

lec11

Creation Date: 11/01/2020 16:31

Last Modified Date: 12/01/2020 00:46

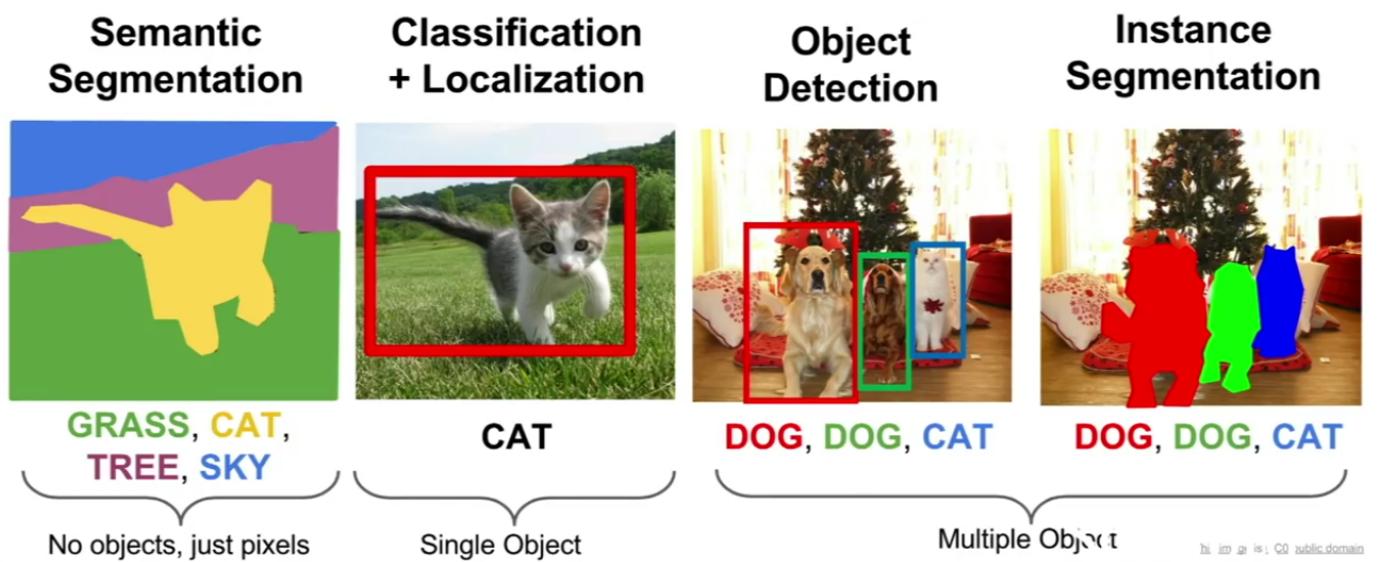
Lec 11: Segmentation, Localization and Detection

Today: Segmentation, Localization, Detection

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 15 University May 10, 2017

Other Computer Vision Tasks



Fei-Fei Li & Justin Johnson & Serena Yeung

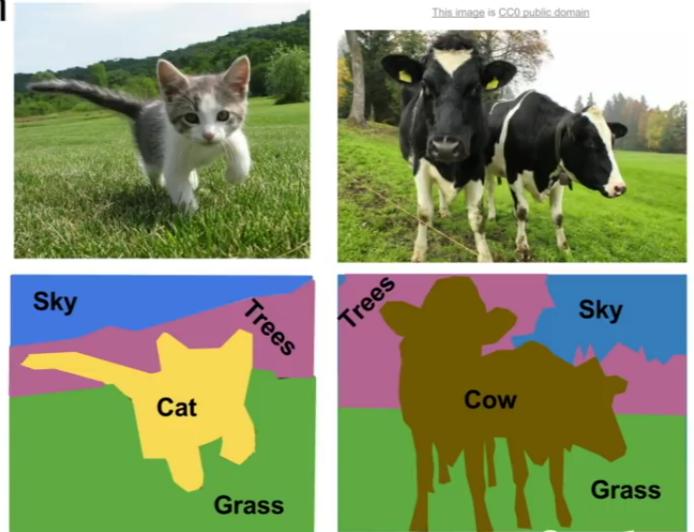
Lecture 11 - 17 University May 10, 2017

- Semantic Segmentation:
 - The goal is to classify each pixel in the given image.

Semantic Segmentation

Label each pixel in the image with a category label

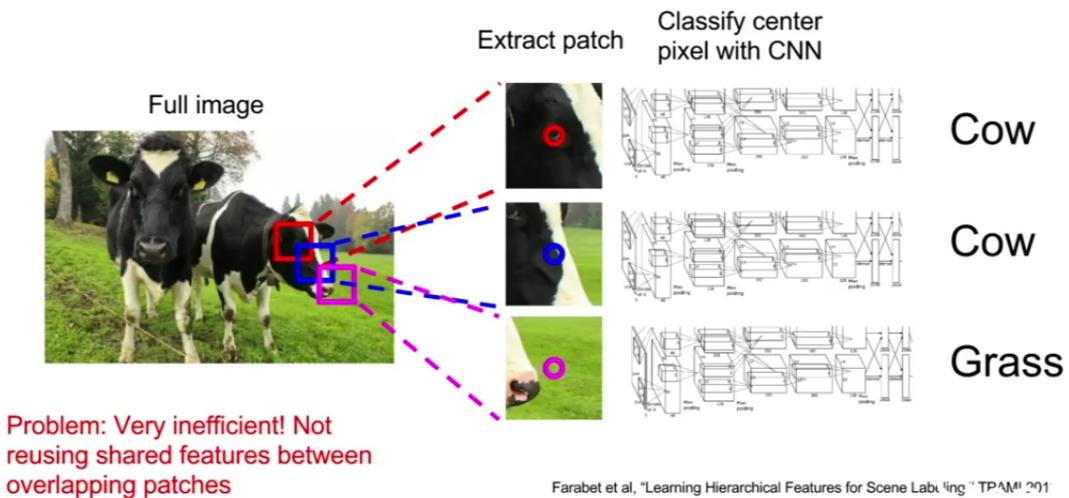
Don't differentiate instances, only care about pixels



○ Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - University 19 May 10, 2017

Semantic Segmentation Idea: Sliding Window



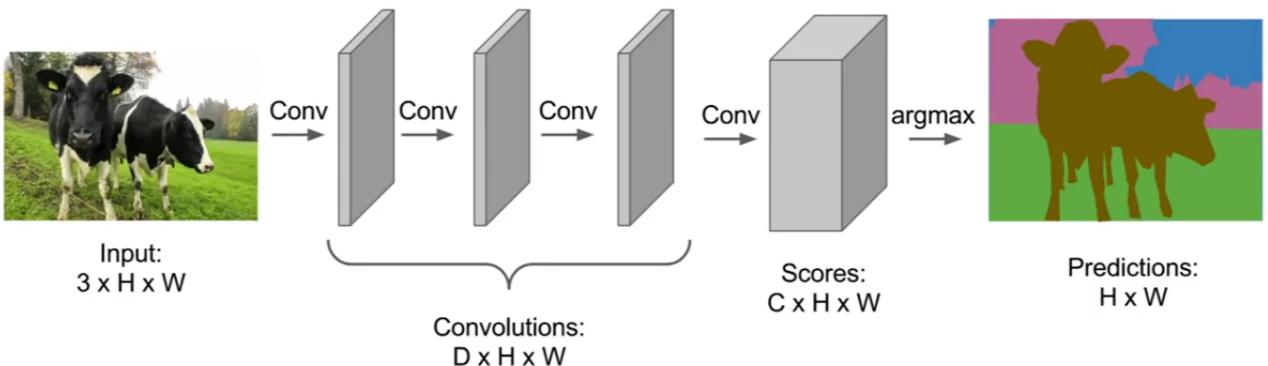
○ Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - University 21 May 10, 2017

- In sliding window technique we want to know what is the category label for the center pixel of the small slide/patch.

Semantic Segmentation Idea: Fully Convolutional

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



Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - University 22 May 10, 2017

- Here we are preserving the spatial size of the image passing through the network and in the end we will have a tensor where the depth will have C channels which shows the probability map of each class in its respective layer.
- During training we have cross entropy loss for each pixel and then sum or average over the minibatch.
- Drawbacks:
 - Extremely heavy model since we will have lots of channels in between and we are also maintaining the spatial resolution of the image through the network.
 - Training time will be high since there are lots of convolutional operations being performed.
 - Solution: **Down-sample and then up-sample the image.**

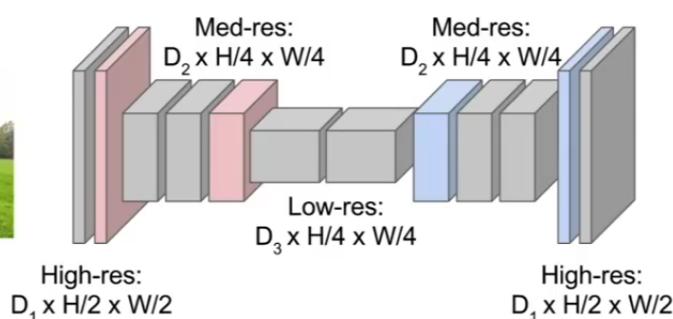
Semantic Segmentation Idea: Fully Convolutional

Downsampling:
Pooling, strided
convolution



Input:
3 x H x W

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



Upsampling:
???



Predictions:
H x W

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

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - University 25 May 10, 2017

- For Down-sampling we have various techniques like pooling, strided convolutions.
- Most of the networks have a different way of up-sampling.

In-Network upsampling: “Unpooling”

Nearest Neighbor

1	2
1	2
3	4

Input: 2 x 2

1	1	2	2
1	1	2	2
3	3	4	4
3	3	4	4

Output: 4 x 4

“Bed of Nails”

1	2
3	4

Input: 2 x 2

1	0	2	0
0	0	0	0
3	0	4	0
0	0	0	0

Output: 4 x 4

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - University 26 May 10, 2017

In-Network upsampling: “Max Unpooling”

Max Pooling

Remember which element was max!

1	2	6	3
3	5	2	1
1	2	2	1
7	3	4	8

Input: 4 x 4

5	6
7	8

Output: 2 x 2

Max Unpooling

Use positions from pooling layer

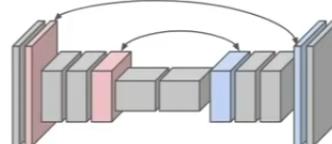
1	2
3	4

Input: 2 x 2

0	0	2	0
0	1	0	0
0	0	0	0
3	0	0	4

Output: 4 x 4

Corresponding pairs of
downsampling and
upsampling layers



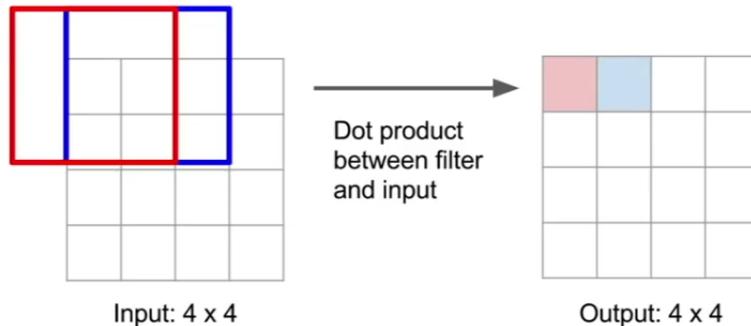
Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - University 27 May 10, 2017

- If we do not use the max-pool indices we won't be knowing where to put the value, because in up-sampling the dimension will increase and since we have a down-sampled image, we have lost the spatial information from the local receptive field.
- The techniques such as Nearest-Neighbours, Bed-of-Nails, Max-Unpooling with indices are fixed functions.**
- Why don't we learn the Up-sampling?
- Ans: **Transpose Convolution**

Learnable Upsampling: Transpose Convolution

Recall: Normal 3×3 convolution, stride 1 pad 1

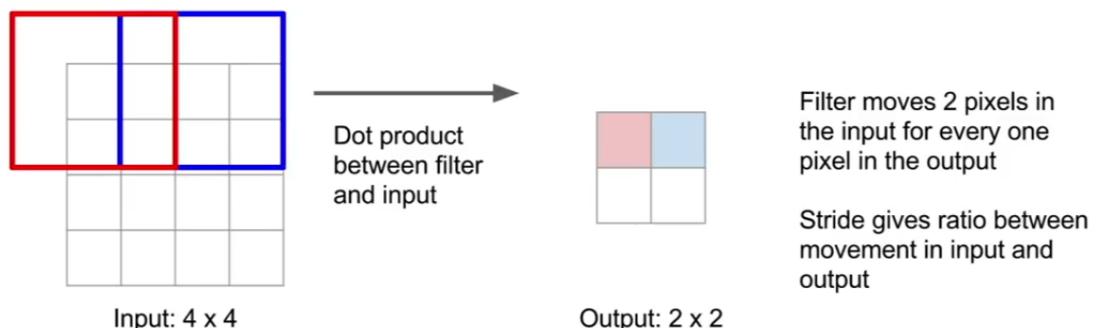


○ Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 30 May 10, 2017 University

Learnable Upsampling: Transpose Convolution

Recall: Normal 3×3 convolution, stride 2 pad 1



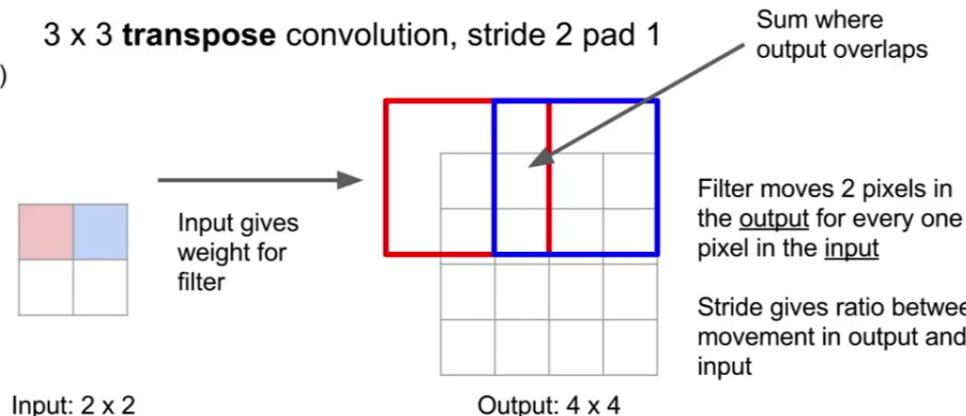
○ Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 33 May 10, 2017 University

- Transpose Convolution:

Learnable Upsampling: Transpose Convolution

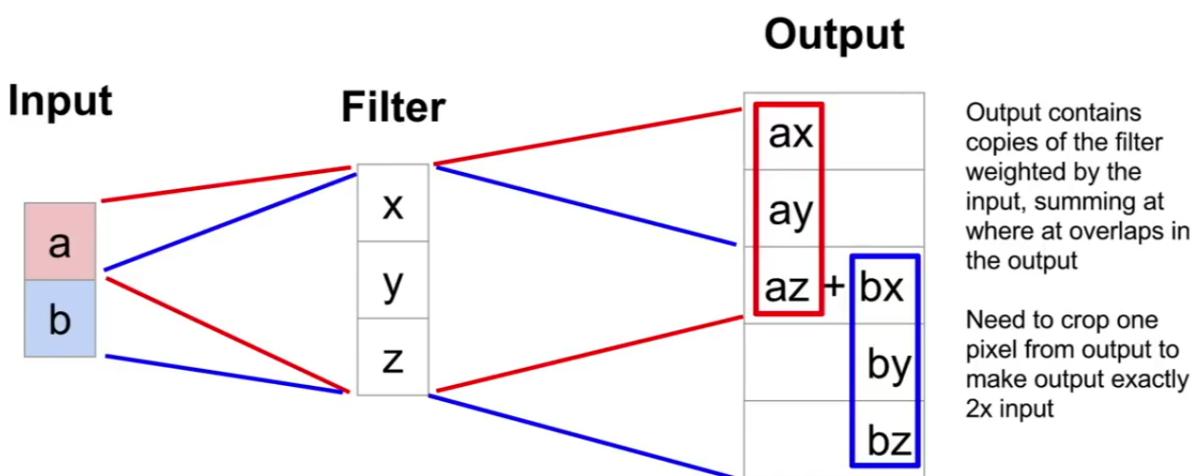
Other names:
-Deconvolution (bad)
-Upconvolution
-Fractionally strided convolution
-Backward strided convolution



Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 38 University, May 10, 2017

Transpose Convolution: 1D Example



Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 39 University, May 10, 2017

Convolution as Matrix Multiplication (1D Example)

We can express convolution in terms of a matrix multiplication

$$\vec{x} * \vec{a} = X\vec{a}$$

$$\begin{bmatrix} x & y & x & 0 & 0 & 0 \\ 0 & x & y & x & 0 & 0 \\ 0 & 0 & x & y & x & 0 \\ 0 & 0 & 0 & x & y & x \end{bmatrix} \begin{bmatrix} 0 \\ a \\ b \\ c \\ d \\ 0 \end{bmatrix} = \begin{bmatrix} ay + bz \\ ax + by + cz \\ bx + cy + dz \\ cx + dy \end{bmatrix}$$

Example: 1D conv, kernel size=3, stride=1, padding=1

Convolution transpose multiplies by the transpose of the same matrix:

$$\vec{x} *^T \vec{a} = X^T\vec{a}$$

$$\begin{bmatrix} x & 0 & 0 & 0 \\ y & x & 0 & 0 \\ z & y & x & 0 \\ 0 & z & y & x \\ 0 & 0 & z & y \\ 0 & 0 & 0 & z \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} ax \\ ay + bx \\ az + by + cx \\ bz + cy + dx \\ dz + dy \\ dy \end{bmatrix}$$

When stride=1, convolution transpose is just a regular convolution (with different padding rules)

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 41 University May 10, 2017

- Here $[a,b,c,d]$ is the input vector padded with 0, and the kernel is $[x,y,z]$.

Convolution as Matrix Multiplication (1D Example)

We can express convolution in terms of a matrix multiplication

$$\vec{x} * \vec{a} = X\vec{a}$$

$$\begin{bmatrix} x & y & x & 0 & 0 & 0 \\ 0 & 0 & x & y & x & 0 \end{bmatrix} \begin{bmatrix} 0 \\ a \\ b \\ c \\ d \\ 0 \end{bmatrix} = \begin{bmatrix} ay + bz \\ bx + cy + dz \end{bmatrix}$$

Example: 1D conv, kernel size=3, stride=2, padding=1

Convolution transpose multiplies by the transpose of the same matrix:

$$\vec{x} *^T \vec{a} = X^T\vec{a}$$

$$\begin{bmatrix} x & 0 \\ y & 0 \\ z & x \\ 0 & y \\ 0 & z \\ 0 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} ax \\ ay \\ az + bx \\ by \\ bz \\ 0 \end{bmatrix}$$

When stride>1, convolution transpose is no longer a normal convolution!

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 43 University May 10, 2017

- We see **checkerboard** patterns for higher strides and for filter of odd sizes.

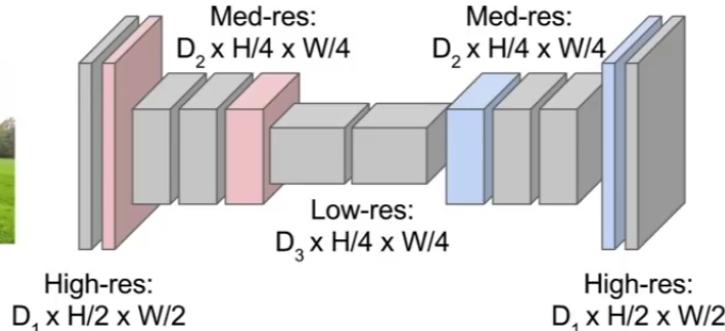
Semantic Segmentation Idea: Fully Convolutional

Downsampling:
Pooling, strided convolution



Input:
 $3 \times H \times W$

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



Upsampling:
Unpooling or strided transpose convolution



Predictions:
 $H \times W$

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

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 44 University, May 10, 2017

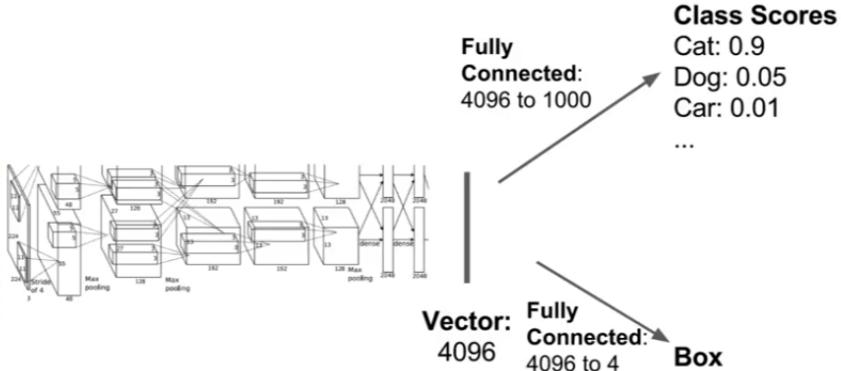
Classification + Localization

- Here along with classifying an object we also want to draw a bounding box to localize the object present in the image

Classification + Localization



This image is CC0 public domain



Treat localization as a regression problem!

Fei-Fei Li & Justin Johnson & Serena Yeung

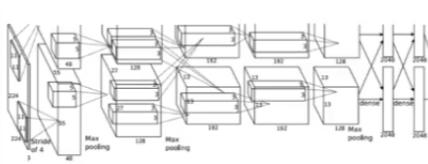
Lecture 11 - 46 University, May 10, 2017

- One set of FC layer can perform the classification whereas the other can generate the Box Coordinates.

Classification + Localization



This image is CC0 public domain



Fully Connected:
4096 to 1000

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

Correct label:
Cat

Softmax Loss

Vector: Fully Connected:
4096 4096 to 4

Box Coordinates → L2 Loss
(x, y, w, h)

Correct box:
(x^*, y^*, w^*, h^*)

Treat localization as a
regression problem!

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 47 University, May 10, 2017

- The L2 loss can be used for the bounding box coordinates and Since we are having only 1 object in the image we can use Softmax Loss.
- These kind of losses in the task is called as **Multi-Task Loss**.
- In Multi-Task Loss, if we have k loss terms(scalars), we perform a weighted sum to create a single loss scalar. And we can use this weighted loss terms for finding the gradient.
- So since the weighting parameter is itself a hyperparameter in such scenarios, we are manipulating the absolute value of the losses. So the other hyperparameters like learning rate etc will vary.
- Hence various approaches look at optimizing the metric associated with the tasks **

Aside: Human Pose Estimation



Represent pose as a
set of 14 joint
positions:

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

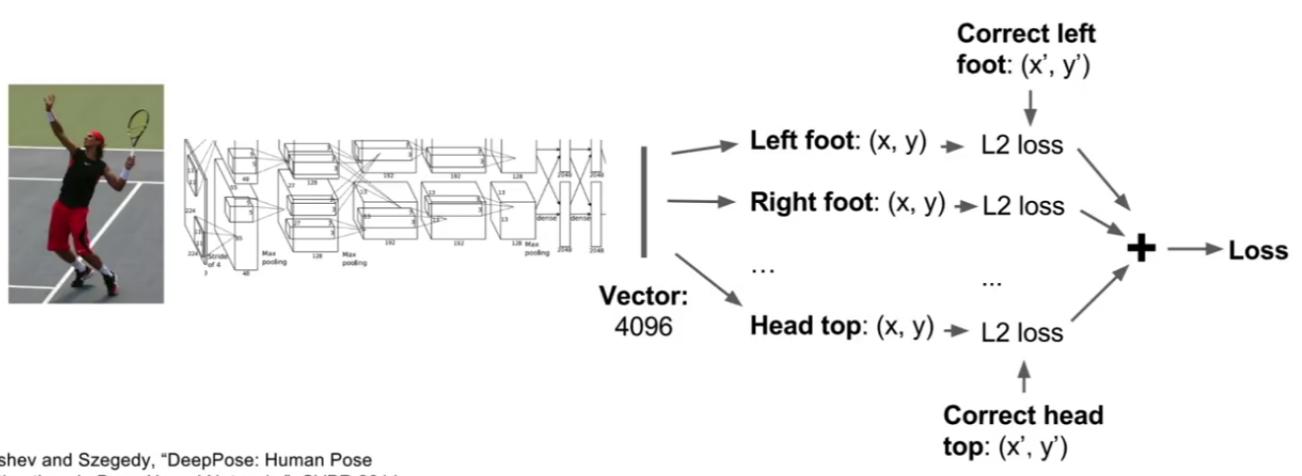
This image is licensed under CC-BY 2.0.

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

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 50 University, May 10, 2017

Aside: Human Pose Estimation

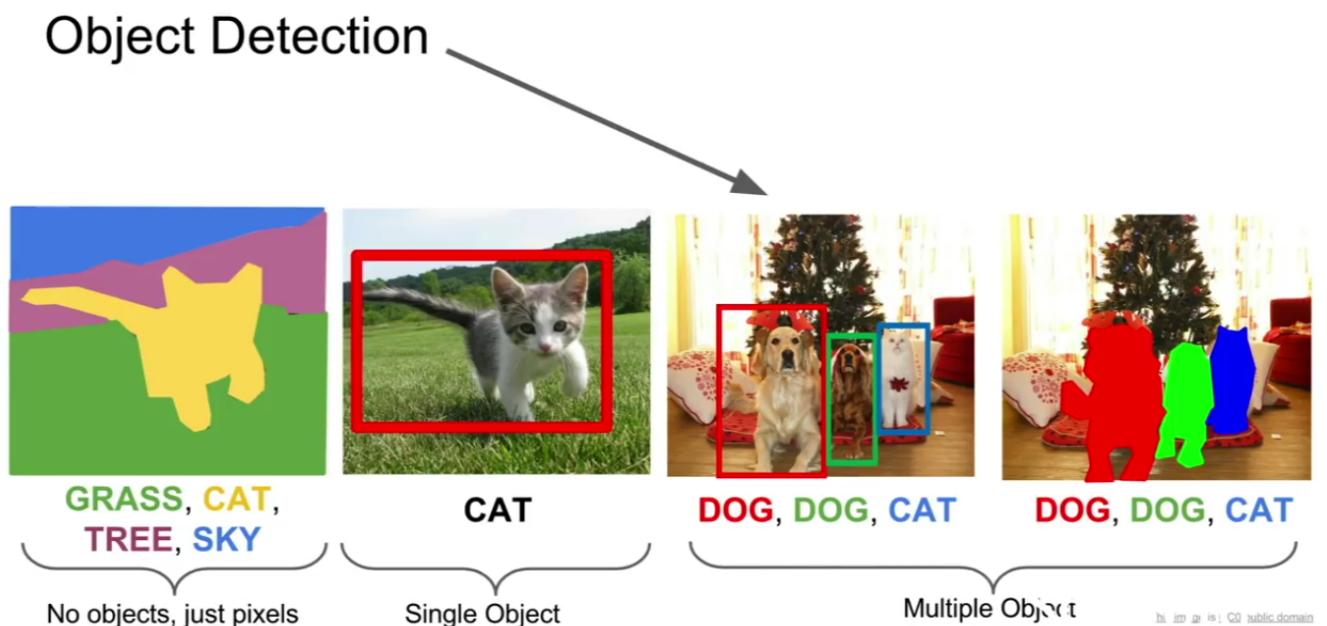


Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 52 University, May 10, 2017

- Regression loss basically means L1,L2 loss etc where the values used are continuous in nature.
- Categorical losses are Cross-Entropy, Softmax, SVM margin loss, are used when we have a fixed number of categories.

Object Detection



Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 53 University, May 10, 2017

- Object Detection: We have a fixed number of categories, and when a image has any object from those categories, we are supposed to classify it and draw a bounding box around it.
- **Detection varies from Classification+Localization because in detection we will have varying number of detection in each image, i.e. you do not expect how many number of objects are there in the image.**

Object Detection: Impact of Deep Learning

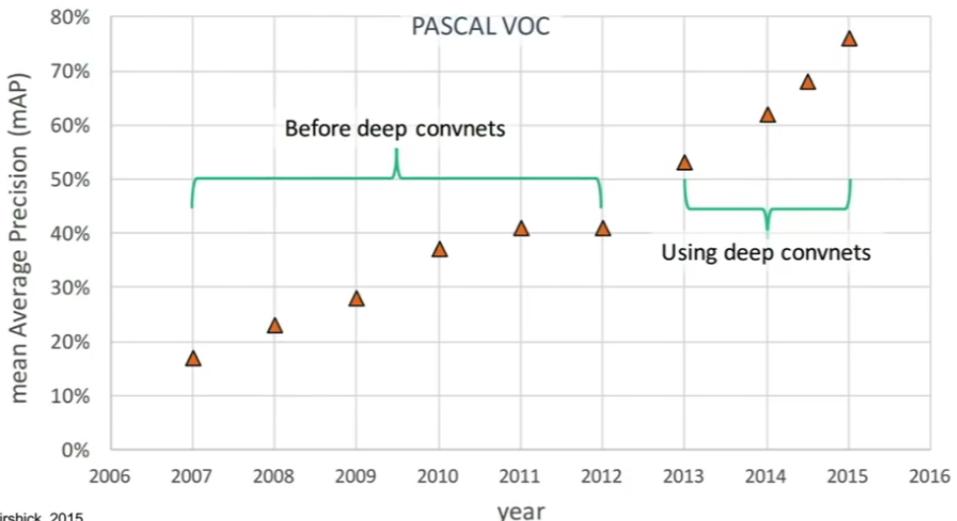


Figure copyright Ross Girshick, 2015.
Reproduced with permission.

Fei-Fei Li & Justin Johnson & Serena Yeung

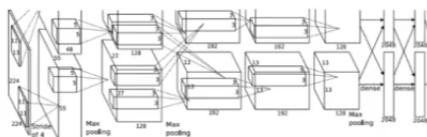
Lecture 11 - 54 University, May 10, 2017

-

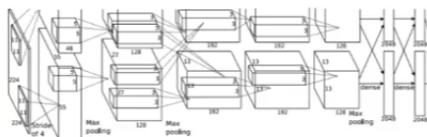
- Before 2013(deep-learning) the best methods were having almost similar results.

Object Detection as Regression?

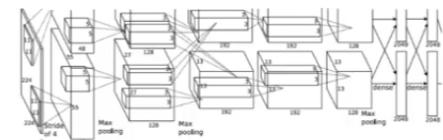
Each image needs a different number of outputs!



CAT: (x, y, w, h) 4 numbers



DOG: (x, y, w, h)
DOG: (x, y, w, h) 16 numbers
CAT: (x, y, w, h)



DUCK: (x, y, w, h) Many
DUCK: (x, y, w, h) numbers!
....

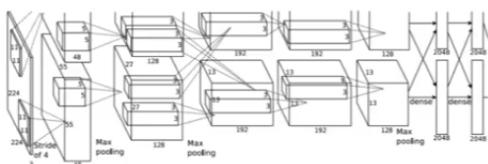
Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 56 University, May 10, 2017

- If we try to solve the Object Detection problem as a regression problem then we have the inherent issue that each image should have varying number of output values.

Object Detection as Classification: 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

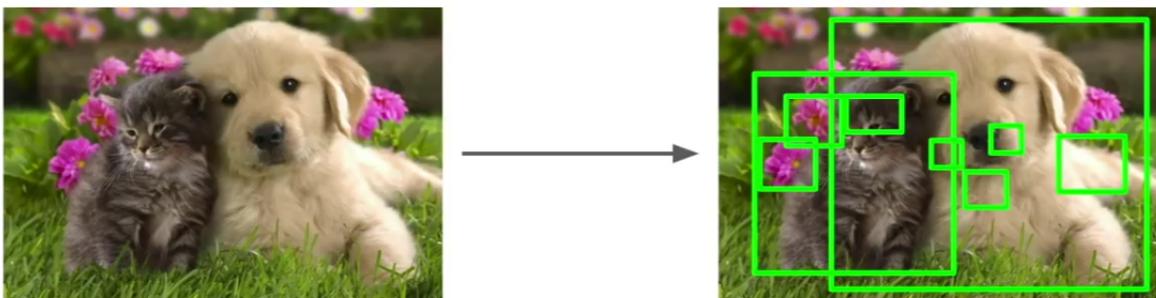
Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 57 University, May 10, 2017

- In sliding window we try to classify the window slice only. Time consuming approach and lot of spatial redundancy while sliding the window.
- Problem:
 - We need to choose sliding window size.
 - Too much of convolution operation!
 - Too time consuming!
- Solution: **Region Proposals** to provide slides of images, reducing the unnecessary computations.
- Region Proposals:

Region Proposals

- Find “blobby” image regions that are likely to contain objects
- Relatively fast to run; e.g. Selective Search gives 1000 region proposals in a few seconds on CPU



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

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 62 University, May 10, 2017

- R-CNN

R-CNN



Input image

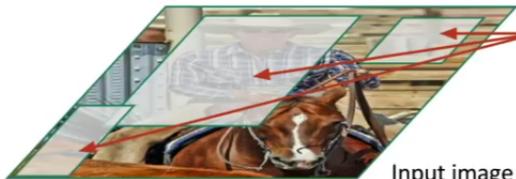
Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; 用于教学目的.

○ Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 63 University of Washington, May 10, 2017

- Given an image we generate Region Proposals also called as Region of Interest(ROIs).

R-CNN



Input image

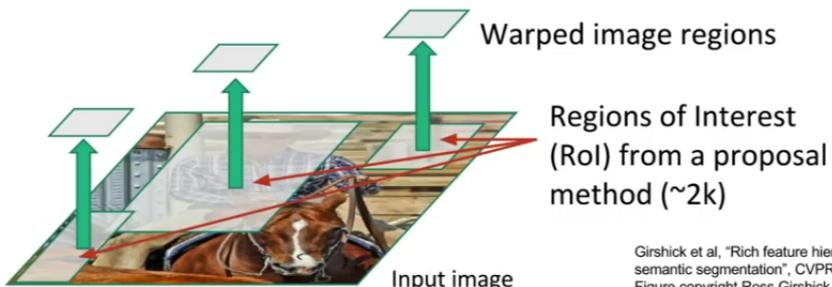
Regions of Interest
(RoI) from a proposal
method (~2k)

Girshick et al, "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; 用于教学目的.

○ Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 64 University of Washington, May 10, 2017

R-CNN



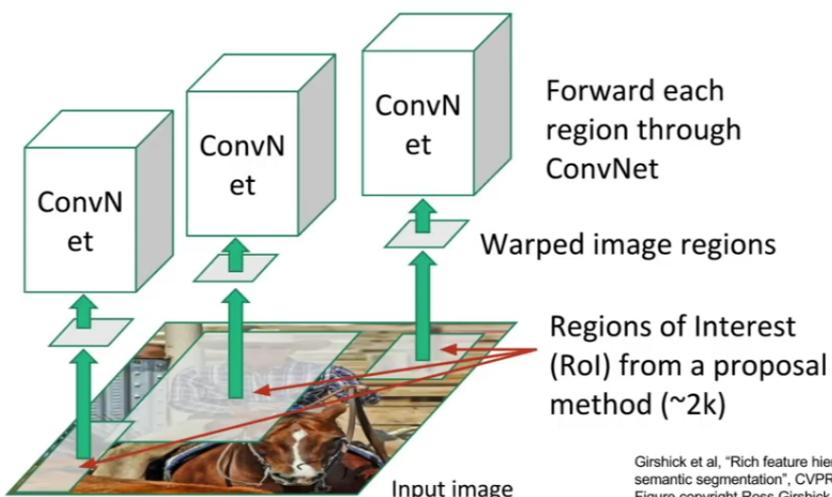
Girshick et al., "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; [http:// Ross Girshick.com](#)

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 65 University May 10, 2017

- Since the CNN take images of single size and the ROIs are of different sizes, we have to warp them to a common size which is required by the CNNs.

R-CNN



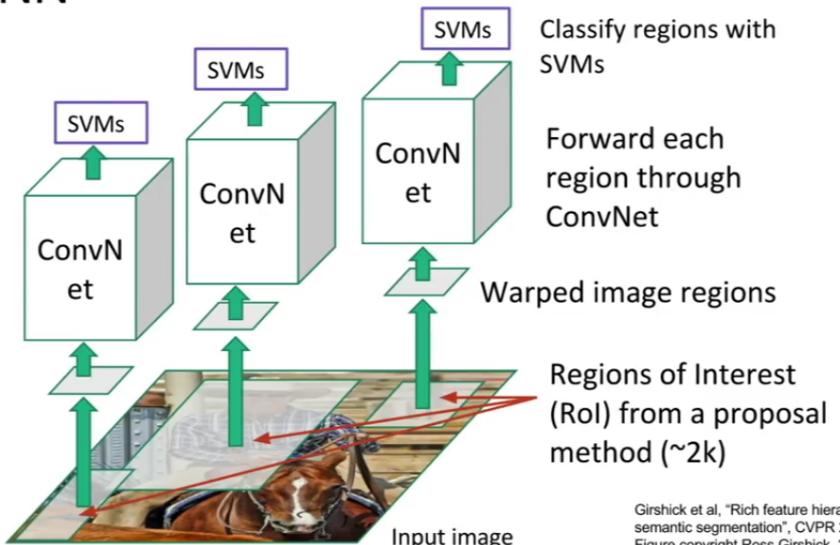
Girshick et al., "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Figure copyright Ross Girshick, 2015; [http:// Ross Girshick.com](#)

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 66 University May 10, 2017

- Then we pass these warped image regions through the CNNs

R-CNN



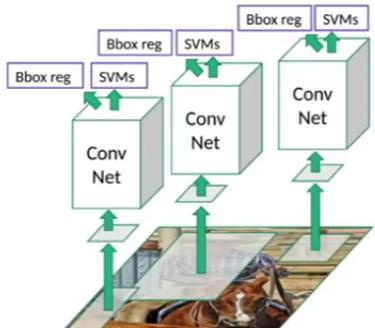
Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 67 University of California, Berkeley, May 10, 2017

- The output of CNNs are passed to SVM for classification.
- Along with classification we also generate the bounding box coordinates. (missing in the slide)

R-CNN: Problems

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



Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 68 University of California, Berkeley, May 10, 2017

- Fast R-CNN

Girshick et al., "Rich feature hierarchies for accurate object detection and semantic segmentation", CVPR 2014.
Slide copyright Ross Girshick, 2015; [http://csail.mit.edu/people/girshick/pubs/cvpr15.pdf](#).

Fast R-CNN



Input image

Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015. All rights reserved. Reproduction or redistribution prohibited without permission.

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 69 University of California, Berkeley, May 10, 2017

Fast R-CNN



"conv5" feature map of image

Forward whole image through ConvNet

Input image

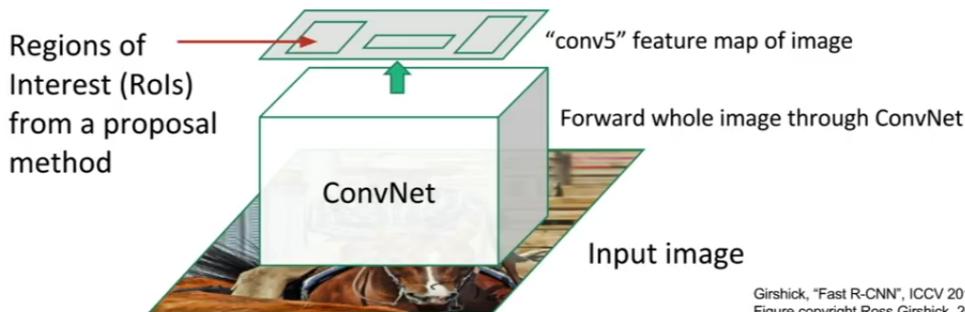
Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015. All rights reserved. Reproduction or redistribution prohibited without permission.

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 70 University of California, Berkeley, May 10, 2017

- In this technique instead of using Region Proposal generator to generate ROIs, we pass the entire image through a CNN and use Region Proposal to generate ROIs.

Fast R-CNN



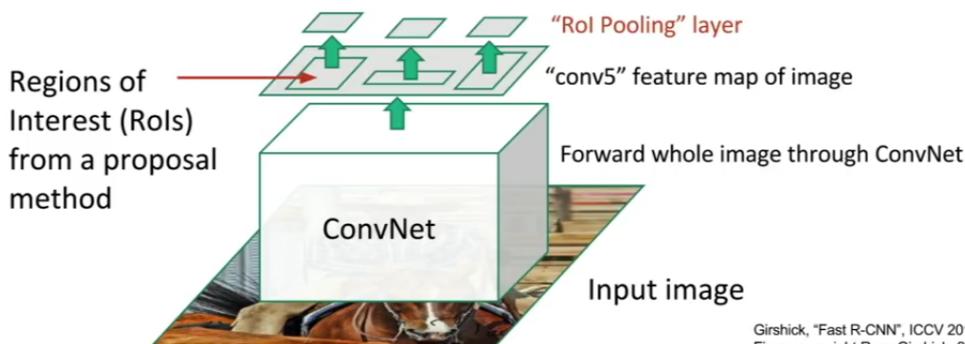
Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015. All rights reserved. Reproduction or redistribution prohibited without permission.

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 71 University May 10, 2017

- Advantage is that we are reducing the use of Convolution operation.

Fast R-CNN



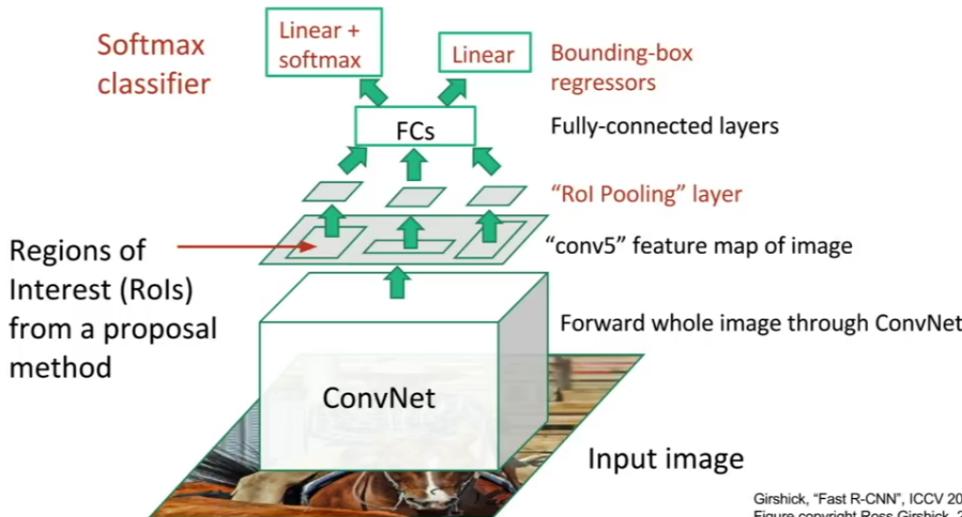
Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015. All rights reserved. Reproduction or redistribution prohibited without permission.

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 72 University May 10, 2017

- Since the ROIs are of different sizes, we use ROI Pooling to make them of same size.

Fast R-CNN



Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015. All rights reserved.

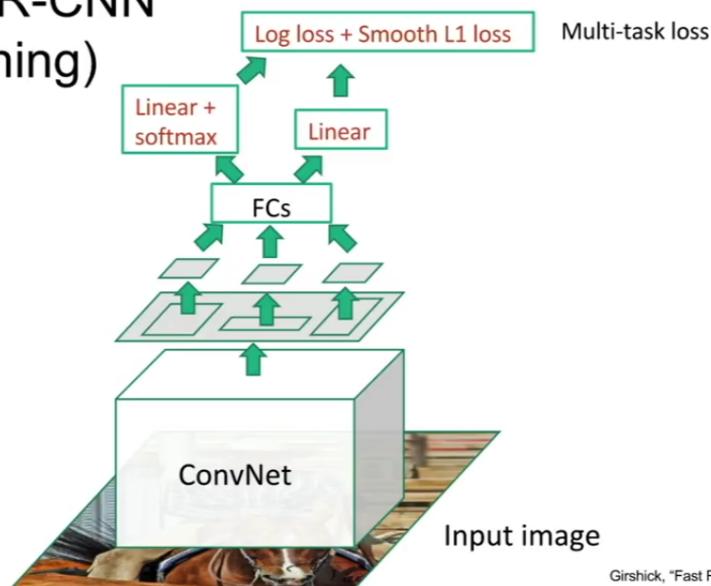
Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 74 University, May 10, 2017

-

- We can use the FC layer output for classification and regression to generate Category label and Bounding box co-ordinates.

Fast R-CNN (Training)



Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015. All rights reserved.

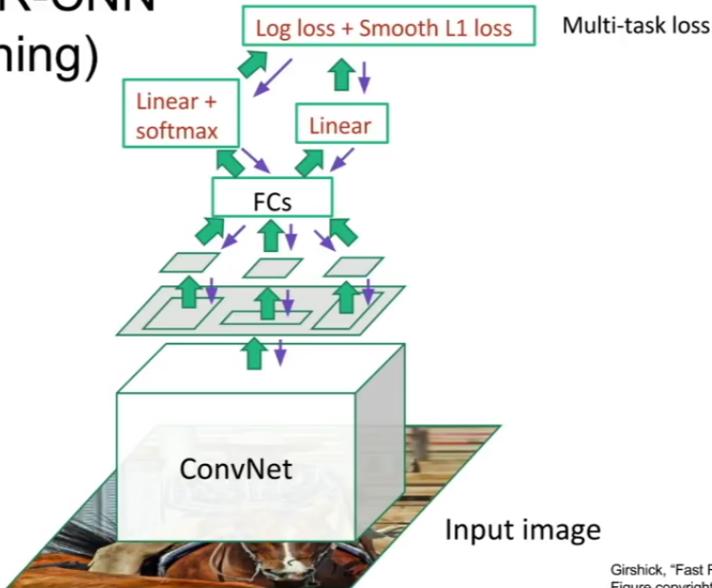
Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 75 University, May 10, 2017

-

- Since we have 2 different loss terms we combine them.

Fast R-CNN (Training)



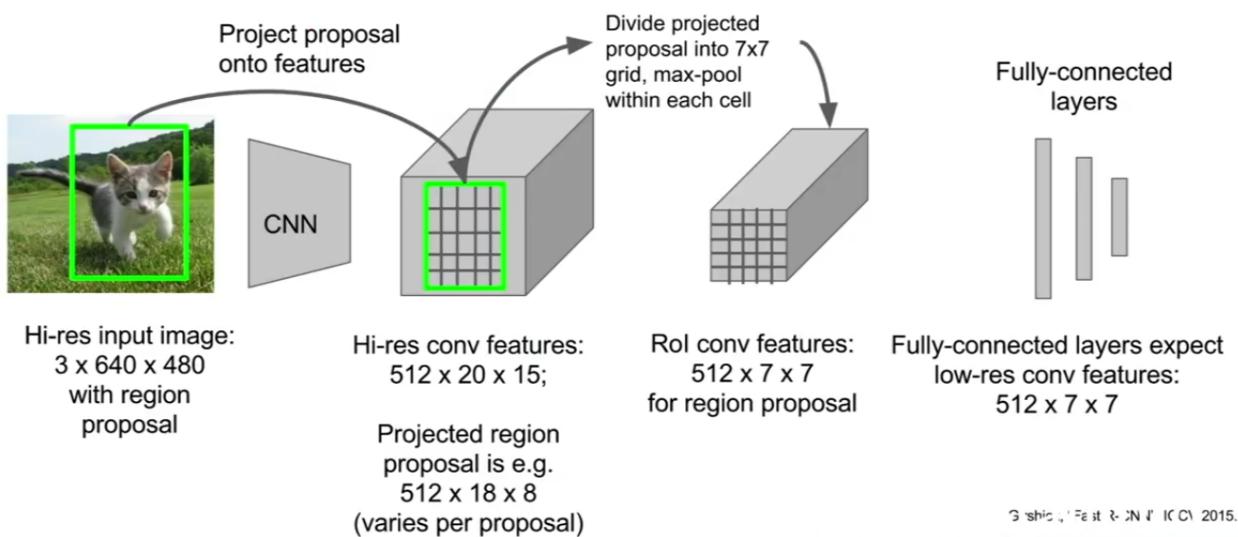
Girshick, "Fast R-CNN", ICCV 2015.
Figure copyright Ross Girshick, 2015. All rights reserved. Reproduced with permission.

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - University of California, Berkeley, May 10, 2017

- We can use this combined Multi-Task loss to backpropagate.

Faster R-CNN: ROI Pooling

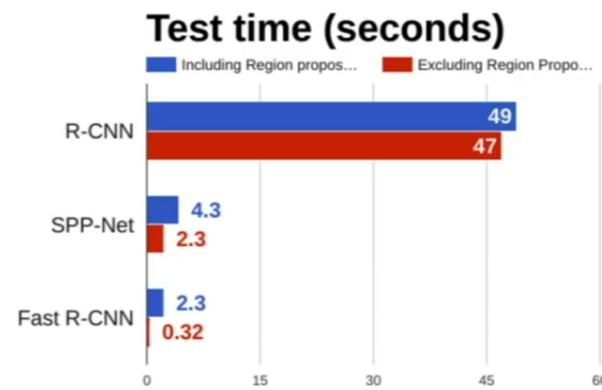


Girshick, "Faster R-CNN", ICCV 2015.

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - University of California, Berkeley, May 10, 2017

R-CNN vs SPP vs Fast R-CNN



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

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 78 University May 10, 2017

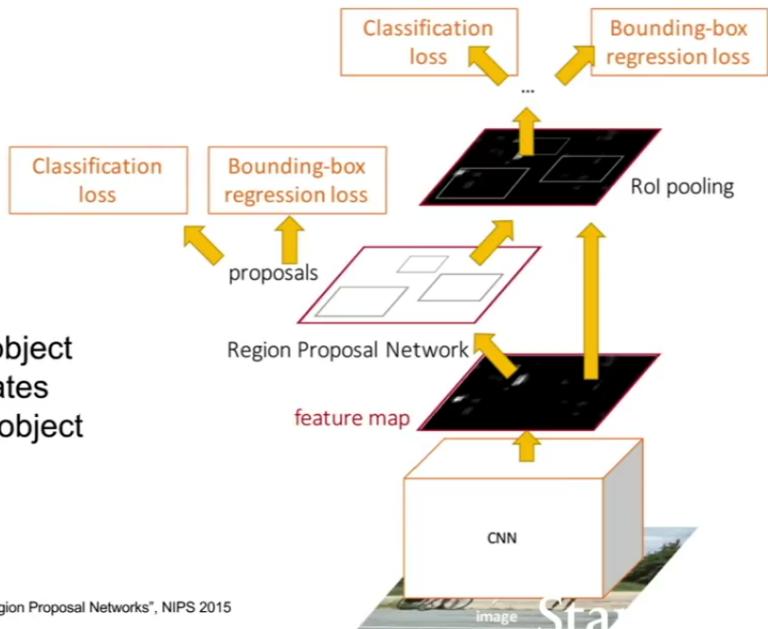
- Faster R-CNN:

Faster R-CNN: Make CNN do proposals!

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

Jointly train with 4 losses:

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



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

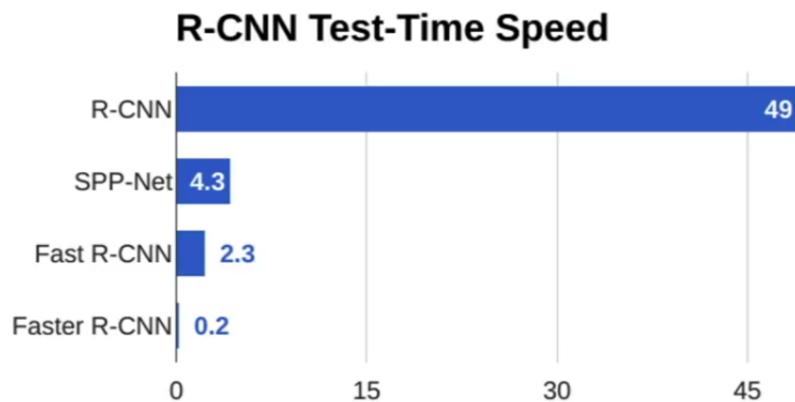
Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 80 University May 10, 2017

- Computing the Region Proposals using a fixed technique was the time consuming task in previous networks.
- So in Faster R-CNN we make the network does it region proposals.
- So here, we run the image through a CNN, and then use a higher feature map as input to the RPN to generate the ROIs.

Faster R-CNN:

Make CNN do proposals!



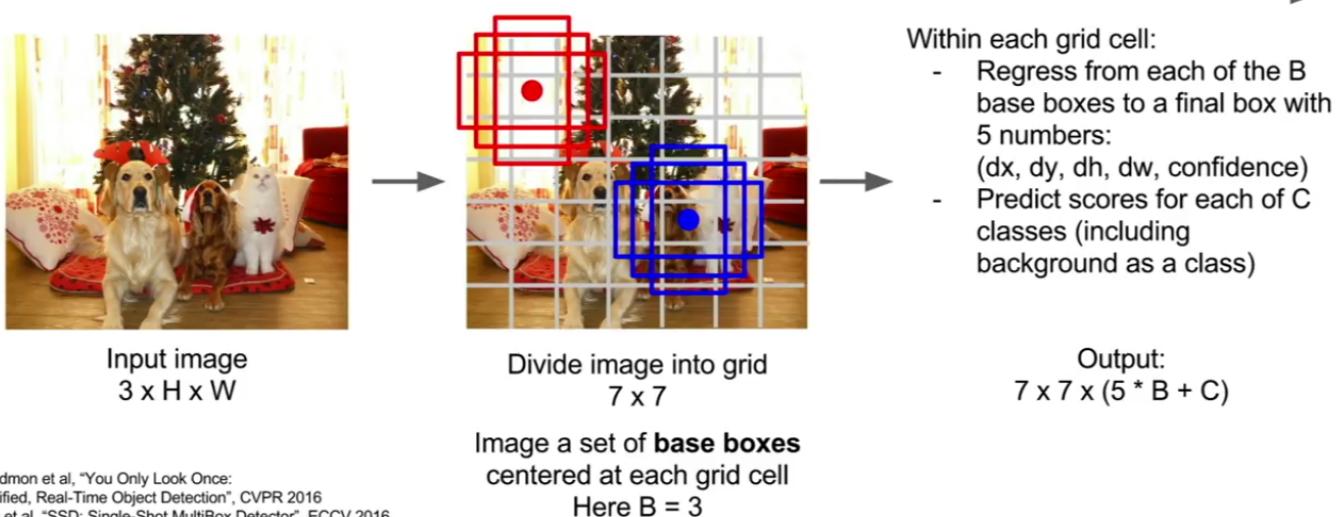
○ Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 81 University May 10, 2017

- R-CNN, Fast R-CNN, Faster R-CNN are **region-based methods**
- There is another set of approaches which are in a sense of all feed forward in a single pass.

Detection without Proposals: YOLO / SSD

Go from input image to tensor of scores with one big convolutional network!



Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 83 University, May 10, 2017

Object Detection: Lots of variables ...

Base Network

VGG16
ResNet-101
Inception V2
Inception V3
Inception
ResNet
MobileNet

Object Detection architecture

Faster R-CNN
R-FCN
SSD

Image Size
Region Proposals

...

Takeaways

Faster R-CNN is slower but more accurate

SSD is much faster but not as accurate

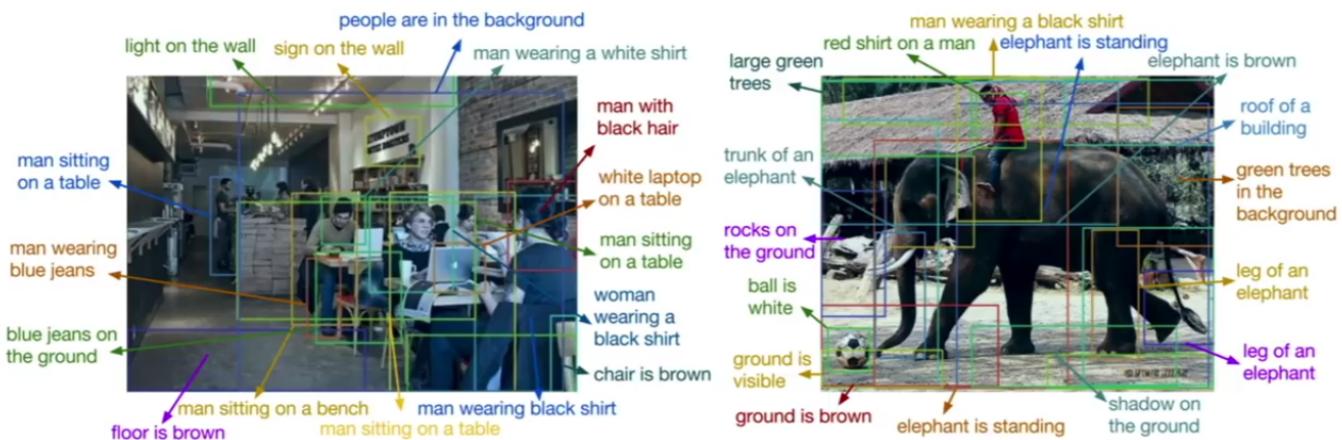
Huang et al, "Speed/accuracy trade-offs for modern convolutional object detectors", CVPR 2017

R-FCN: Dai et al, "R-FCN: Object Detection via Region-based Fully Convolutional Networks", NIPS 2016
Inception-V2: Ioffe and Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift", ICML 2015
Inception V3: Szegedy et al, "Rethinking the Inception Architecture for Computer Vision", arXiv 2016
Inception ResNet: Szegedy et al, "Inception-V4, Inception-ResNet and the Impact of Residual Connections on Learning", arXiv 2016
MobileNet: Howard et al, "Efficient Convolutional Neural Networks for Mobile Vision Applications", arXiv 2017

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 84 University, May 10, 2017

Aside: Object Detection + Captioning = Dense Captioning

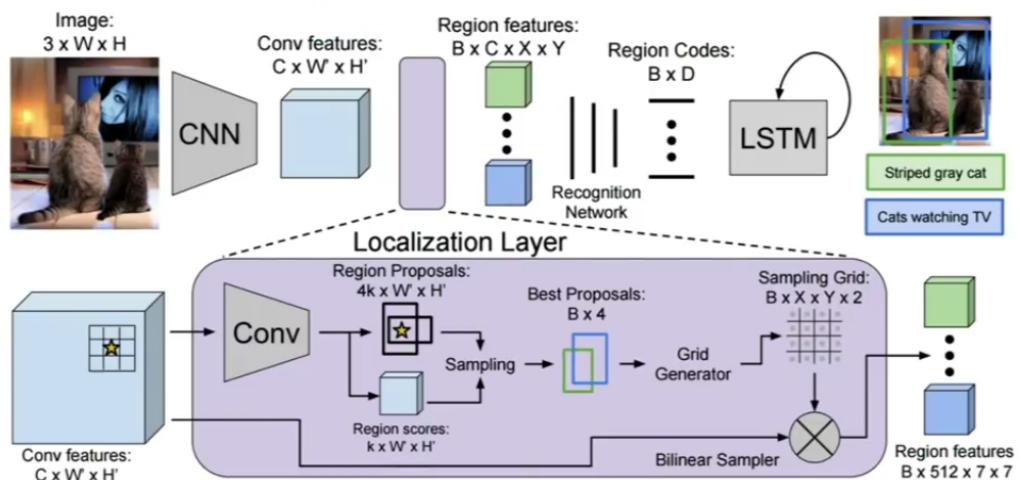


Johnson, Karpathy, and Fei-Fei, "DenseCap: Fully Convolutional Localization Networks for Dense Captioning", CVPR 2016
Figure copyright IEEE, 2016. Reproduced for educational purposes.

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 85 University, May 10, 2017

Aside: Object Detection + Captioning = Dense Captioning



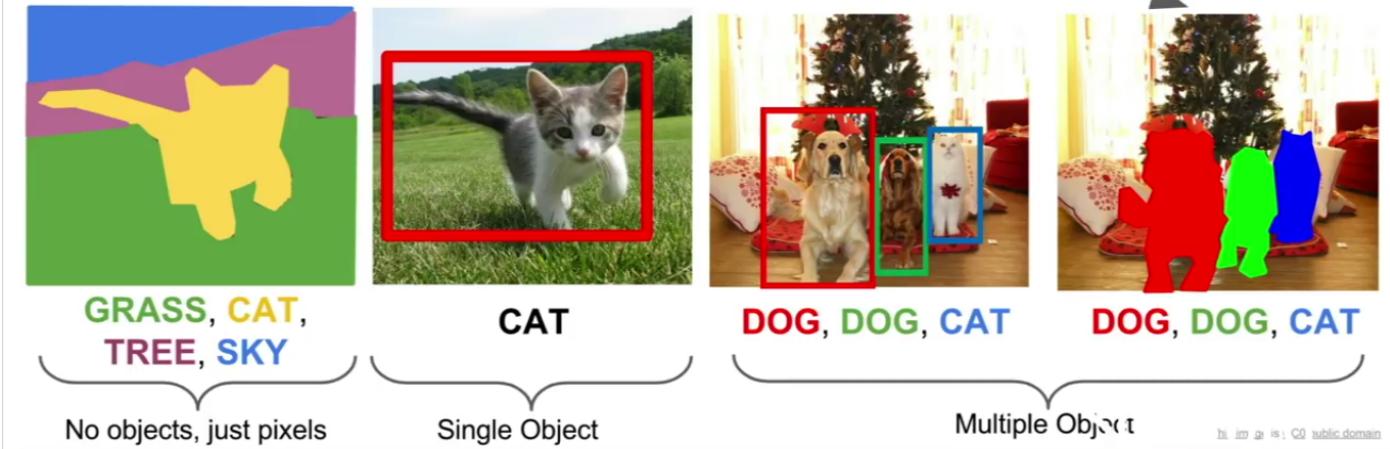
Johnson, Karpathy, and Fei-Fei, "DenseCap: Fully Convolutional Localization Networks for Dense Captioning", CVPR 2016
Figure copyright IEEE, 2016. Reproduced for educational purposes.

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 86 University, May 10, 2017

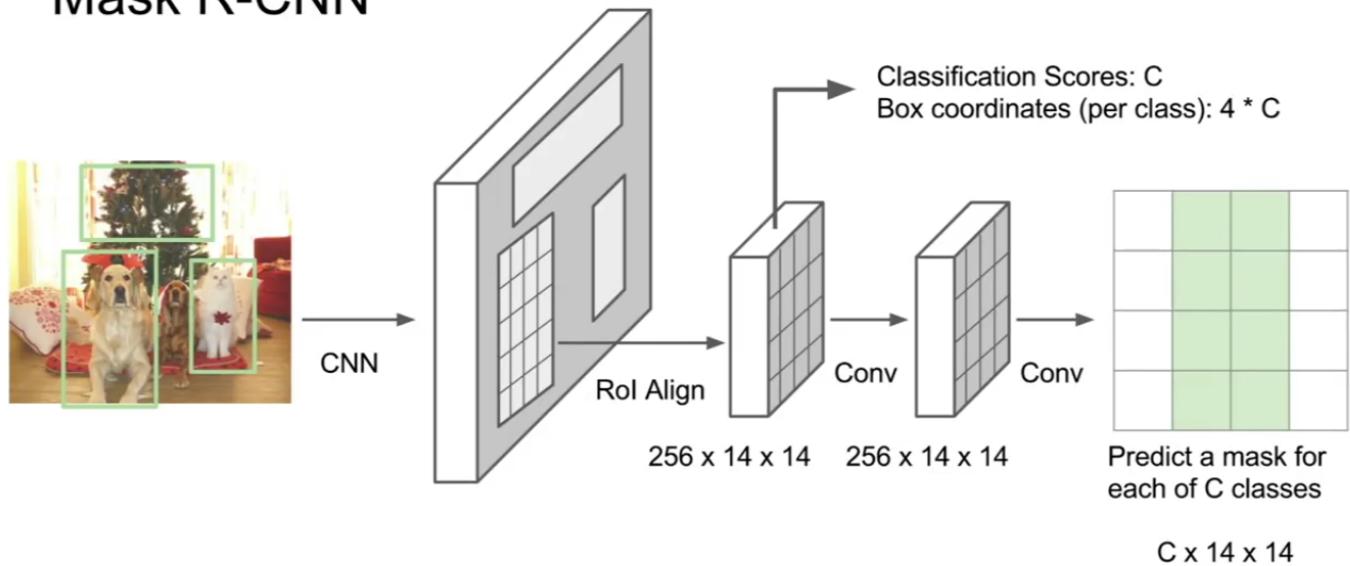
Instance Segmentation

Instance Segmentation



Lecture 11 - 88 University of California, Berkeley May 10, 2017

Mask R-CNN

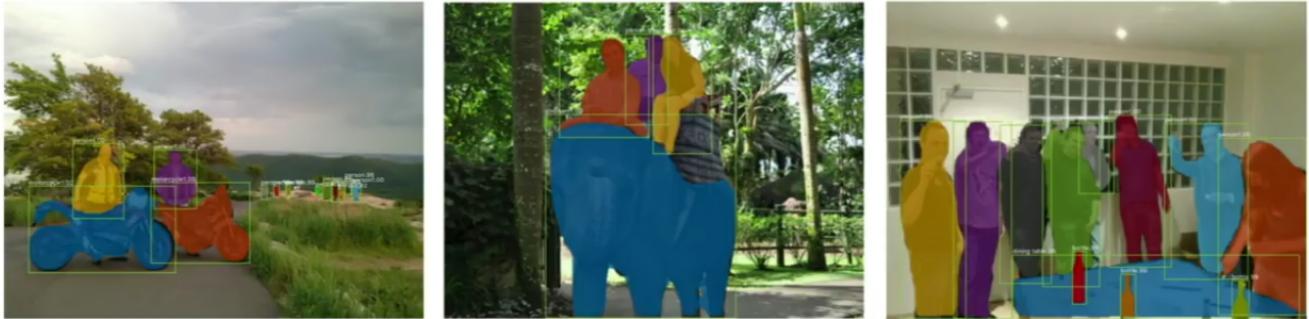


Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 89 University of California, Berkeley May 10, 2017

Mask R-CNN: Very Good Results!

Subtitle track: Disable

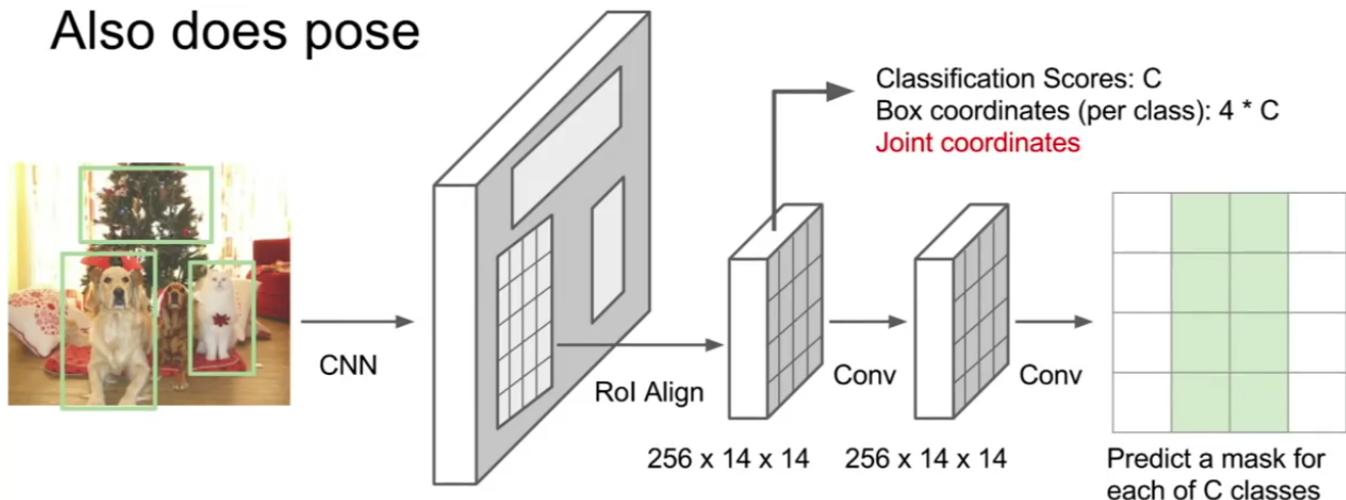


He et al, "Mask R-CNN", arXiv 2017
Figures copyright Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, 2017.
Reproduced with permission.

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 90 University, May 10, 2017

Mask R-CNN Also does pose



So we talked about, you
can do pose estimation

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

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 91 University, May 10, 2017

- To do pose detection along with classification and box coordinates we have to generate joint coordinates.

Mask R-CNN

Also does pose



He et al, "Mask R-CNN", arXiv 2017
Figures copyright Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, 2017.
Reproduced with permission.

Fei-Fei Li & Justin Johnson & Serena Yeung

Lecture 11 - 92 University, May 2017

- Mask R-CNN with pose runs at 5 fps on GPUs.