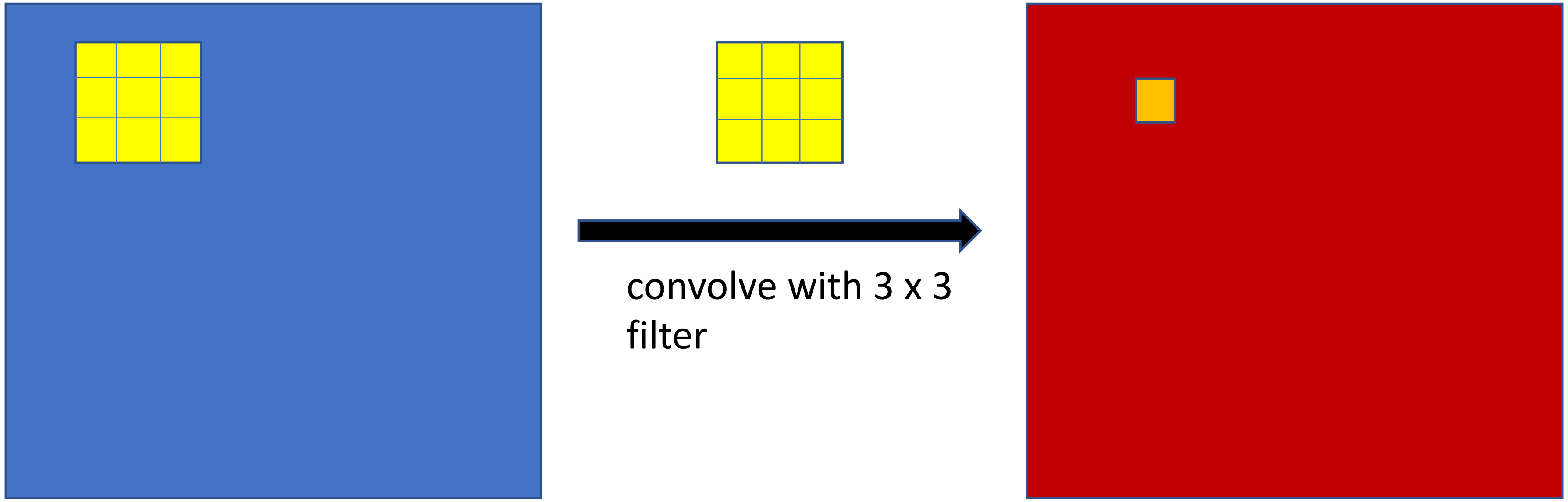
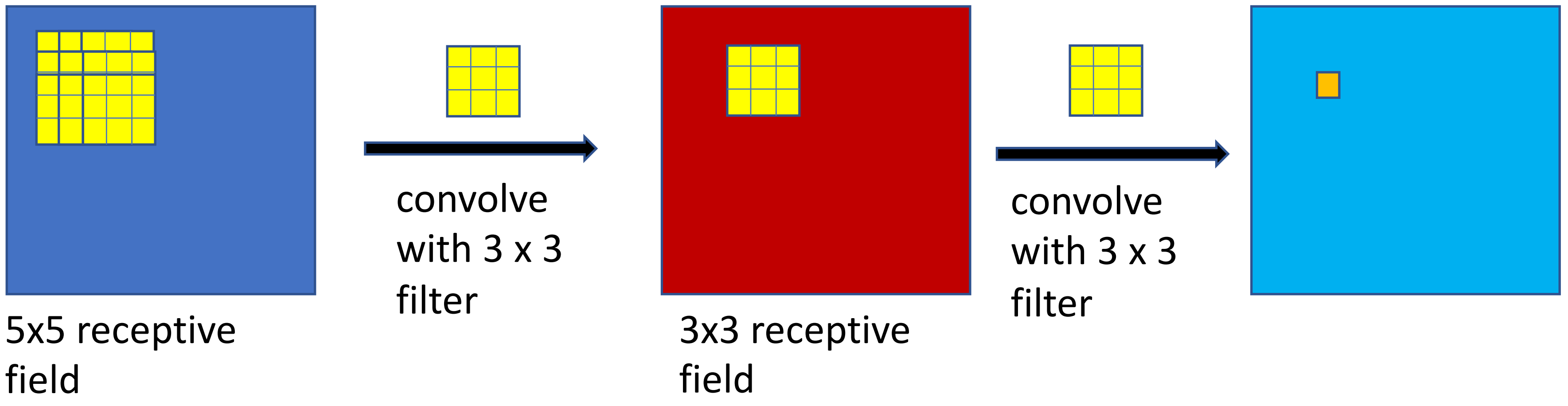
# Visualizing convolutional networks

# Receptive field

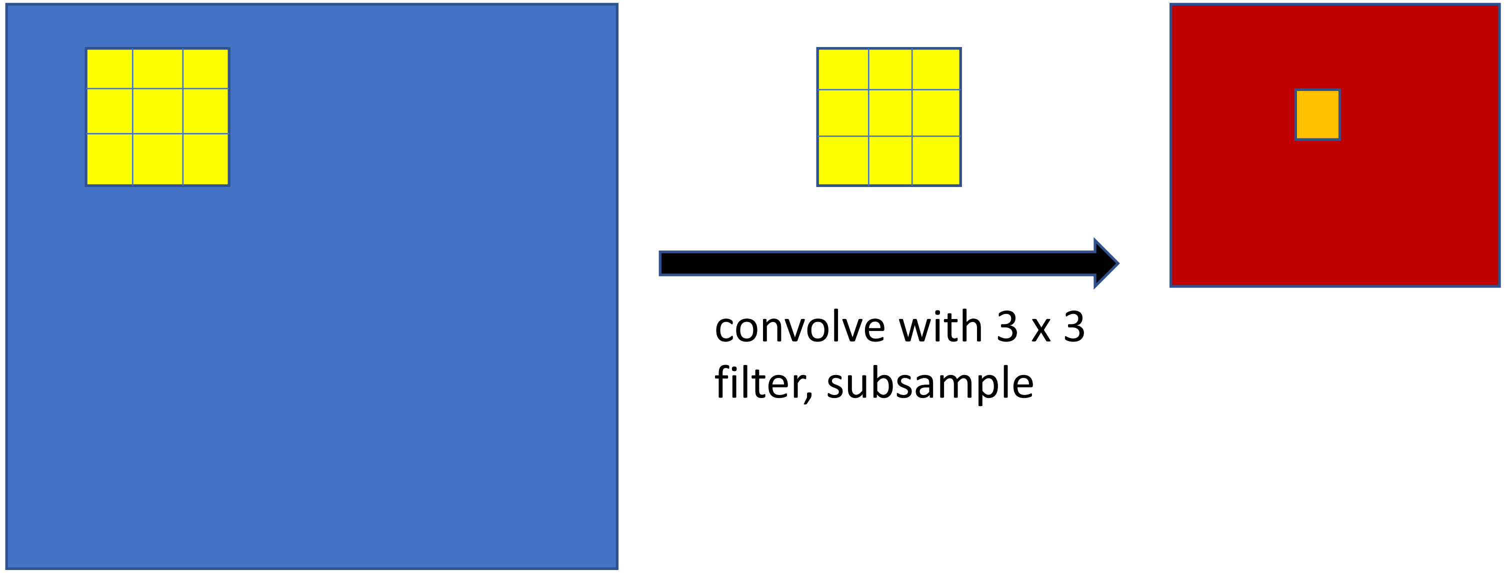
- Which input pixels does a particular unit in a feature map depends on



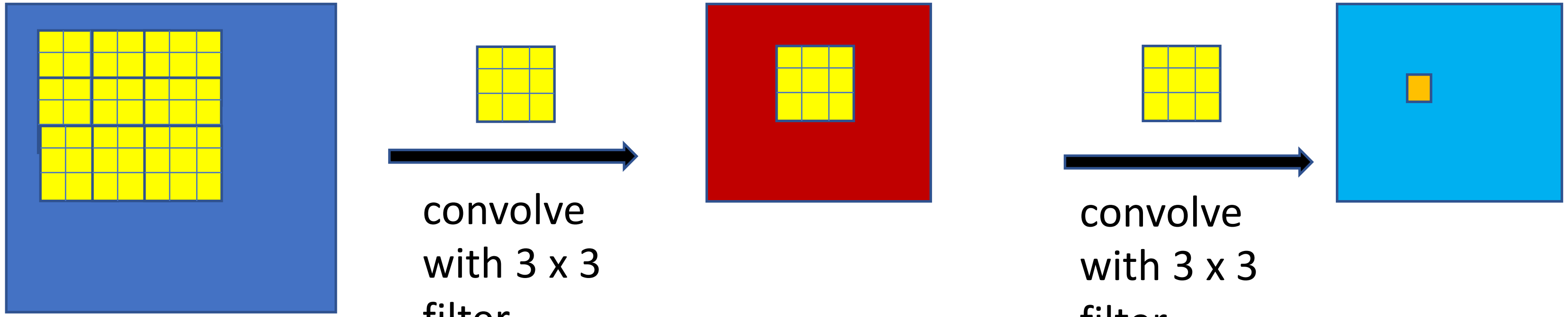
# Receptive field



# Receptive field



# Receptive field



7x7 receptive field: union of 9 3x3 fields with stride of 2

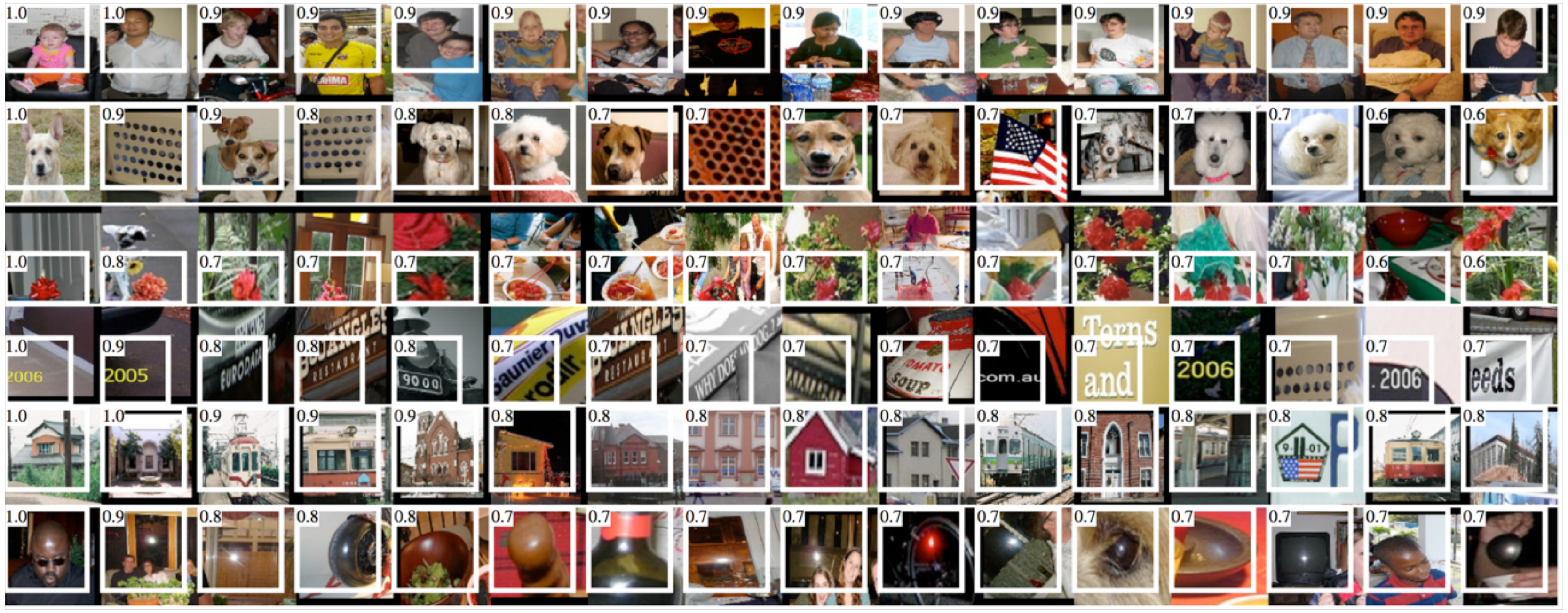
convolve with 3 x 3 filter, subsample by factor 2

3x3 receptive field

convolve with 3 x 3 filter

# Visualizing convolutional networks

- Take images for which a given unit in a feature map scores high
- Identify the receptive field for each.




Rich feature hierarchies for accurate object detection and semantic segmentation. R. Girshick, J. Donahue, T. Darrell, J. Malik. In *CVPR,* 2014.

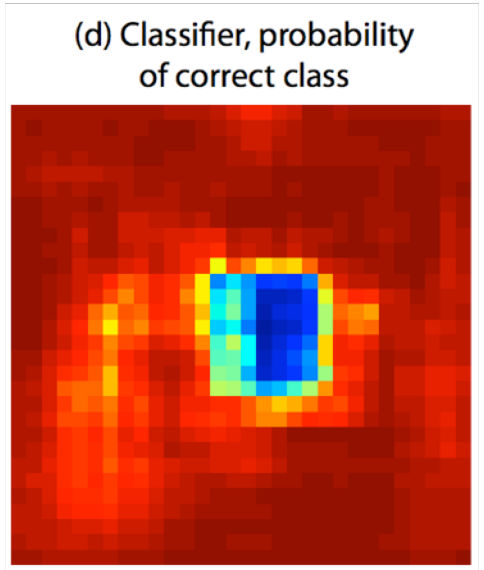
# Visualizing convolutional networks II

- Block regions of the image and classify

Visualizing and Understanding Convolutional Networks. M. Zeiler and R. Fergus. In *ECCV 2014.*

# Visualizing convolutional networks II

- Image pixels important for classification = pixels when blocked cause misclassification


Visualizing and Understanding Convolutional Networks. M. Zeiler and R. Fergus. In *ECCV 2014.*

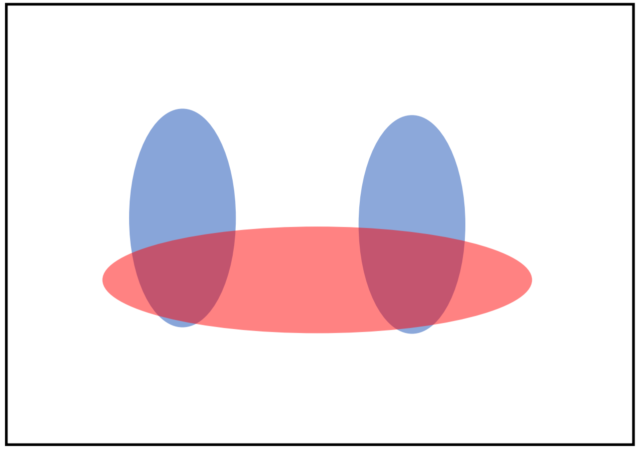
# Semantic Segmentation

# The Task

person
grass
trees
motorbike
road

# Evaluation metric

- Pixel classification!
- Accuracy?
  - Heavily unbalanced
  - Common classes are over-emphasized
- *Intersection over Union*
  - Average across classes and images
- Per-class accuracy
  - Compute accuracy for every class and then average



# Things vs Stuff

THINGS

- Person, cat, horse, etc
- Constrained shape
- Individual instances with separate identity
- May need to look at objects

STUFF

- Road, grass, sky etc
- Amorphous, no shape
- No notion of instances
- Can be done at pixel level
- “texture”

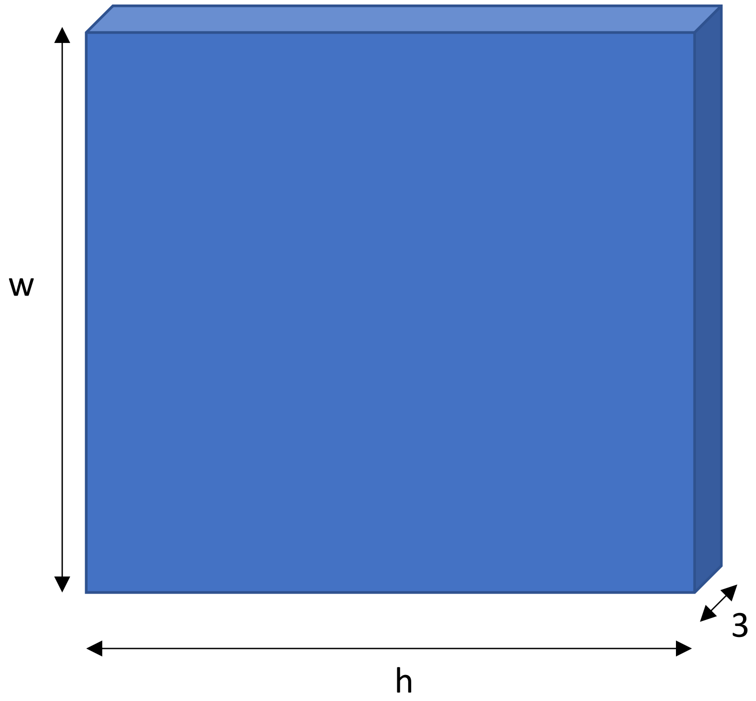
# Challenges in data collection

- Precise localization is hard to annotate
- Annotating every pixel leads to heavy tails
- Common solution: annotate few classes (often things), mark rest as “Other”
- Common datasets: PASCAL VOC 2012 (~1500 images, 20 categories), COCO (~100k images, 20 categories)

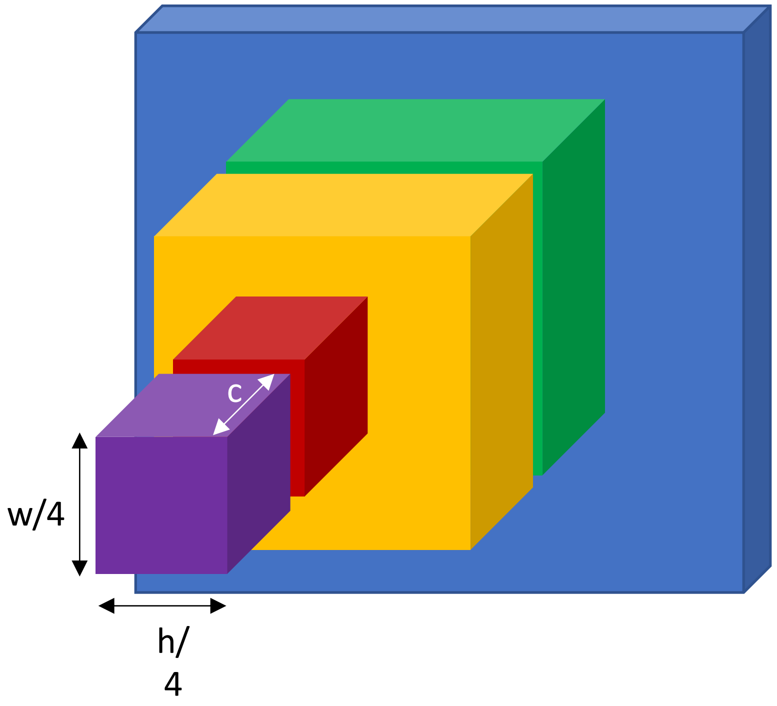
# Pre-convnet semantic segmentation

- Things
  - Do object detection, then segment out detected objects
- Stuff
  - ”Texture classification”
  - Compute histograms of filter responses
  - Classify local image patches

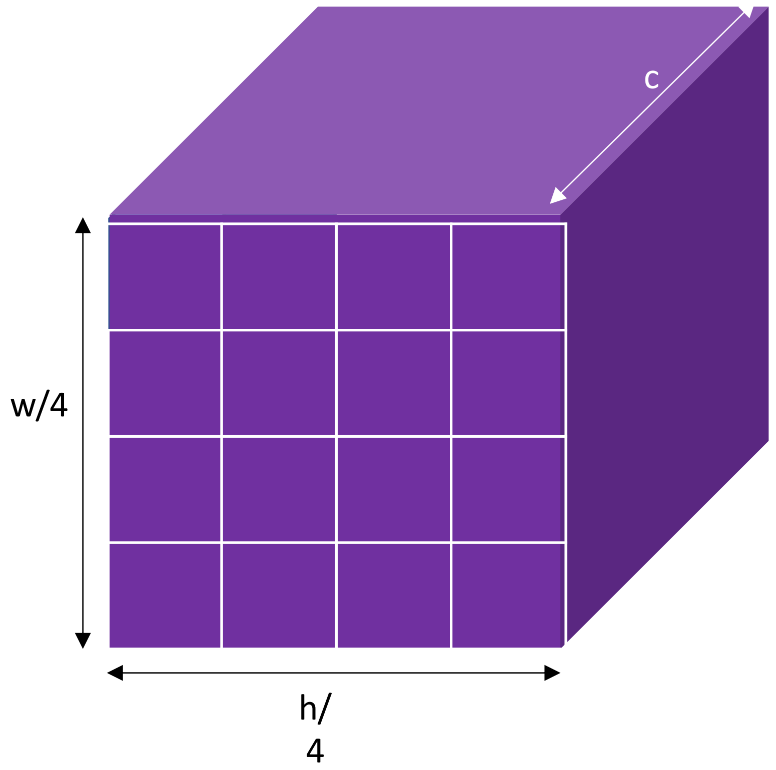
# Semantic segmentation using convolutional networks



# Semantic segmentation using convolutional networks

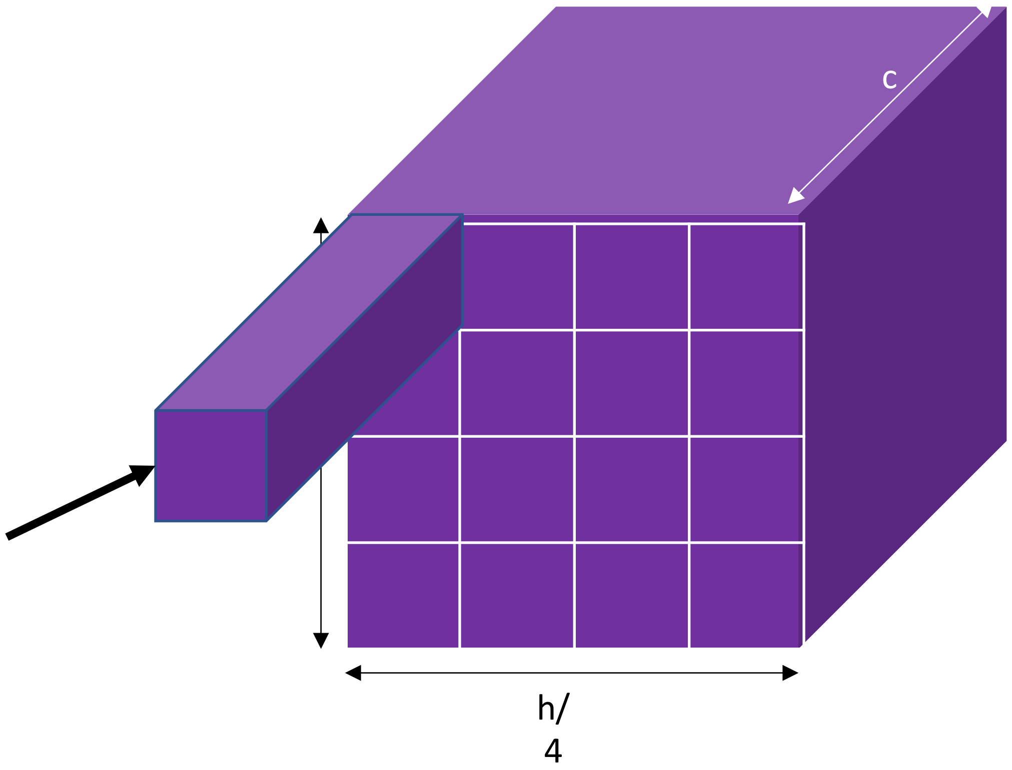


# Semantic segmentation using convolutional networks

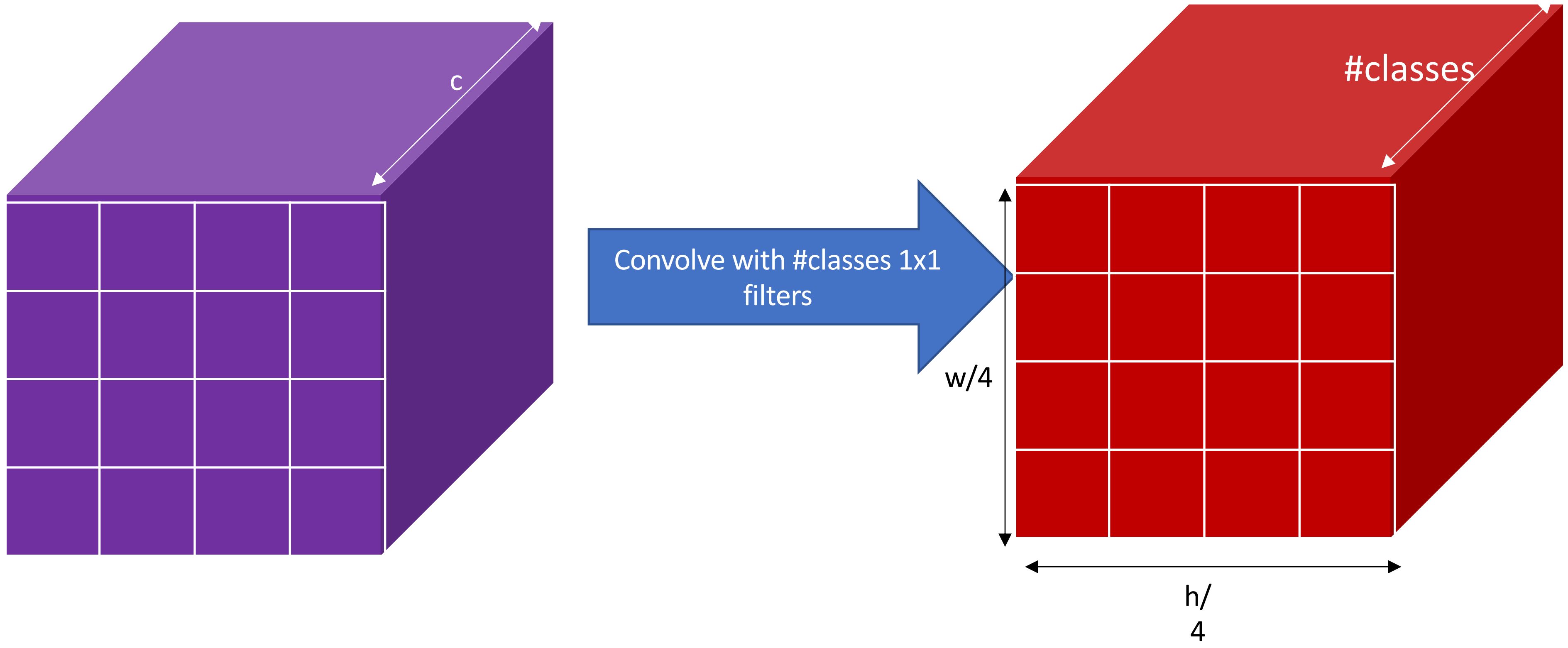


# Semantic segmentation using convolutional networks

Can be considered as a feature vector for a pixel



# Semantic segmentation using convolutional networks



# Semantic segmentation using convolutional networks

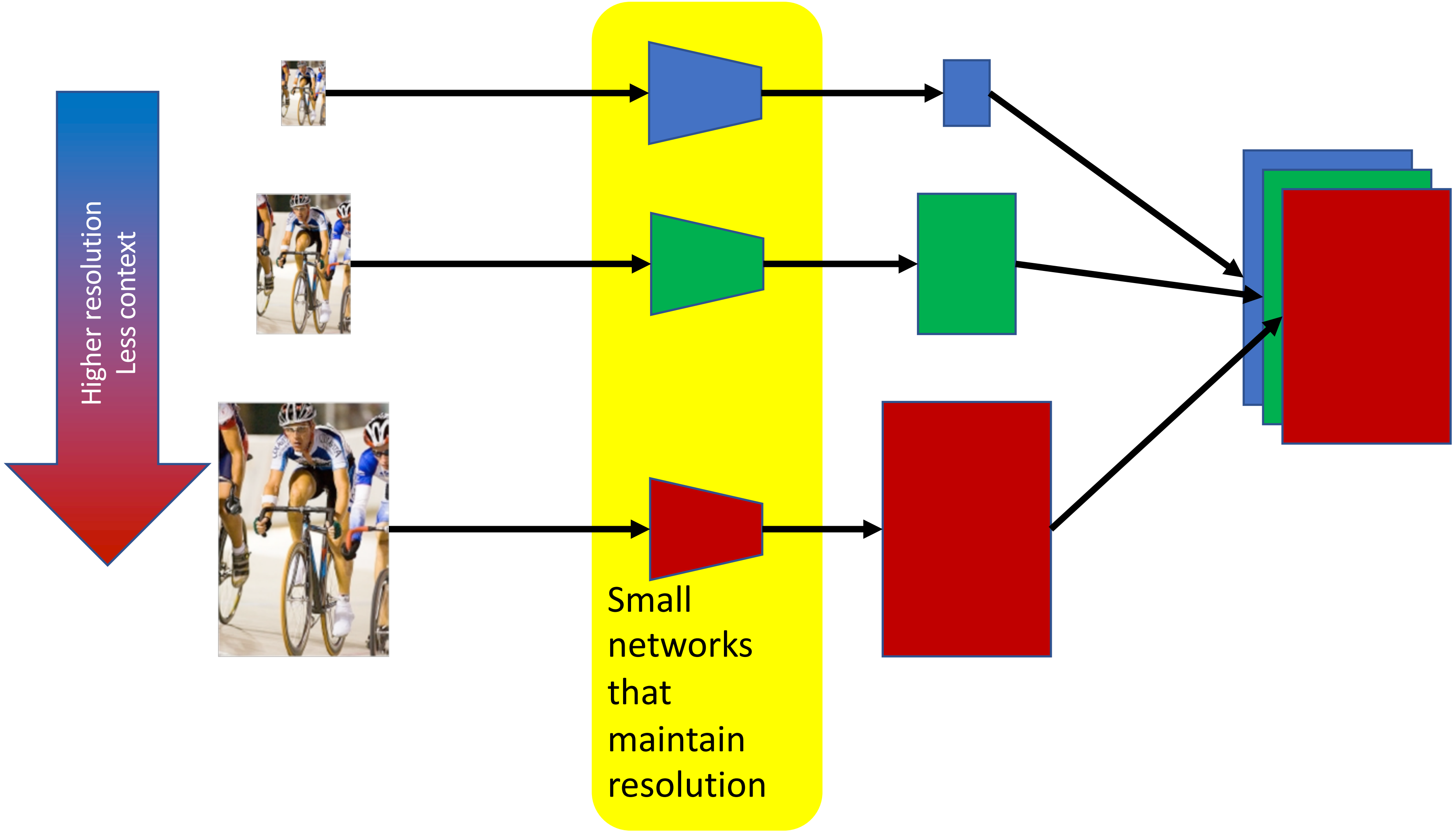
- Pass image through convolution and subsampling layers
- Final convolution with #classes outputs
- Get scores for *subsampled* image
- Upsample back to original size

# Semantic segmentation using convolutional networks

# The resolution issue

- Problem: Need fine details!
- Shallower network / earlier layers?
  - Deeper networks work better: more abstract concepts
  - Shallower network => Not very semantic!
- Remove subsampling?
  - Subsampling allows later layers to capture larger and larger patterns
  - Without subsampling => Looks at only a small window!

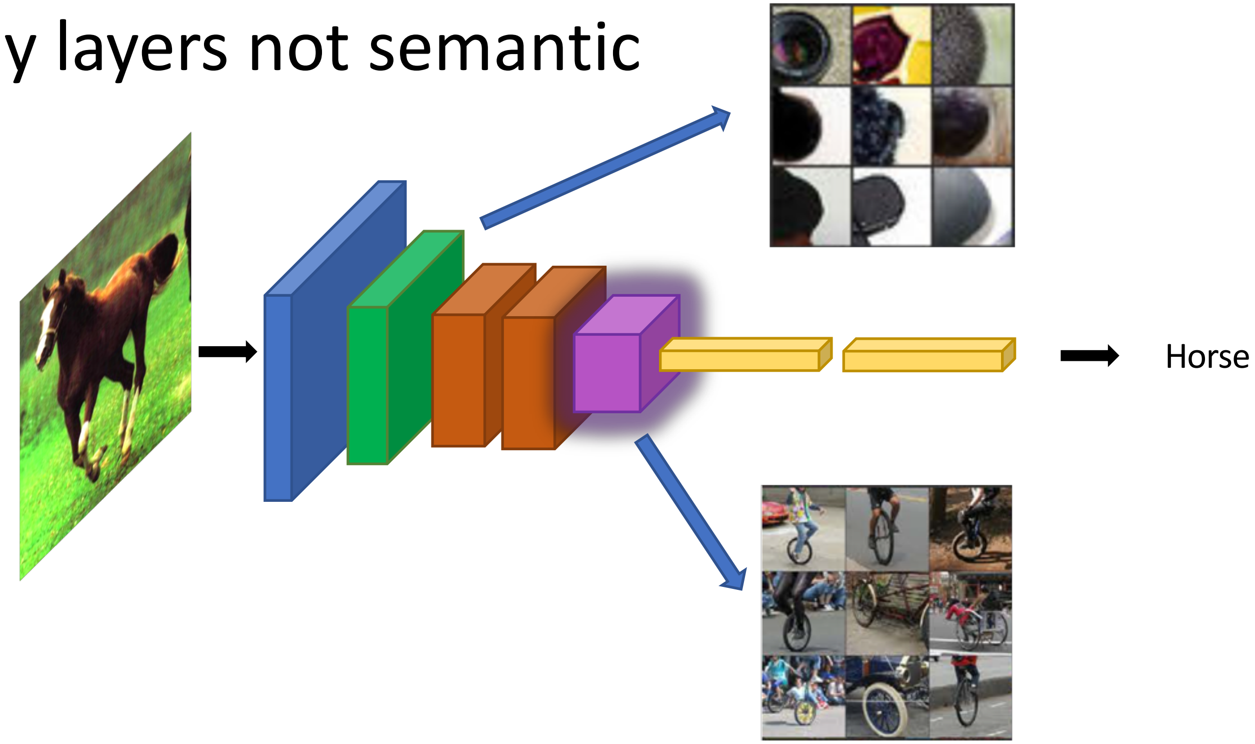
# Solution 1: Image pyramids




Learning Hierarchical Features for Scene Labeling. Clement Farabet, Camille Couprie, Laurent Najman, Yann LeCun. In *TPAMI,* 2013

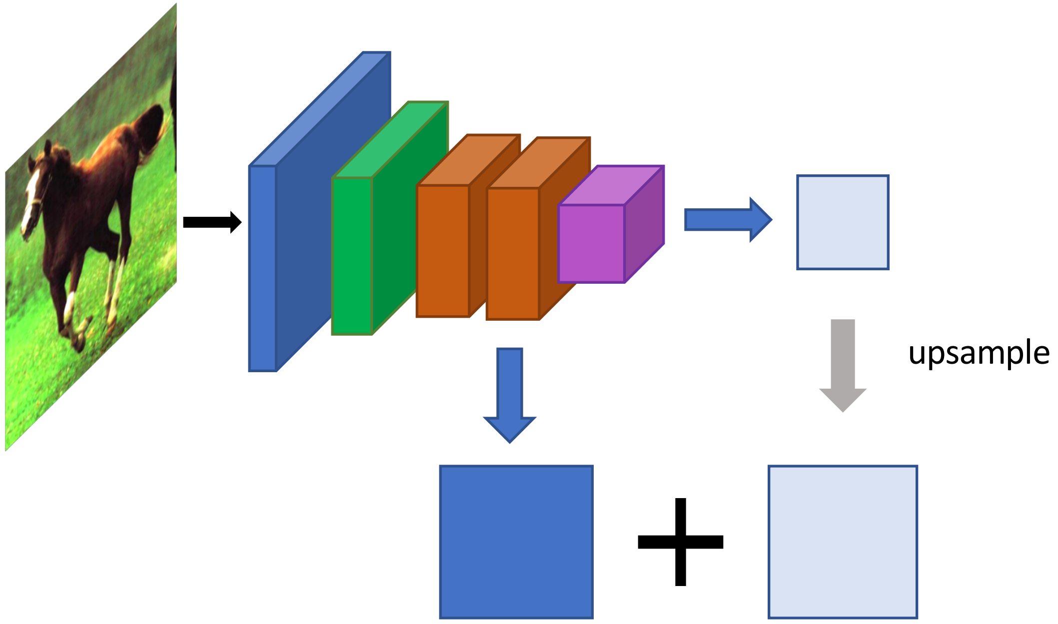
# Solution 2: Skip connections

- Problem: early layers not semantic



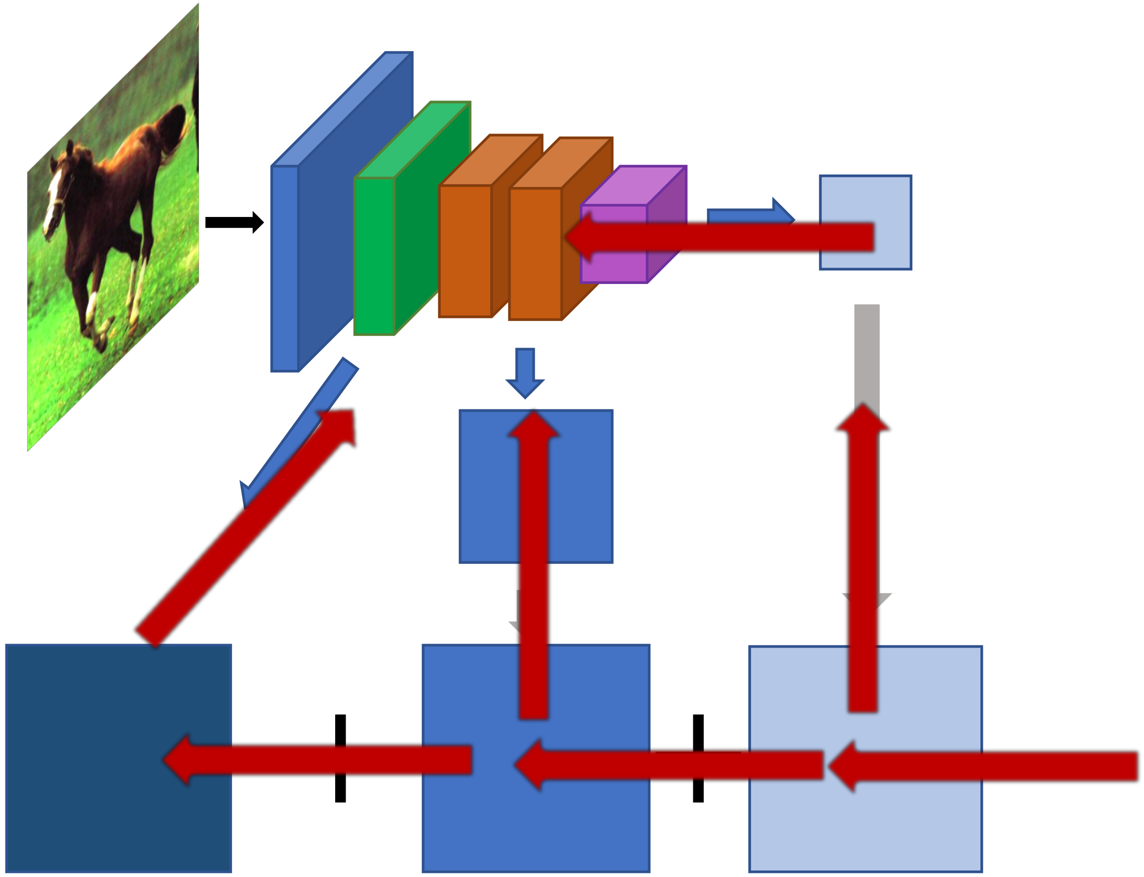

Visualizations from : M. Zeiler and R. Fergus. Visualizing and Understanding Convolutional Networks. In *ECCV* 2014.

# Solution 2: Skip connections




Compute class scores at multiple layers, then upsample and add

# Solution 2: Skip connections



Red arrows indicate backpropagation

# Skip connections

without skip with skip

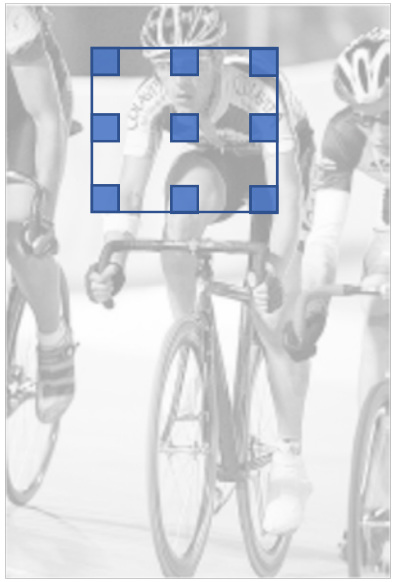
Fully convolutional networks for semantic segmentation. Evan Shelhamer, Jon Long, Trevor Darrell. In *CVPR* 2015

# Solution 3: Dilation

- Need subsampling to allow convolutional layers to capture large regions with small filters
  - Can we do this without subsampling?

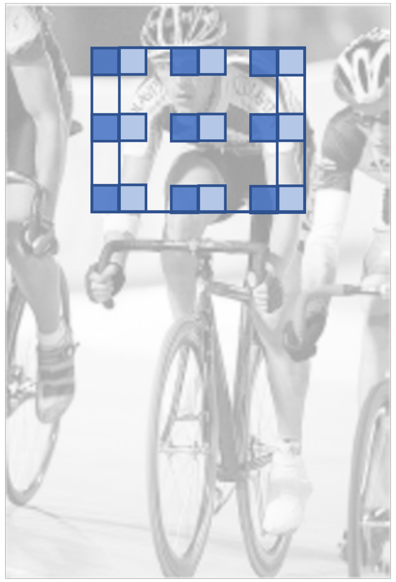
# Solution 3: Dilation

- Need subsampling to allow convolutional layers to capture large regions with small filters
  - Can we do this without subsampling?



# Solution 3: Dilation

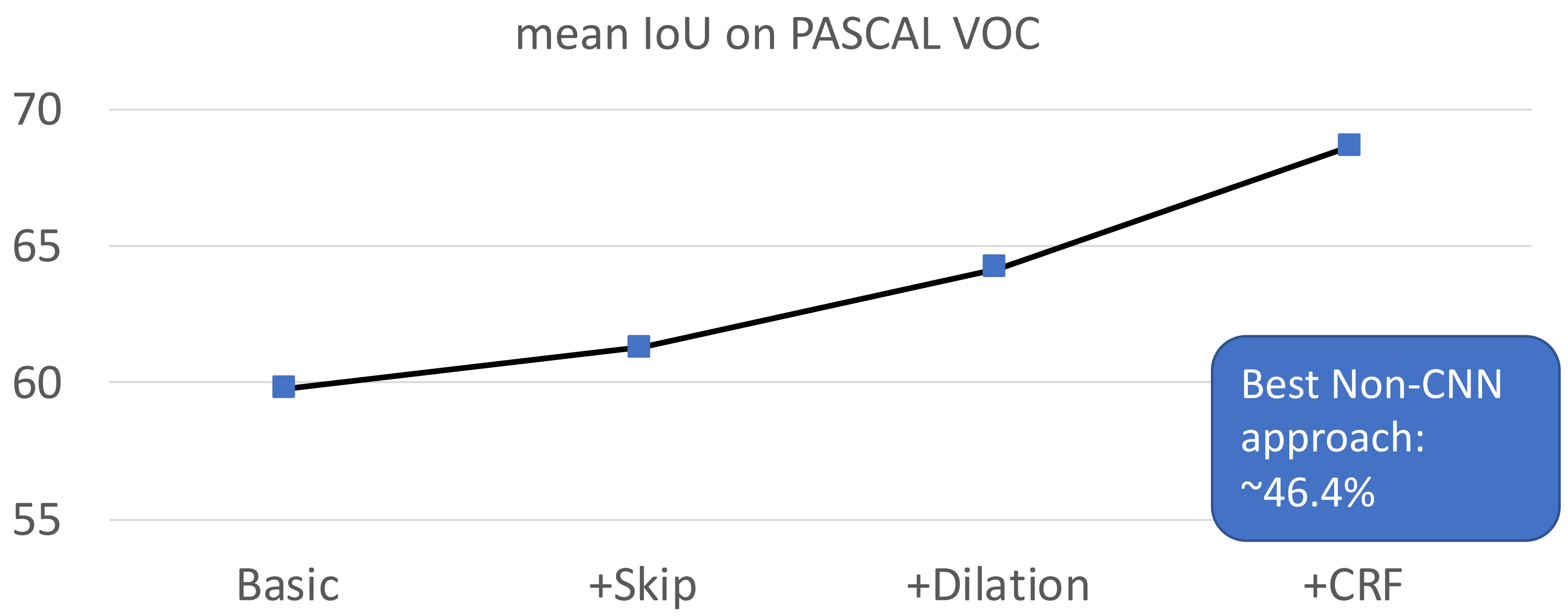
- Need subsampling to allow convolutional layers to capture large regions with small filters
  - Can we do this without subsampling?



# Solution 3: Dilation

- Instead of subsampling by factor of 2: dilate by factor of 2
- Dilation can be seen as:
  - Using a much larger filter, but with most entries set to 0
  - Taking a small filter and “exploding”/ “dilating” it
- Not panacea: without subsampling, feature maps are much larger: memory issues

# Putting it all together




Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, Alan Yuille. In *ICLR,* 2015.