Today: Outline

- ***Advanced Topics I (Slides from Stanford CS231; Kate Saenko DL)***

- Announcements/Reminders:

  - Advanced Topics II (Mon Jun 28)
  - *"Efficient Deep Learning in Production, From Virtual Reality to Full Self-Driving" Alvin Wan, Tesla, UC Berkeley*

  - *"Training GANs to be representative of humanity" Tabitha Oanda, BU*

  - Five bonus points will be offered.

  - Problem Set 2 is graded (regrade requests accepted until Mon Jun 28)

  - Problem Set 3, due Fri Jun 25

  - Last Pre-lec Material Posted, Due Mon Jun 28

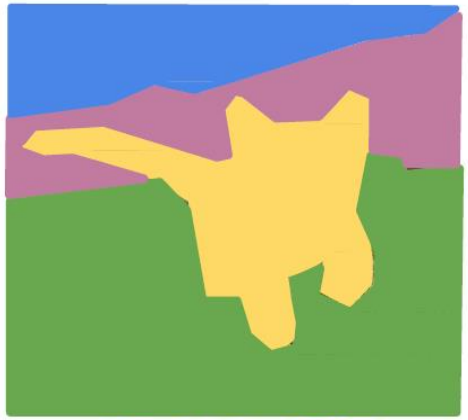
Today: Applications of CNNs to Computer Vision

Classification

CAT

No spatial extent

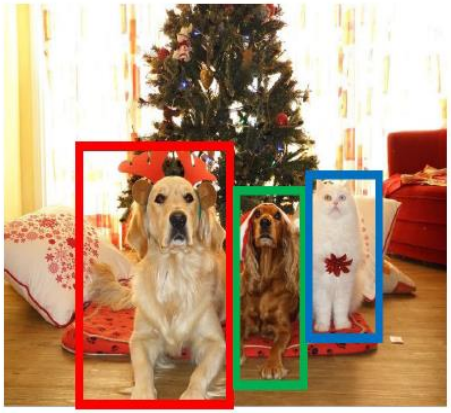
Semantic Segmentation



GRASS, CAT, TREE, SKY

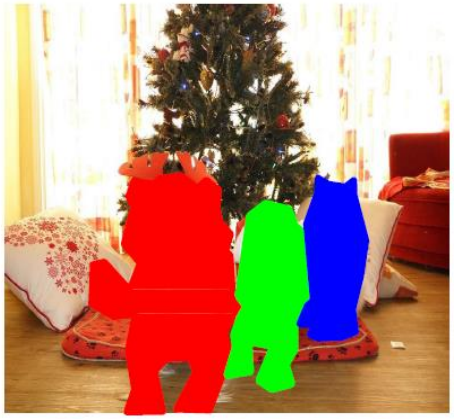
No objects, just pixels

Object Detection



DOG, DOG, CAT

Instance Segmentation



DOG, DOG, CAT

Multiple Object

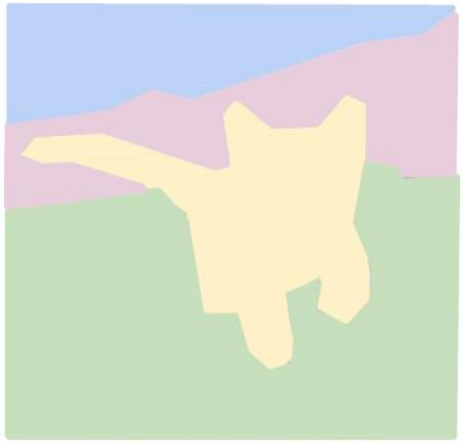
So far: CNN for classification

Classification

CAT

No spatial extent

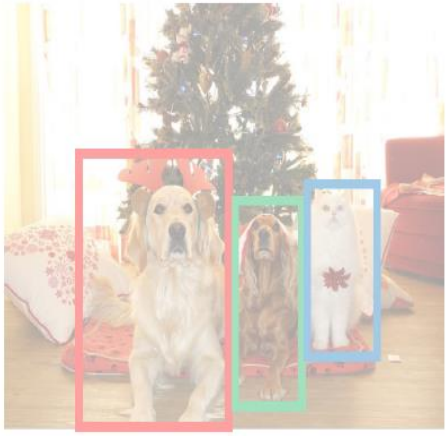
Semantic Segmentation



GRASS, CAT, TREE, SKY

No objects, just pixels

Object Detection



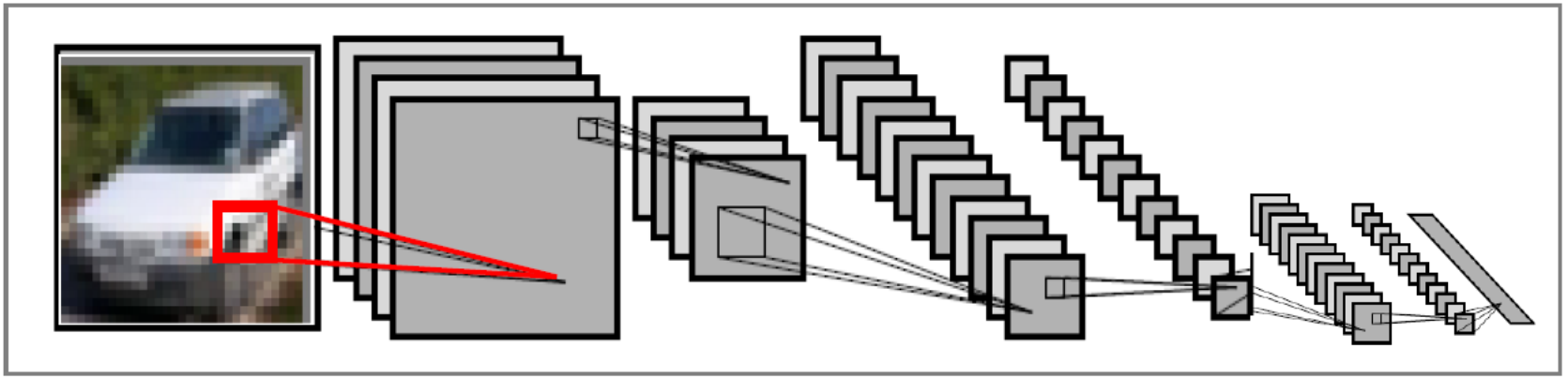
DOG, DOG, CAT

Instance Segmentation

DOG, DOG, CAT

Multiple Object

Example: CNN architecture for classification



Output:
"car"

CIFAR-10 Demo ConvJS Network

Today: Segmentation, Detection

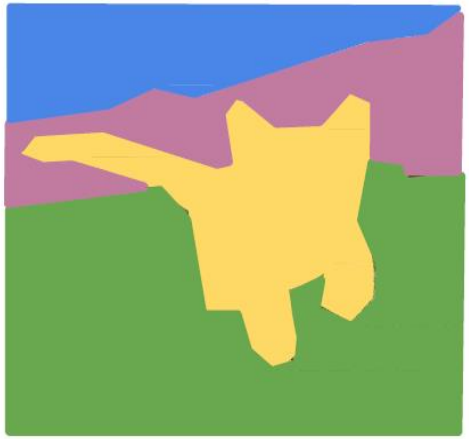
Semantic Segmentation

Classification

Semantic Segmentation

Object Detection

Instance Segmentation



CAT

GRASS, CAT, TREE, SKY

DOG, DOG, CAT

DOG, DOG, CAT

No spatial extent

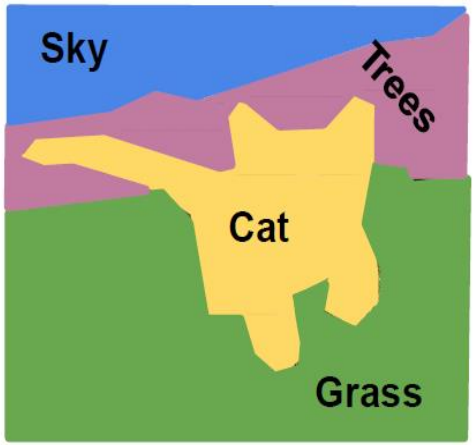
No objects, just pixels

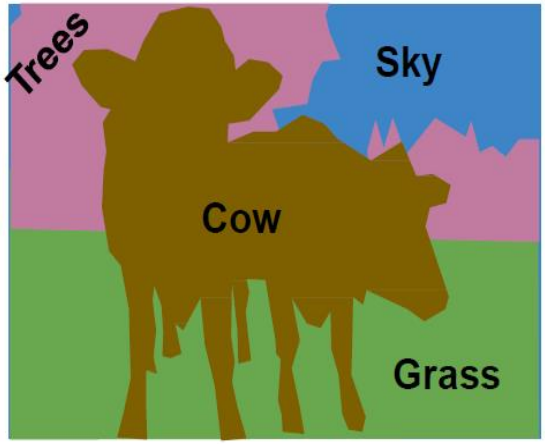
Multiple Object

Semantic Segmentation

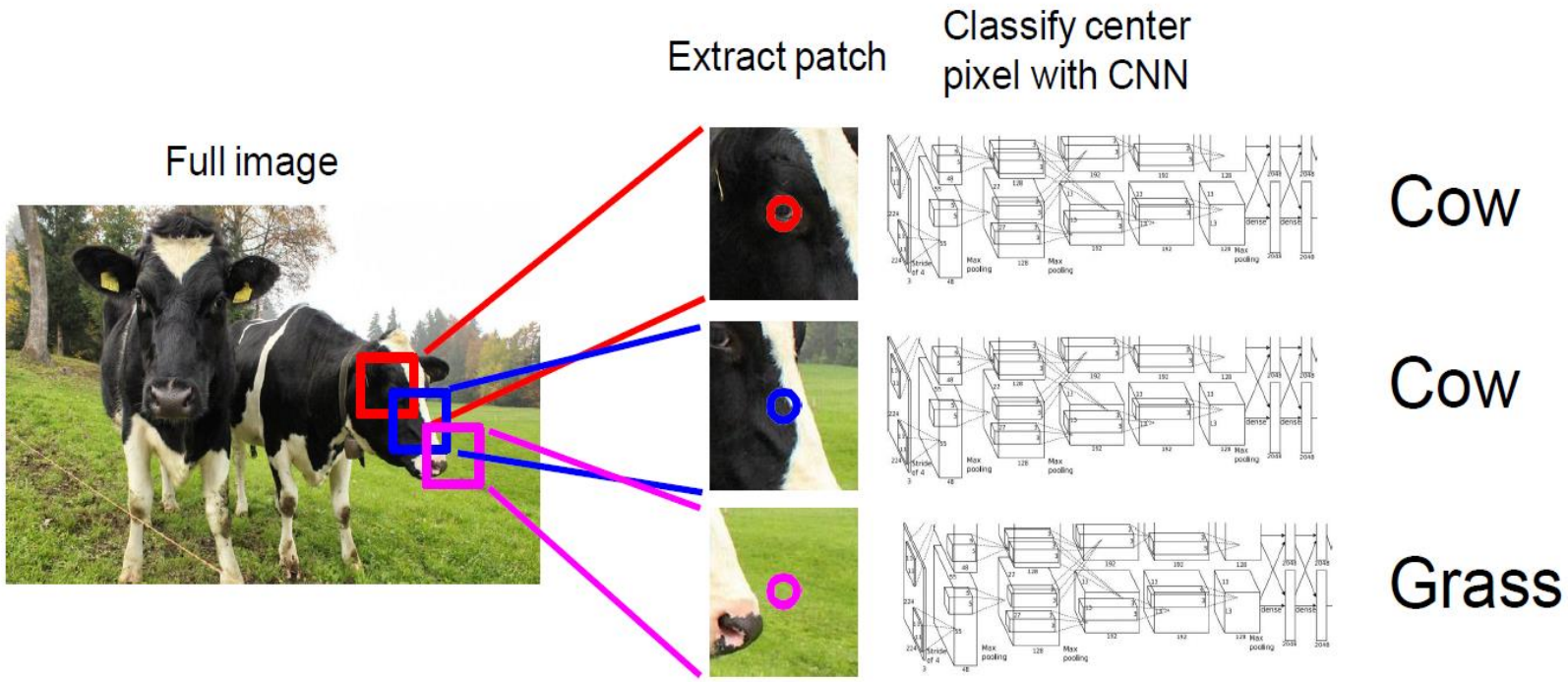
Label each pixel in the image with a category label

Don't differentiate instances, only care about pixels



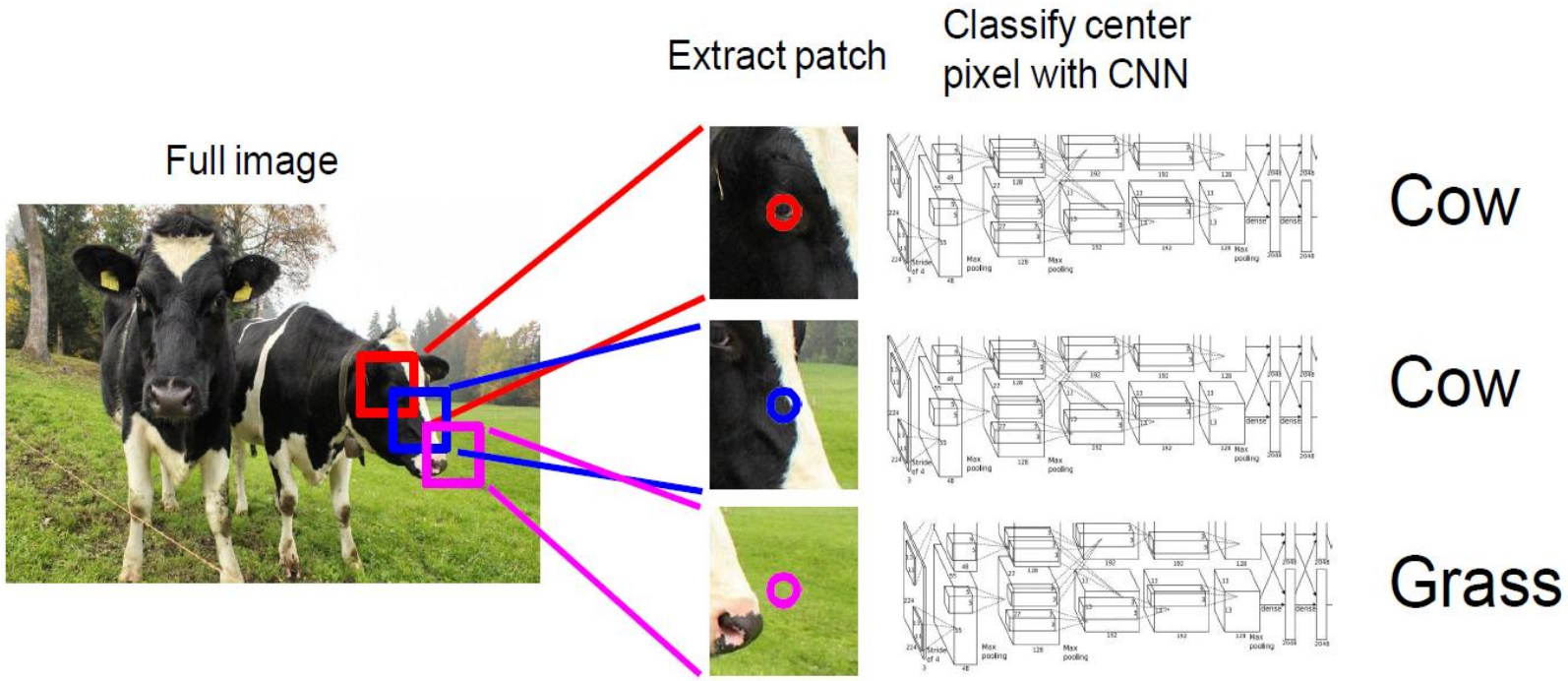


Semantic Segmentation Idea: Sliding Window



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

Semantic Segmentation Idea: Sliding Window

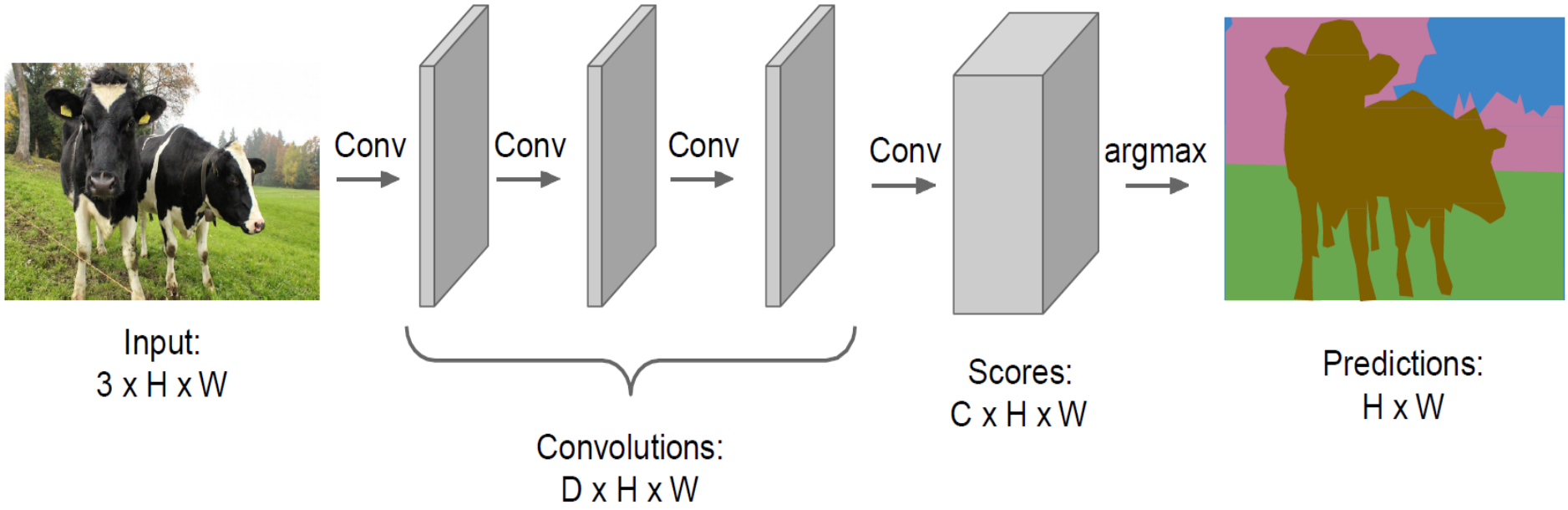


Problem: Very inefficient! Not reusing shared features between overlapping patches

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

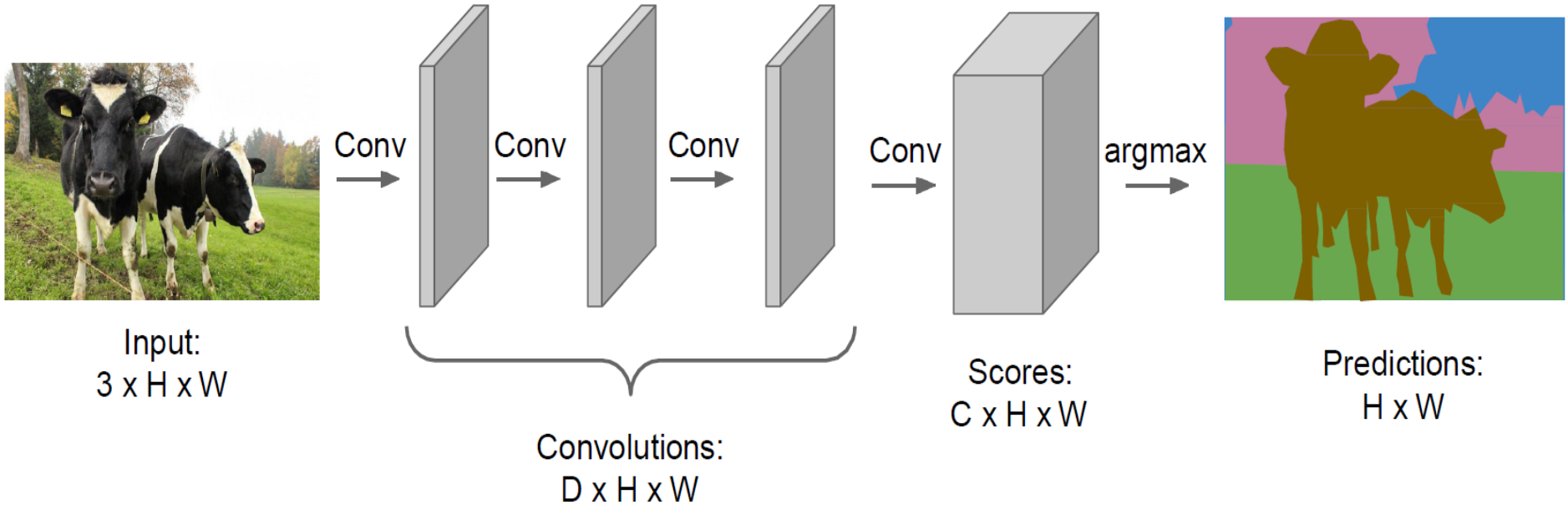
Semantic Segmentation Idea: Fully Convolutional

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



Semantic Segmentation Idea: Fully Convolutional

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

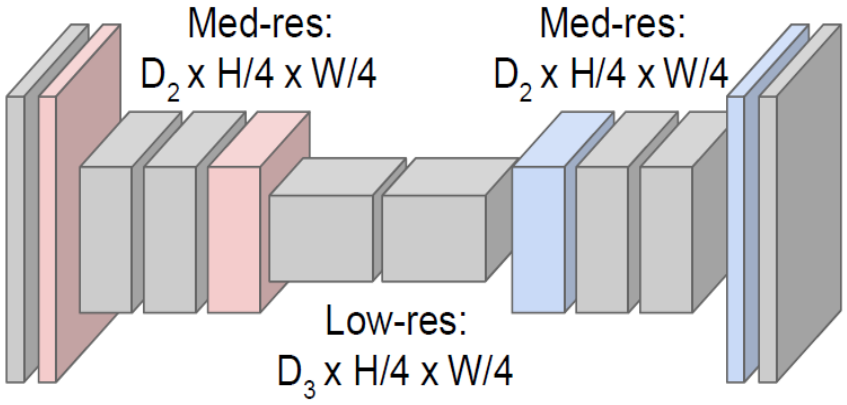


Problem: convolutions at original image resolution will be very expensive ...

Semantic Segmentation Idea: Fully Convolutional

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

Input:
3 x H x W



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

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

Predictions:
H x W

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

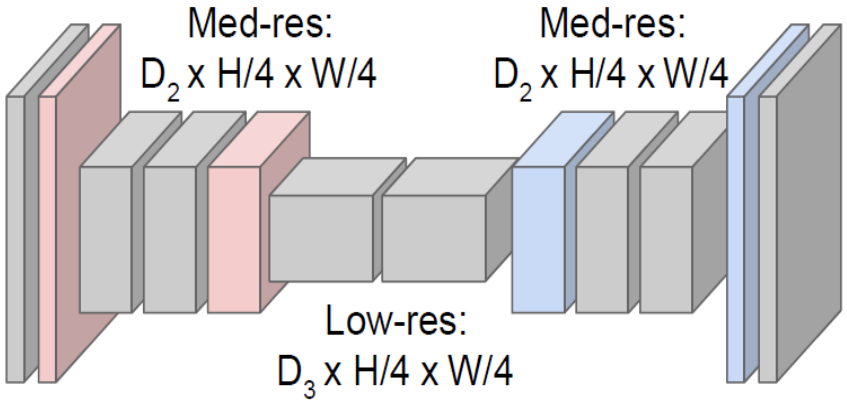
Semantic Segmentation Idea: Fully Convolutional

Downsampling:
Pooling, strided convolution

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

Upsampling:
???

Input:
3 x H x W



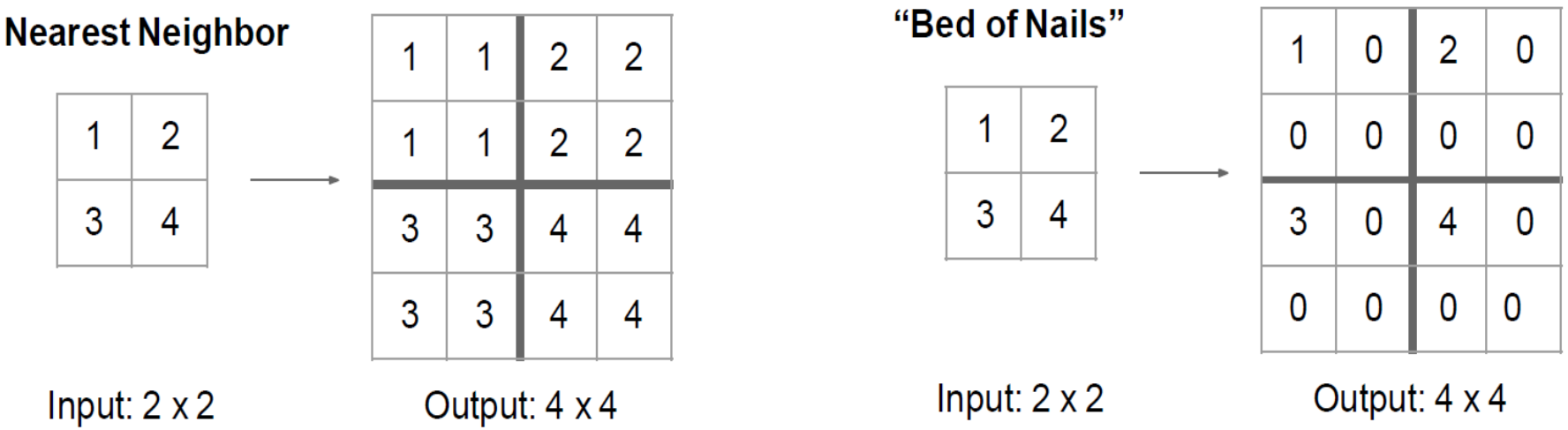
High-res:
D_1 x H/2 x W/2

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

Predictions:
H x W

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

In-Network upsampling: “Unpooling”



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

. . .

Rest of the network

Max Unpooling
Use positions from pooling layer

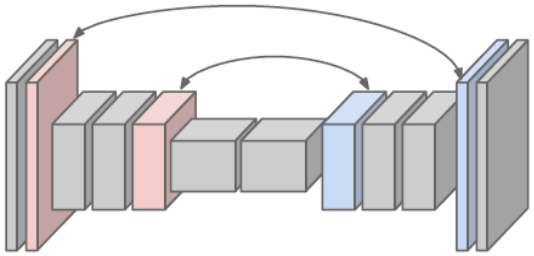
1	2
3	4

Input: 2 x 2

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

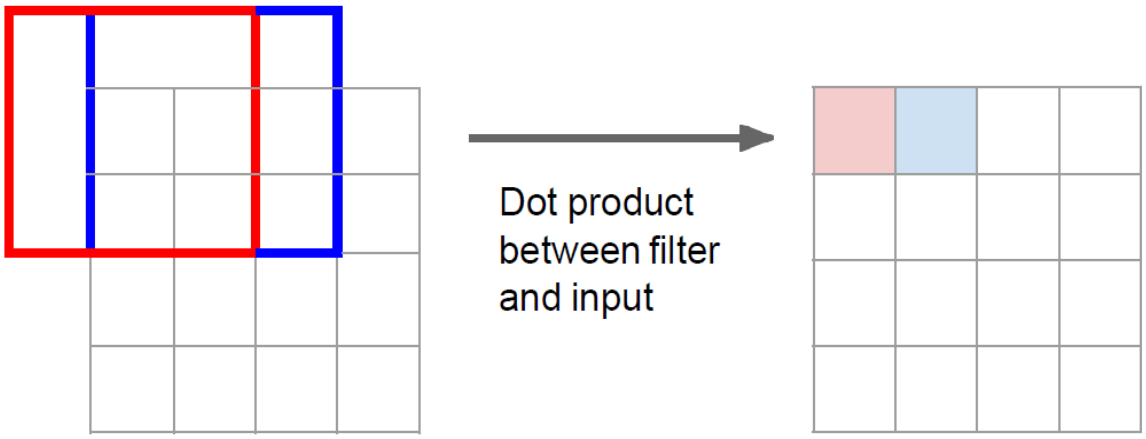
Output: 4 x 4

Corresponding pairs of downsampling and upsampling layers



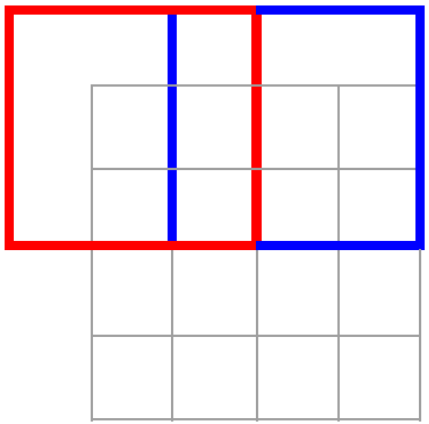
Learnable Upsampling: Transpose Convolution

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



Learnable Upsampling: Transpose Convolution

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



Input: 4 x 4

Dot product between filter and input

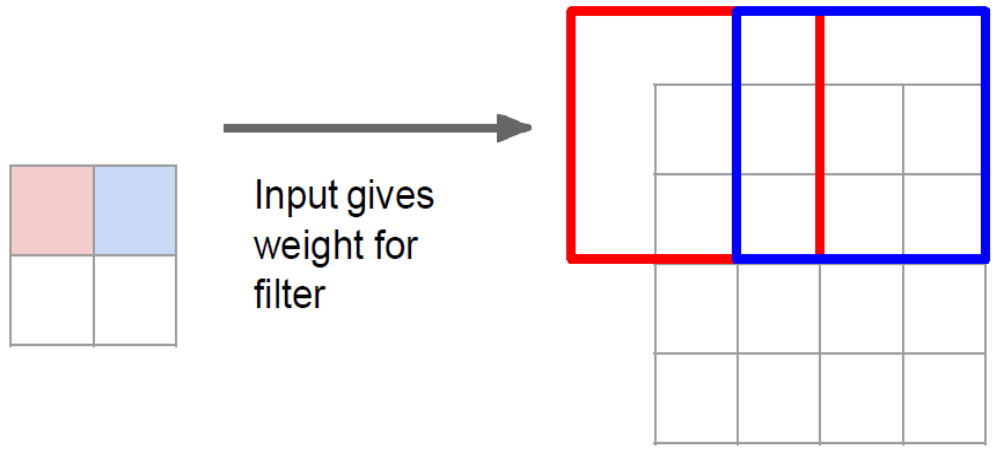
Output: 2 x 2

Filter moves 2 pixels in the input for every one pixel in the output

Stride gives ratio between movement in input and output

Learnable Upsampling: Transpose Convolution

3 x 3 **transpose** convolution, stride 2 pad 1



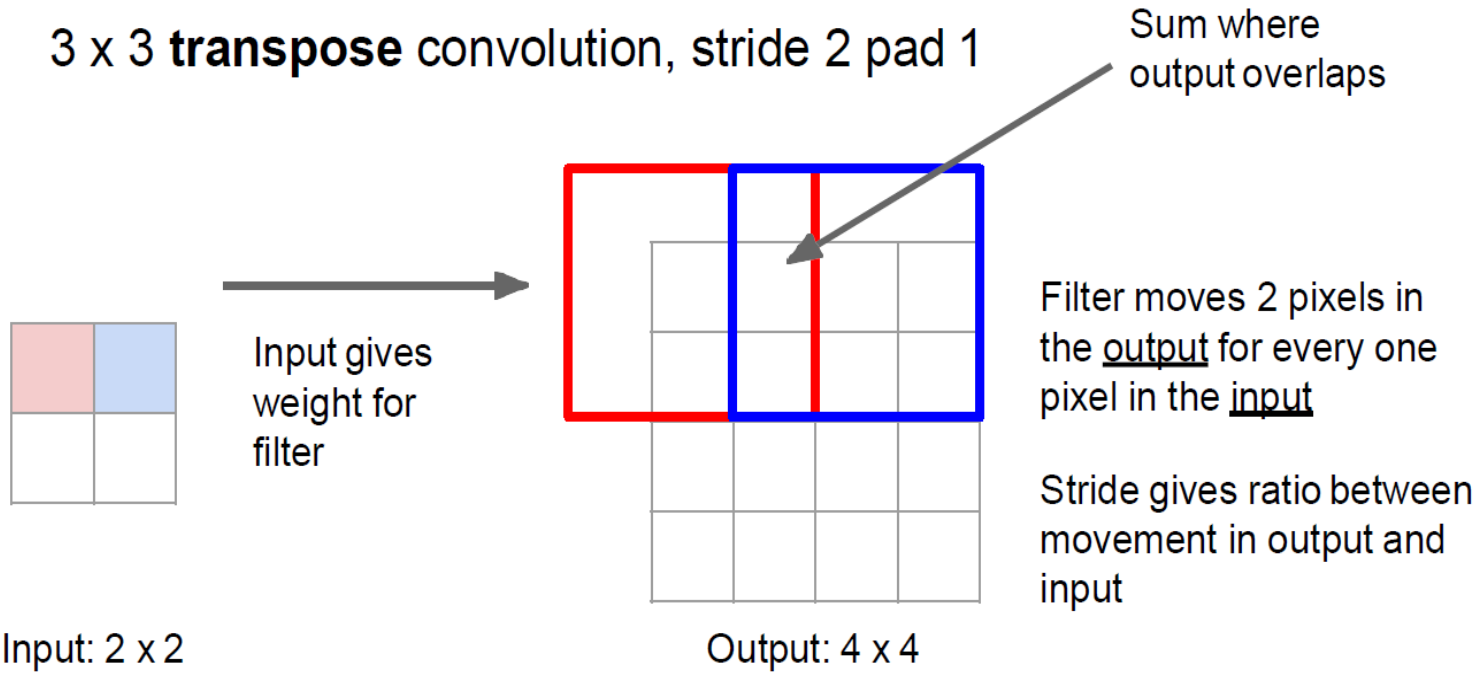
Filter moves 2 pixels in the output for every one pixel in the input

Stride gives ratio between movement in output and input

Learnable Upsampling: Transpose Convolution

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

3 x 3 **transpose** convolution, stride 2 pad 1



Object Detection

Classification

CAT

Semantic Segmentation

GRASS, CAT, TREE, SKY

Object Detection

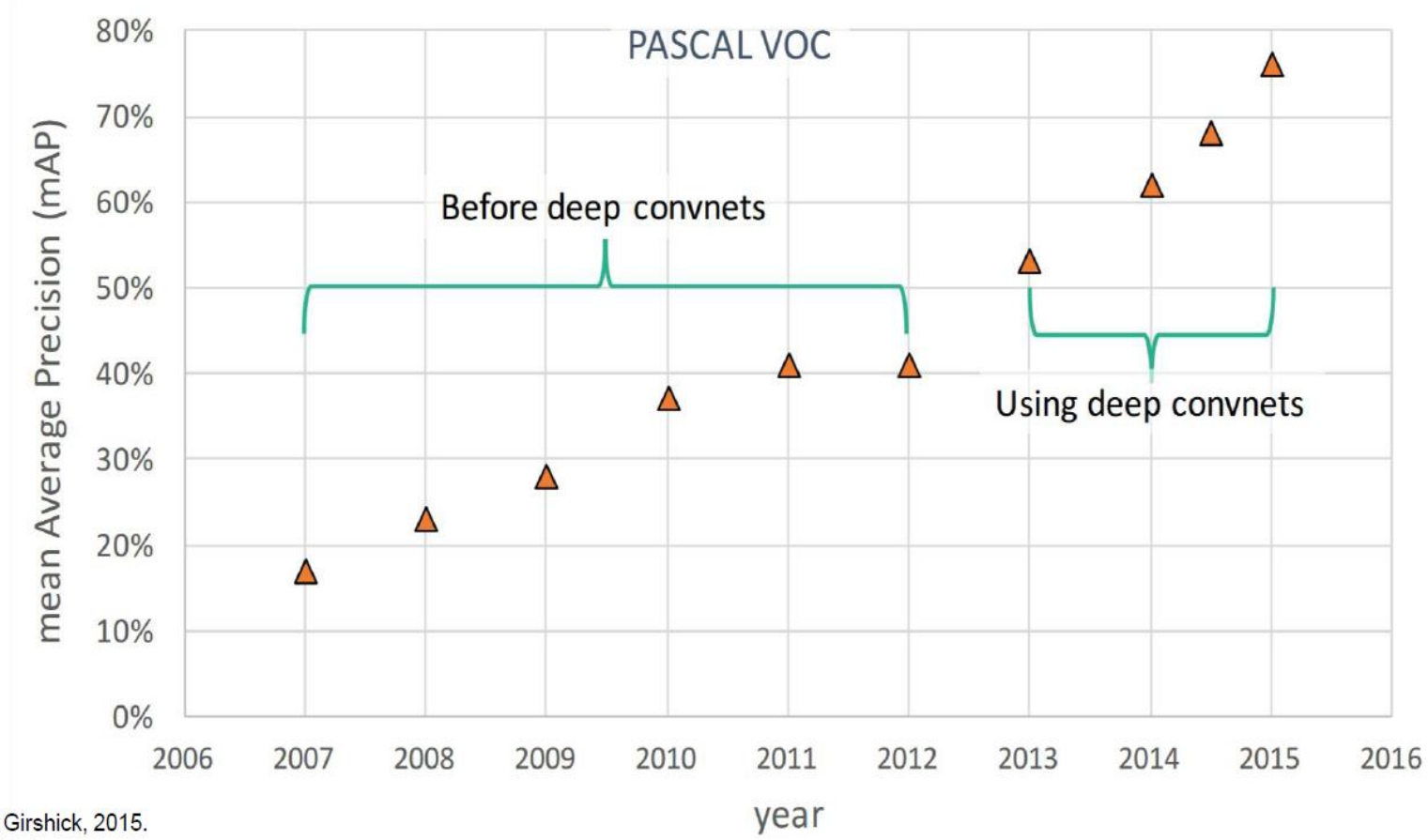
DOG, DOG, CAT

Instance Segmentation

DOG, DOG, CAT

Multiple Object

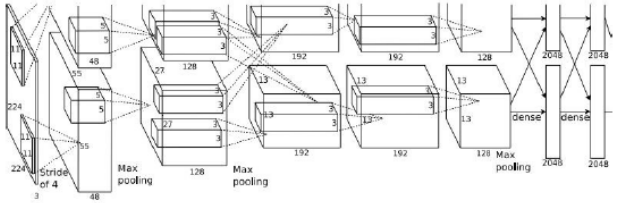
Object Detection: Impact of Deep Learning



Object Detection: Single Object

(Classification + Localization)

This image is CC0 public domain



Fully Connected: 4096 to 1000

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

Vector: 4096

Fully Connected: 4096 to 4

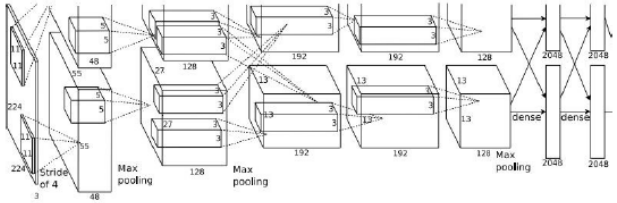
Box Coordinates
(x, y, w, h)

Treat localization as a regression problem!

Object Detection: Single Object

(Classification + Localization)

This image is CC0 public domain



Fully Connected: 4096 to 1000

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

Softmax Loss

Correct label: Cat

Vector: 4096

Fully Connected: 4096 to 4

Box Coordinates (x, y, w, h)

L2 Loss

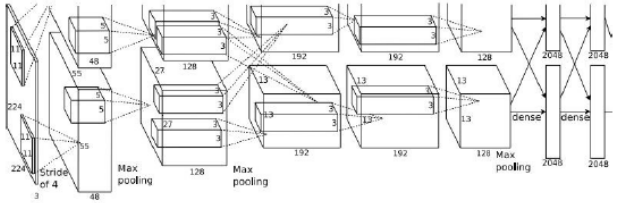
Correct box: (x', y', w', h')

Treat localization as a regression problem!

Object Detection: Single Object

(Classification + Localization)

This image is CC0 public domain



Vector: 4096

Fully Connected: 4096 to 1000

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

Softmax Loss

Correct label: Cat

Multitask Loss

+ → Loss

Fully Connected: 4096 to 4

Box Coordinates (x, y, w, h)

L2 Loss

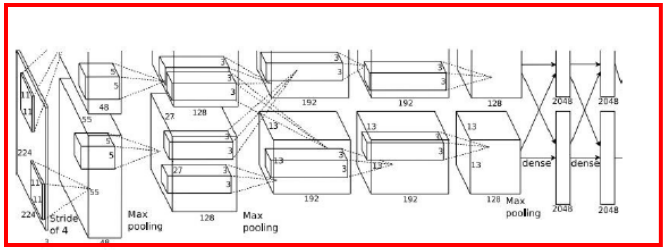
Correct box: (x', y', w', h')

Treat localization as a regression problem!

Object Detection: Single Object

(Classification + Localization)

This image is CC0 public domain



Often pre-trained on ImageNet (Transfer Learning)

Vector: 4096

Fully Connected: 4096 to 1000

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

Softmax Loss

Correct label: Cat

Multitask Loss

Loss

Fully Connected: 4096 to 4

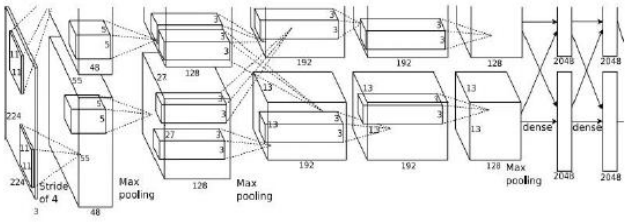
Box Coordinates (x, y, w, h)

L2 Loss

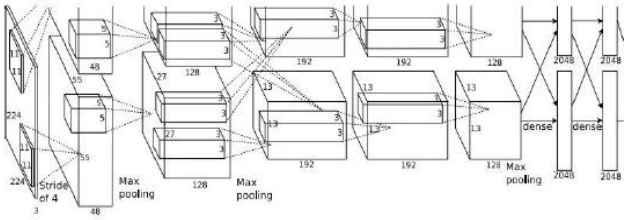
Correct box: (x', y', w', h')

Treat localization as a regression problem!

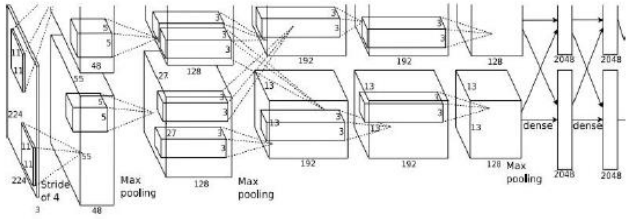
Object Detection: Multiple Objects



CAT: (x, y, w, h)



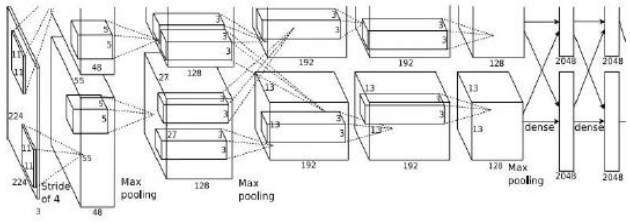
DOG: (x, y, w, h)
DOG: (x, y, w, h)
CAT: (x, y, w, h)



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

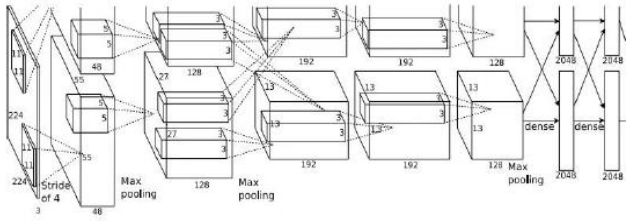
Object Detection: Multiple Objects

Each image needs a different number of outputs!



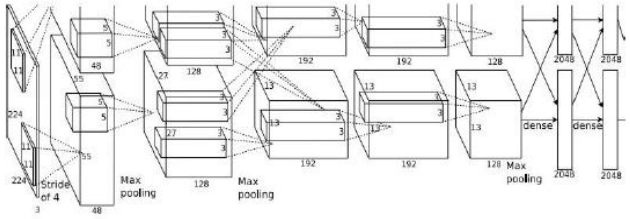
CAT: (x, y, w, h)

4 numbers



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

16 numbers

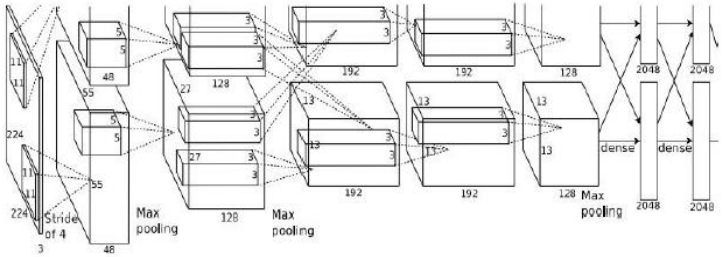


DUCK: (x, y, w, h)
DUCK: (x, y, w, h)
….

Many numbers!

Object Detection: Multiple Objects

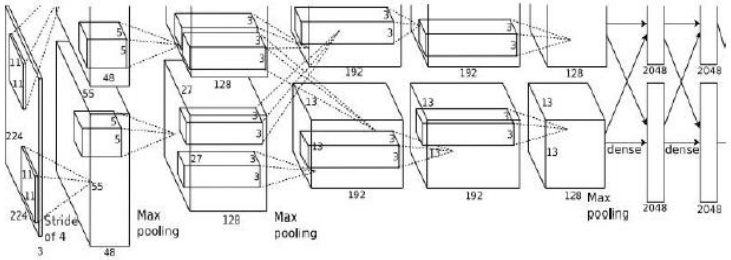
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background



Dog? NO
Cat? NO
Background? YES

Object Detection: Multiple Objects

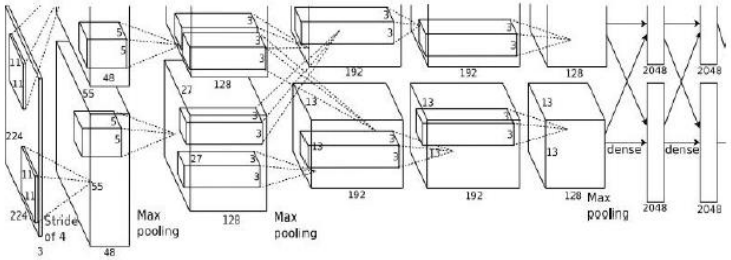
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background



Dog? YES
Cat? NO
Background? NO

Object Detection: Multiple Objects

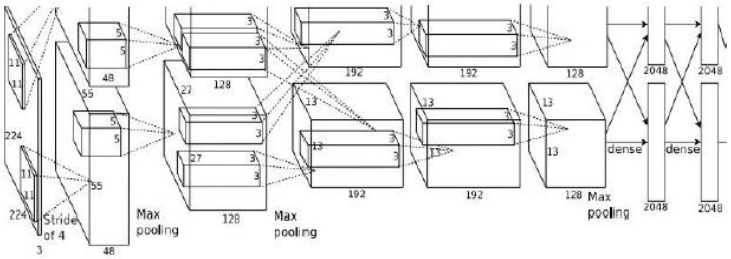
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background



Dog? YES
Cat? NO
Background? NO

Object Detection: Multiple Objects

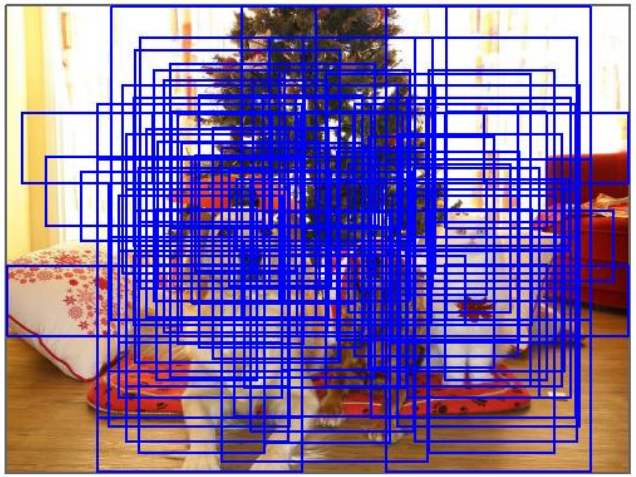
Apply a CNN to many different crops of the image, CNN classifies each crop as object or background

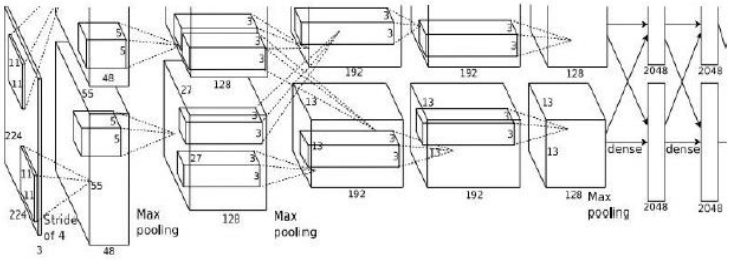


Dog? NO
Cat? YES
Background? NO

Object Detection: Multiple Objects

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



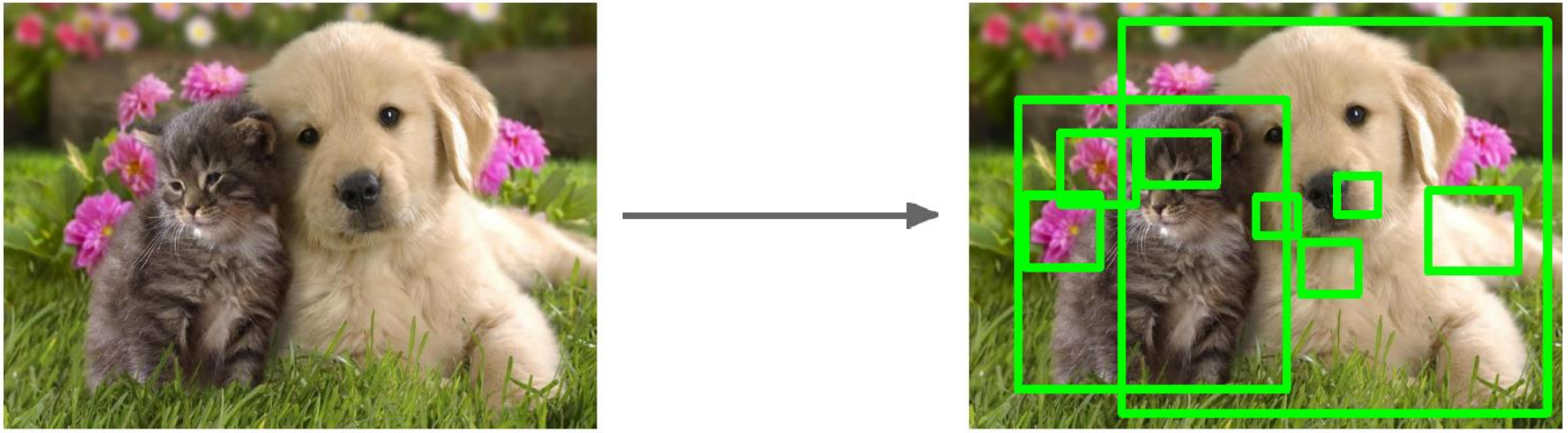


Dog? NO
Cat? YES
Background? NO

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

Region Proposals: Selective Search

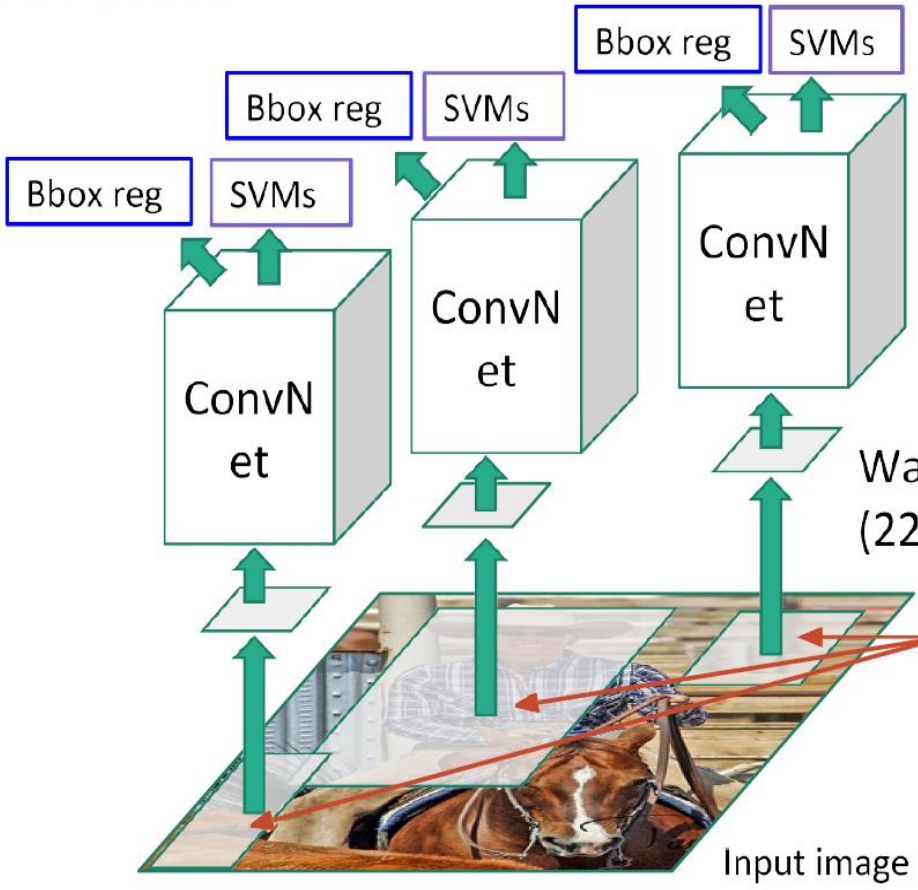
- Find "blobby" image regions that are likely to contain objects
- Relatively fast to run; e.g. Selective Search gives 2000 region proposals in a few seconds on CPU



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

R-CNN

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



Classify regions with SVMs

Forward each region through ConvNet

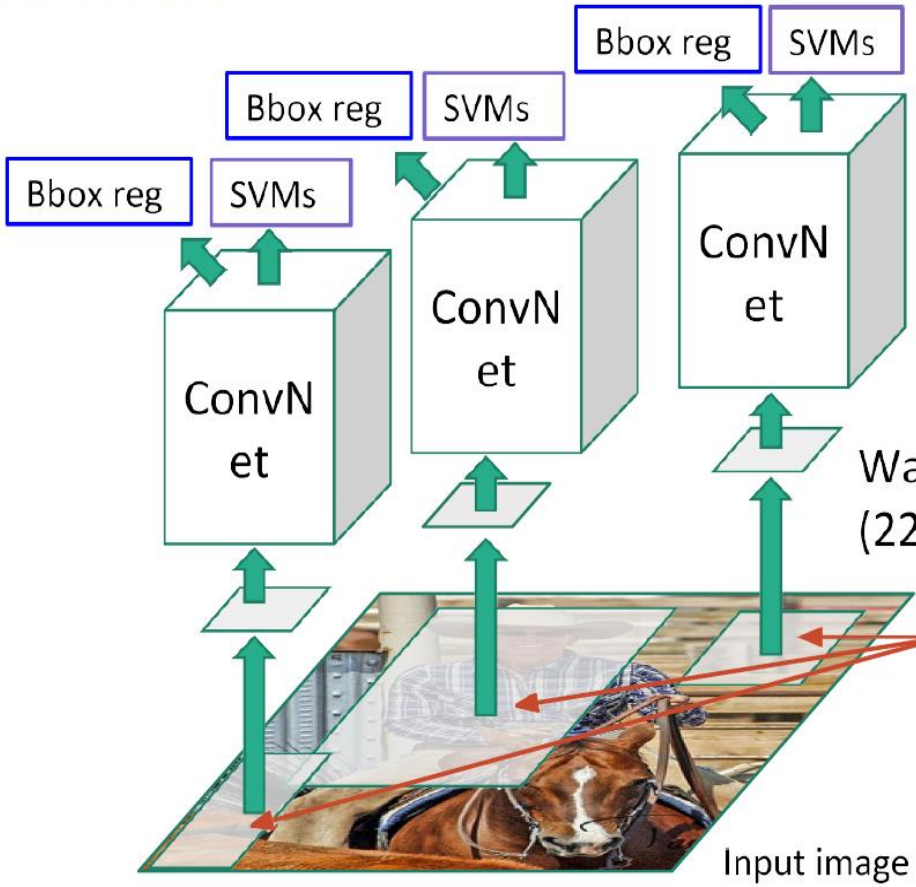
Warped image regions (224x224 pixels)

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

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

R-CNN

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



Classify regions with SVMs

Forward each region through ConvNet

Warped image regions (224x224 pixels)

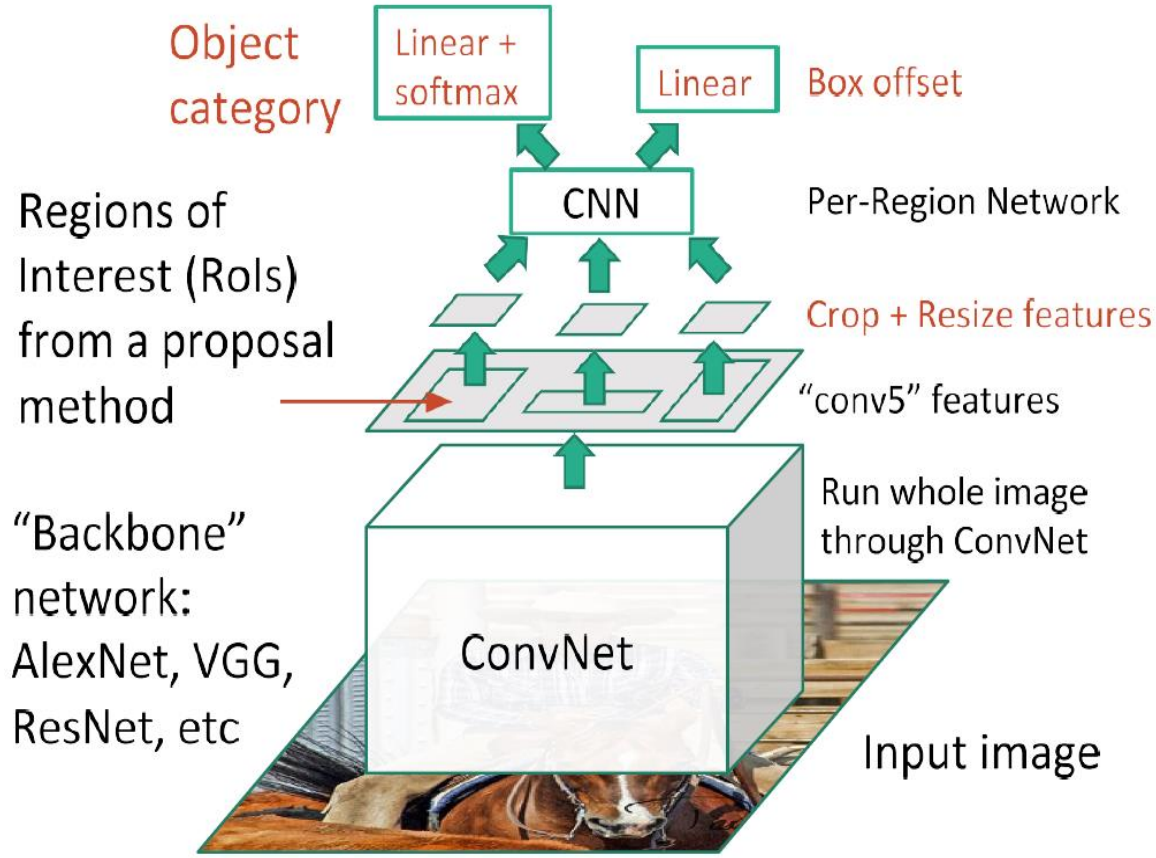
Regions of Interest (RoI) from a proposal method (~2k)

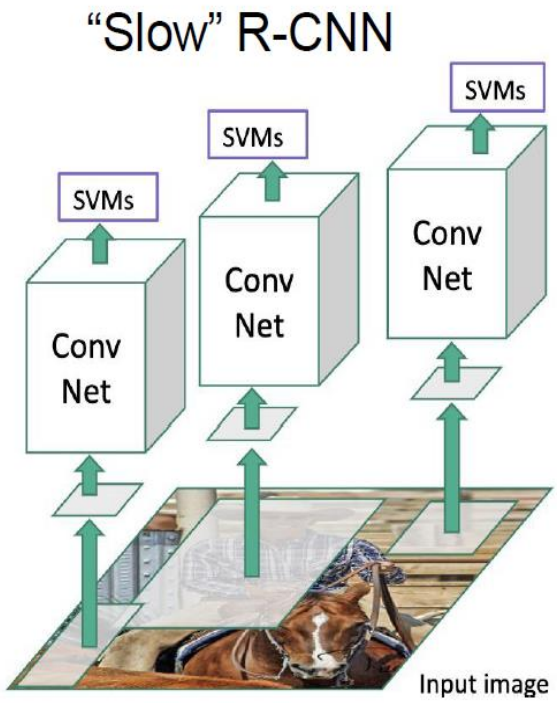
Problem: Very slow! Need to do ~2k independent forward passes for each image!

Idea: Process image before cropping! Swap convolution and cropping!

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

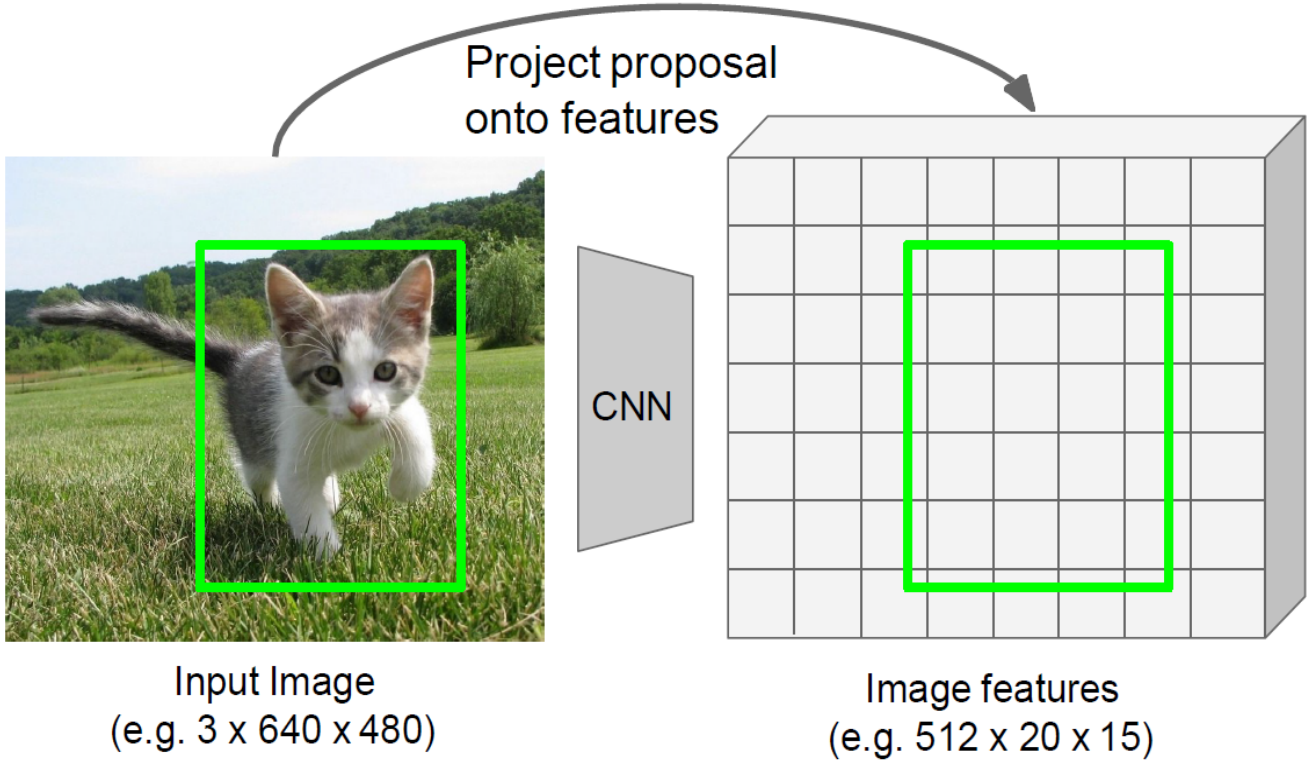
Fast R-CNN





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

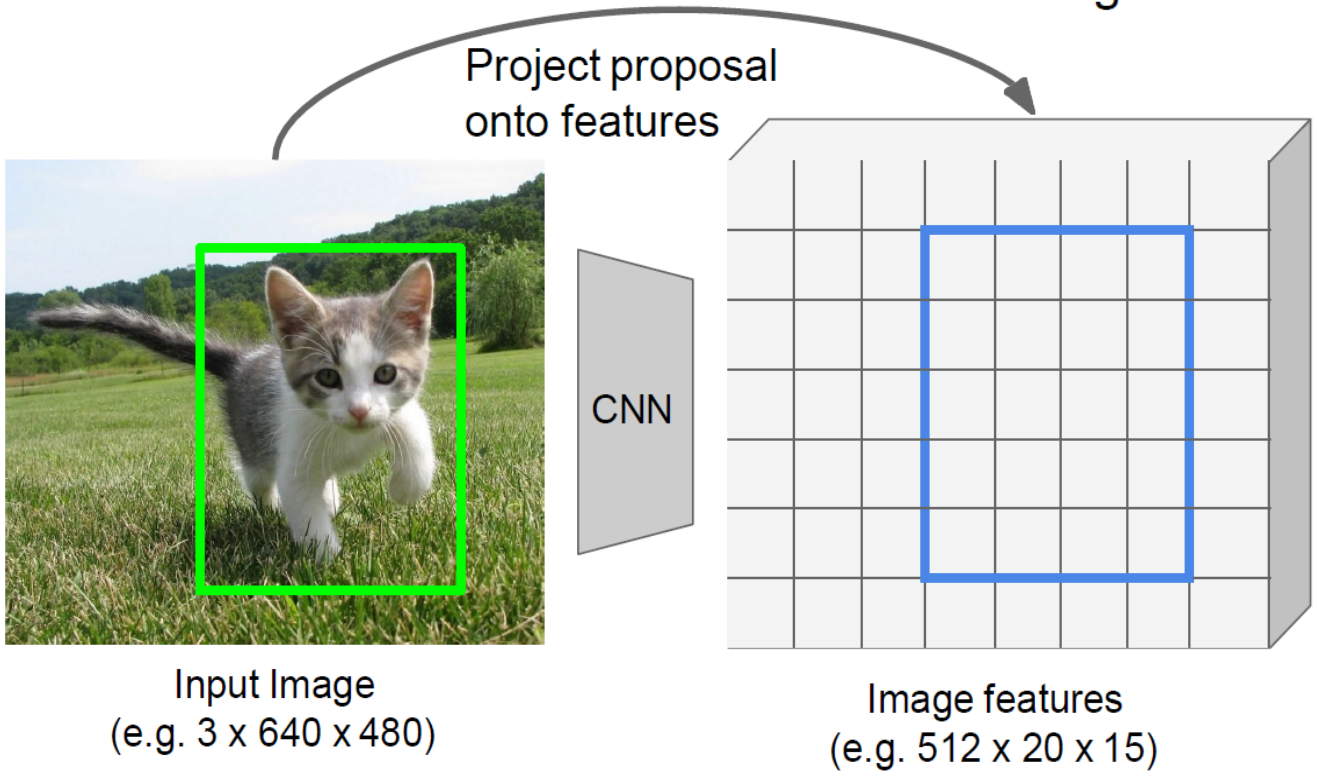
Cropping Features: RoI Pool



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

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

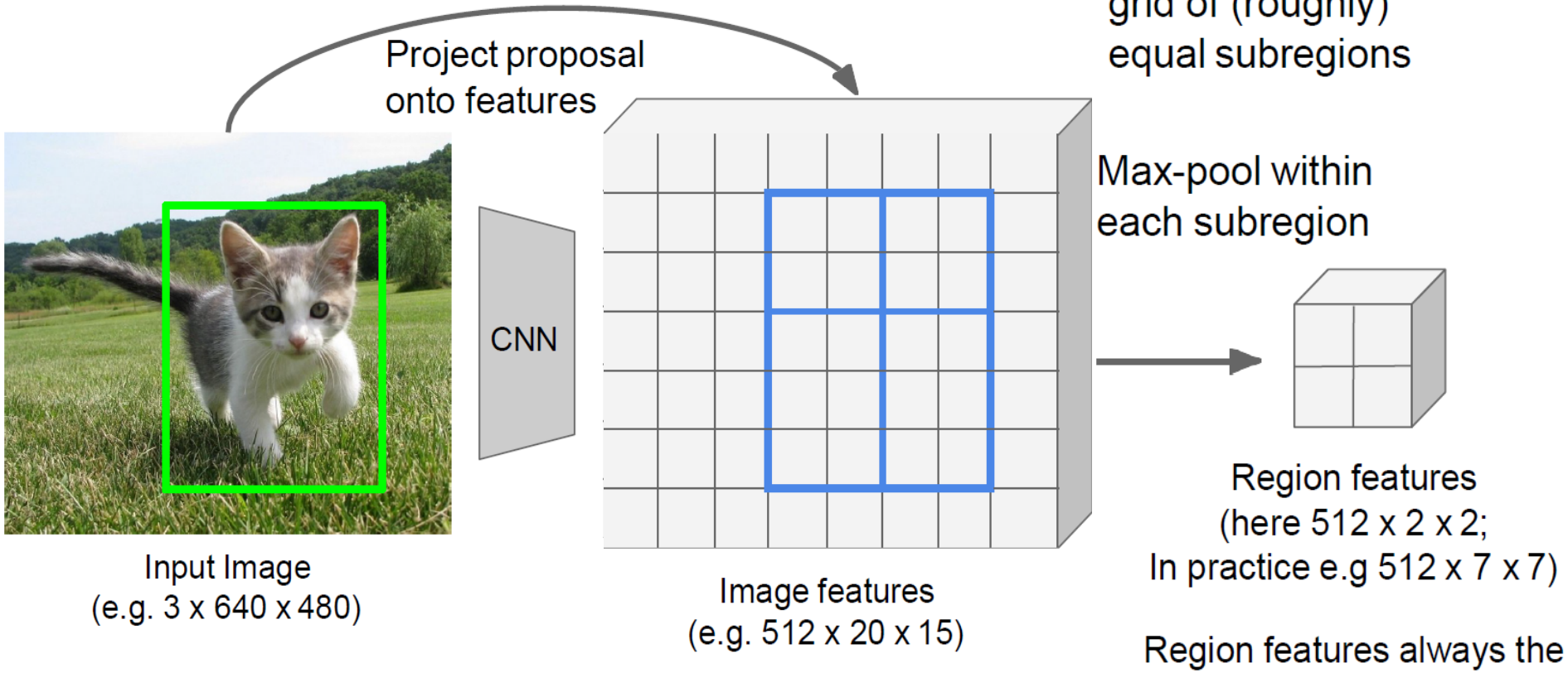
Cropping Features: RoI Pool



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

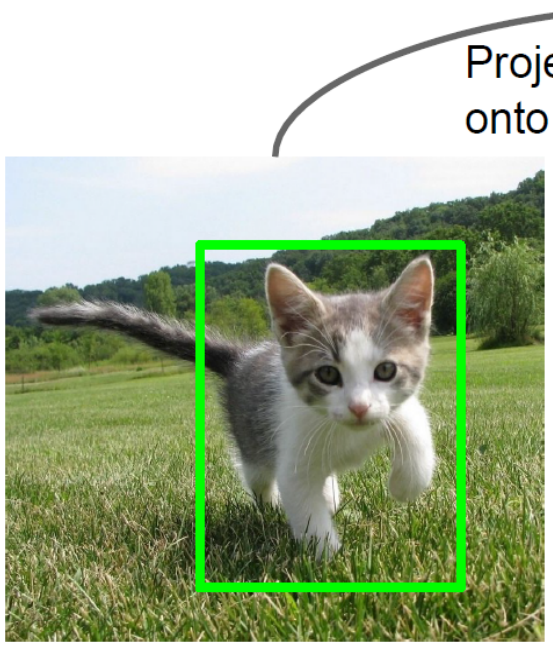
Girshick, "Fast R-CNN", ICCV 2015.

Cropping Features: RoI Pool



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

Cropping Features: RoI Pool



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

Project proposal onto features

CNN

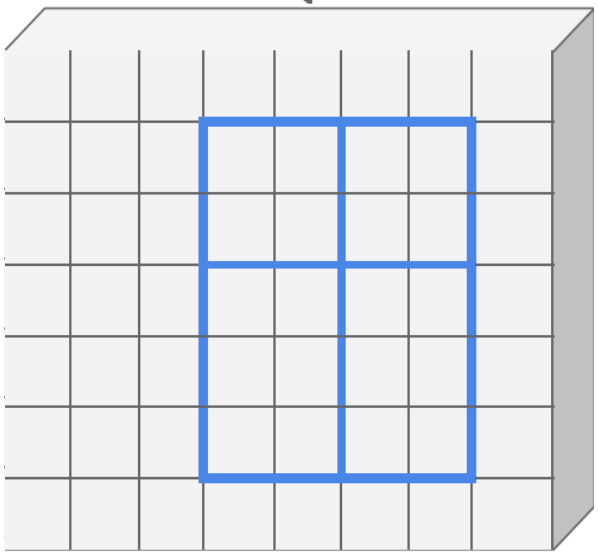
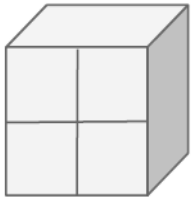


Image features
(e.g. 512 x 20 x 15)

Divide into 2x2 grid of (roughly) equal subregions

Max-pool within each subregion



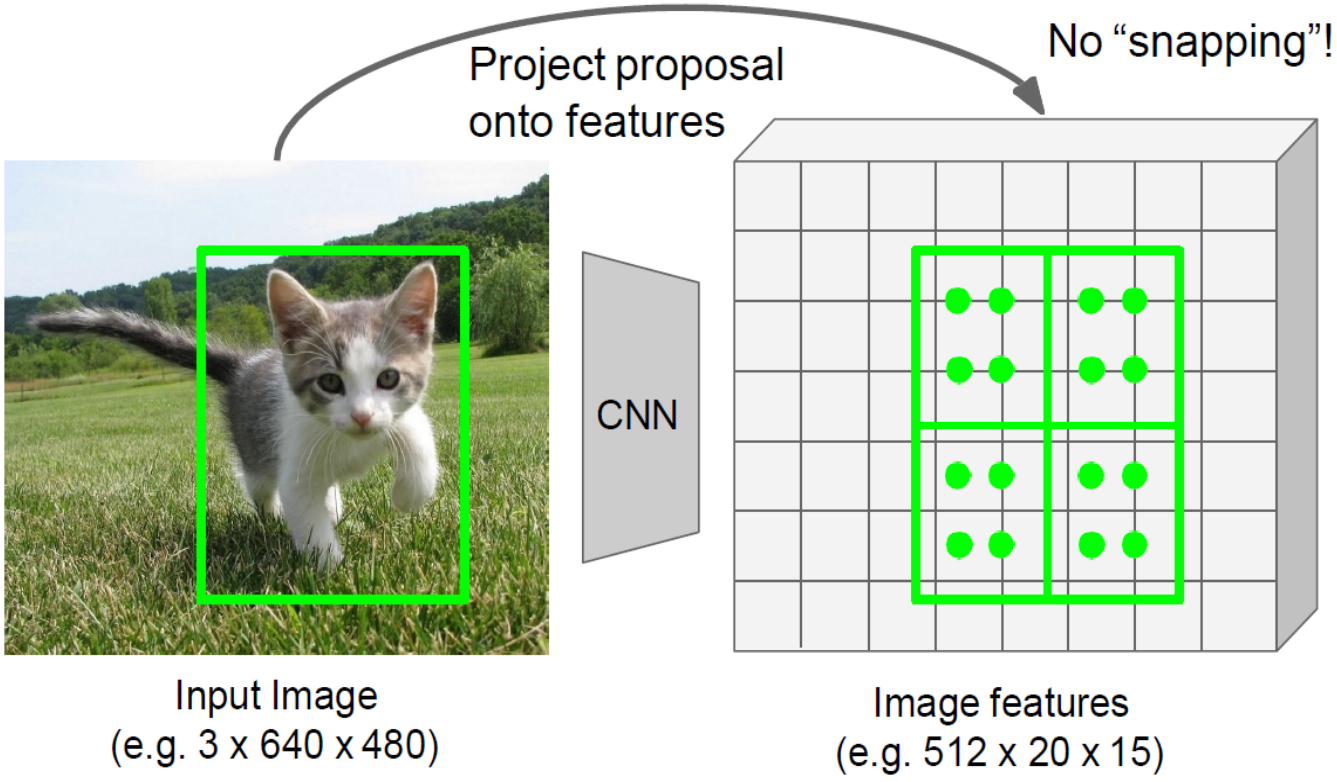
Region features
(here 512 x 2 x 2;
In practice e.g 512 x 7 x 7)

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

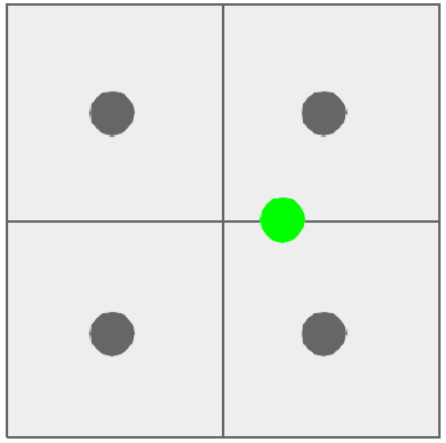
Girshick, "Fast R-CNN", ICCV 2015.

Problem: Region features slightly misaligned

Cropping Features: RoI Align

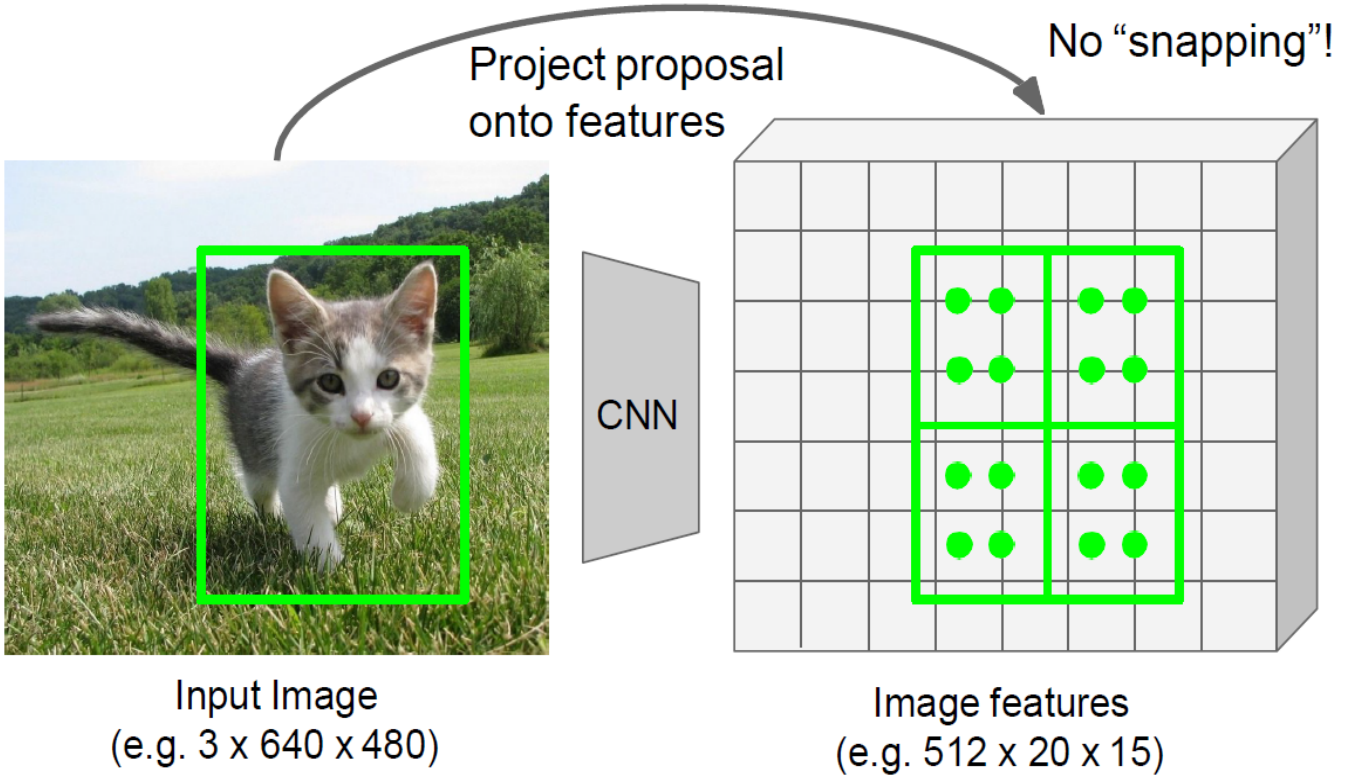


Sample at regular points in each subregion using bilinear interpolation



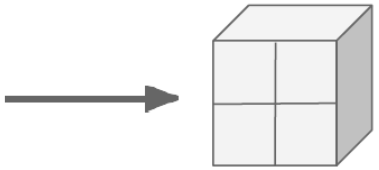
Feature f_{xy} for point (x, y) is a linear combination of features at its four neighboring grid cells:

Cropping Features: RoI Align



Sample at regular points in each subregion using bilinear interpolation

Max-pool within each subregion



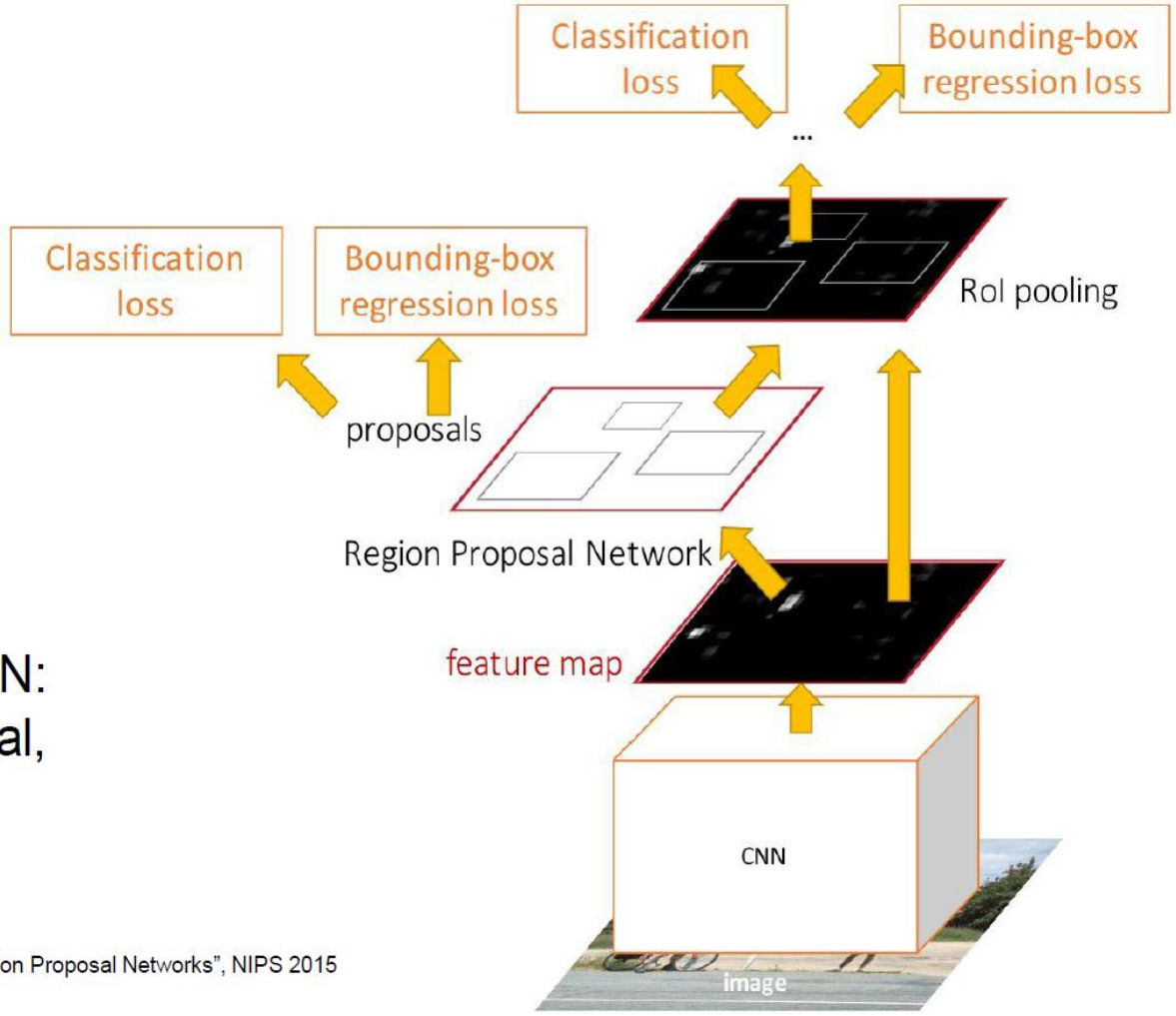
Region features
(here 512 x 2 x 2;
In practice e.g 512 x 7 x 7)

Faster R-CNN:

Make CNN do proposals!

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

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



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

Region Proposal Network

Imagine an **anchor box** of fixed size at each point in the feature map

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

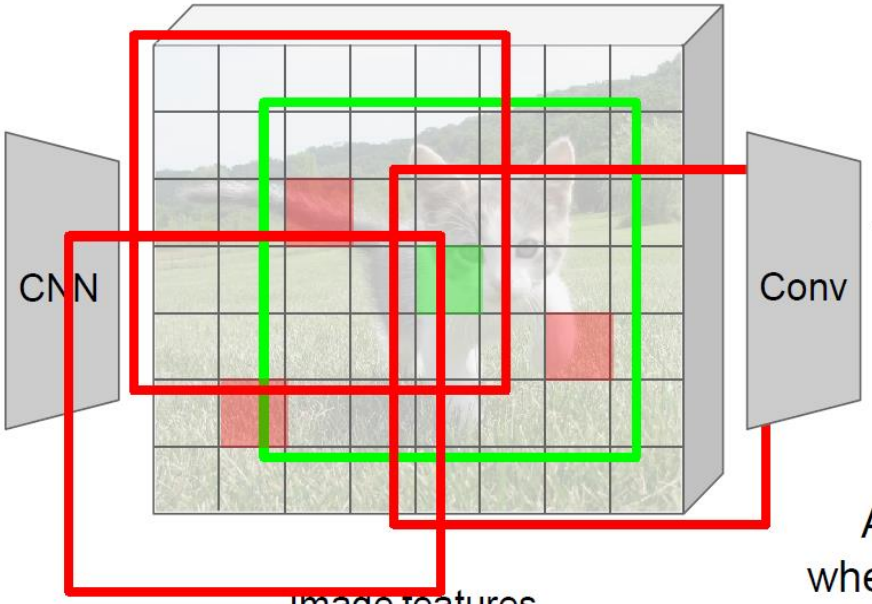


Image features
(e.g. 512 x 20 x 15)

At each point, predict whether the corresponding anchor contains an object (per-pixel logistic regression)

Region Proposal Network

Imagine an **anchor box** of fixed size at each point in the feature map

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

CNN

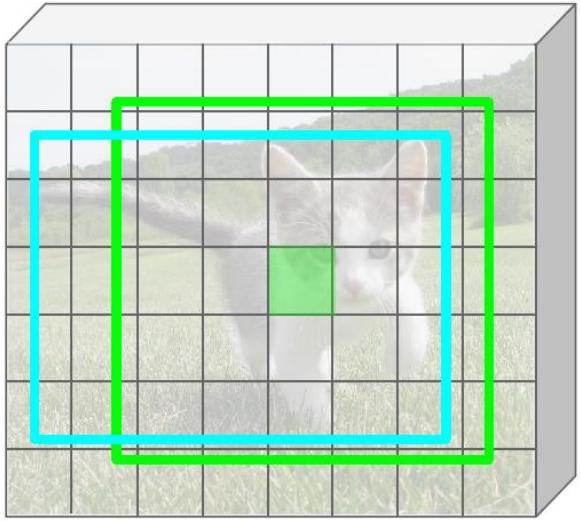


Image features
(e.g. 512 x 20 x 15)

Conv

Anchor is an object?
1 x 20 x 15

Box transforms
4 x 20 x 15

For positive boxes, also predict a transformation from the anchor to the ground-truth box (regress 4 numbers per pixel)

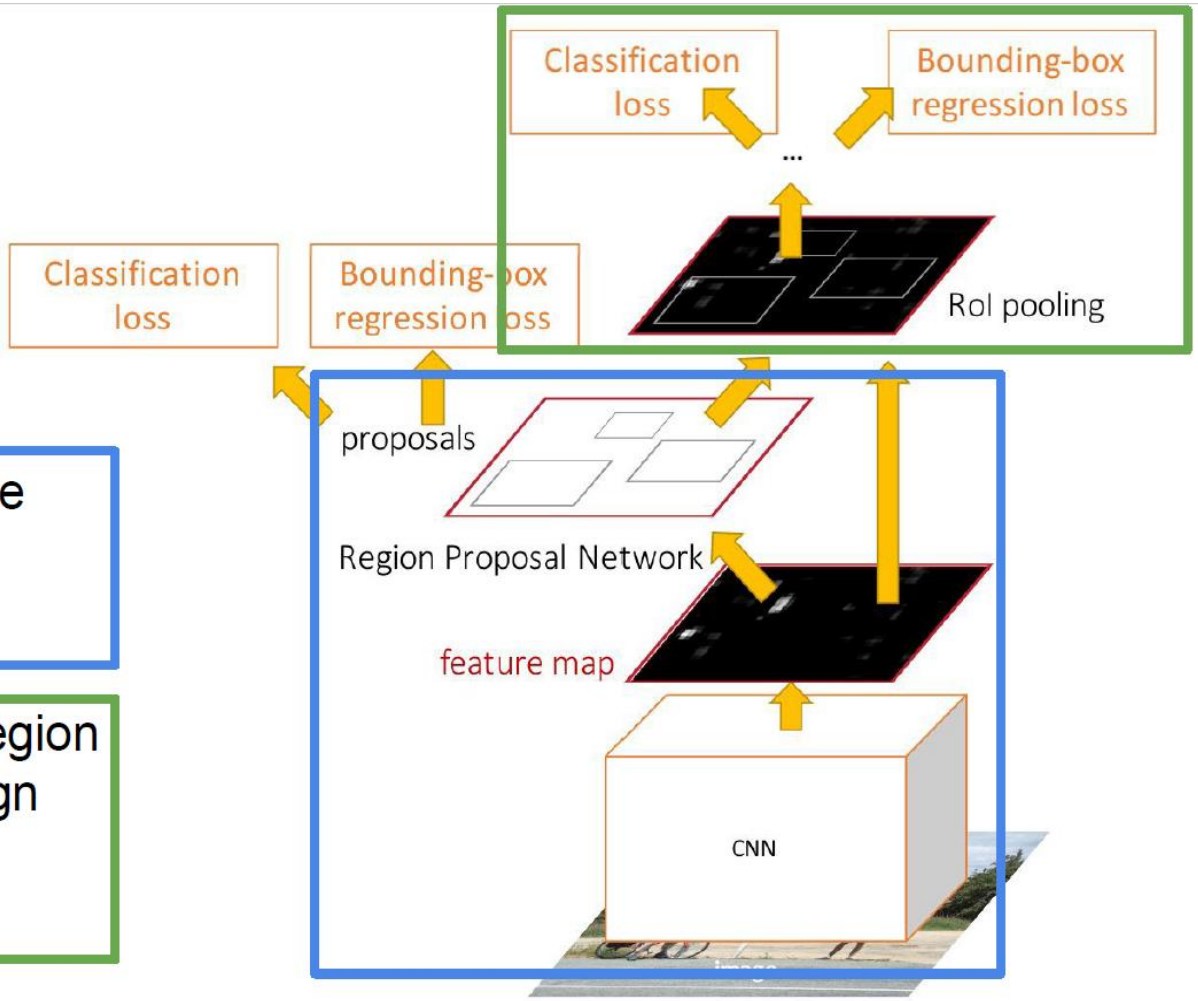
Faster R-CNN:

Make CNN do proposals!

Faster R-CNN is a
Two-stage object detector

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

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



Instance Segmentation

Classification

CAT

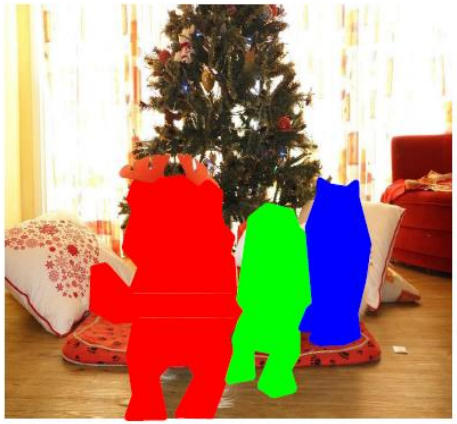
Semantic Segmentation

GRASS, CAT, TREE, SKY

Object Detection

DOG, DOG, CAT

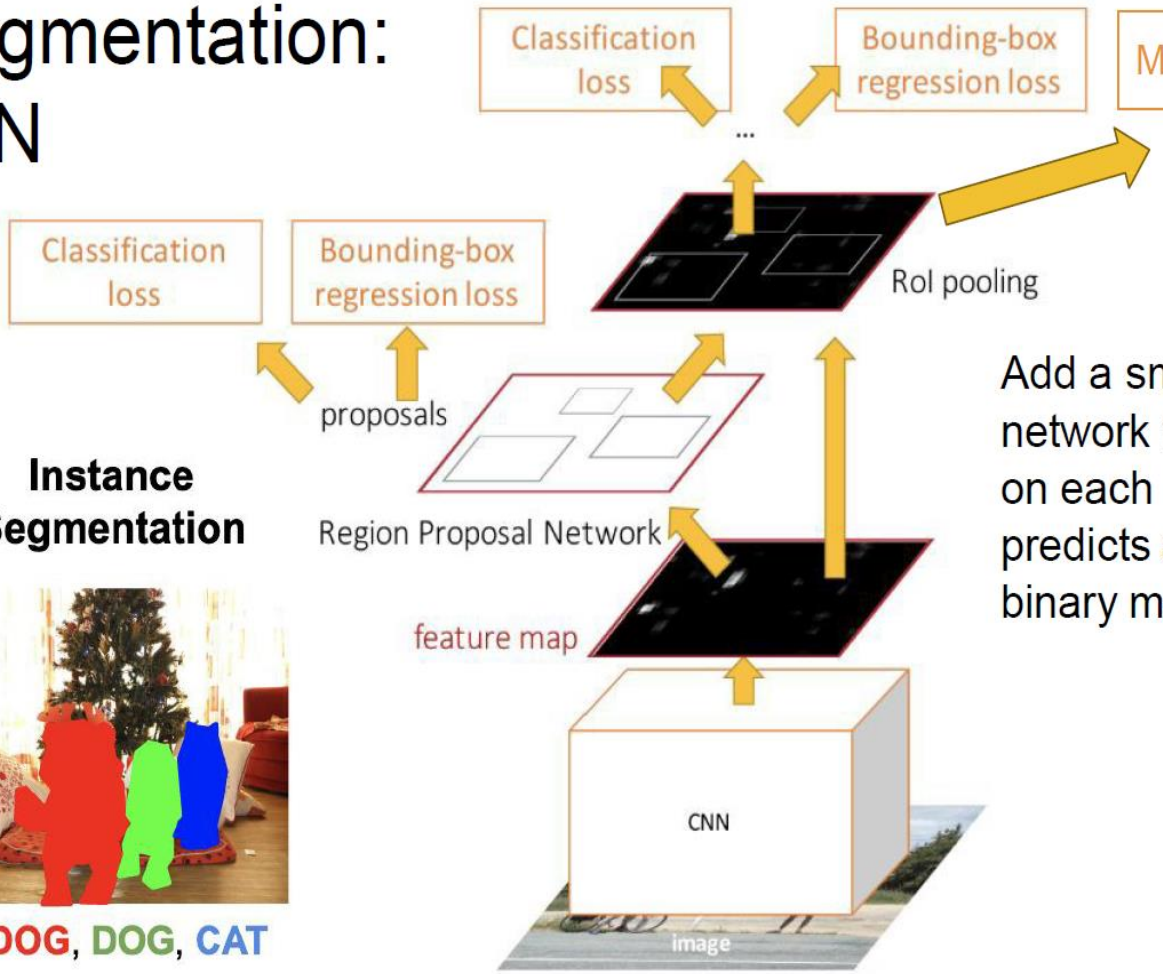
Instance Segmentation



DOG, DOG, CAT

Multiple Object

Instance Segmentation: Mask R-CNN



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

Object Detection

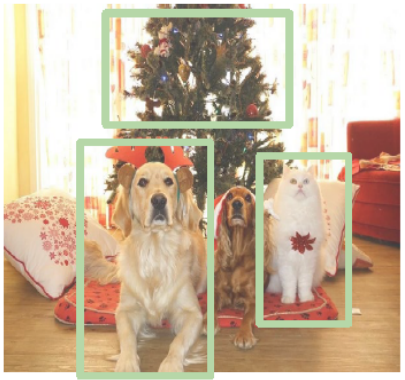
DOG, DOG, CAT

Instance Segmentation

DOG, DOG, CAT

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

Mask R-CNN



CNN
+RPN

RoI Align

Classification Scores: C
Box coordinates (per class): 4 * C

256 x 14 x 14

Conv

256 x 14 x 14

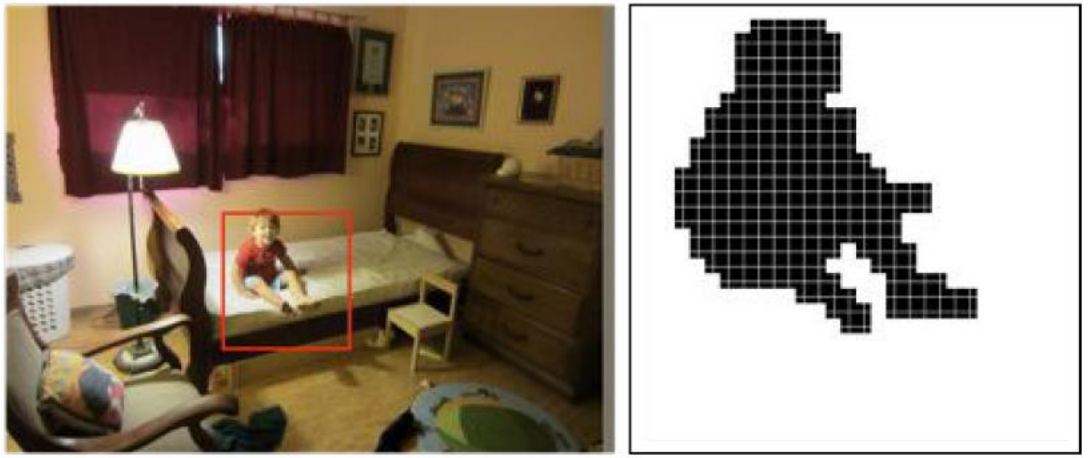
Conv

Predict a mask for each of C classes

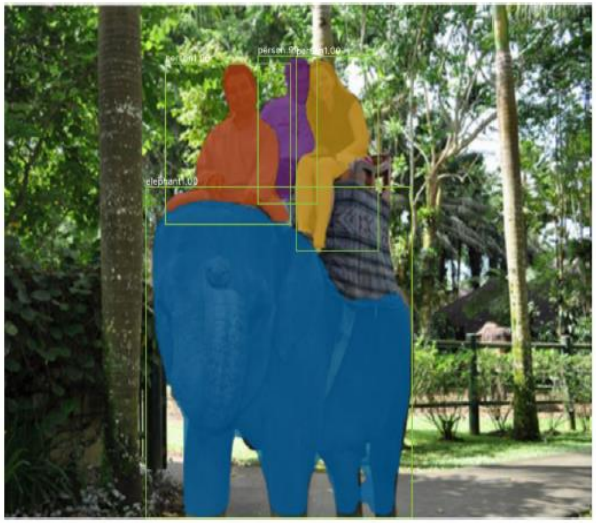
C x 28 x 28

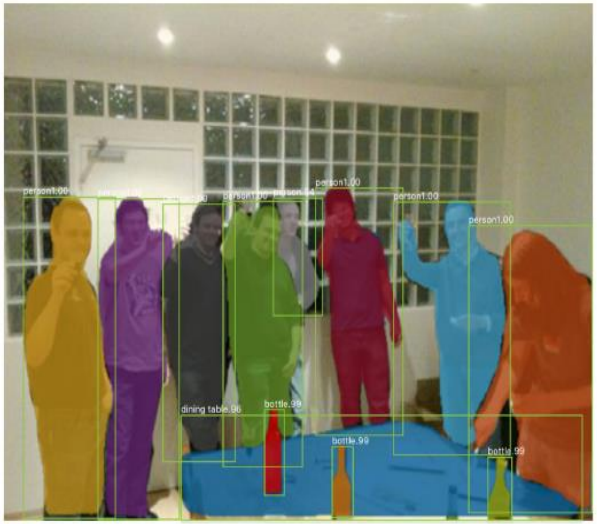
He et al, “Mask R-CNN”, arXiv 2017

Example



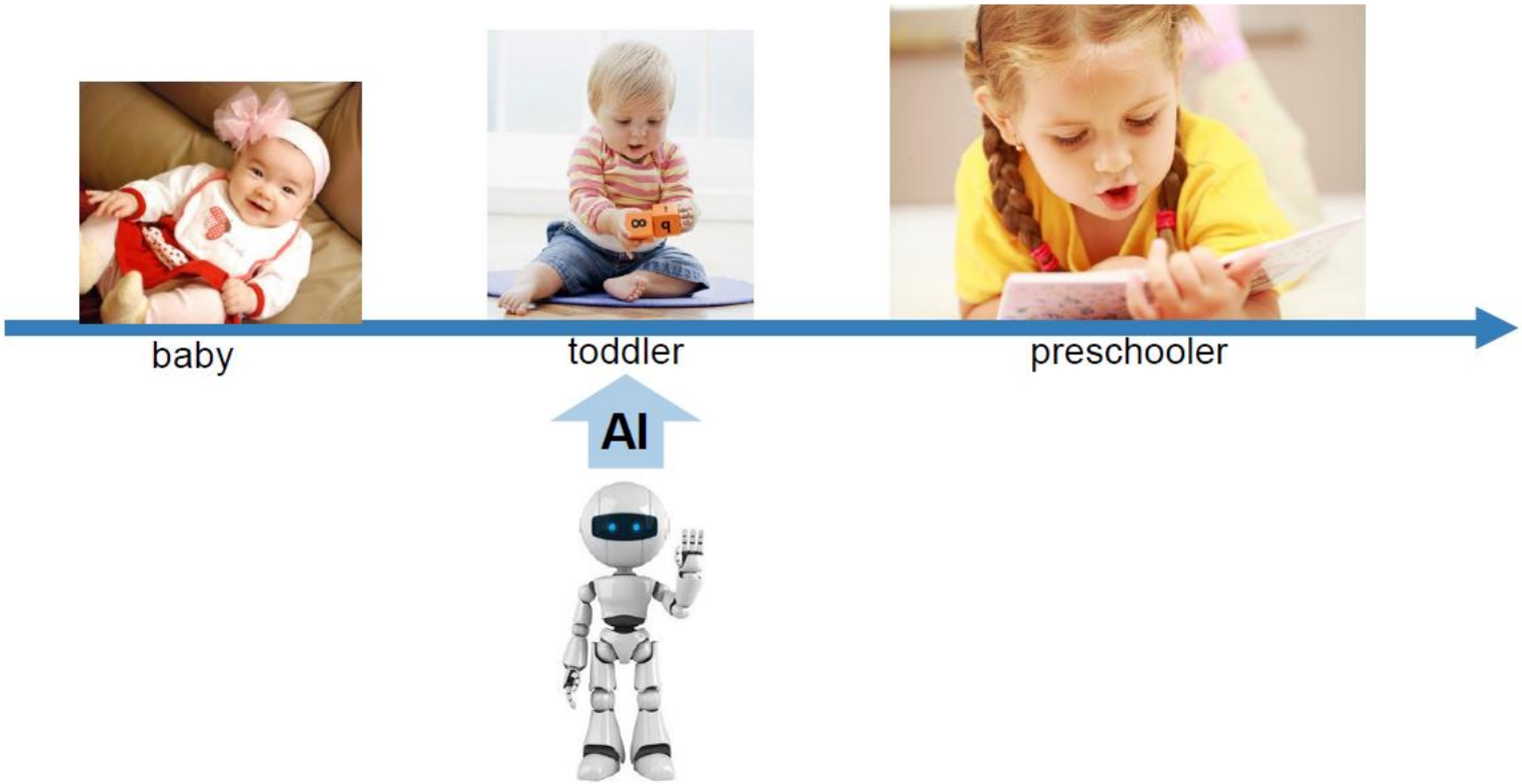
Mask R-CNN: Very Good Results!



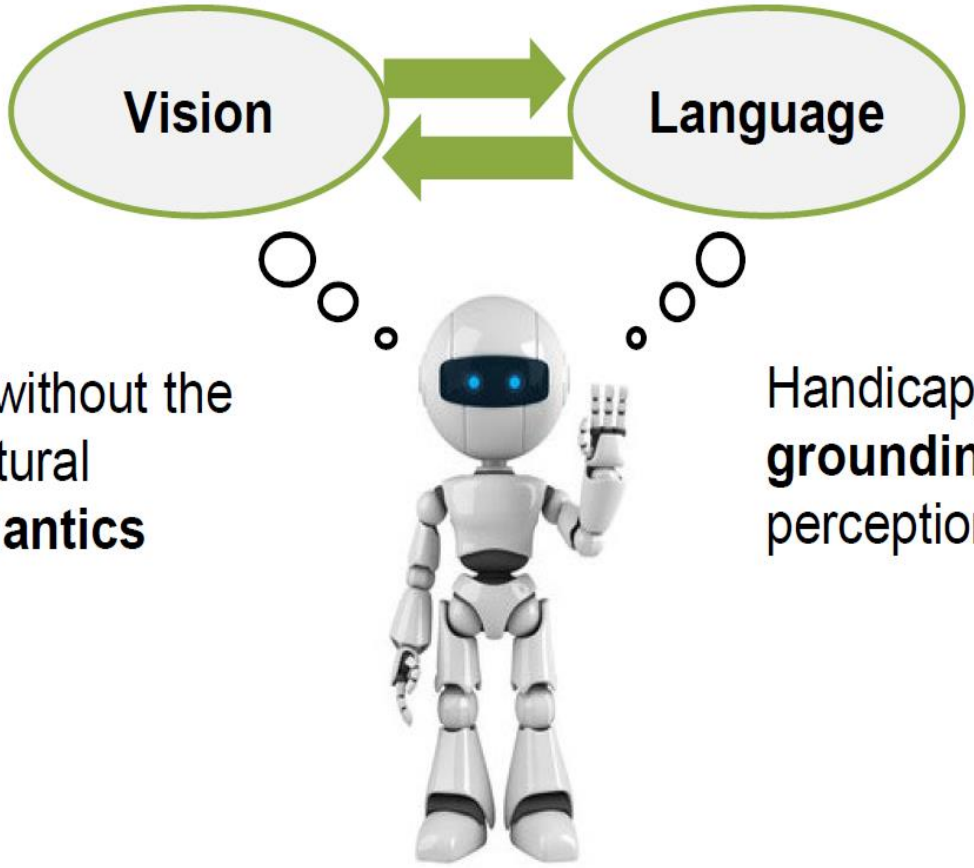


He et al, "Mask R-CNN", ICCV 2017

Problem: the “Toddler” AI



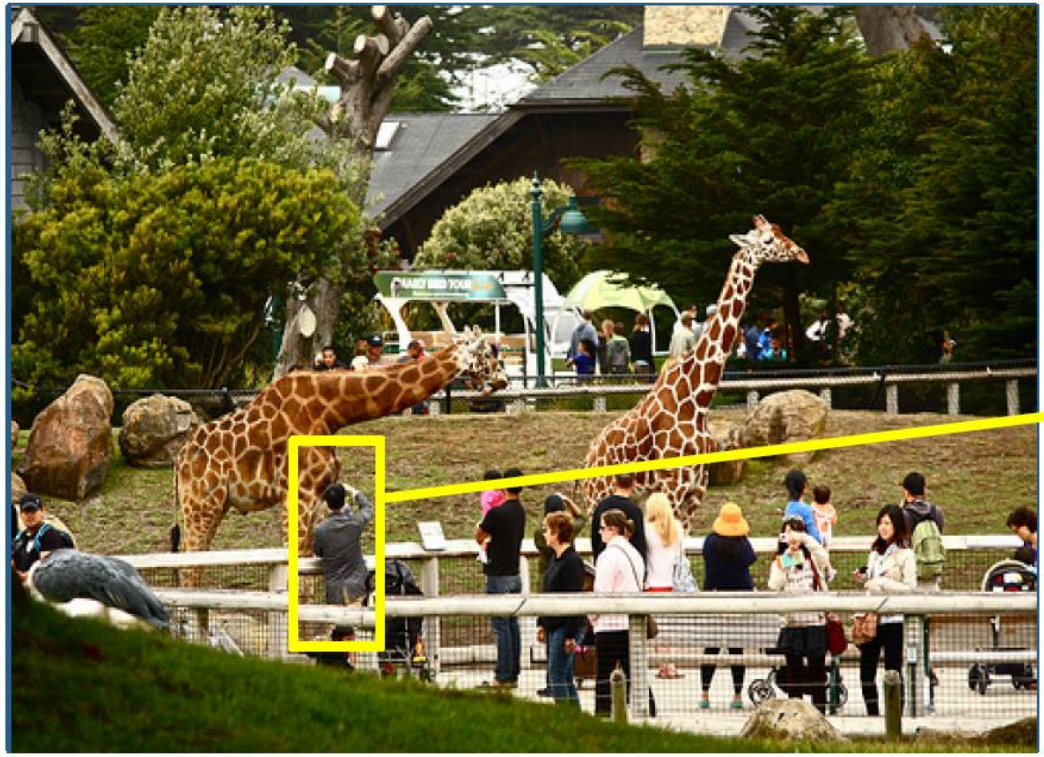
Why combine vision and language?



Handicapped without the richness of natural language **semantics**

Handicapped without **grounding** meaning in perception

How can we connect vision and language?



Captioning Hendricks et al, CVPR16 Ramanishka et al, ACMM16

A crowd of people looking at giraffes in a zoo.

Referring Expressions Hu et al., CVPR16

Person taking a photo?

Question Answering Xu and Saenko ECCV16

What time of year is it?

Answer: summer

Today: connecting language to vision

Image Description

Input image

Output: A close up of a hot dog on a bun.

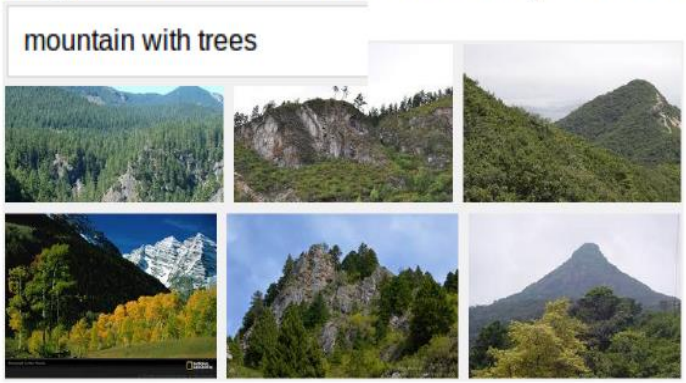
Video Description

Input video

Output: A woman shredding chicken in a kitchen

Applications

Image and video retrieval by content.

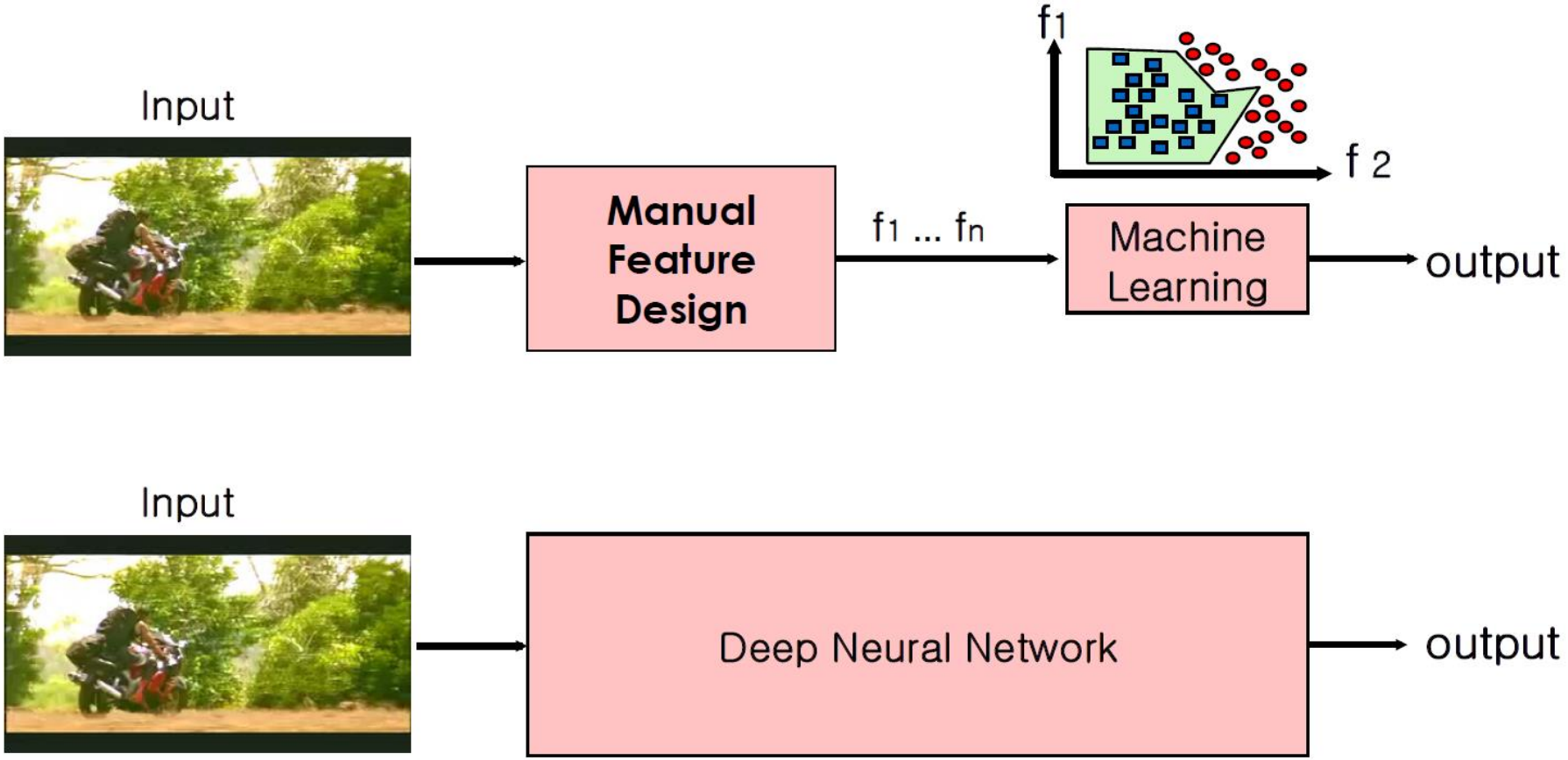


Video description service.

Human Robot Interaction

Video surveillance

Deep Learning Revolution



Deep End-to-End Neural Models based on Recurrent Nets

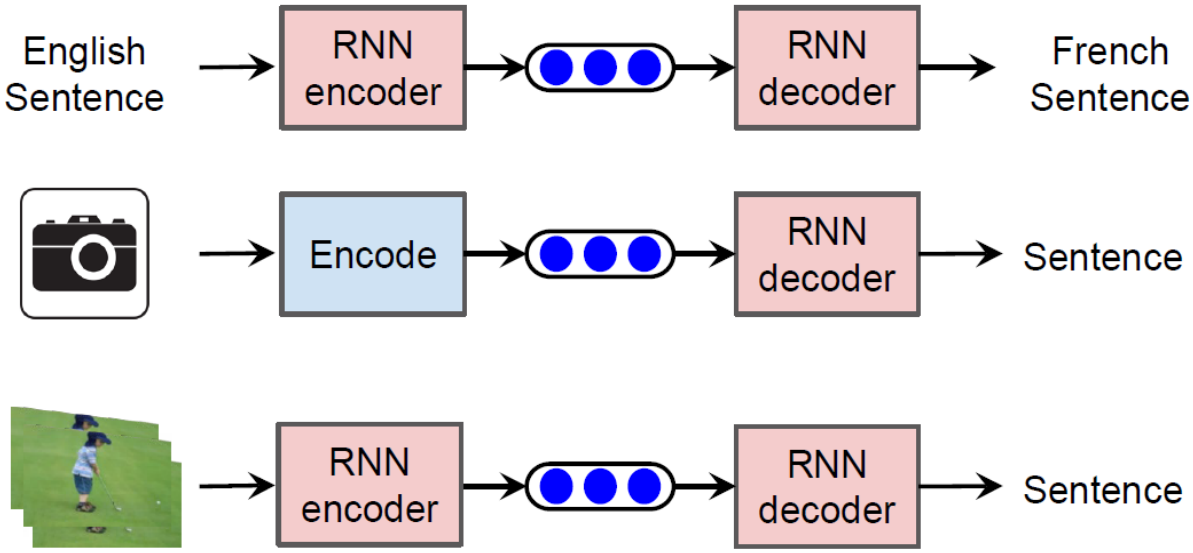
Deep Neural Network → Sentence

[Donahue et al. CVPR'15] (our work) [Vinyals et al. CVPR'15]

Deep Neural Network → Sentence

[Venugopalan et. al. NAACL'15] [Venugopalan et. al. ICCV'15] (our work)

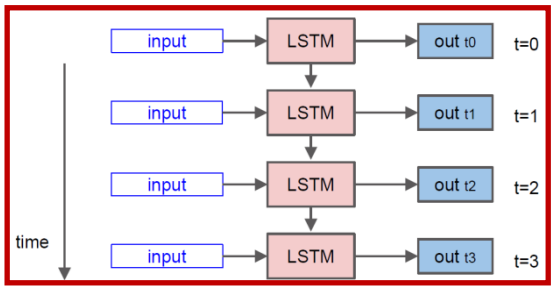
Deep End-to-End Neural Models based on Recurrent Nets



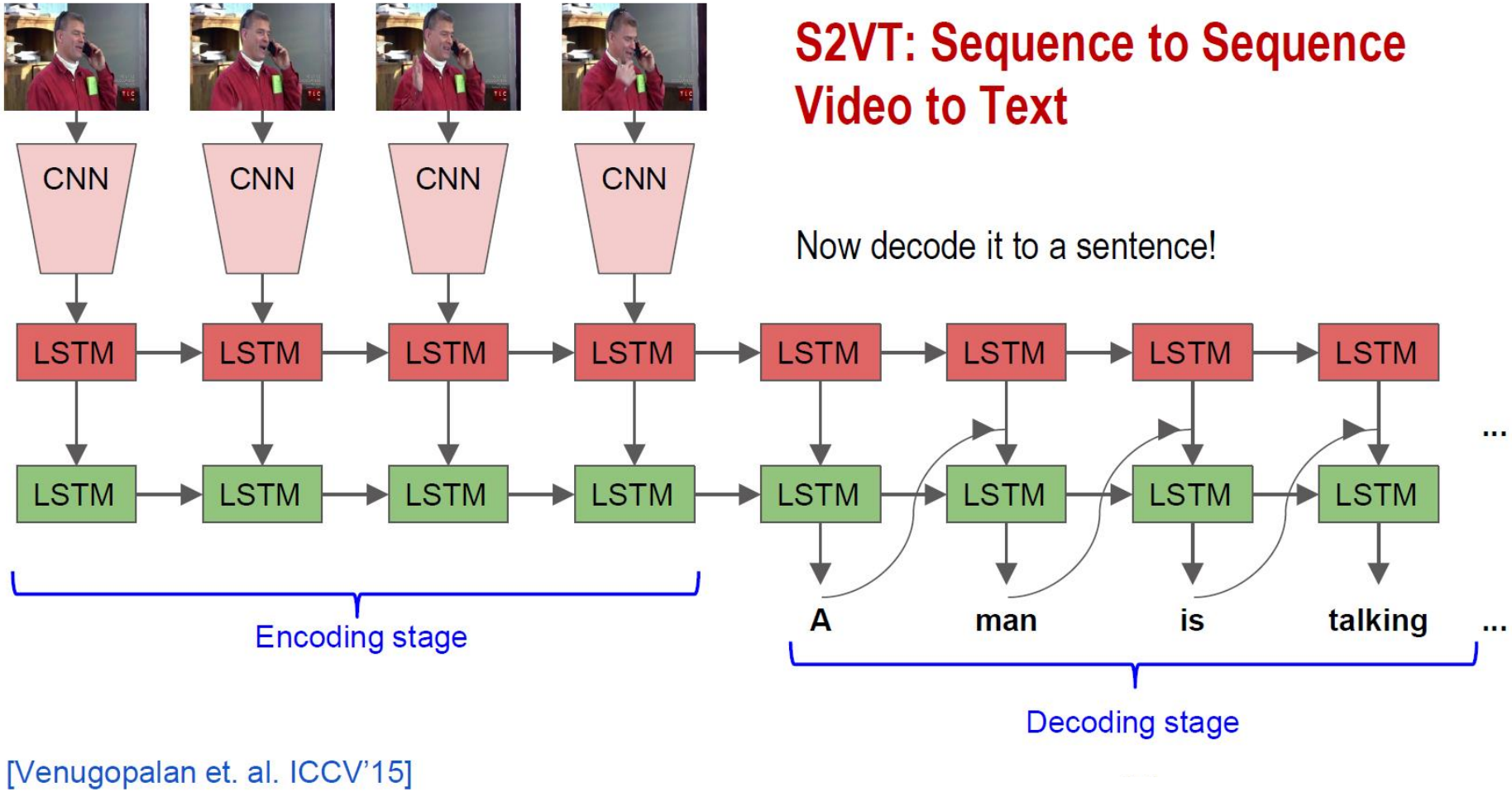
[Sutskever et al. NIPS'14]

[Donahue et al. CVPR'15]
[Vinyals et al. CVPR'15]

[Venugopalan et. al. NAACL'15]
[Venugopalan et. al. ICCV'15]

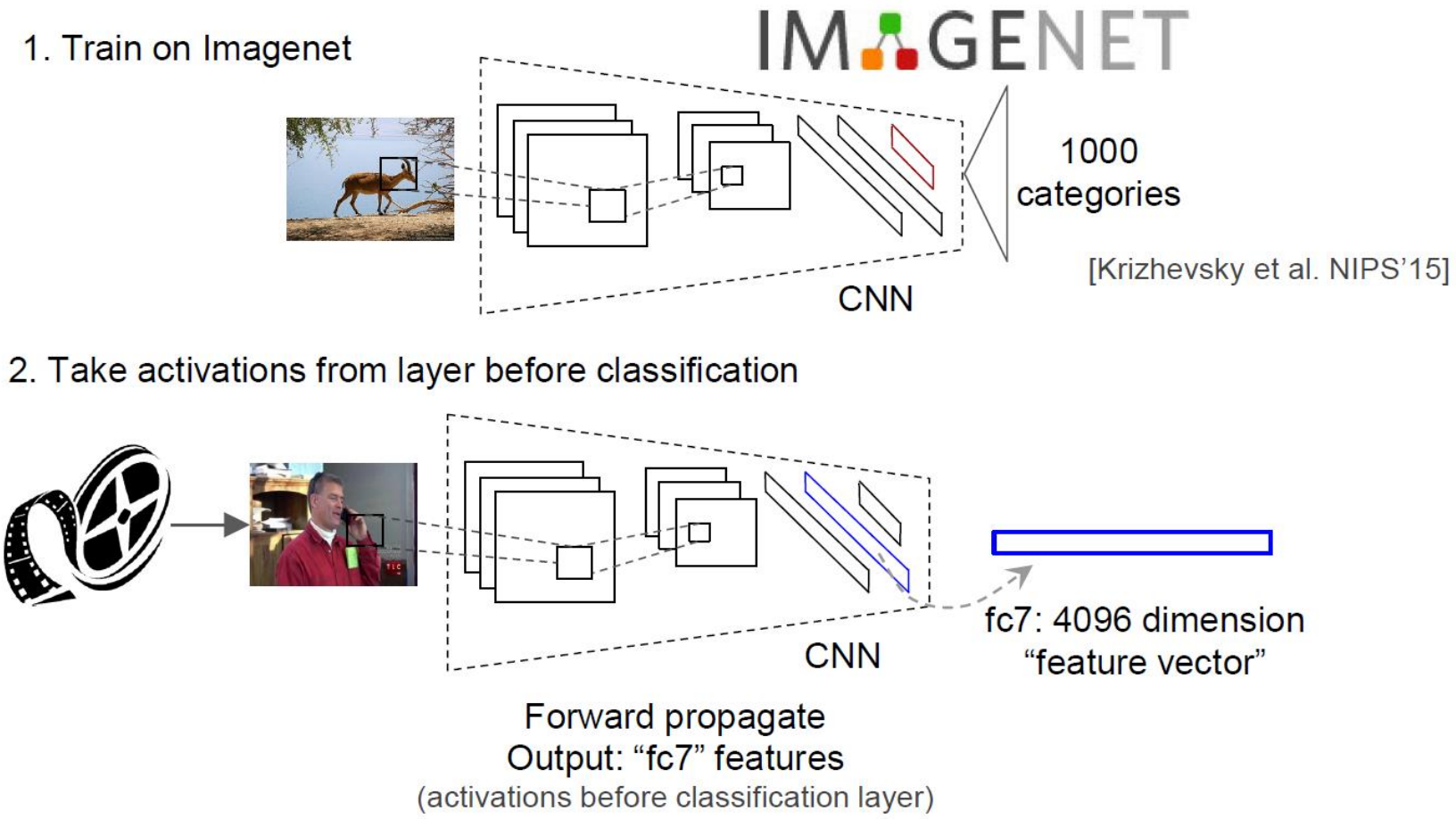


S2VT: Sequence to Sequence Video to Text



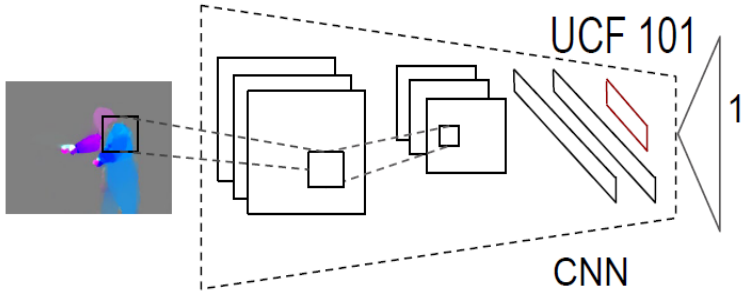
[Venugopalan et. al. ICCV'15]

Objects


1. Train on Imagenet
IMAGENET
1000
categories
CNN
[Krizhevsky et al. NIPS'15]
2. Take activations from layer before classification
CNN
fc7: 4096 dimension
"feature vector"
Forward propagate
Output: "fc7" features
(activations before classification layer)

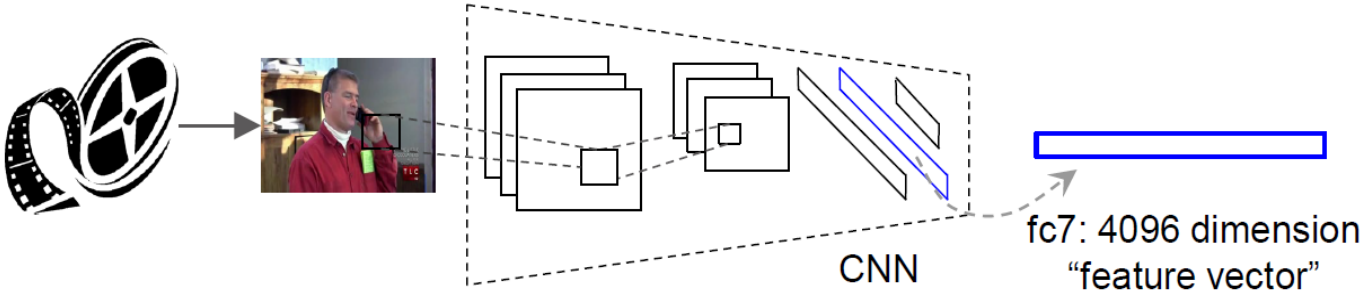
Actions

1. Train CNN on Activity classes



[Donahue et al. CVPR'15]

2. Take activations from layer before classification



Forward propagate
Output: "fc7" features
(activations before classification layer)

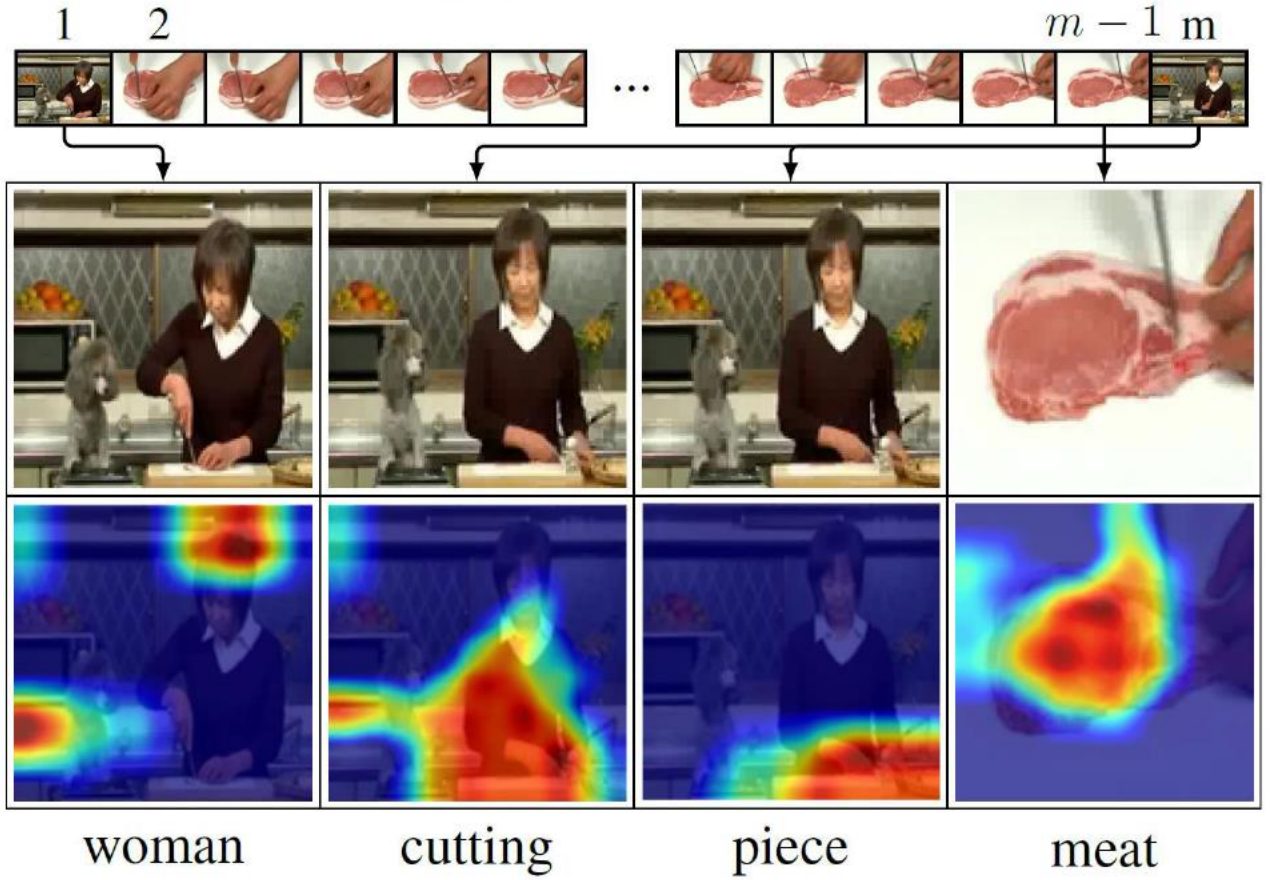
Explaining the network's captions

Predicted sentence: A woman is cutting a piece of meat

can the network localize objects?

Spatiotemporal saliency

Predicted sentence: A **woman** is **cutting** a **piece** of **meat**



Pointing game in Flickr30kEntities