Deep Learning 3
Visualizing Network Internals & Fully Convolutional Networks

Computer Vision

James Hays

Many slides from CVPR 2014 Deep Learning Tutorial (Honglak Lee and Marc'Aurelio especially) and Rob Fergus

https://sites.google.com/site/deeplearningcvpr2014

Project 6 out today

Recap

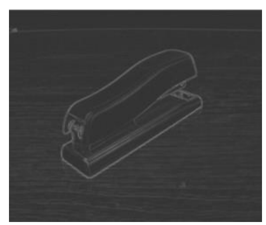
Lecture 1 Neural Networks
Lecture 2 Convolutional Deep Neural Networks (e.g. AlexNet)

Traditional Recognition Approach

Features are not learned

Input data (pixels) → feature representation (hand-crafted) → Learning Algorithm (e.g., SVM)

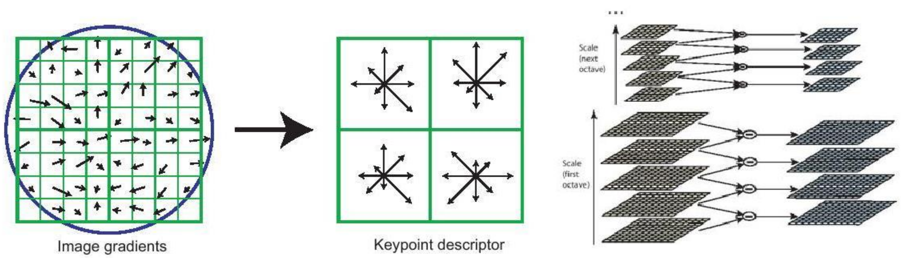
Image



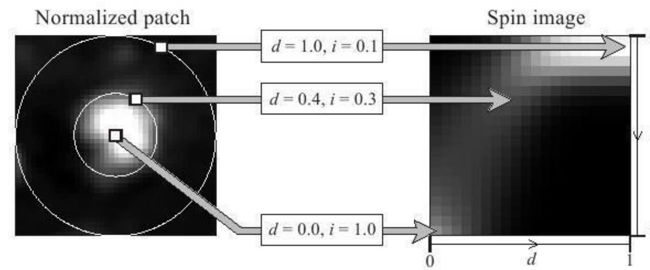
Low-level vision features (edges, SIFT, HOG, etc.)

Object detection / classification

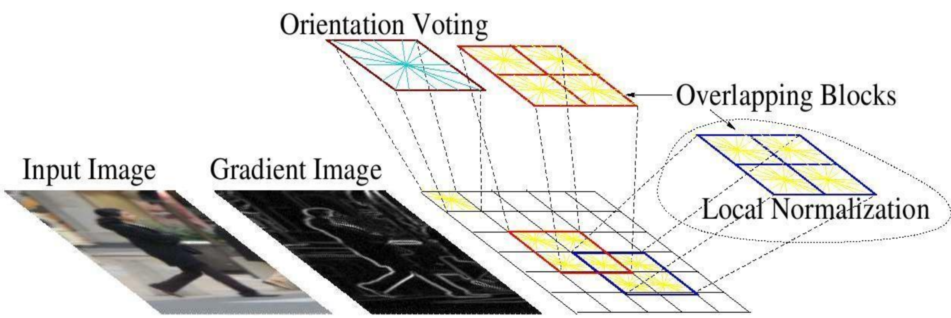
Computer vision features



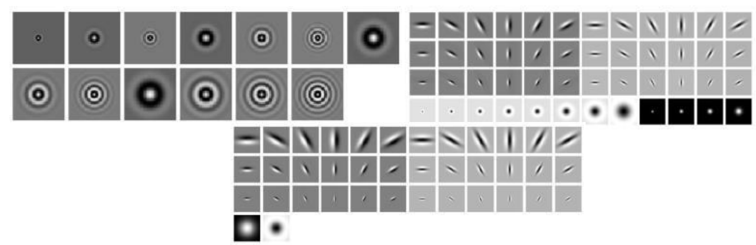
SIFT



Spin image



HoG



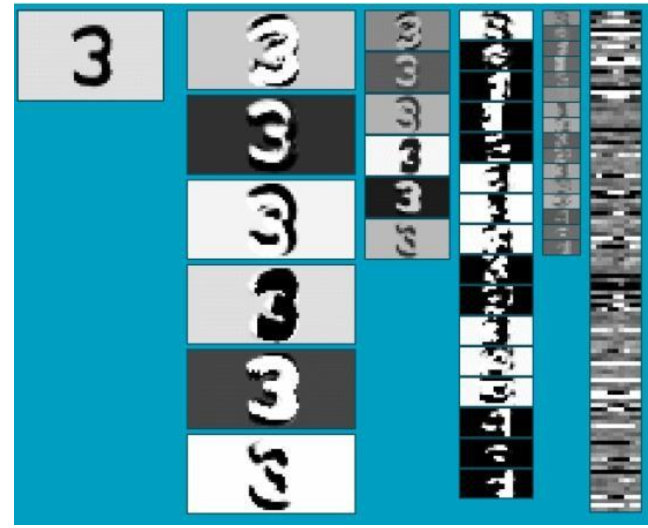
Textons

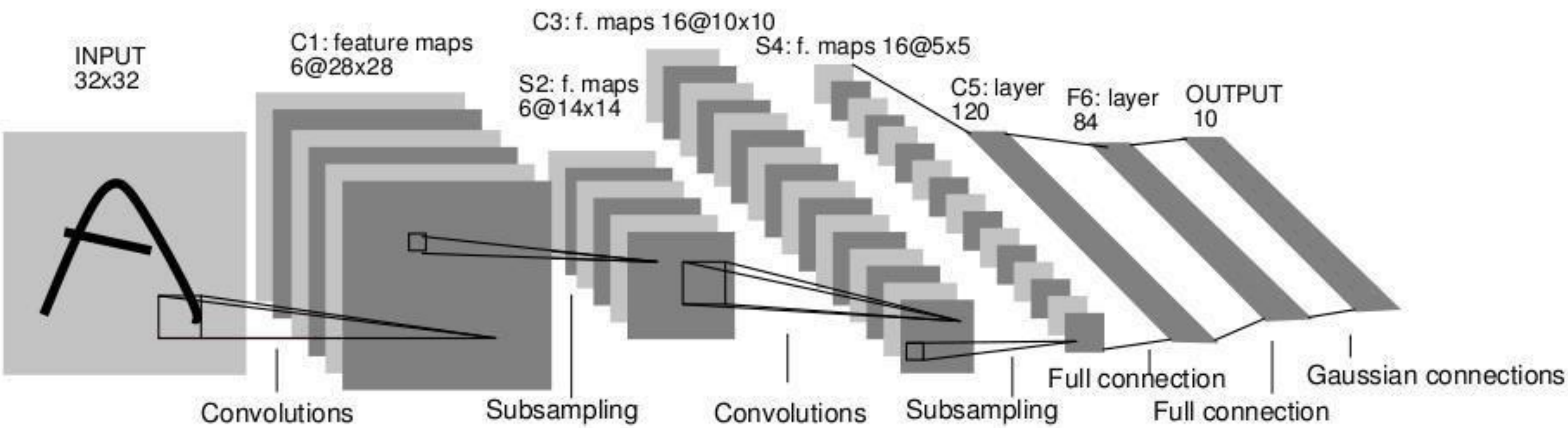
and many others:

SURF, MSER, LBP, Color-SIFT, Color histogram, GLOH, …..

Example: Convolutional Neural Networks

- LeCun et al. 1989
- Neural network with specialized connectivity structure



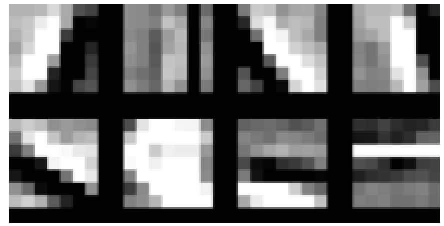


Slide: R. Fergus

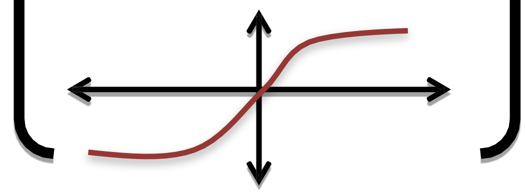
Components of Each Layer

Pixels / Features

Filter with Dictionary (convolutional or tiled)

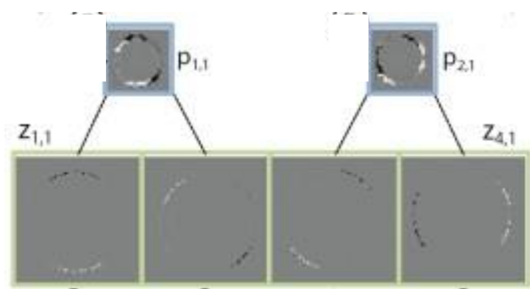


+ Non-linearity



[Optional]

Spatial/Feature (Sum or Max)



[Optional]

Normalization between feature responses

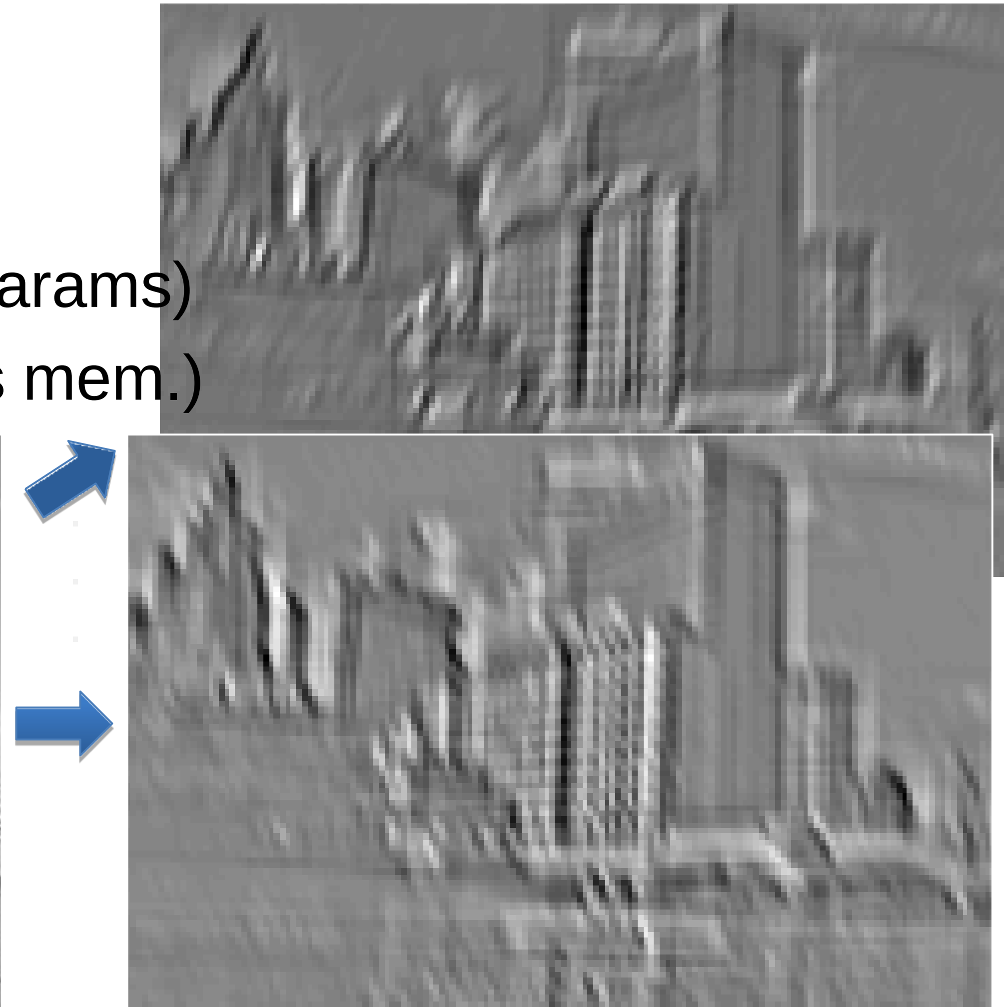
Output Features

Slide: R. Fergus

Filtering

- Convolutional
  - Dependencies are local
  - Translation equivariance
  - Tied filter weights (few params)
  - Stride 1,2,… (faster, less mem.)

Input



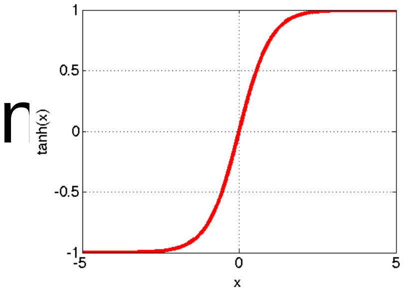
Feature Map

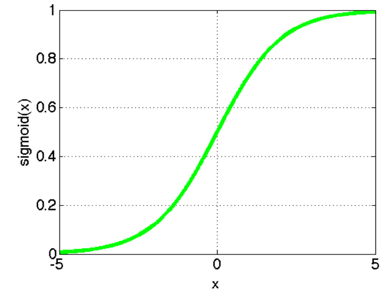
Slide: R. Fergus

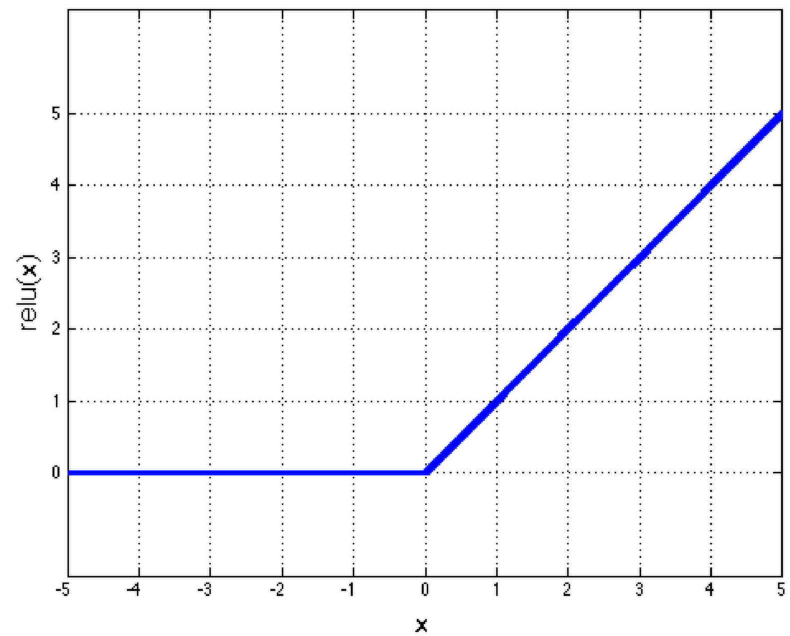
Non-Linearity

- Non-linearity
  - Per-element (independen
  - Tanh
  - Sigmoid: 1/(1+exp(-x))
  - Rectified linear
    - Simplifies backprop
    - Makes learning faster
    - Avoids saturation issues

    → Preferred option



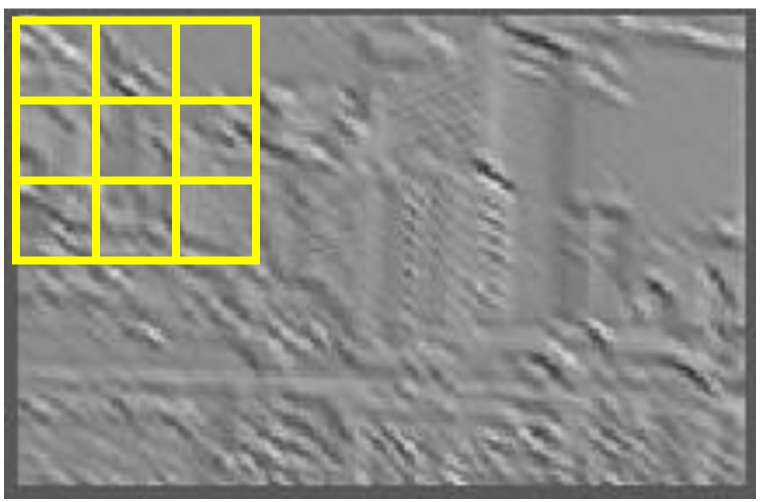




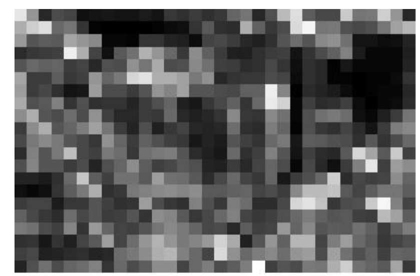
Slide: R. Fergus

Pooling

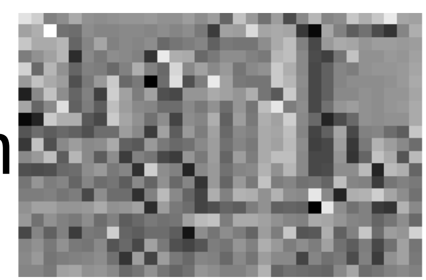
- Spatial Pooling
  - Non-overlapping / overlapping regions
  - Sum or max
  - Boureau et al. ICML’10 for theoretical analysis



Max



Sum

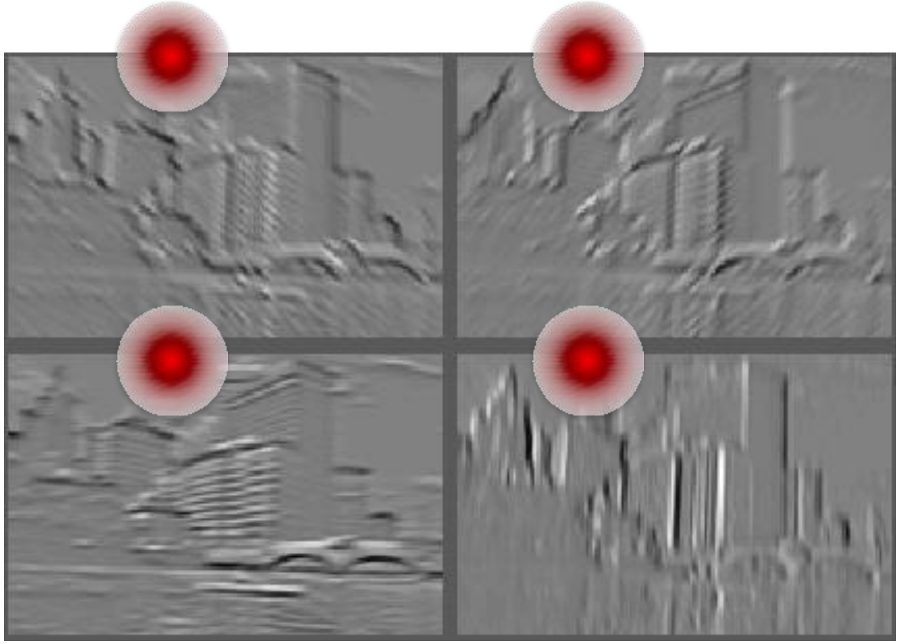


Slide: R. Fergus

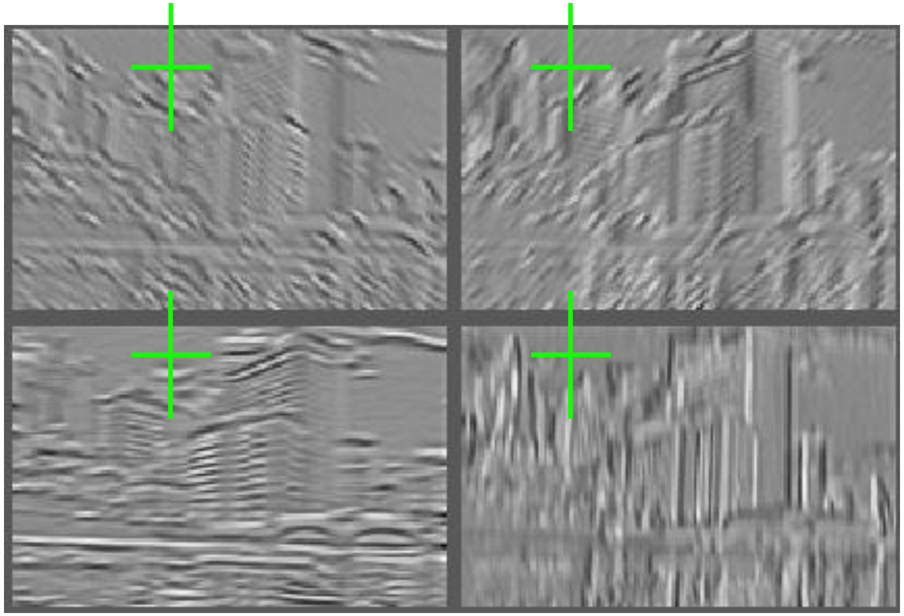
Normalization

- Contrast normalization (across feature maps)
  - Local mean = 0, local std. = 1, “Local” → 7x7 Gaussian
  - Equalizes the features maps

Feature Maps



Feature Maps
After Contrast Normalization

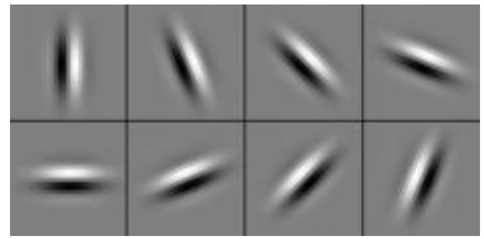


Slide: R. Fergus

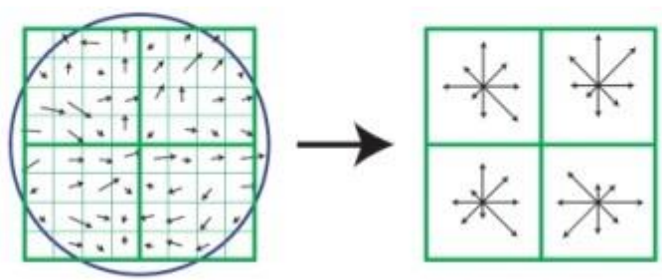
Compare: SIFT Descriptor

Image Pixels

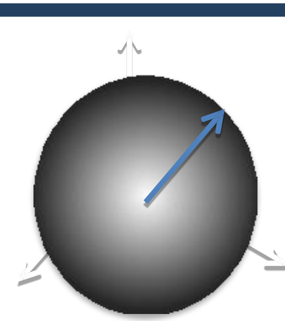
Apply Gabor filters



Spatial pool (Sum)



Normalize to unit length



Feature Vector

Slide: R. Fergus

Visualizing Deep Networks

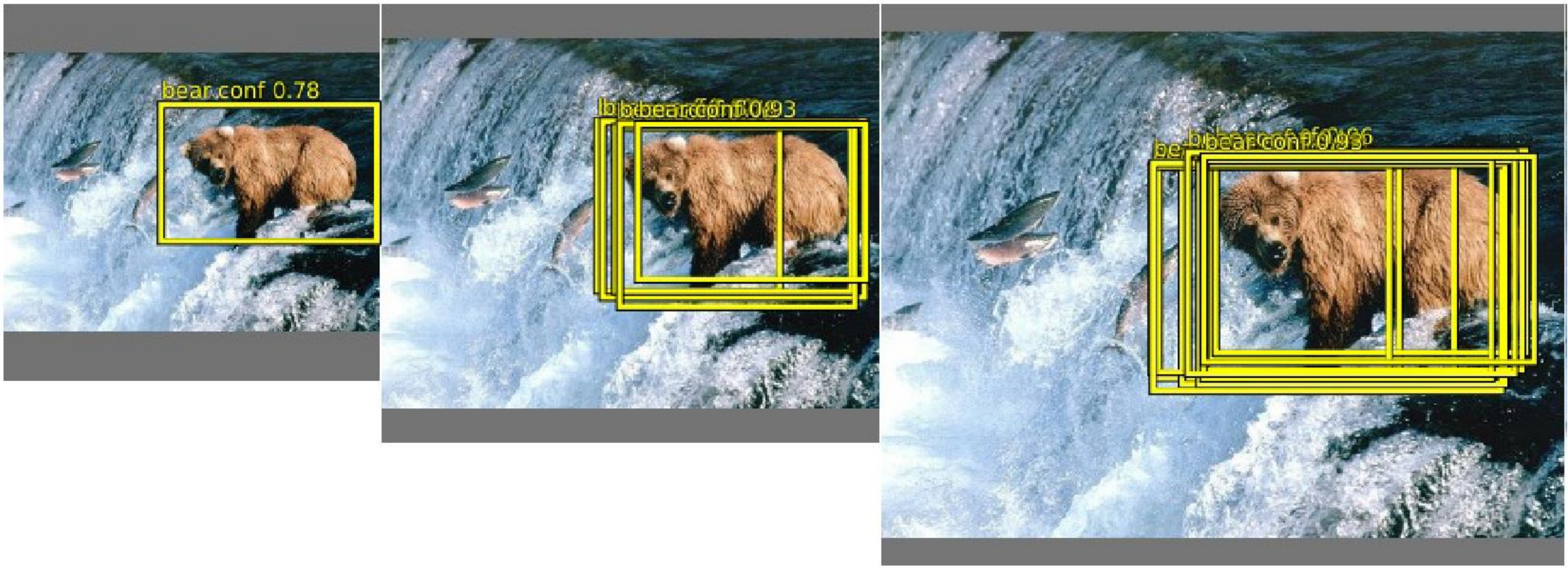
See slides here:

http://places.csail.mit.edu/slide_iclr2015.pdf

Fully Convolutional Networks

CONV NETS: EXAMPLES

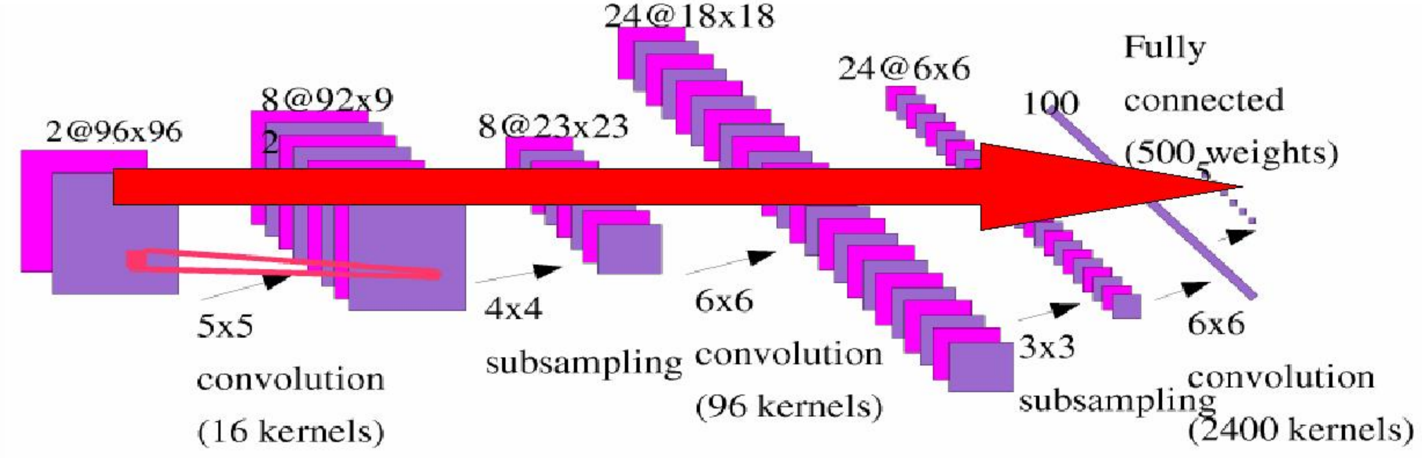
- **Object detection**



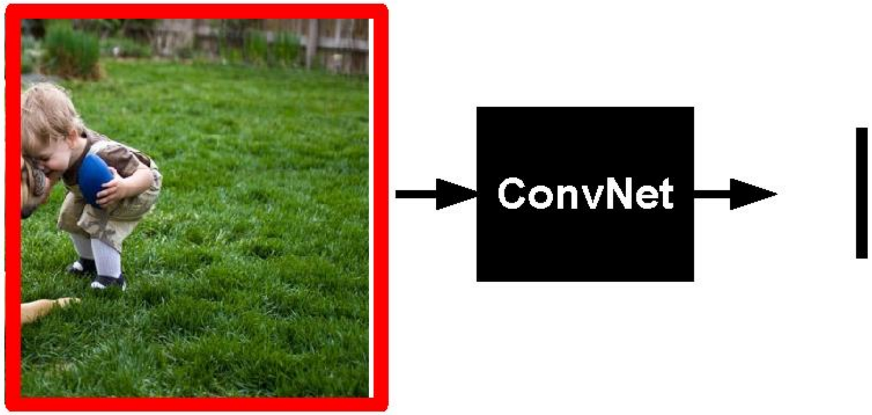
Sermanet et al. "OverFeat: Integrated recognition, localization, ..." arxiv 2013
Girshick et al. "Rich feature hierarchies for accurate object detection..." arxiv 2013
Szegedy et al. "DNN for object detection" NIPS 2013

ConvNets: Test

At test time, run only is forward mode (FPROP).



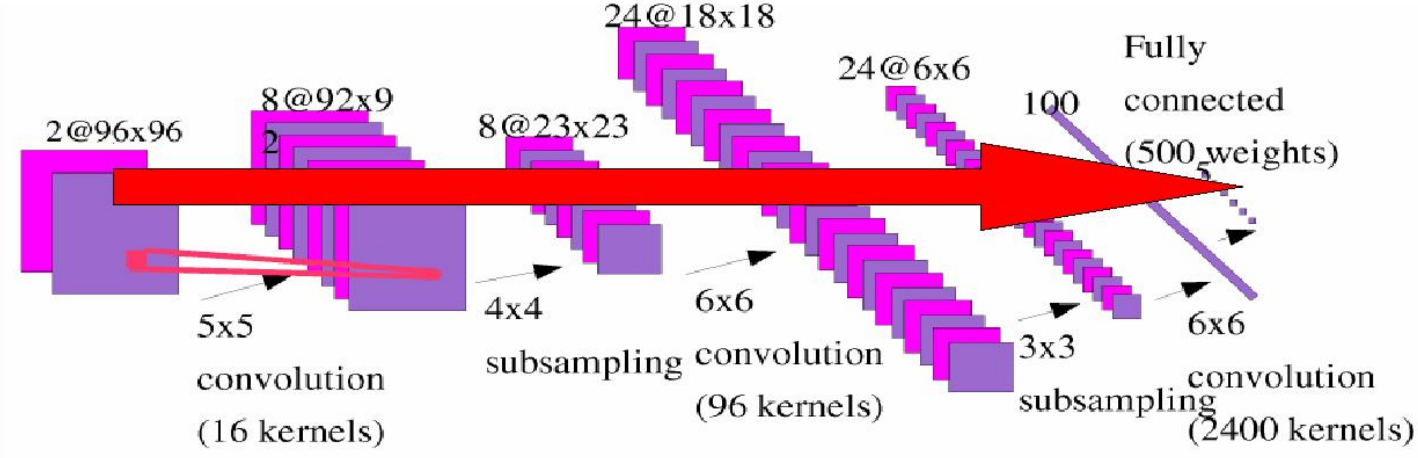
Naturally, convnet can process larger images at little cost.



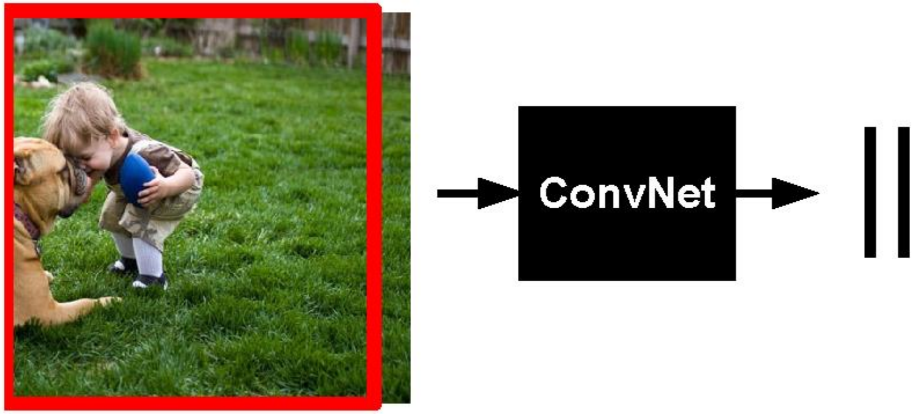
Traditional methods use inefficient sliding windows.

ConvNets: Test

At test time, run only is forward mode (FPROP).



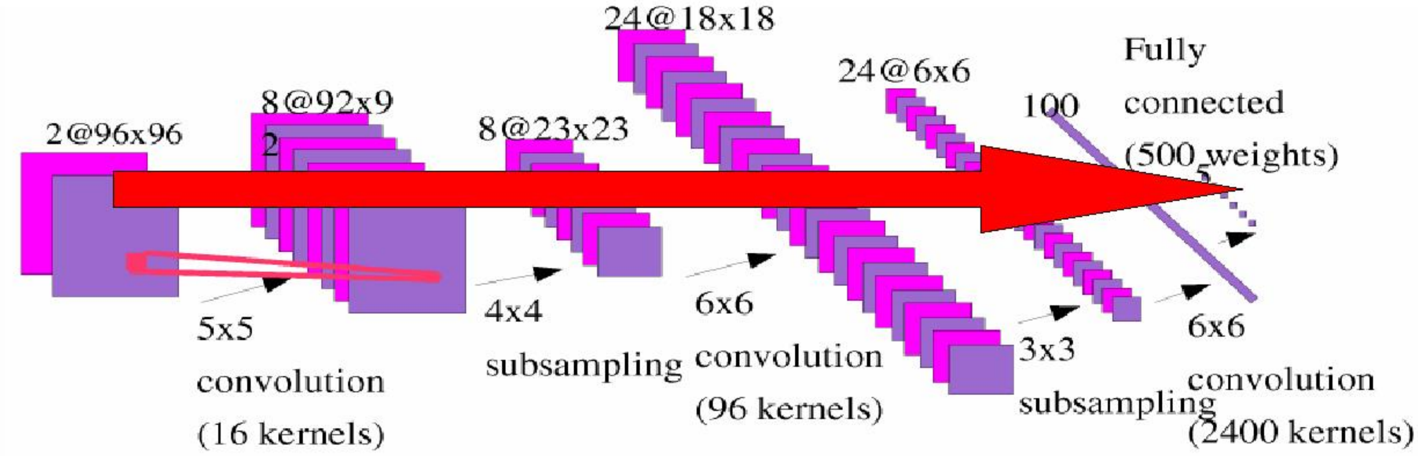
Naturally, convnet can process larger images at little cost.



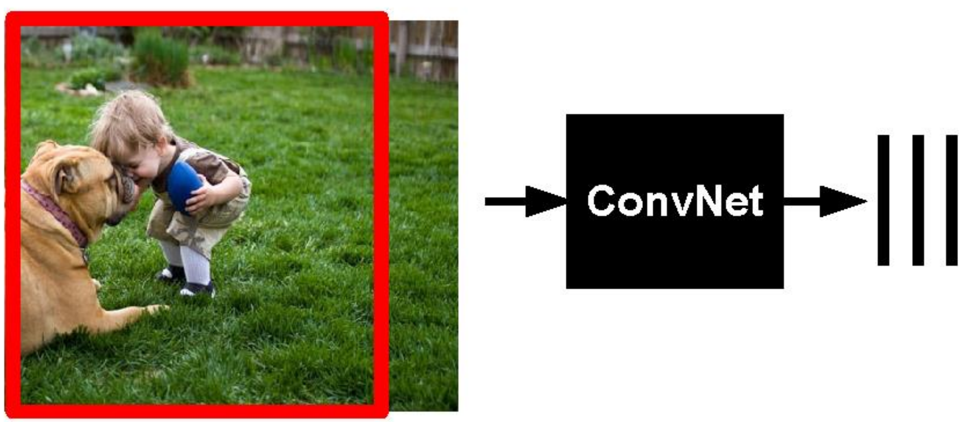
Traditional methods use inefficient sliding windows.

ConvNets: Test

At test time, run only is forward mode (FPROP).



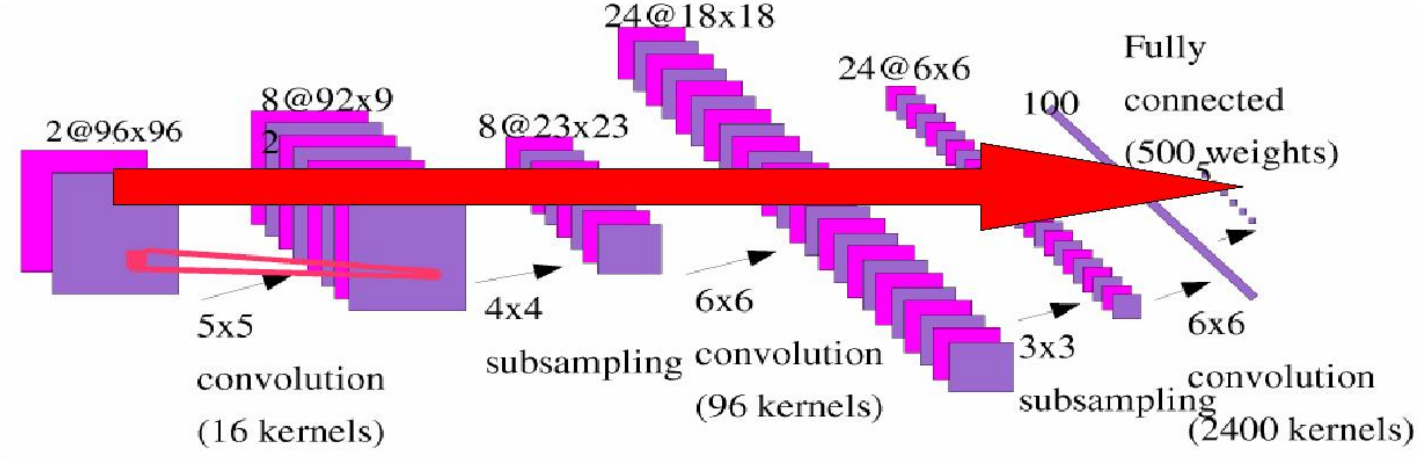
Naturally, convnet can process larger images at little cost.



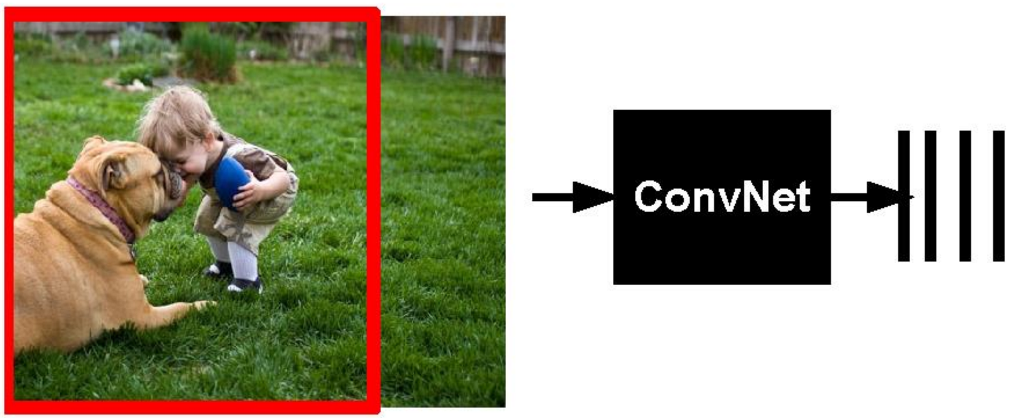
Traditional methods use inefficient sliding windows.

ConvNets: Test

At test time, run only is forward mode (FPROP).



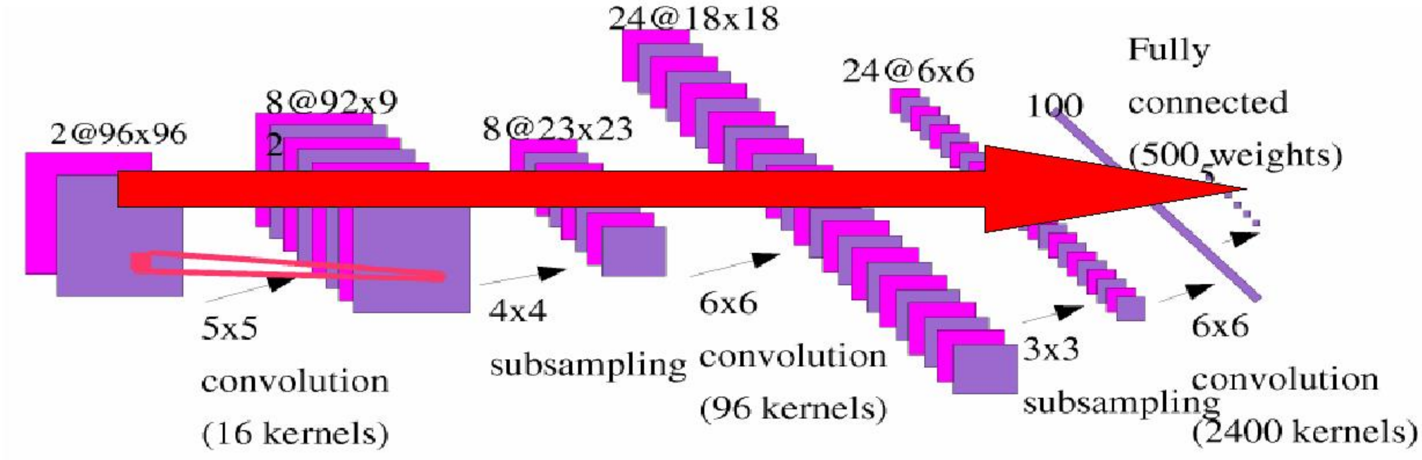
Naturally, convnet can process larger images at little cost.



Traditional methods use inefficient sliding windows.

ConvNets: Test

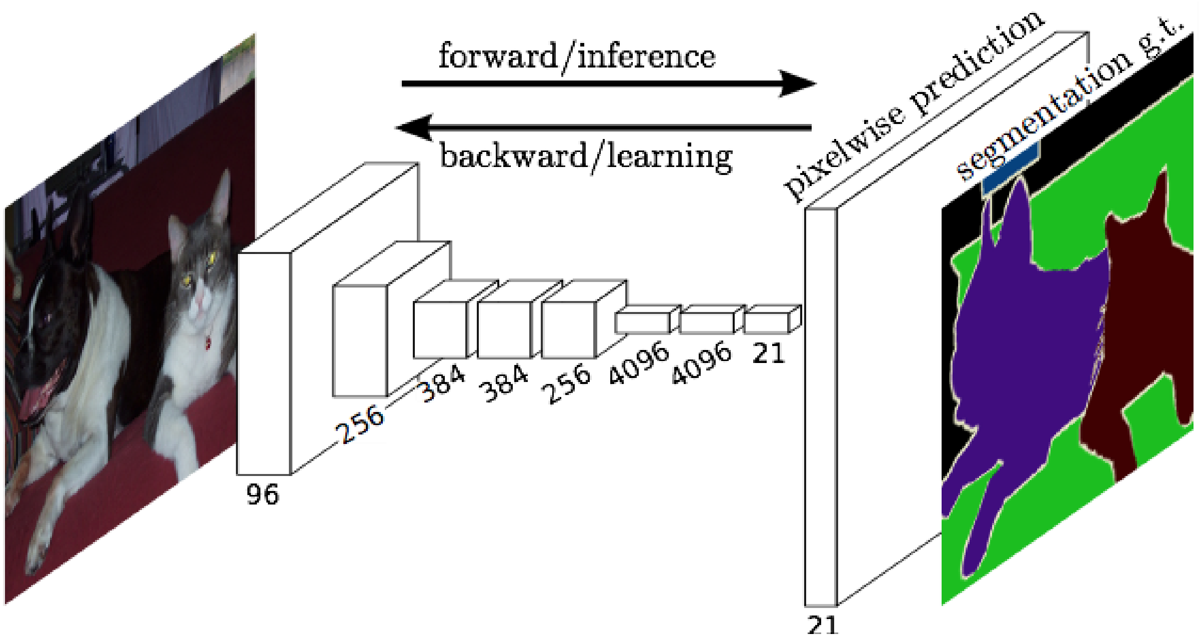
At test time, run only is forward mode (FPROP).



Naturally, convnet can process larger images at little cost.

ConvNet: unrolls convolutions over bigger images and produces outputs at several locations.

Fully Convolutional Networks for Semantic Segmentation

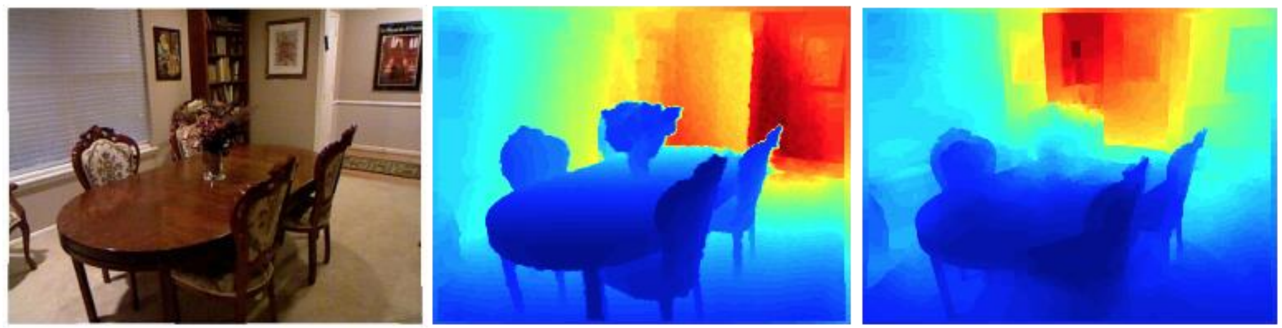


Jonathan Long* Evan Shelhamer* Trevor Darrell
UC Berkeley

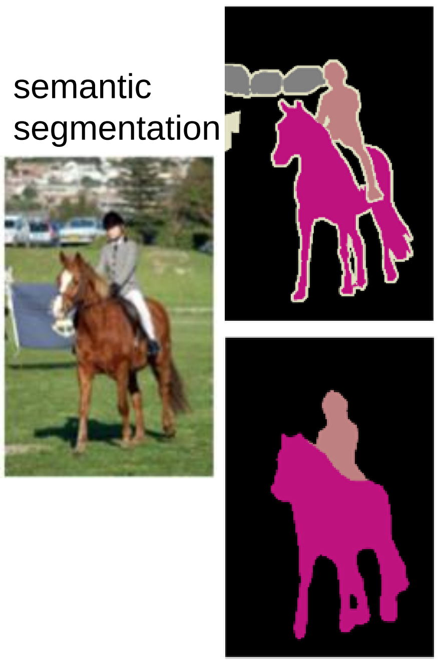
Slides from Long, Shelhamer, and Darrell

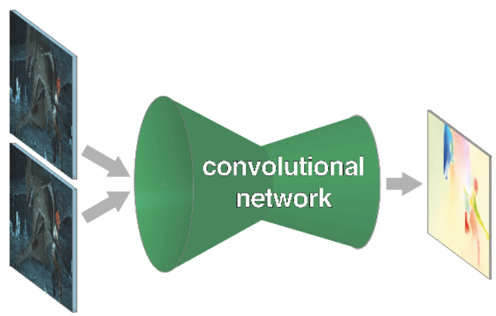
pixels in, pixels out

monocular depth estimation Eigen & Fergus 2015



semantic
segmentation

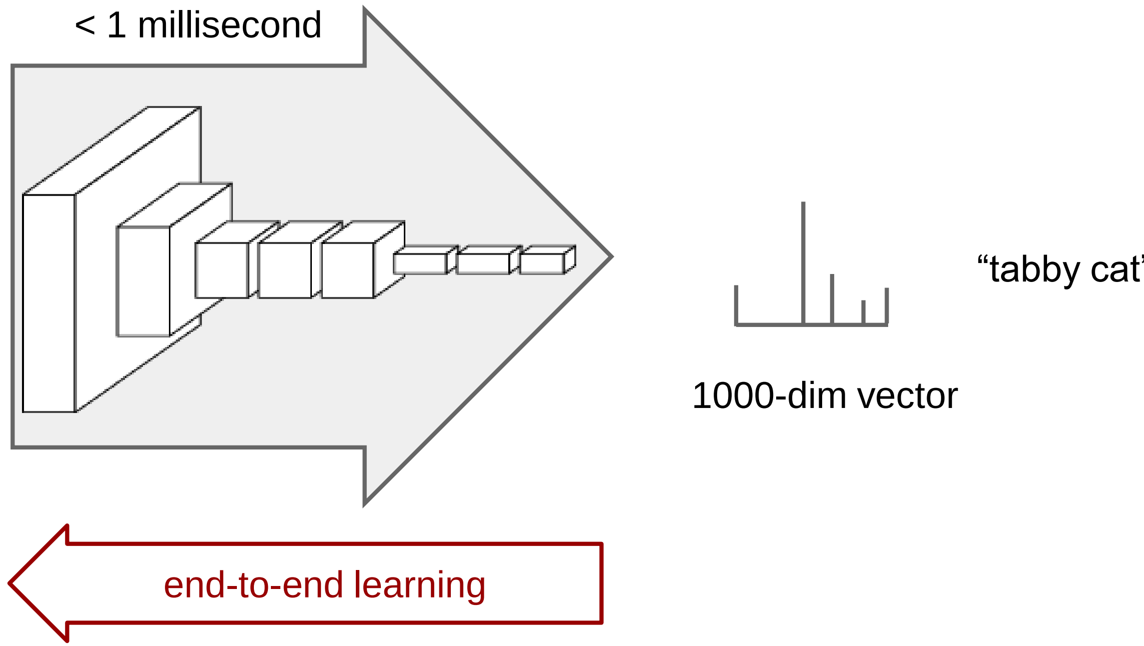




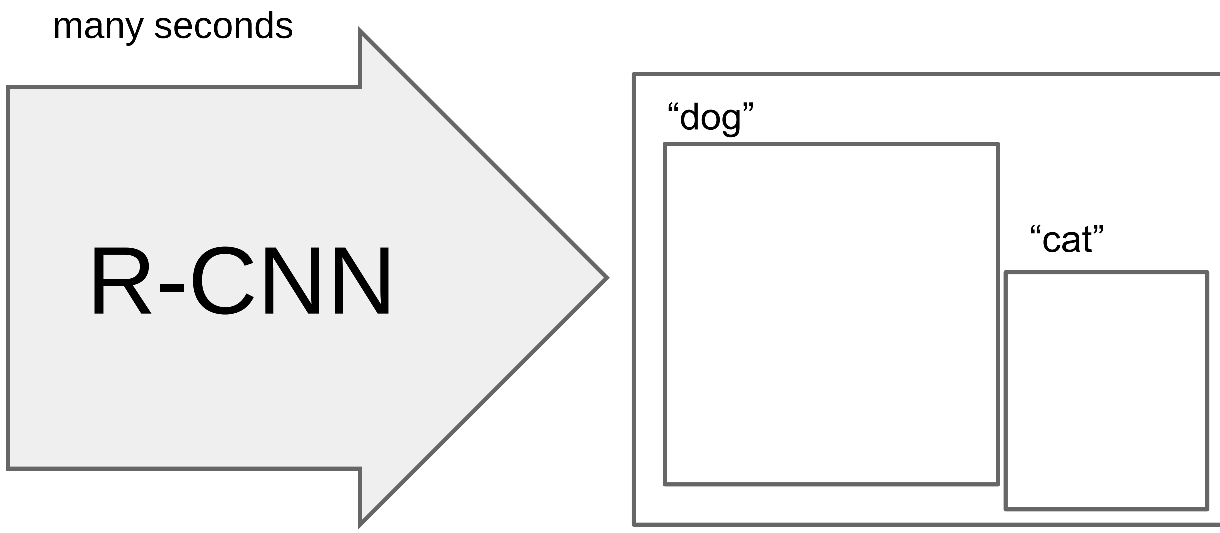
optical flow Fischer et al. 2015

boundary prediction Xie & Tu 2015

convnets perform classification



R-CNN does detection



R-CNN

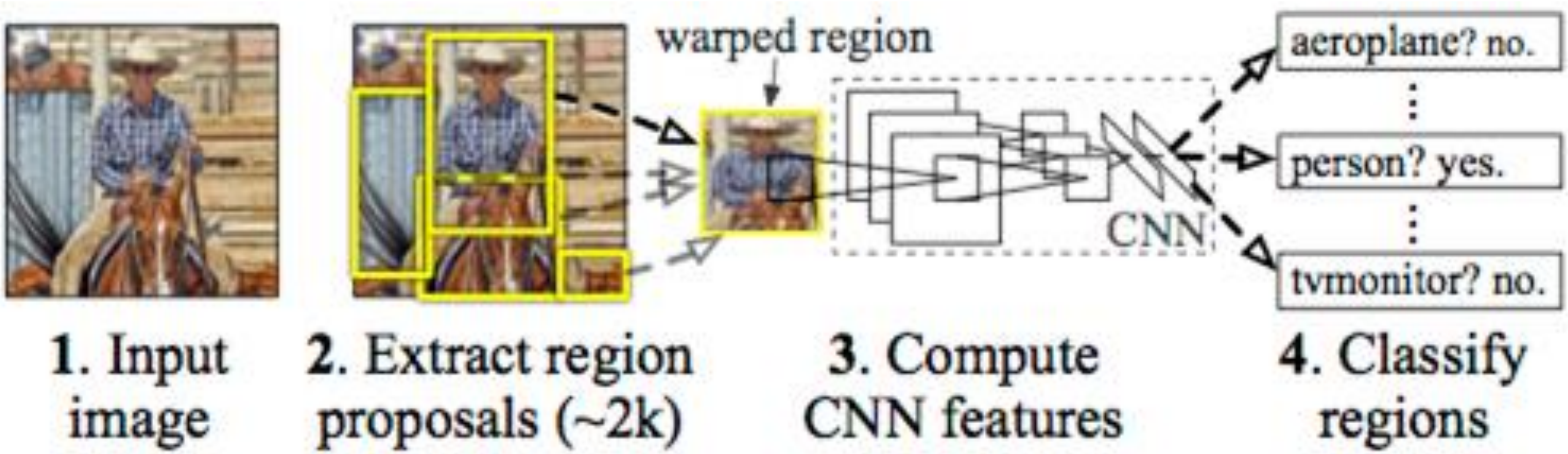
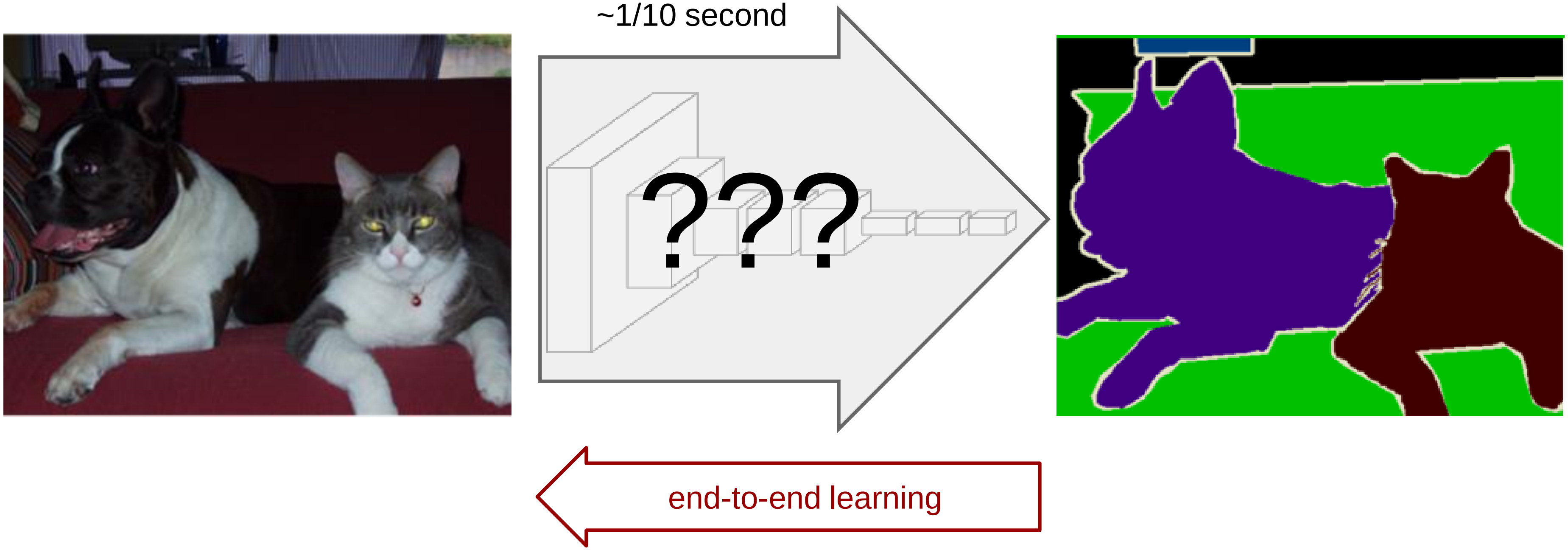
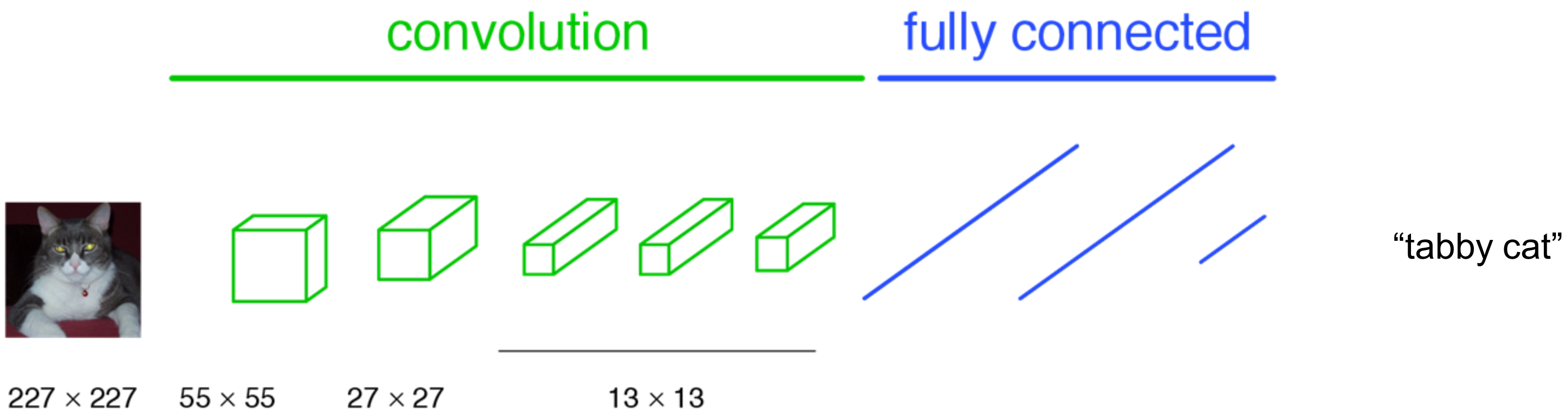


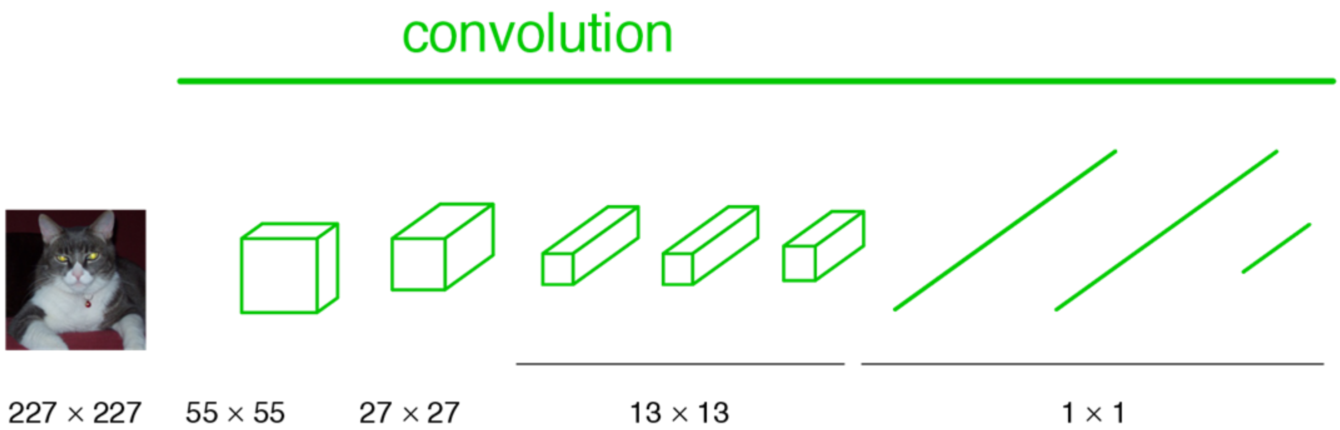
figure: Girshick et al.


~1/10 second
???
end-to-end learning

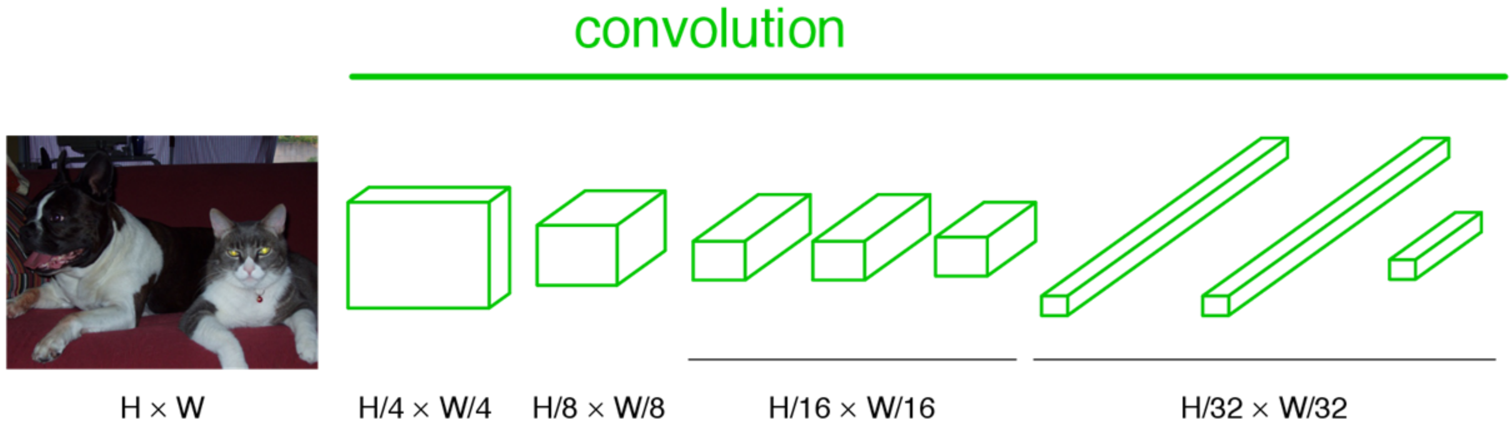
a classification network



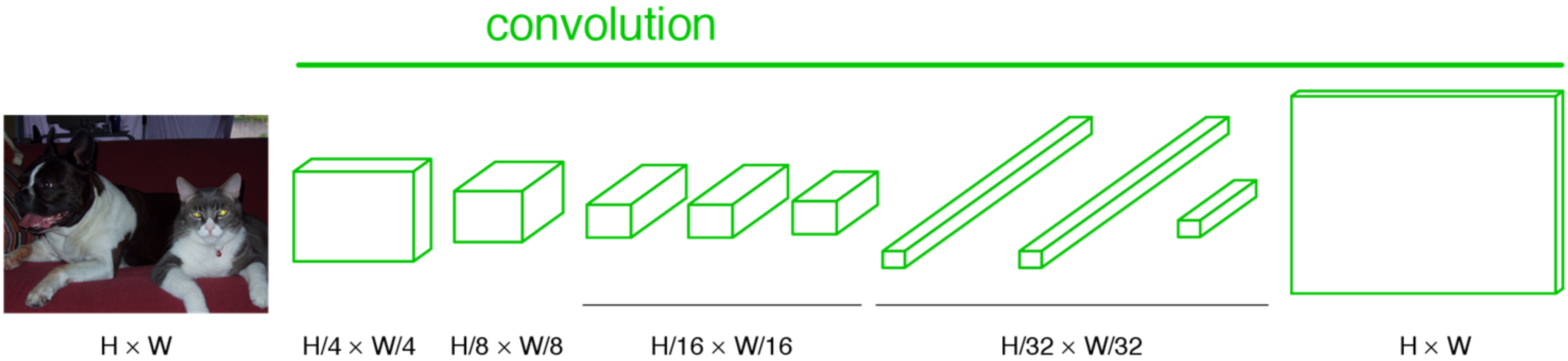
becoming fully convolutional



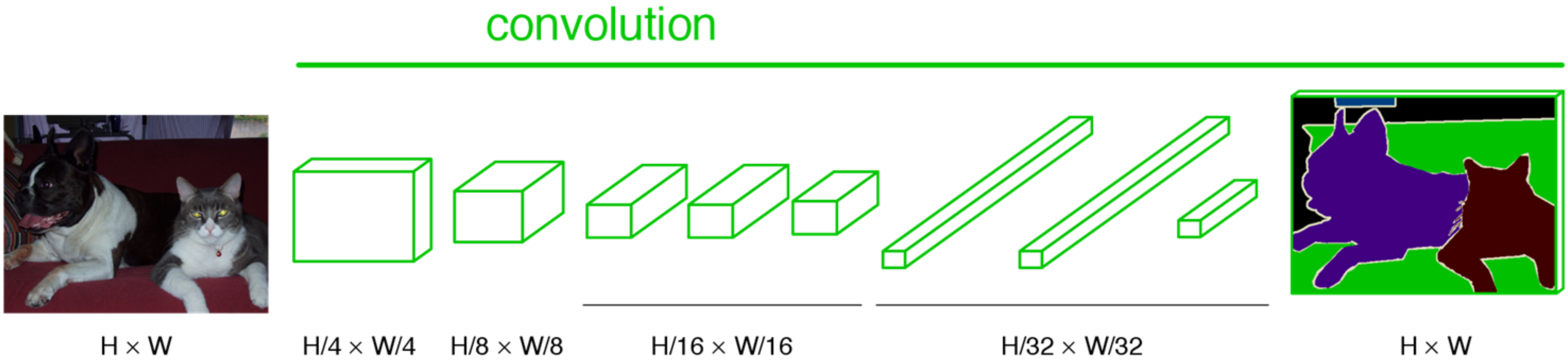
becoming fully convolutional



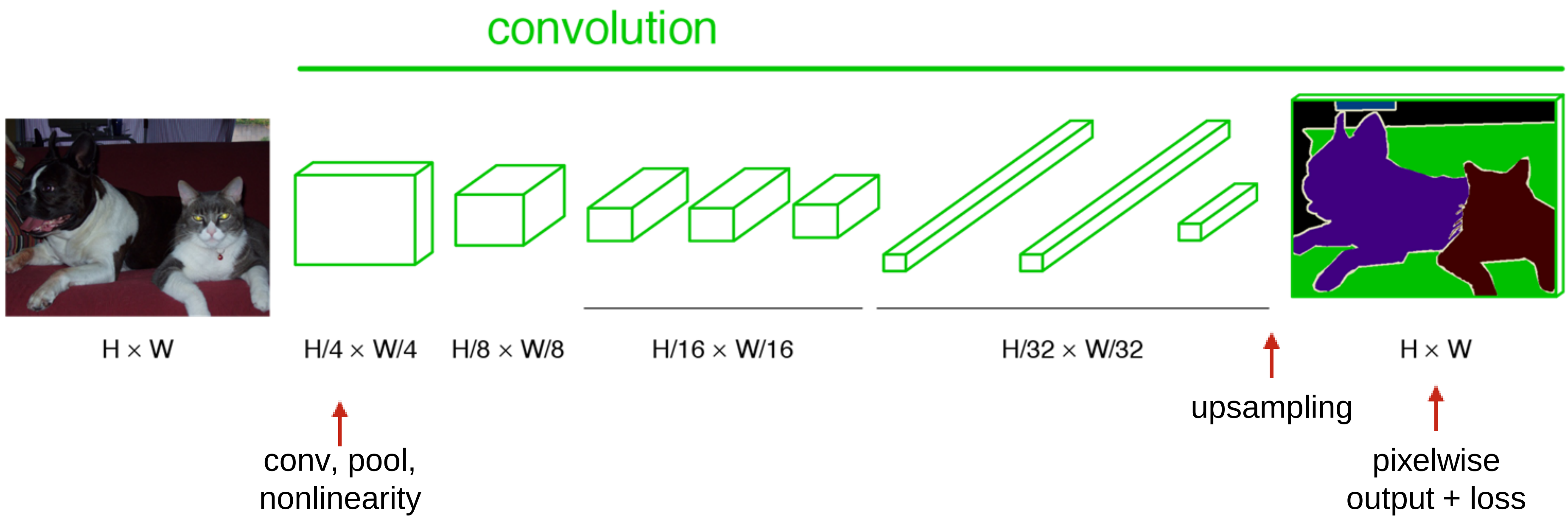
upsampling output



end-to-end, pixels-to-pixels network

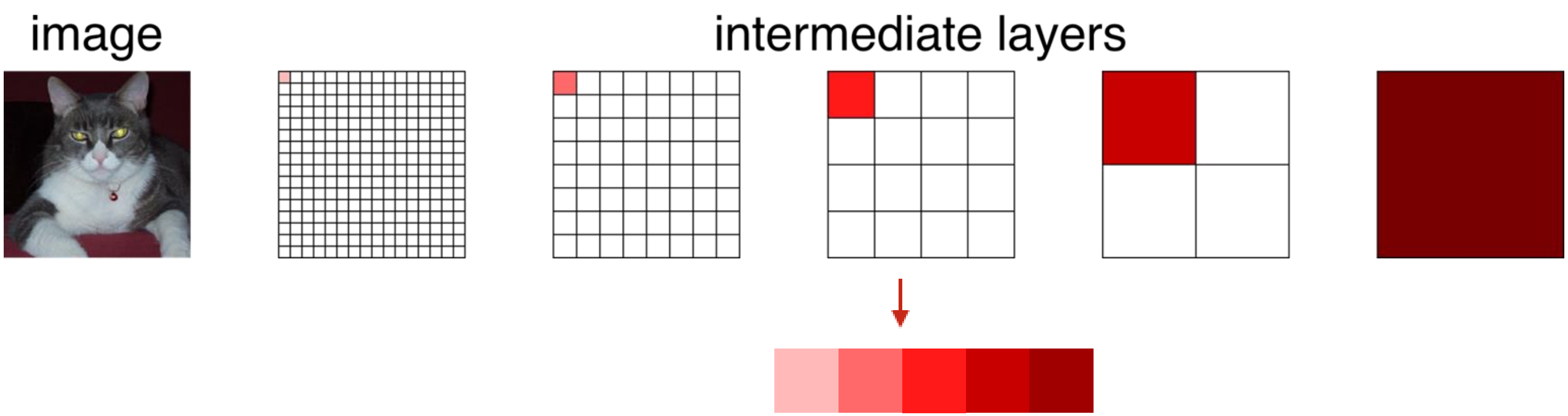


end-to-end, pixels-to-pixels network



spectrum of deep features

combine *where* (local, shallow) with *what* (global, deep)



fuse features into **deep jet**

(cf. Hariharan et al. CVPR15 “hypercolumn”)

skip layers

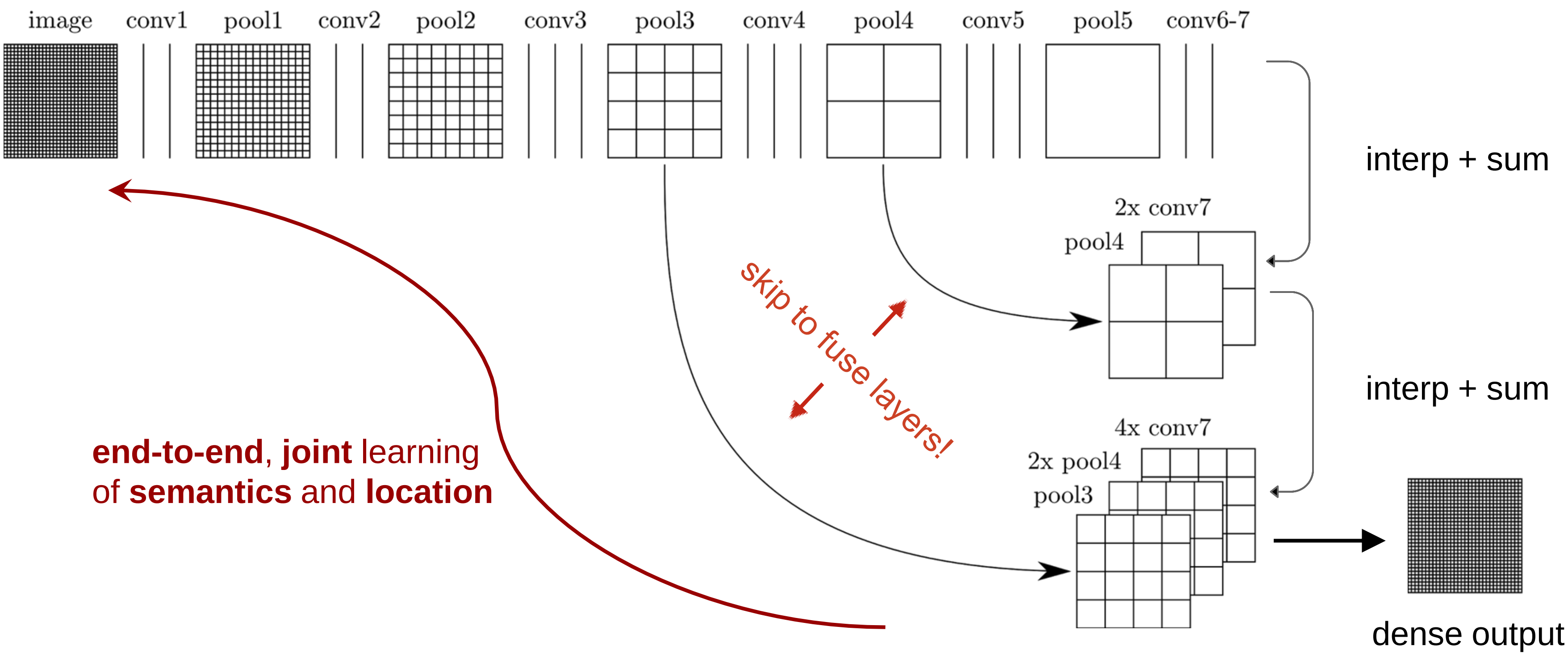
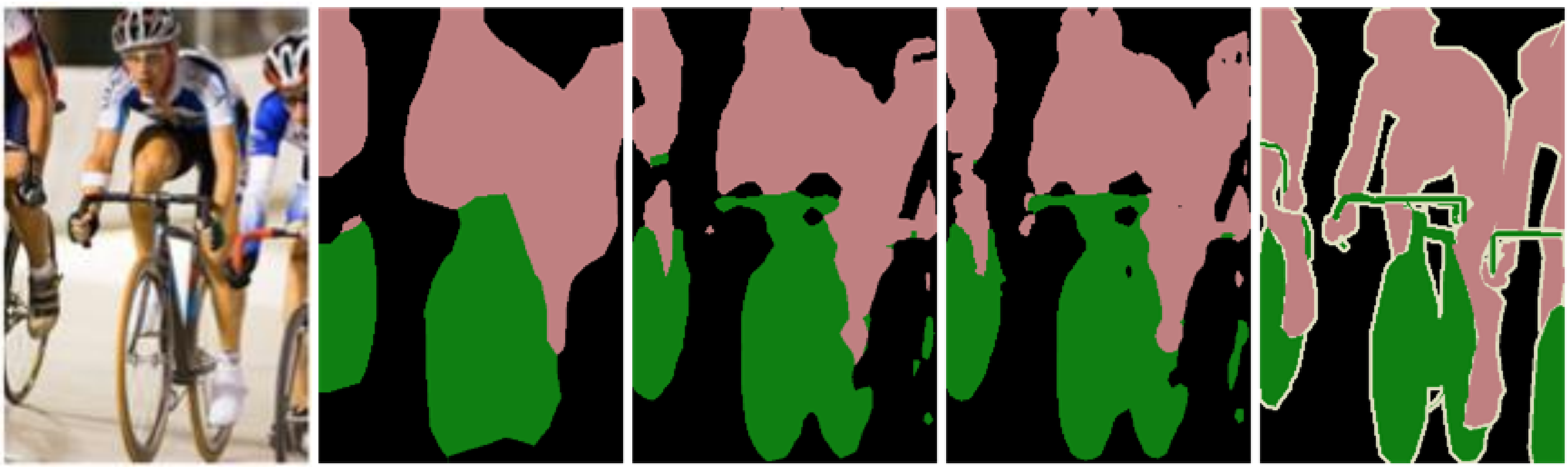


image
conv1
pool1
conv2
pool2
conv3
pool3
conv4
pool4
conv5
pool5
conv6-7
interp + sum
2x conv7
pool4
skip to fuse layers!
interp + sum
4x conv7
2x pool4
pool3
end-to-end, joint learning
of semantics and location
dense output

skip layer refinement

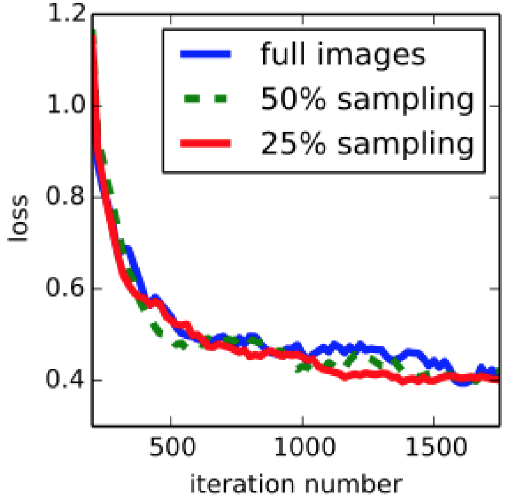
input image | stride 32 | stride 16 | stride 8 | ground truth

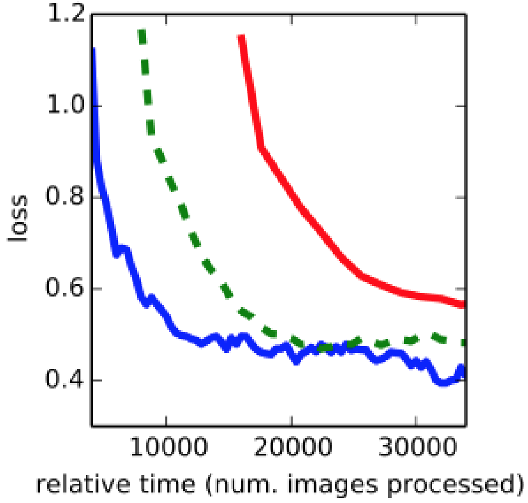


no skips | 1 skip | 2 skips

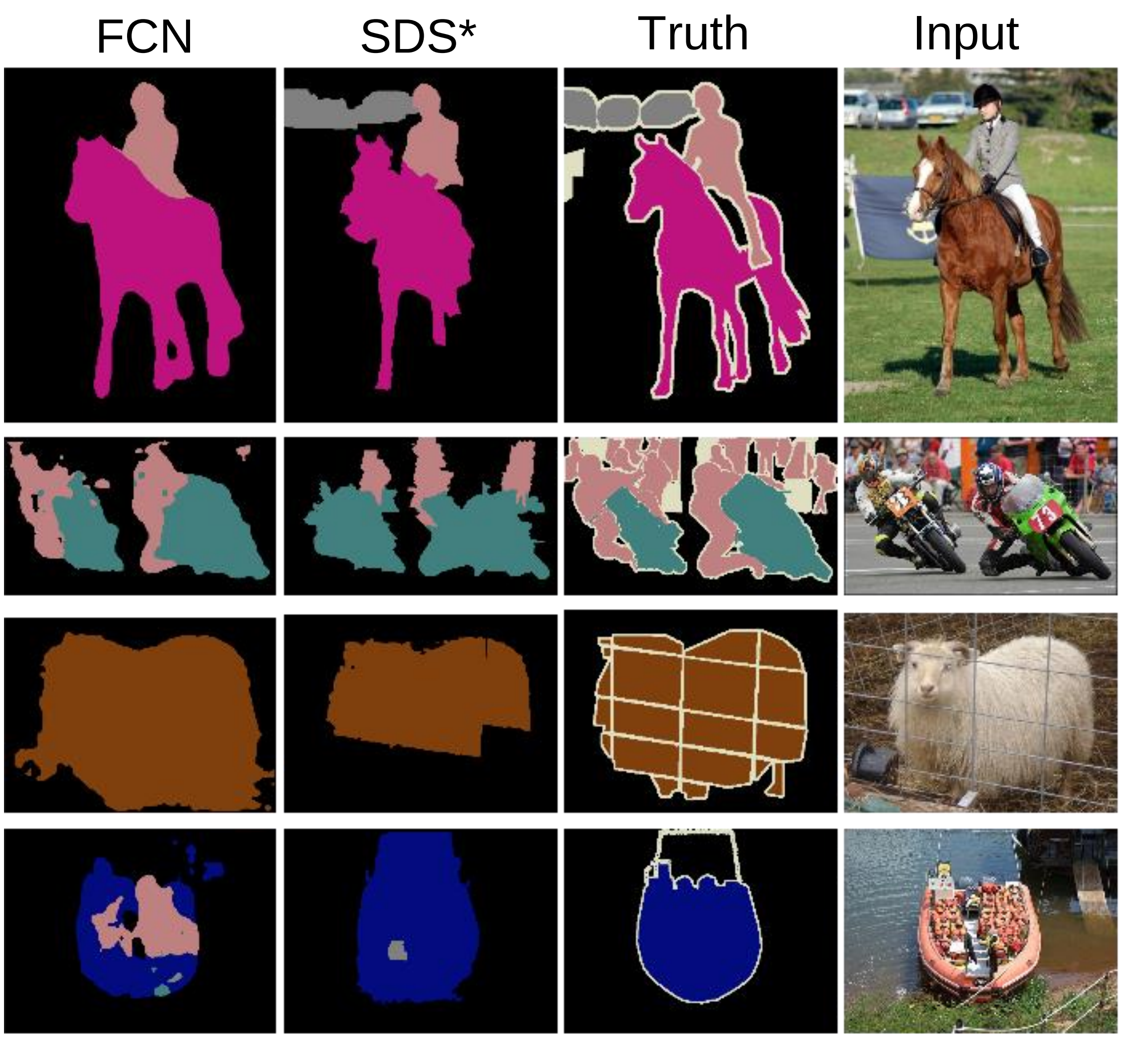
training + testing

- train full image at a time *without patch sampling*
- reshape network to take input of any size
- forward time is ~100ms for 500 x 500 x 21 output





results



Relative to prior state-of-the-art SDS:

- 30% relative improvement for mean IoU
- 286× faster

*Simultaneous Detection and Segmentation
Hariharan et al. ECCV14