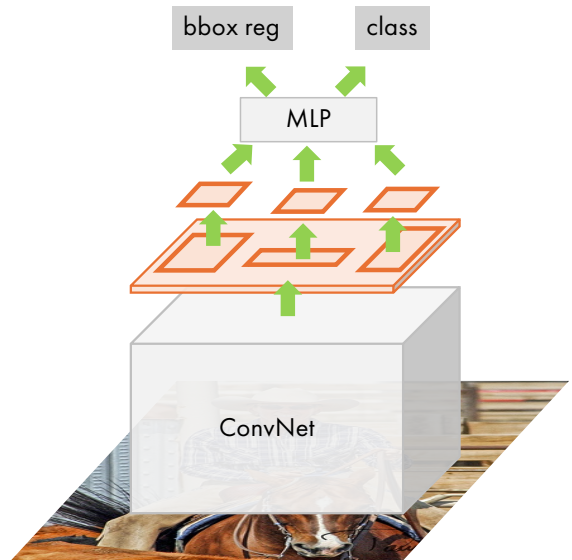
Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition", ECCV 2014

Towards Fast R-CNN (SPPnet)

- What's still wrong? Not yet "Differentiable Programming"

batching is w.r.t. regions (not images)

frozen pre-trained ConvNet



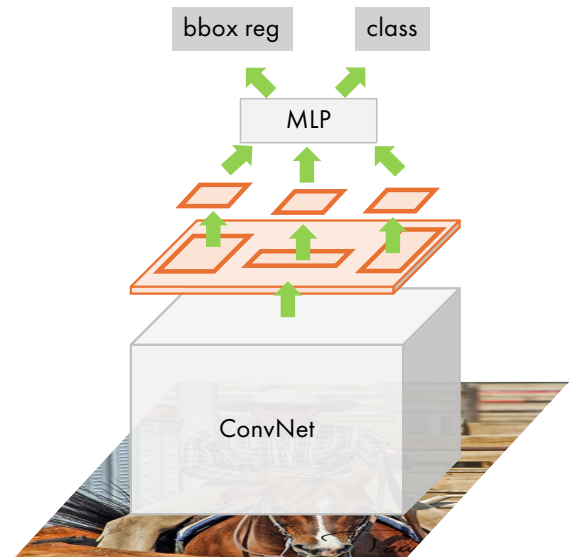
- Classify and regress boxes
- Apply MLP on regions
- Crop and pool features
- Map proposed RoIs to feature maps
- Apply ConvNet to whole image

Slides adapted from: Ross Girshick, ICCV 2015

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition", ECCV 2014

Fast R-CNN

- Differentiable Programing

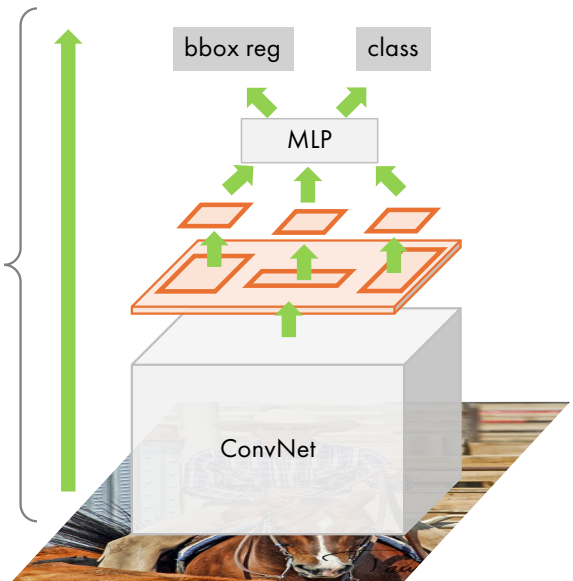


Slides adapted from: Ross Girshick, ICCV 2015
Ross Girshick, "Fast R-CNN", ICCV 2015

Fast R-CNN

- Differentiable Programing

forward path
(a program)

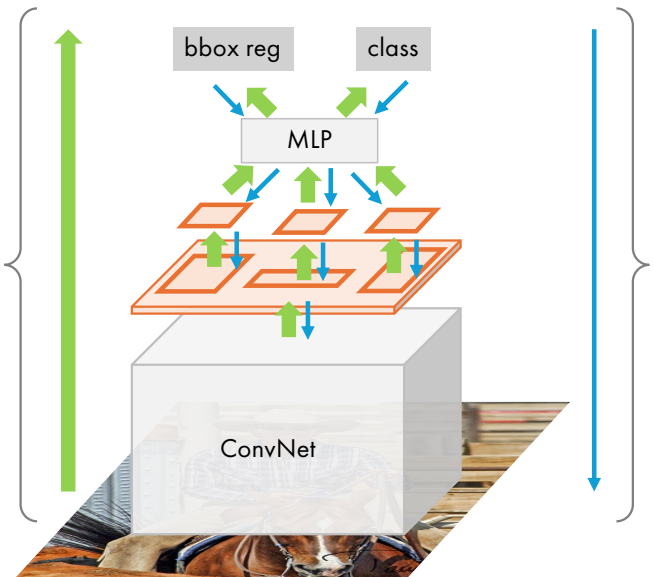


Slides adapted from: Ross Girshick, ICCV 2015
Ross Girshick, "Fast R-CNN", ICCV 2015

Fast R-CNN

- Differentiable Programing

forward path
(a program)



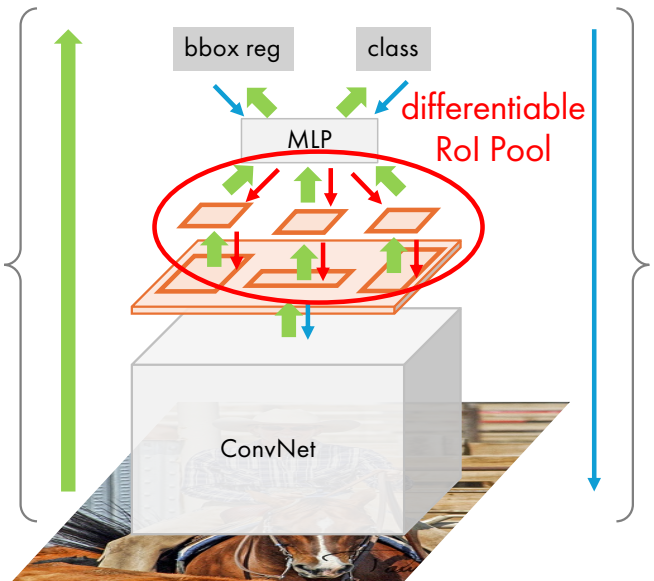
backward path
(differentiable)

Slides adapted from: Ross Girshick, ICCV 2015
Ross Girshick, "Fast R-CNN", ICCV 2015

Fast R-CNN

- Differentiable Programing

forward path
(a program)

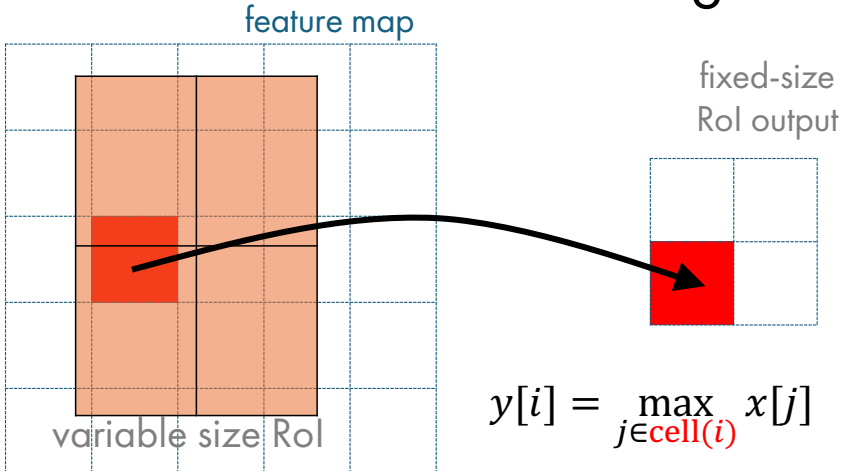


backward path
(differentiable)

Slides adapted from: Ross Girshick, ICCV 2015
Ross Girshick, "Fast R-CNN", ICCV 2015

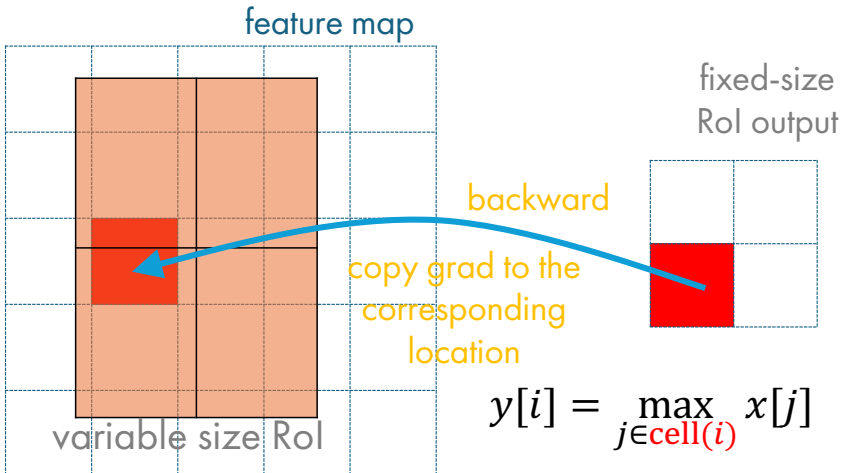
Fast R-CNN

Differentiable RoI Pooling



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

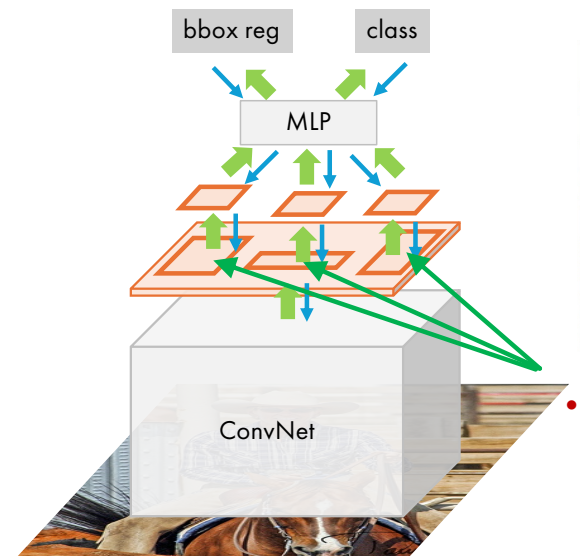
Differentiable RoI Pooling



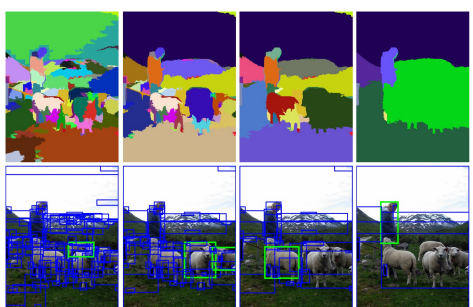
Ross Girshick, "Fast R-CNN", ICCV 2015

Fast R-CNN

- What's still wrong? Region proposals are not part of the program



recap: Selective Search



- Propose Regions of Interest (RoI)

Slides adapted from: Ross Girshick, ICCV 2015
Ross Girshick, "Fast R-CNN", ICCV 2015

Faster R-CNN

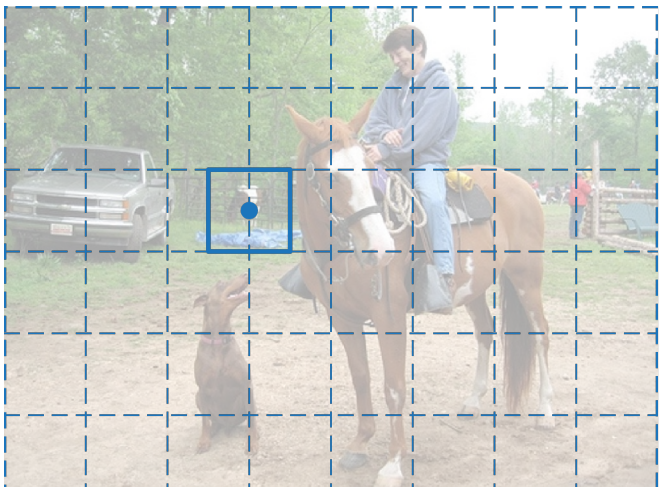
- How can neural nets produce regions?

- ConvNets are Fully Convolutional!
- Search objects w/ sliding windows (a.k.a., convolutions)

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN

- How can neural nets produce regions?

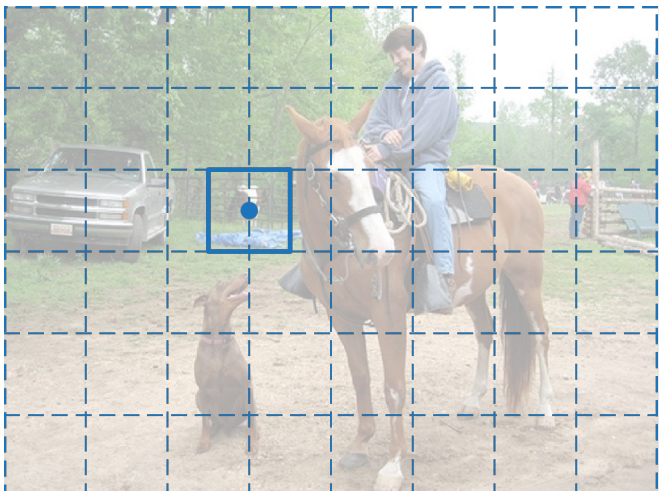


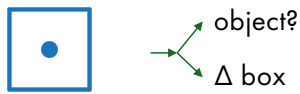
- ConvNets are Fully Convolutional!
- Search objects w/ sliding windows (a.k.a., convolutions)

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN

- How can neural nets produce regions?





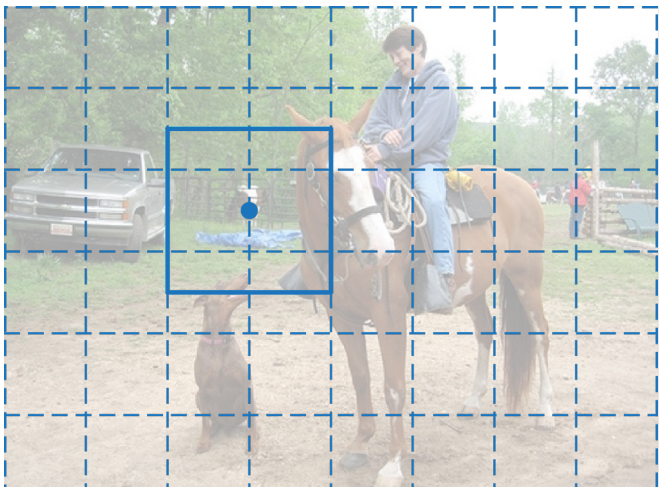
For each window, predict:

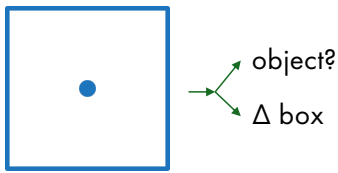
- Is it an object?
  - binary classification
- Object location w.r.t. window?
  - bbox regression

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN

- How can neural nets produce regions?





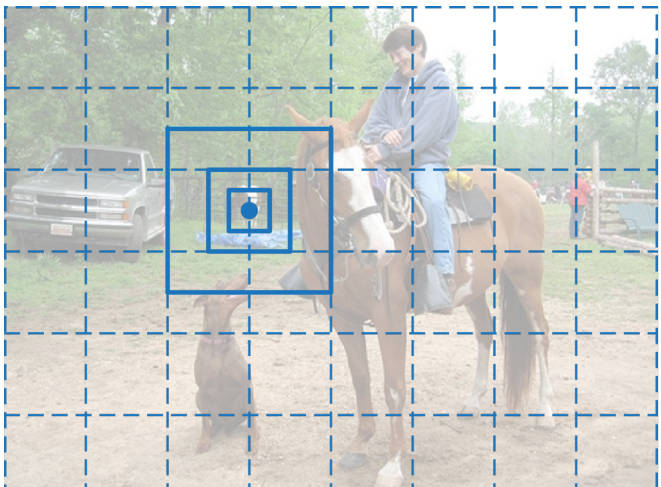
Anchor

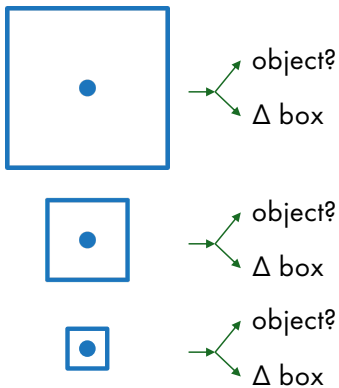
- pred w.r.t. a reference window
- reference is called an "anchor"
- be different w/ sliding window

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN

- How can neural nets produce regions?





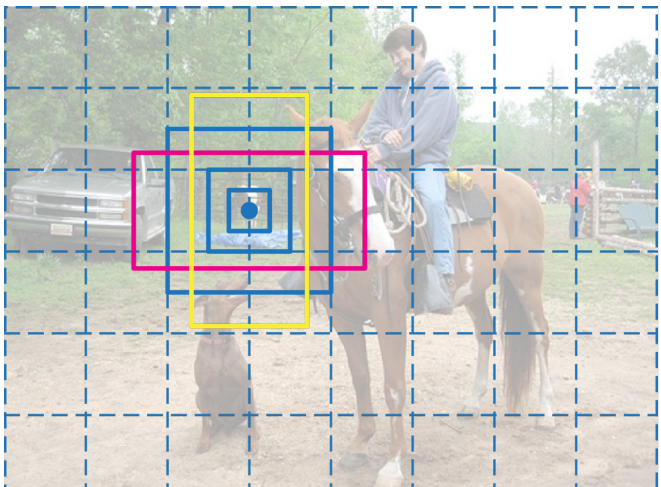
Anchors

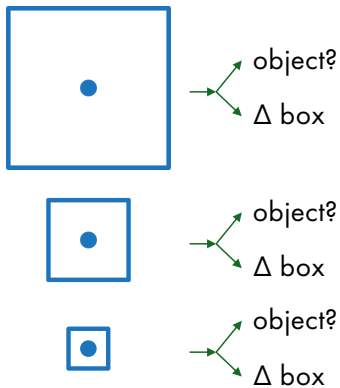
- one sliding window, many anchors
- multiple scales

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN

- How can neural nets produce regions?





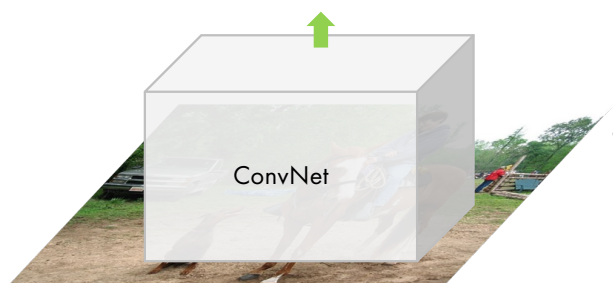
Anchors

- one sliding window, many anchors
- multiple scales
- multiple aspect ratios

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN: Region Proposal Network (RPN)

- How can neural nets produce regions?

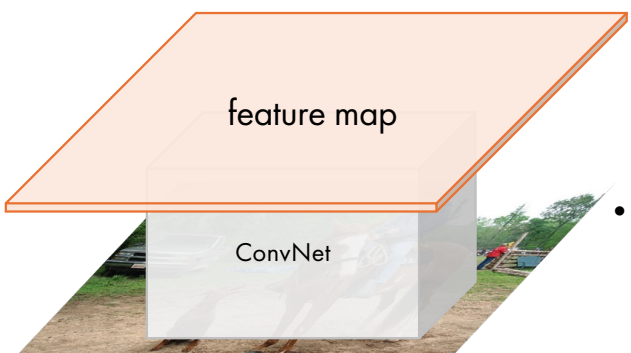


- Apply ConvNet to whole image

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN: Region Proposal Network (RPN)

- How can neural nets produce regions?

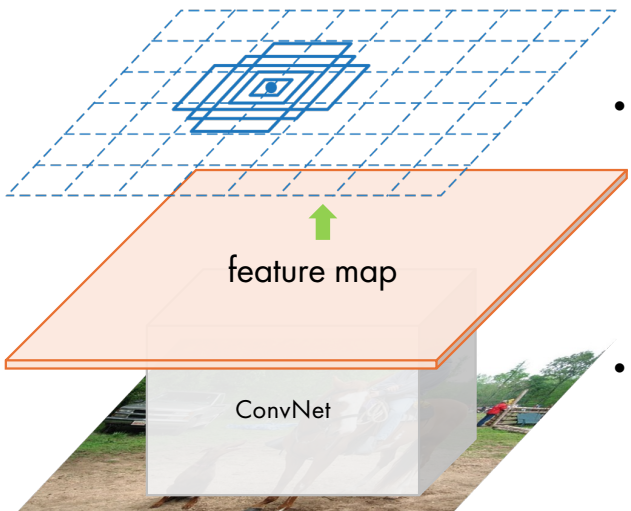


- Apply ConvNet to whole image

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN: Region Proposal Network (RPN)

- How can neural nets produce regions?

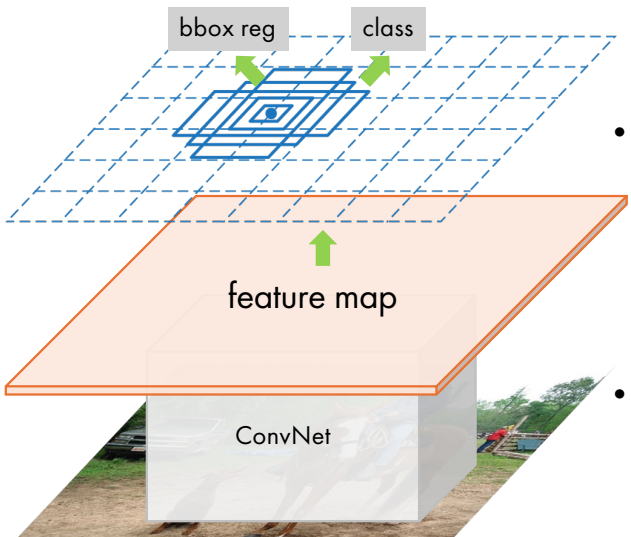


- Convs for *k* anchors
  - *k*-d: binary classification
  - *4k*-d: bbox regression
- Apply ConvNet to whole image

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN: Region Proposal Network (RPN)

- How can neural nets produce regions?



- Loss: reg + class
  - over all locations
  - over all anchors
- Convs for *k* anchors
  - *k*-d: binary classification
  - *4k*-d: bbox regression
- Apply ConvNet to whole image

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN: End-to-End Object Detection

- End-to-End Differentiable Programming

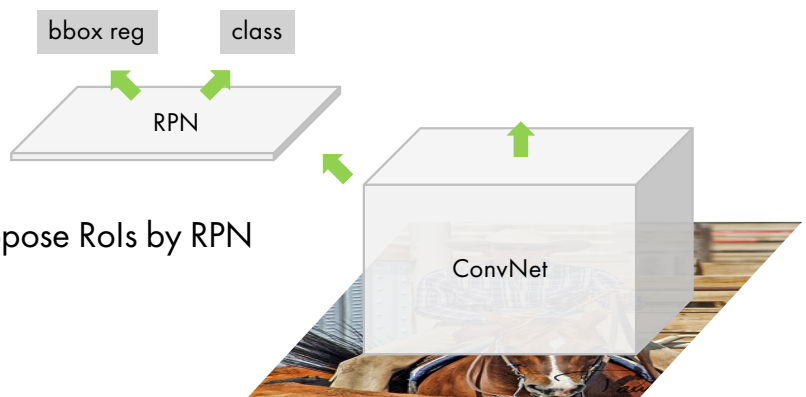
- Shared ConvNet backbone on whole image

Slides adapted from: Ross Girshick, ICCV 2015

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN: End-to-End Object Detection

- End-to-End Differentiable Programming



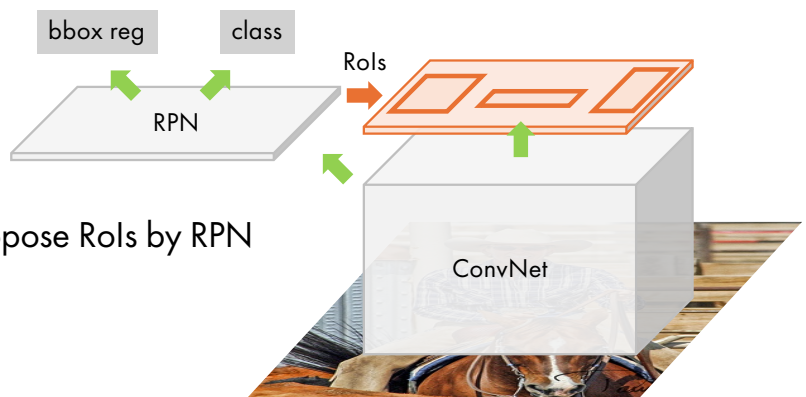
- Propose RoIs by RPN
- Shared ConvNet backbone on whole image

Slides adapted from: Ross Girshick, ICCV 2015

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN: End-to-End Object Detection

- End-to-End Differentiable Programming



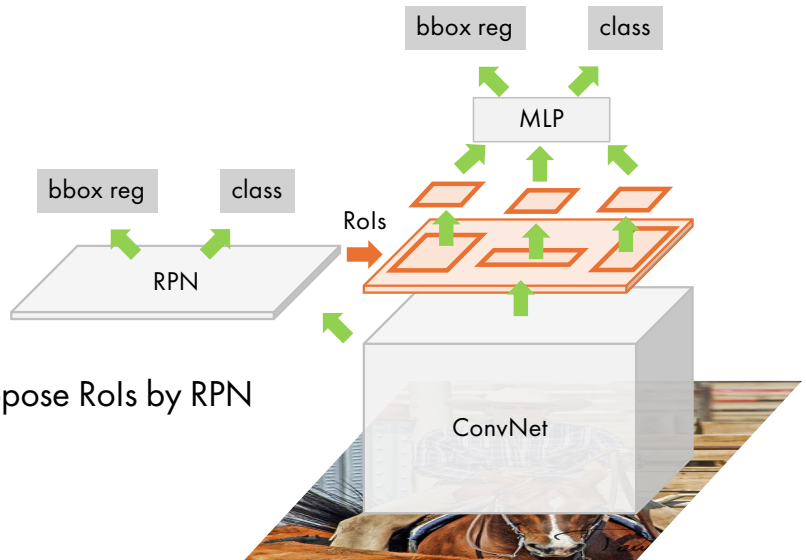
- Propose RoIs by RPN
- Apply RPN's RoIs
- Shared ConvNet backbone on whole image

Slides adapted from: Ross Girshick, ICCV 2015

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN: End-to-End Object Detection

- End-to-End Differentiable Programming



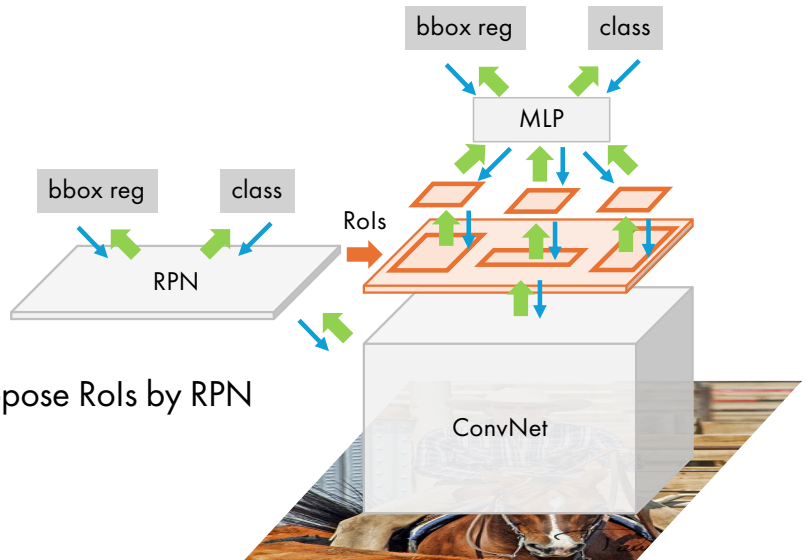
- Propose RoIs by RPN
- Same as Fast R-CNN
- Apply RPN's RoIs
- Shared ConvNet backbone on whole image

Slides adapted from: Ross Girshick, ICCV 2015

Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN: End-to-End Object Detection

- End-to-End Differentiable Programming

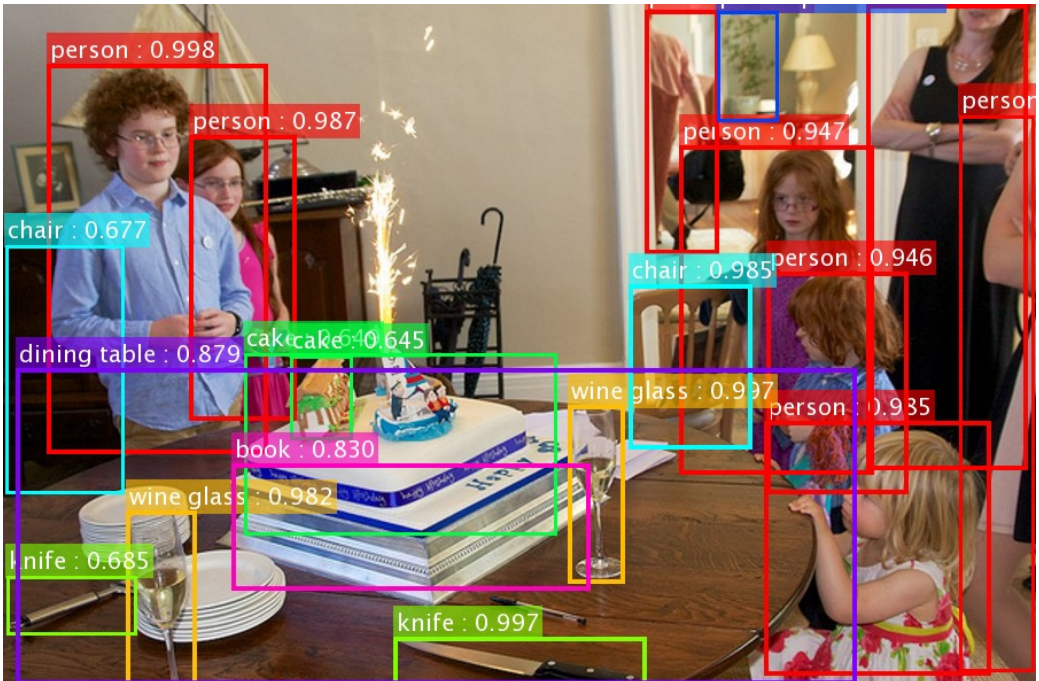


- Propose RoIs by RPN
- Same as Fast R-CNN
- Apply RPN's RoIs
- Shared ConvNet backbone on whole image

Slides adapted from: Ross Girshick, ICCV 2015

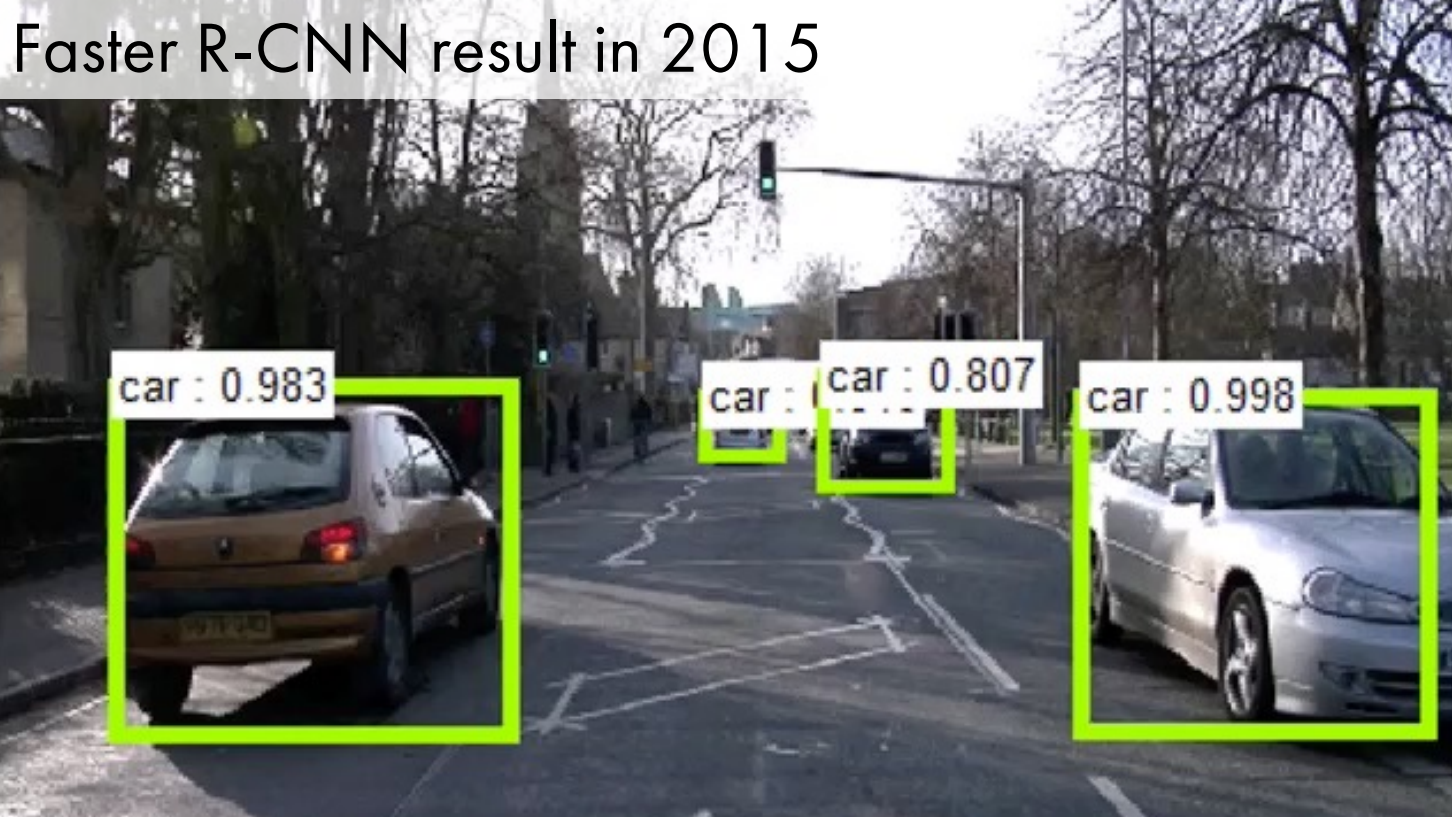
Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN result in 2015



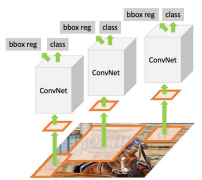
Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Faster R-CNN result in 2015



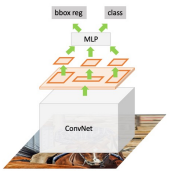
Evolution of R-CNN

R-CNN



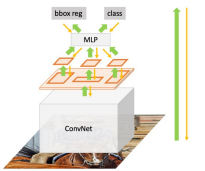
DL for detection made work!

Fast R-CNN "v0.1" (SPPnet)



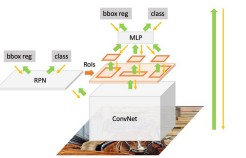
Fully convolutional feature maps

Fast R-CNN



Differentiable Programming

Faster R-CNN



RPN & End-to-End detection

Summary

- motivation / overview of recognition tasks
- CNNs
  - Classification
  - Segmentation
  - Object Detection

Takeaways

- what are the building blocks of CNNs?
- what are advantages of CNNs vs FCNs?
- what are the trainable parameters of CNNs?
- what are the inputs/outputs of recognition tasks?
- how do we measure success?
- what are the tradeoffs between different methods for object detection?

Next Time

- Transformers!