Semantic Segmentation & Object Detection

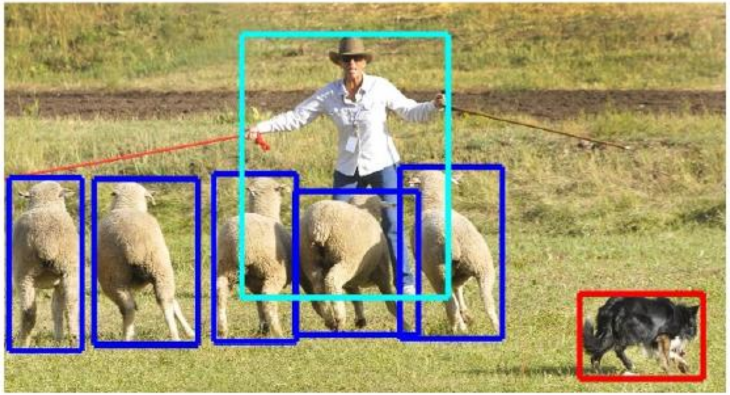
Hakan Bilen

Machine Learning Practical - MLP Lecture 16
13 Feb 2019

Classification is about “what object categories are present in the image?”

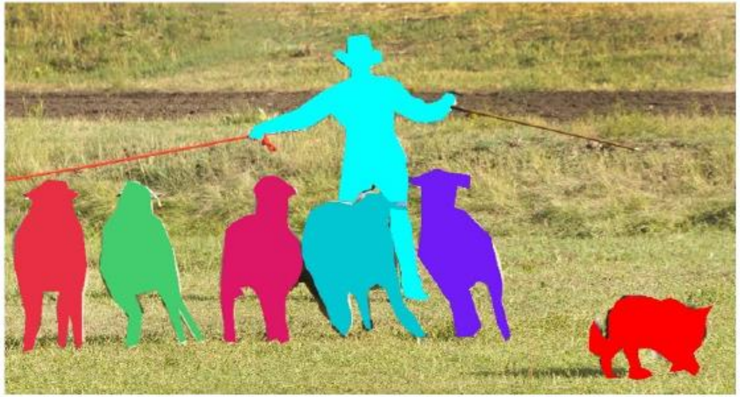
What other questions can we ask about the image?

image classification



object detection

semantic segmentation



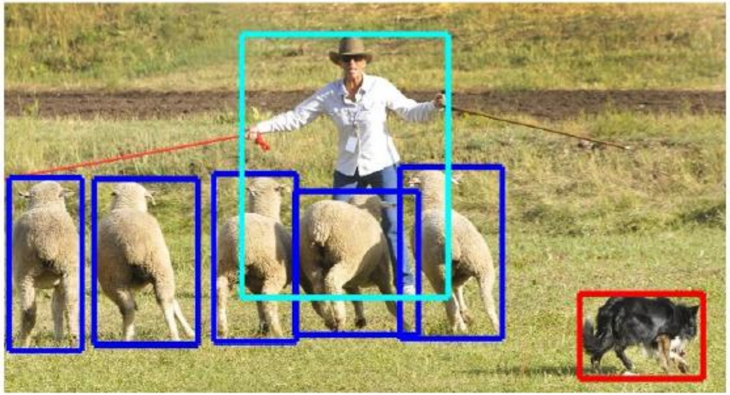
instance segmentation

Image credits: Microsoft COCO

Today’s goal

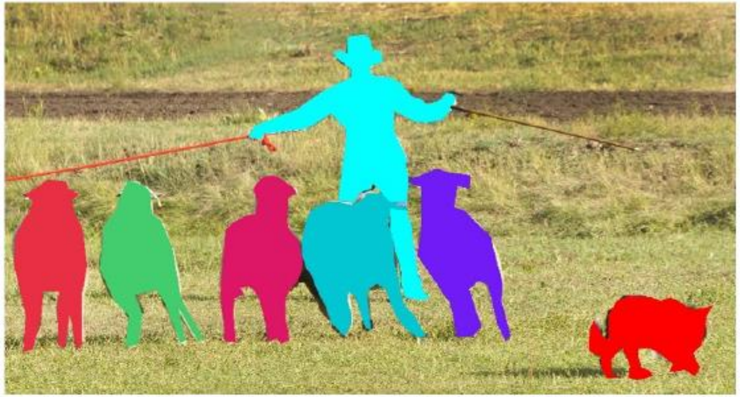
- Tasks beyond image classification
- How to customize the learning machine for the task of interest
  - Customise network architecture
  - Design new layer types and loss functions

image classification



object detection

semantic segmentation



instance segmentation

Image credits: Microsoft COCO

Semantic segmentation

Label each pixel with a category label

Do not differentiate between instances

Evaluation: Mean intersection over union (IoU)

$$\text{IoU} = \frac{\text{true pos}}{\text{true pos} + \text{false neg} + \text{false pos}}$$

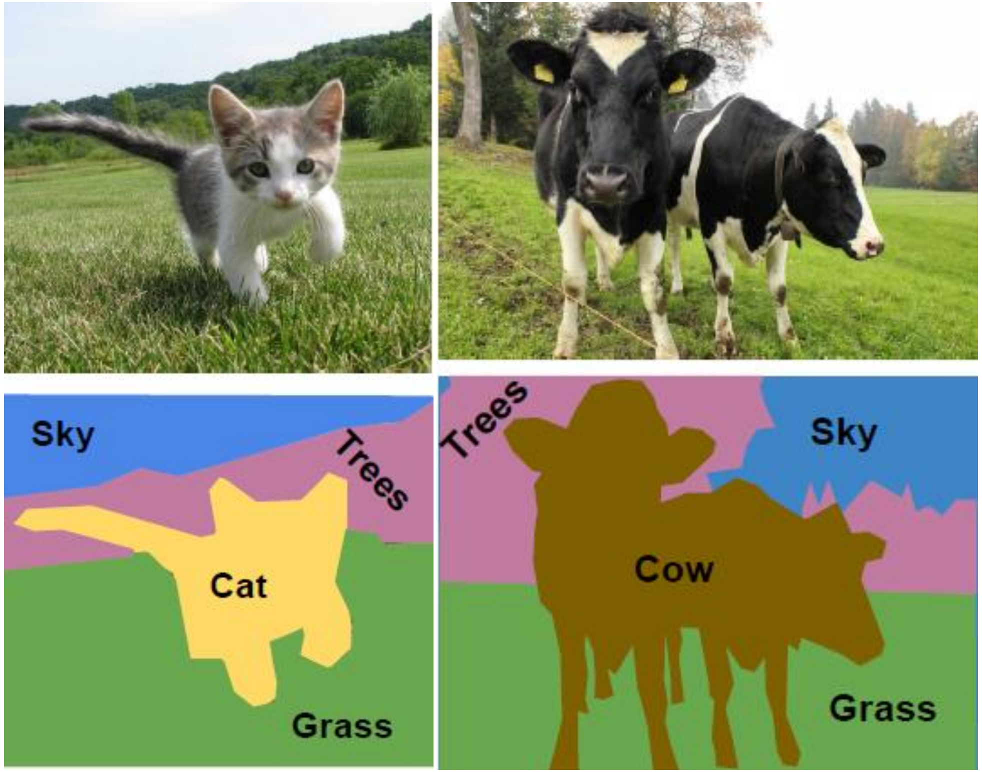
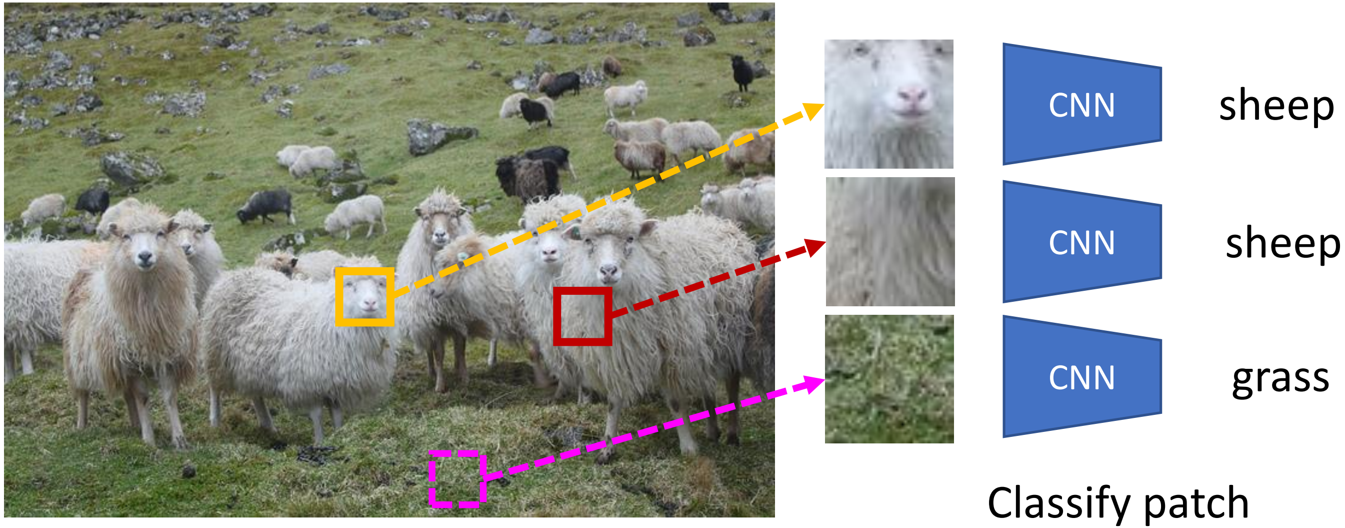


Image credits: CS231 - Stanford - CC0 public domain

Classifying image patches



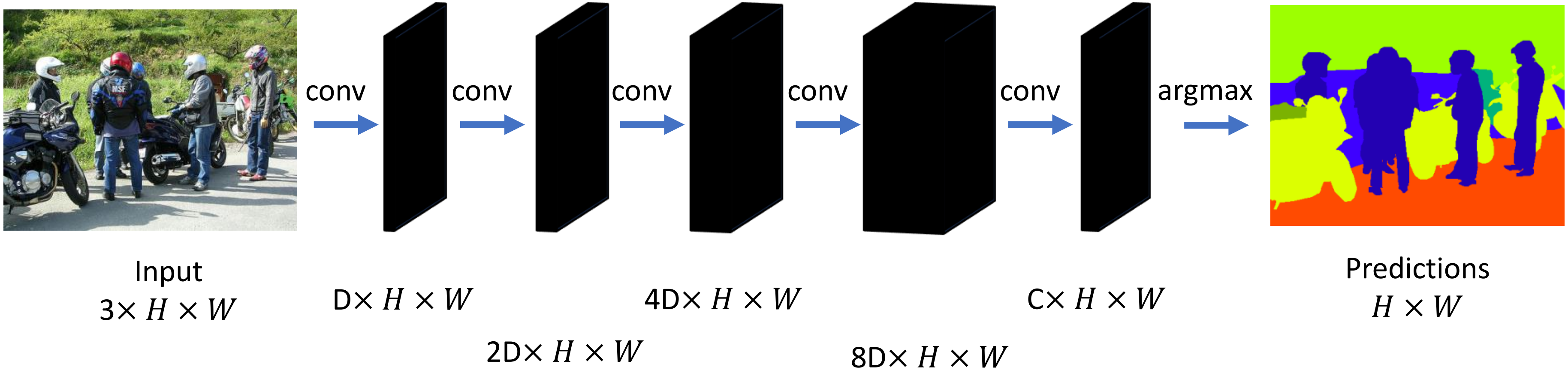
Classify patch
with CNN

☹ Computationally expensive! No feature sharing between overlapping patches.

[Farabet, et al. (2013) "Learning hierarchical features for scene labeling." PAMI]

(Fully) Convolutional Network

- Design a neural network that can generate labels for each pixel at once!
- No spatial dimension reduction (and no fully connected layer)

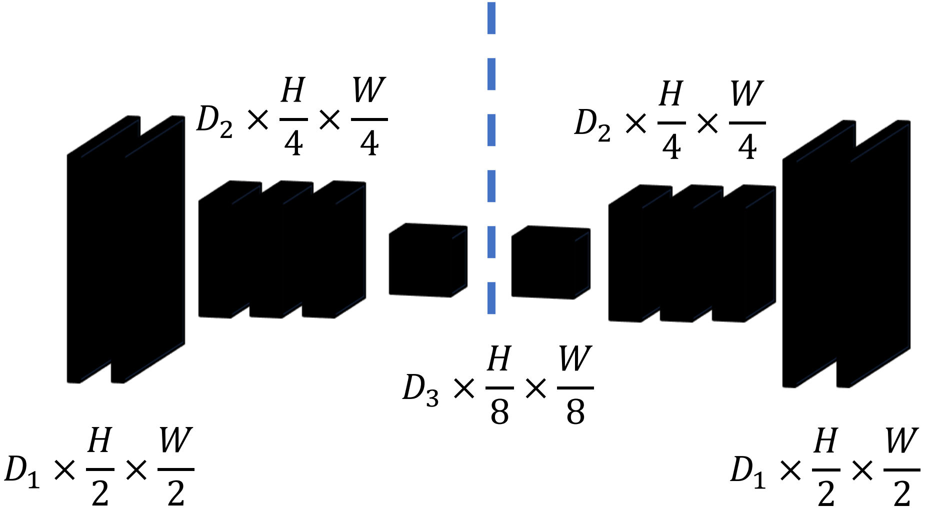


☹ Convolutions at original image resolution is very expensive!

(Fully) Convolutional Network

Design a network that first downsamples and then upsamples to input size!

Input
$3\times H \times W$



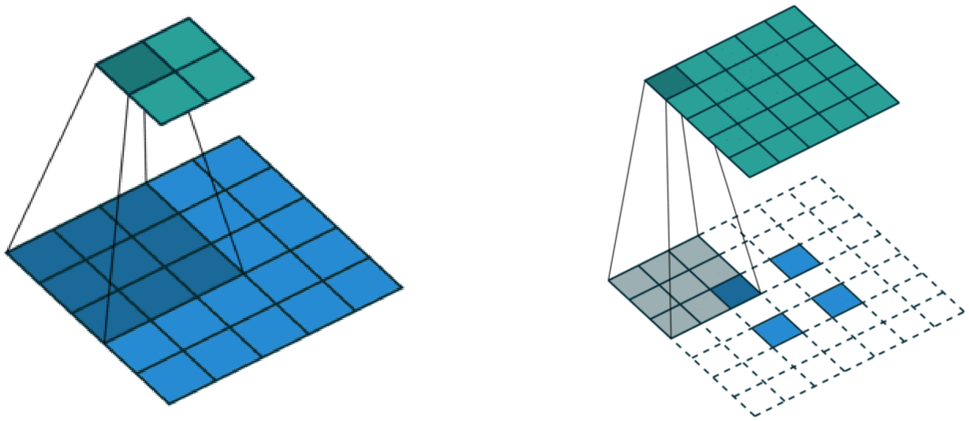
Predictions
$H \times W$

[Long et al. (2015) "Fully Convolutional Networks for Semantic Segmentation", CVPR](#)
[Noh et al. (2015), Learning Deconvolution Network for Semantic Segmentation, ICCV](#)

Upsampling with transpose convolution

convolution vs transpose convolution

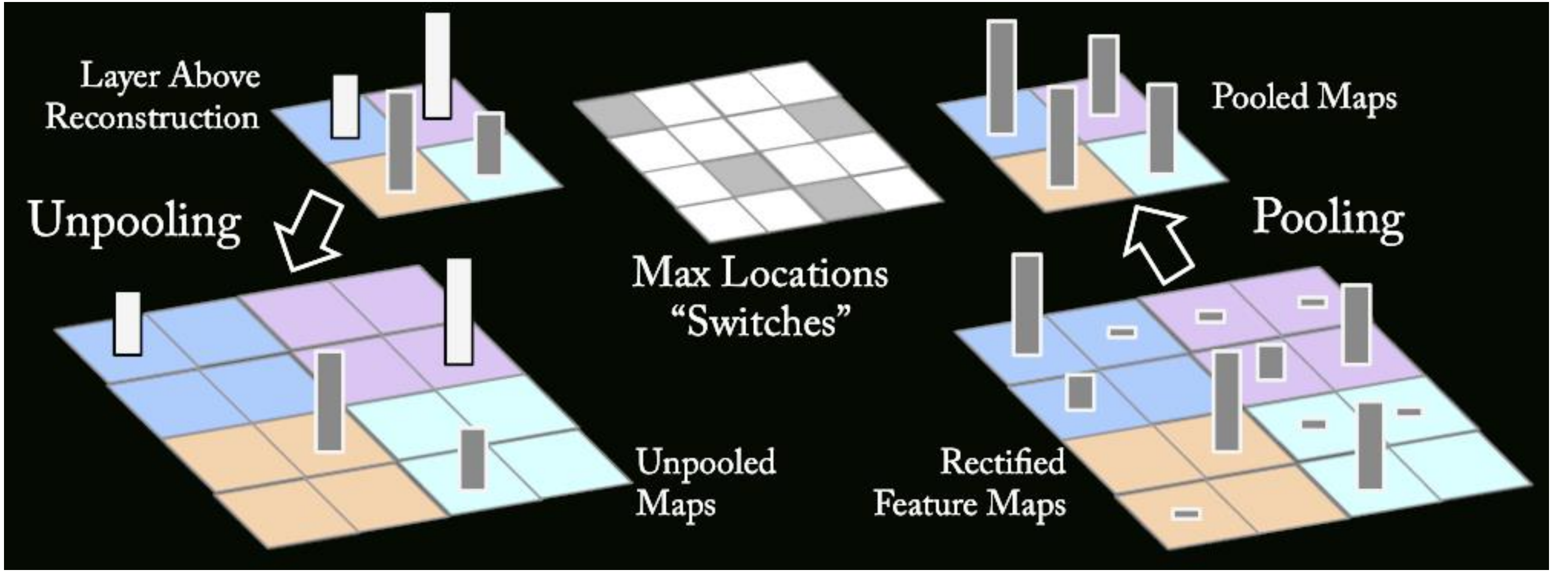
stride=2



☺ It can learn a nonlinear upsampling

☹ Its input feature is low resolution

Upsampling with unpooling

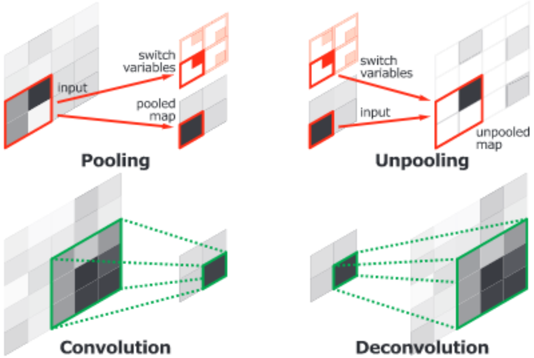


- ☺ It has information about max locations of high resolution features
- ☹ It can't learn to upsample (has no learnable parameters)

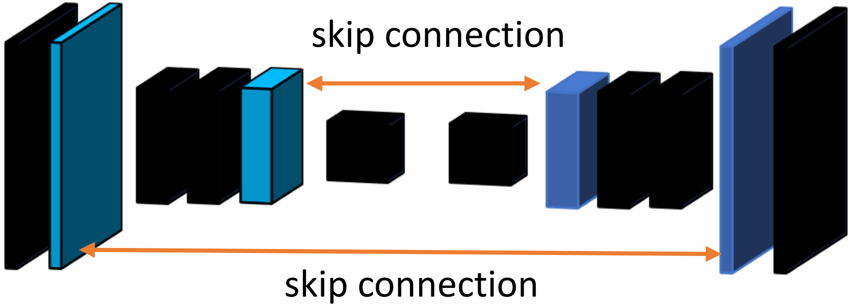
(Fully) Convolutional Network

Use skip connections to transfer pooling switches via skip connections

- learn to upscale
- maintain high resolution shape information



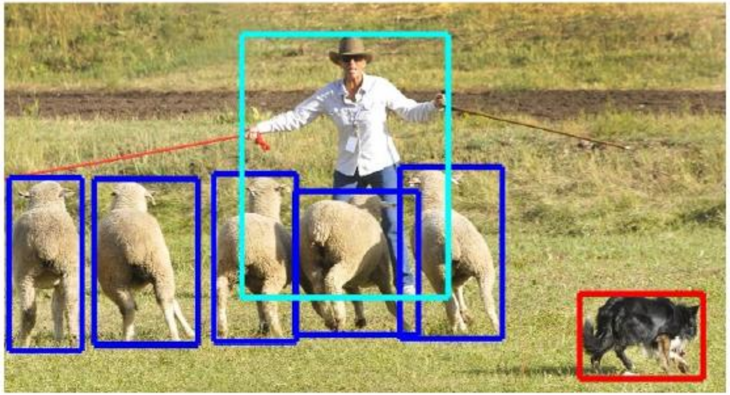
Input
$3\times H \times W$



Predictions
$H \times W$

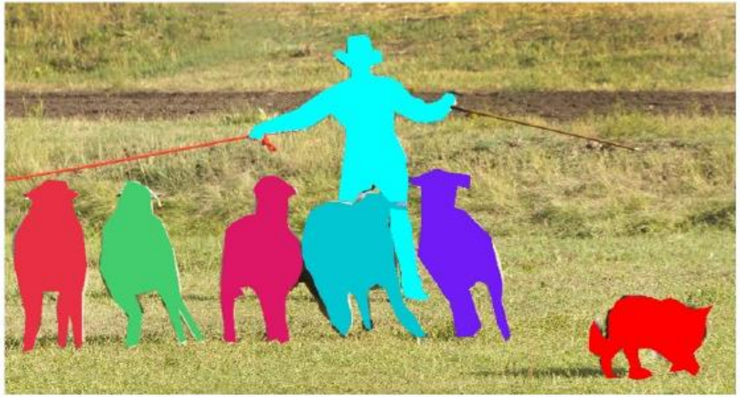
[Noh et al. (2015), Learning Deconvolution Network for Semantic Segmentation, ICCV]

image classification



object detection

semantic segmentation



instance segmentation

Image credits: Microsoft COCO

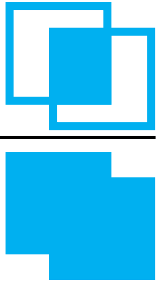
Object detection

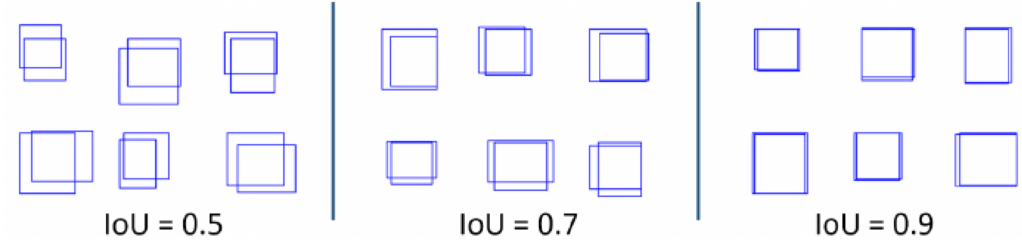
Object detection has two goals

- Classification (*e.g.* cow)
- Localisation (*e.g.* [x y w h])

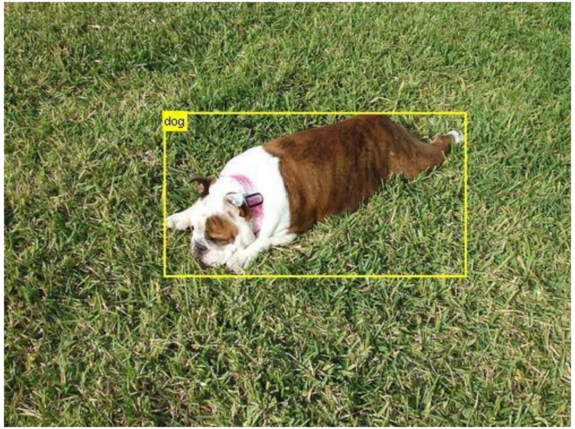
Evaluation metrics is intersection over union

- $\text{IoU} = \dfrac{\text{Area of overlap}}{\text{Area of union}}$



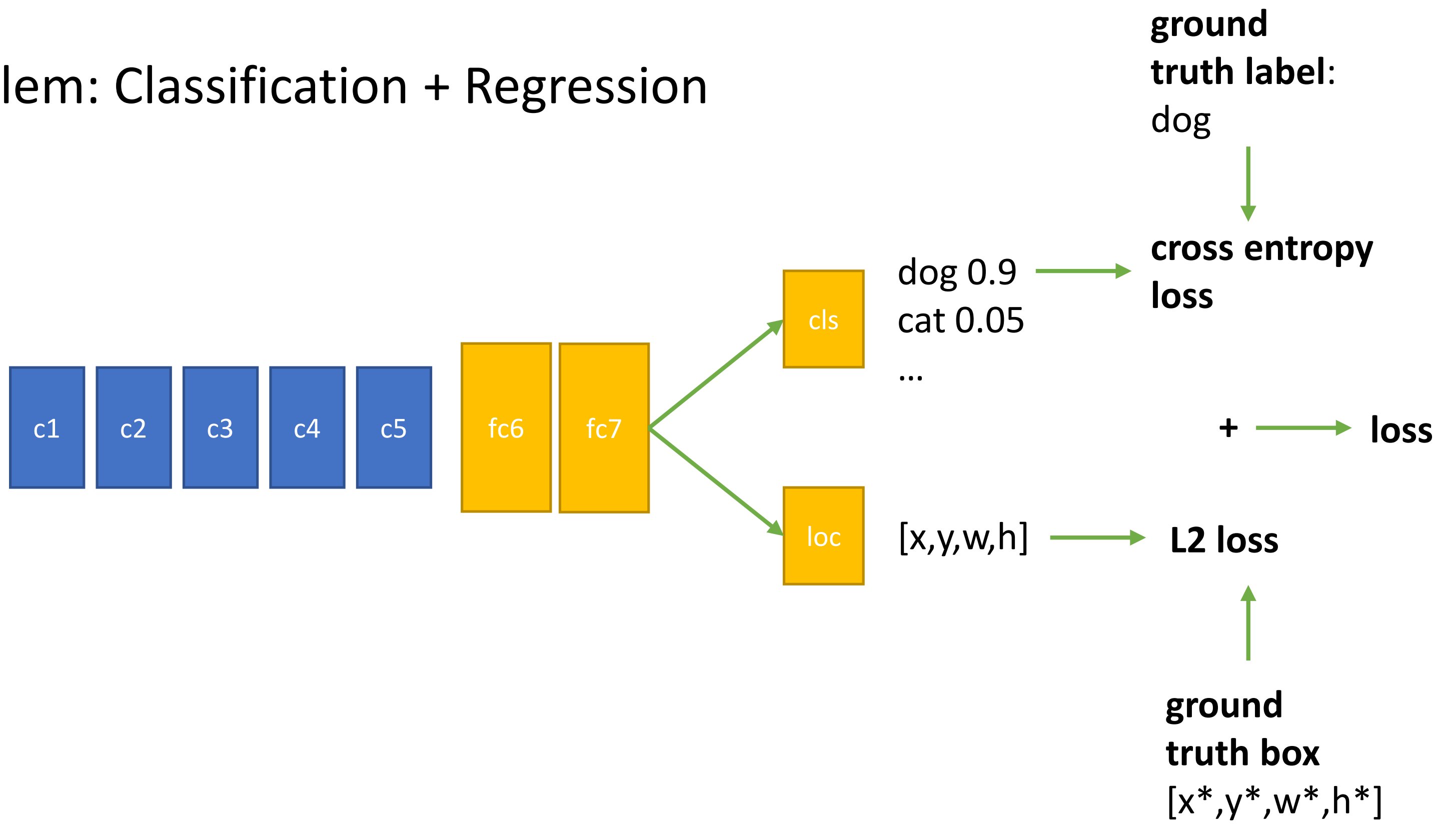


[Figure credit](#)

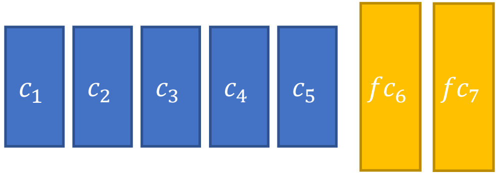


Classification + Localisation

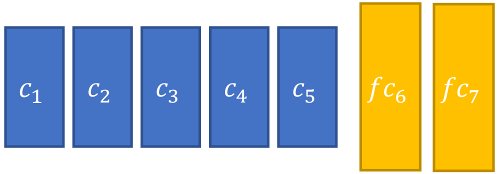
Multi-task problem: Classification + Regression



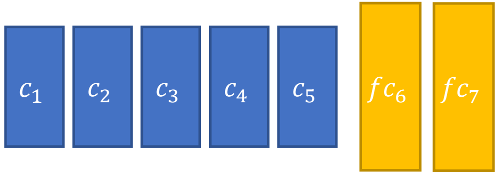
Multi object instances



dog [x,y,w,h]



person [x,y,w,h]
m.bike [x,y,w,h]

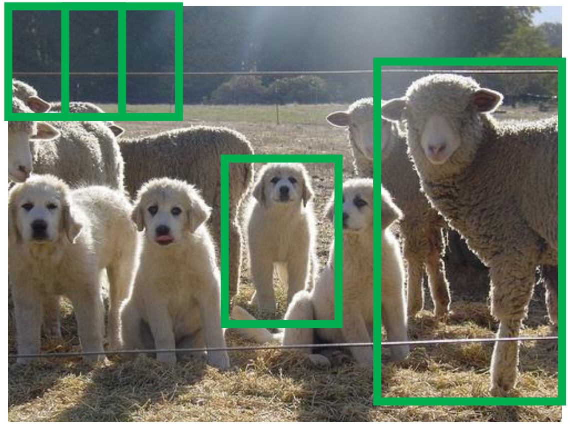


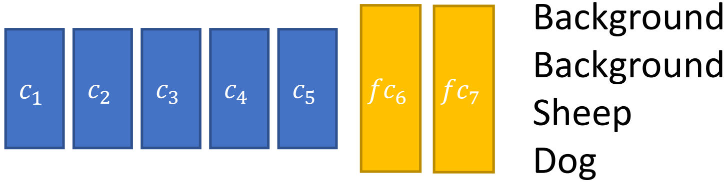
dog [x,y,w,h]
dog [x,y,w,h]
…
many more

What if number of object instances vary?

☹ This will work for one object instance per class!

Classifying sliding windows





1. Crop an image to many regions
2. Apply a CNN to classify each region

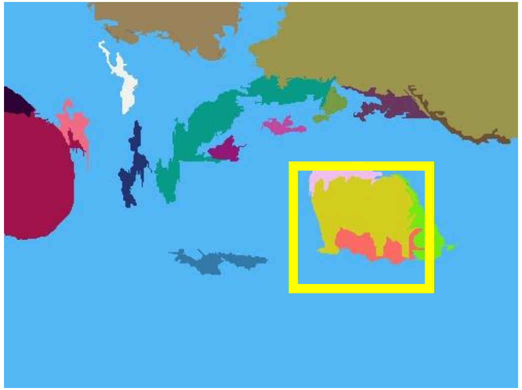
☹ We need to sample ~100,000 regions to get tight boxes around object instances!

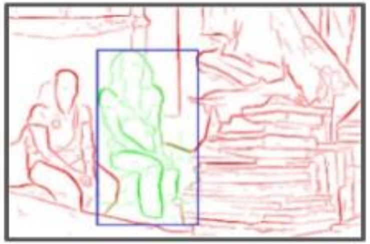
Object proposals

Q. Is there a smart (quick & accurate) way of picking fewer regions that are likely to contain objects?

A. Measure objectness

- Saliency + Contrast + Edge Density + Superpixel/Contour Straddling

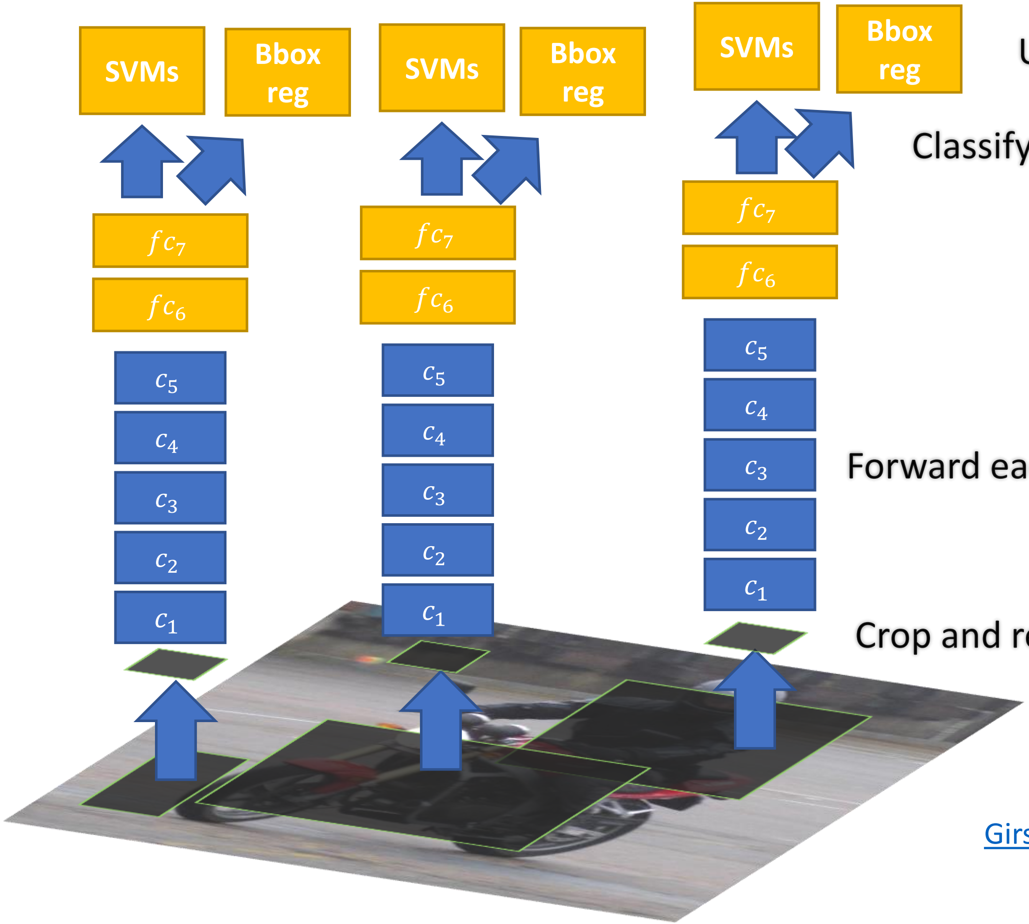




Relatively fast, computes 1-2k regions in few seconds!

Alexe et al. (2010), "What is an object?", CVPR
Uijlings et al. (2013) "Selective search for object recognition." IJCV
Zitnick et al. (2014) "Edge boxes: Locating object proposals from edges." ECCV

(R)egion-CNN



Update box coordinates

Classify regions

Forward each region through CNN

Crop and resize regions

Extract region proposals (~2k)

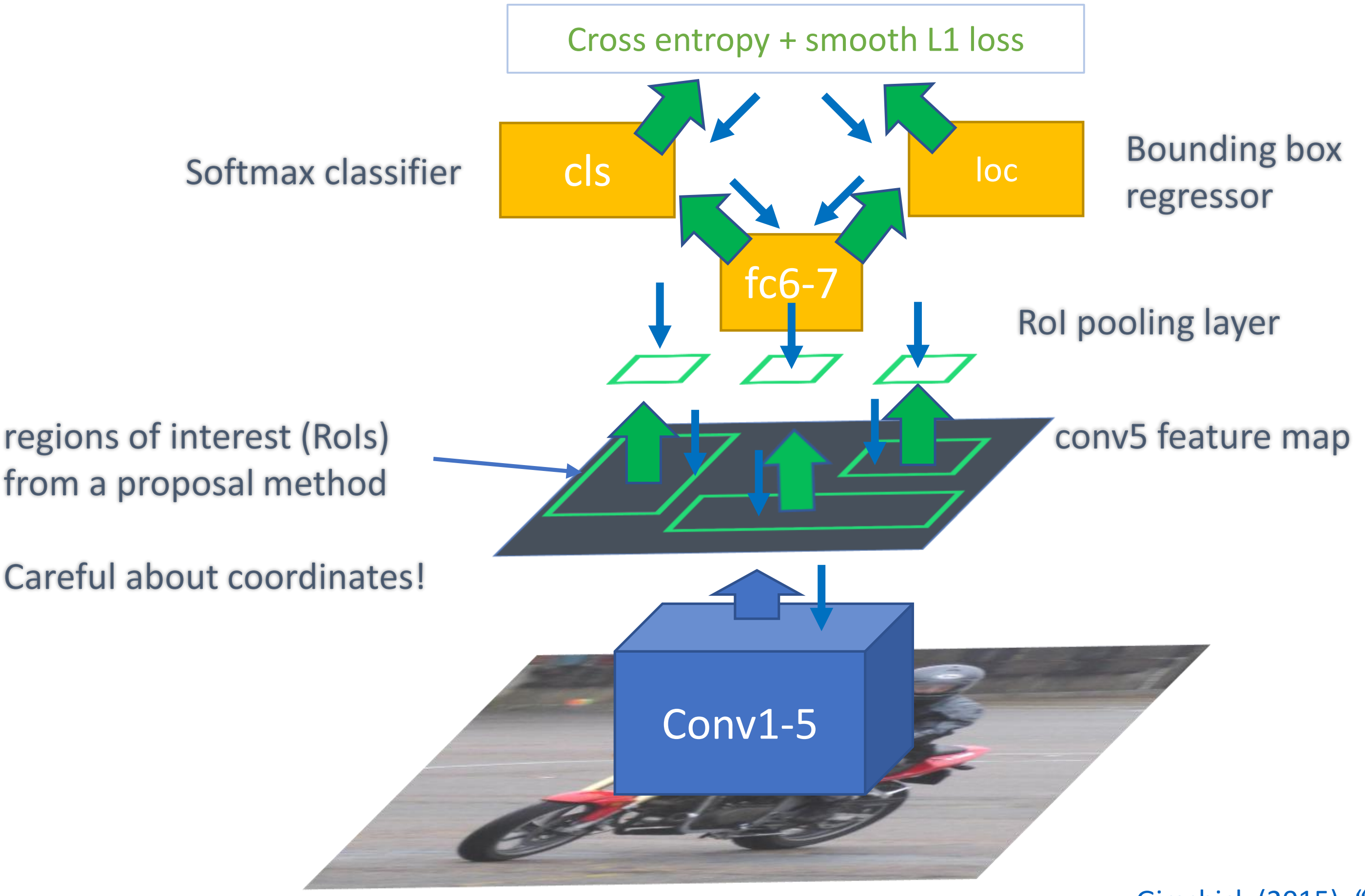
[Girschick et al. (2014), "Rich feature hierarchies for accurate object detection ...", CVPR]

What is wrong with R-CNN?

- Training is multi-stage pipeline
  - Fine-tune network with softmax classifier
  - Train post-hoc linear SVMs
  - Train post-hoc bounding box regressor
- Training is slow (84h), takes a lot of disk space
- Inference (test time) is slow
  - 47s / image with VGG16 [Simonyan & Zisserman ICLR15]
  - Fixed by SPP-net [He et al. ECCV14]

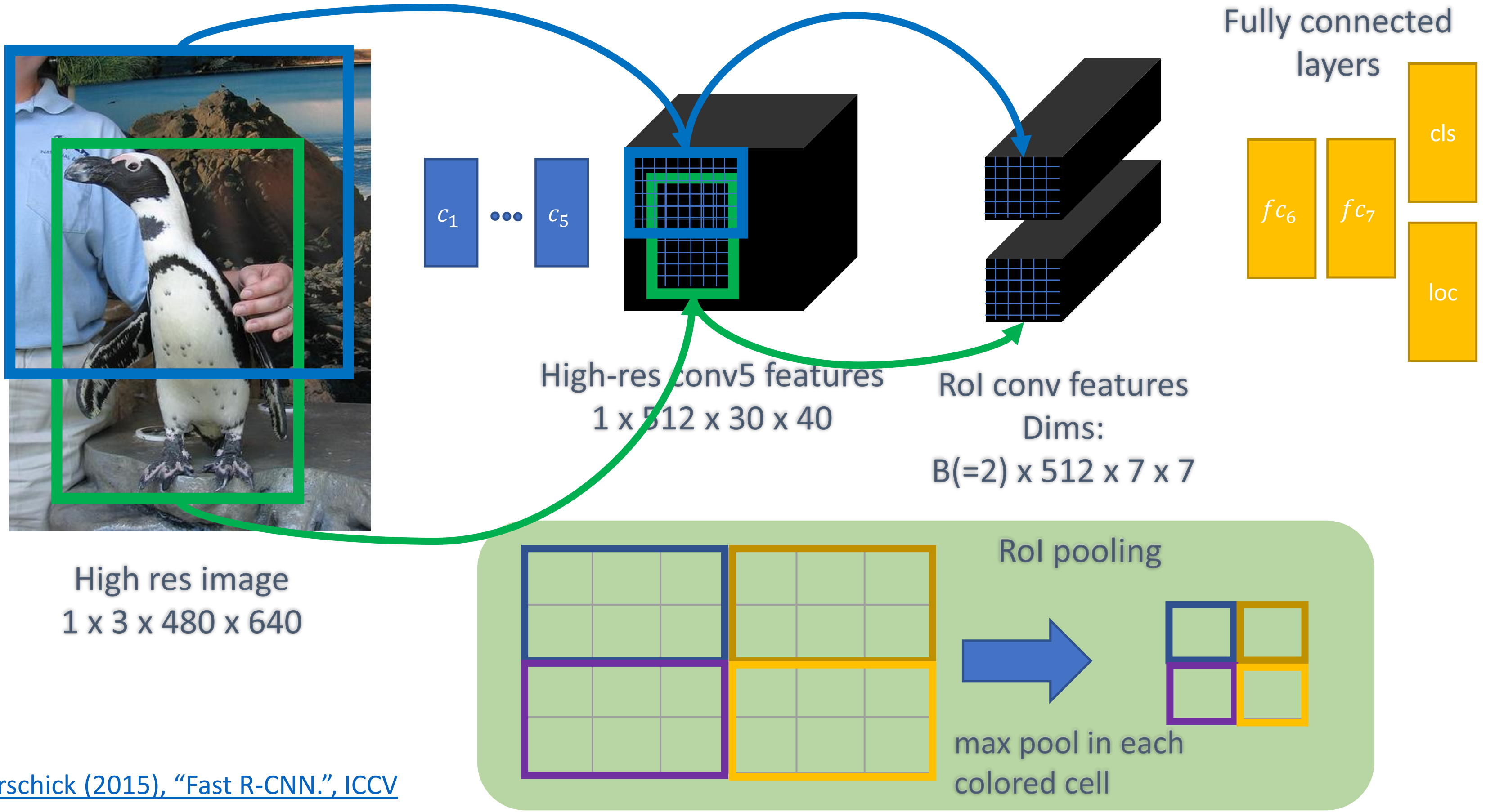
Girschick et al. (2014), "Rich feature hierarchies for accurate object detection and semantic segmentation.", CVPR

Fast R-CNN



[Girschick (2015), "Fast R-CNN.", ICCV](#)

Fast R-CNN: RoI pooling



Girschick (2015), "Fast R-CNN.", ICCV

Fast R-CNN vs R-CNN

	Fast R-CNN	R-CNN
Train time (h)	9.5	84
- Speedup	8.8x	1x
Test time / image	0.32s	47.0s
Test speedup	146x	1x
Accuracy (mean AP)	66.9%	66.0%

- Results on PASCAL VOC 2007 dataset
- Base CNN is VGG16

Girschick (2015), “Fast R-CNN.”, ICCV

What is still wrong with Fast R-CNN?

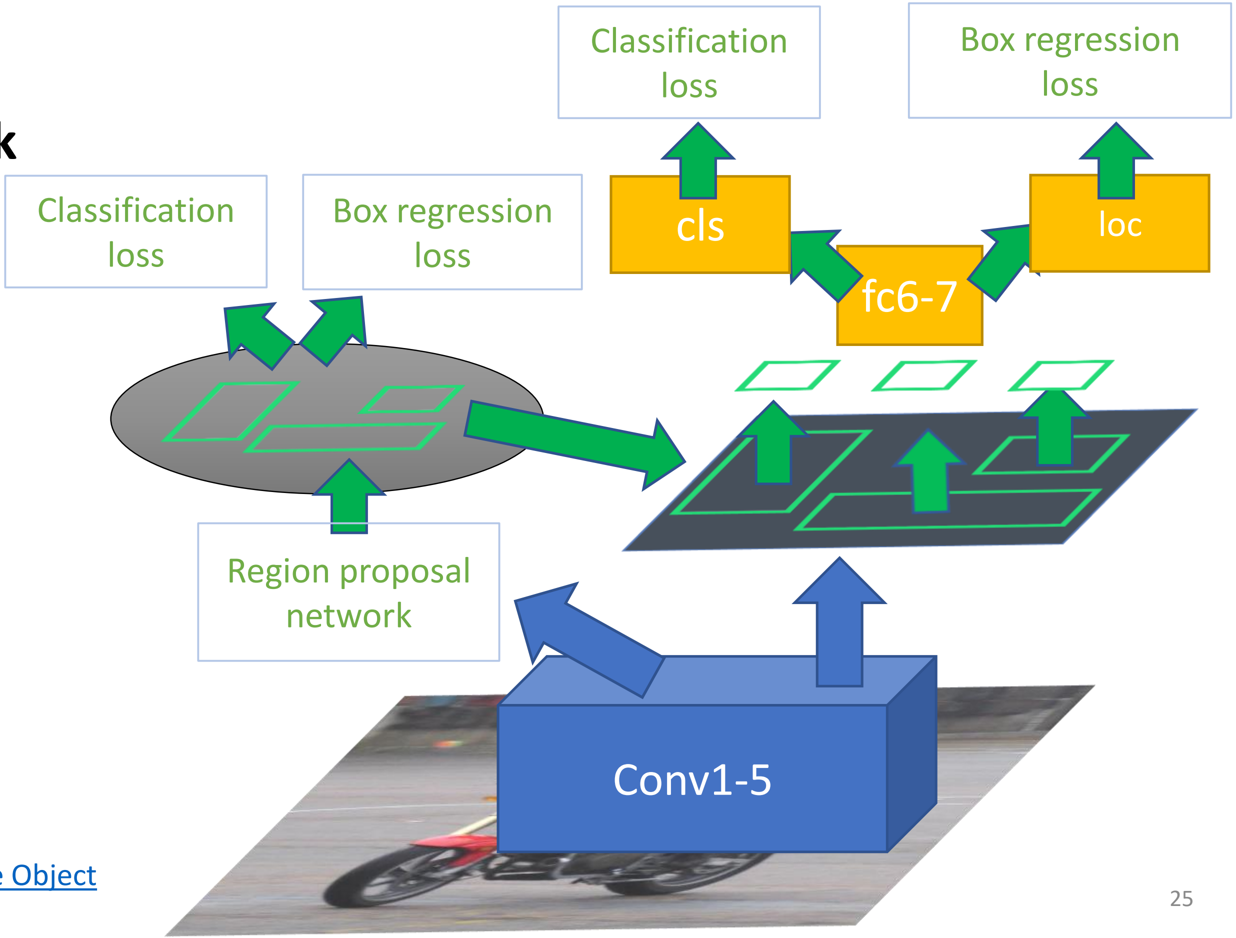
- Out-of-network object proposals
  - Selective search: 2s / im;
    EdgeBoxes: 0.2s / im
- Can we learn better/faster object proposals?
  - Fast(er) R-CNN

Faster R-CNN

A new sub-network: **Region Proposal Network (RPN)** predicts object proposals from features

Jointly train with 4 losses

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



[Ren et al (2015), "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NIPS]

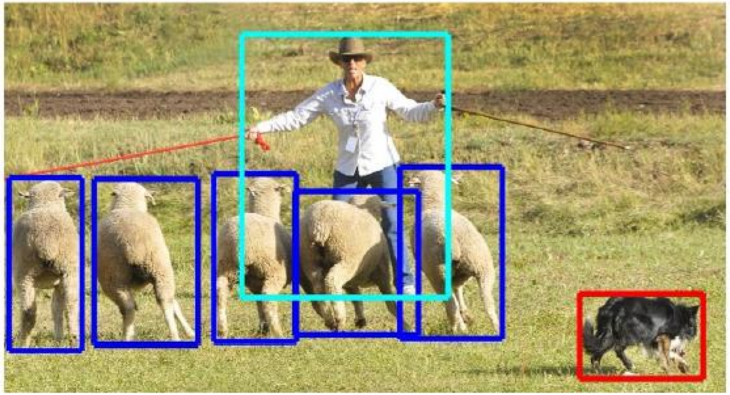
Fast vs Faster R-CNN

	Faster R-CNN	Fast R-CNN	R-CNN
Test time / image	0.2s	2s	47.0s
Test speedup	235x	23.5x	1x
Accuracy (mean AP)	69.9%	66.9%	66.0%

- RPN with 300 proposals can do better than 2k external region proposals
- It is faster due to shared feature computation

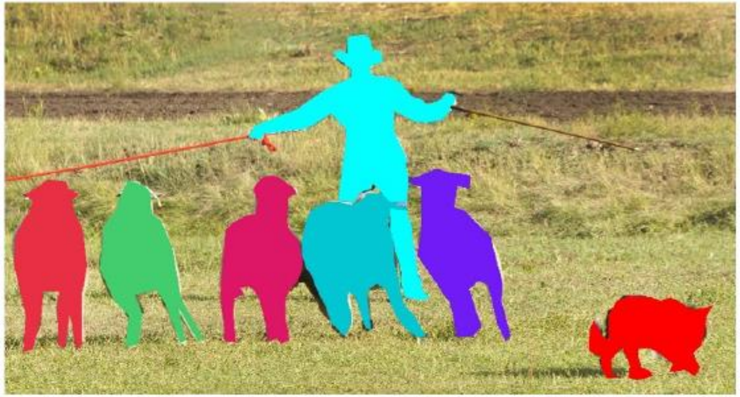
Ren et al (2015), "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NIPS

image classification



object detection

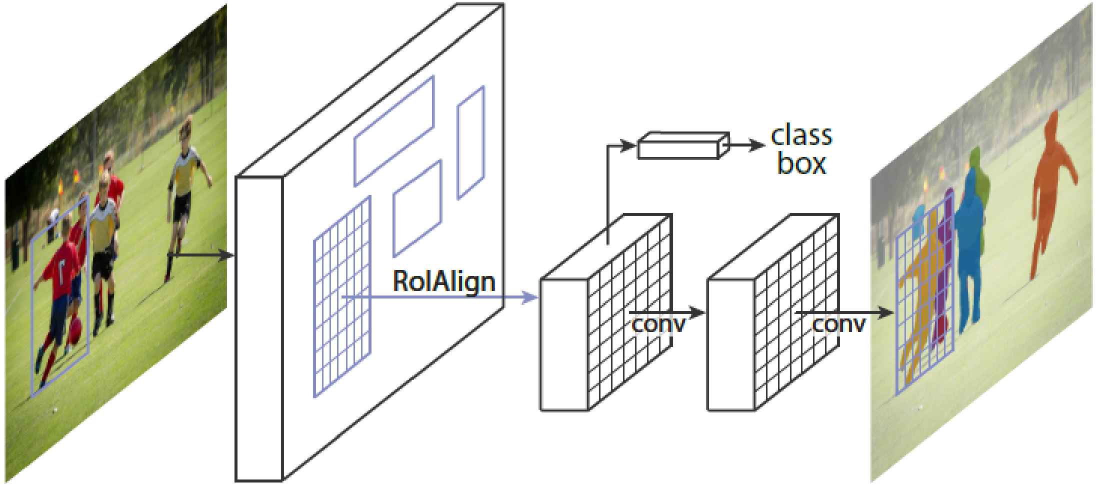
semantic segmentation



instance segmentation

Image credits: Microsoft COCO

Mask R-CNN



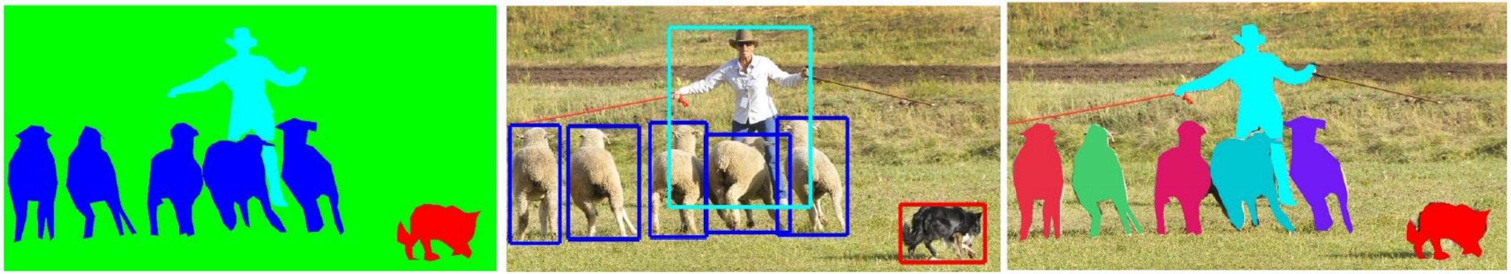
- Builds on Faster R-CNN
- Additionally predicts a mask for each box
- Uses an improved RoI pooling (RoIAlign)

[He et al. (2017) “Mask r-cnn” ICCV](#)

Mask R-CNN

He et al. (2017) "Mask r-cnn" ICCV

Summary



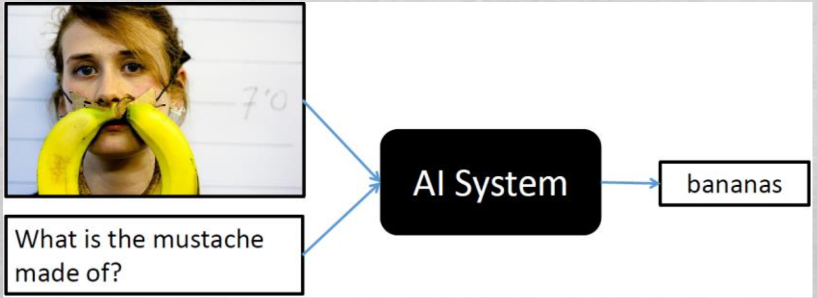
semantic segmentation object detection instance segmentation

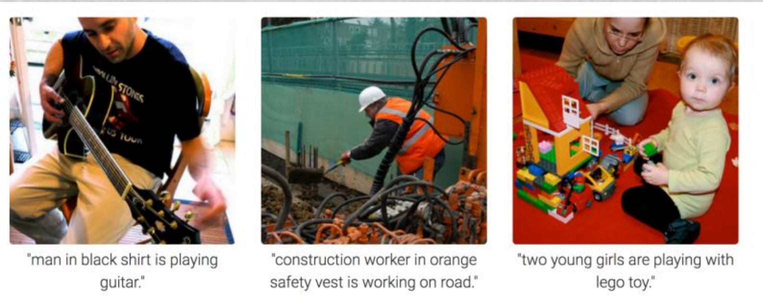
Recommended

- [Girschick (2015), “Faster R-CNN.” ICCV]
- [Nice blog about semantic segmentation by Arthur Ouaknine]

Additional

- [Long et al. (2015) "Fully Convolutional Networks for Semantic Segmentation", CVPR]





Next lecture