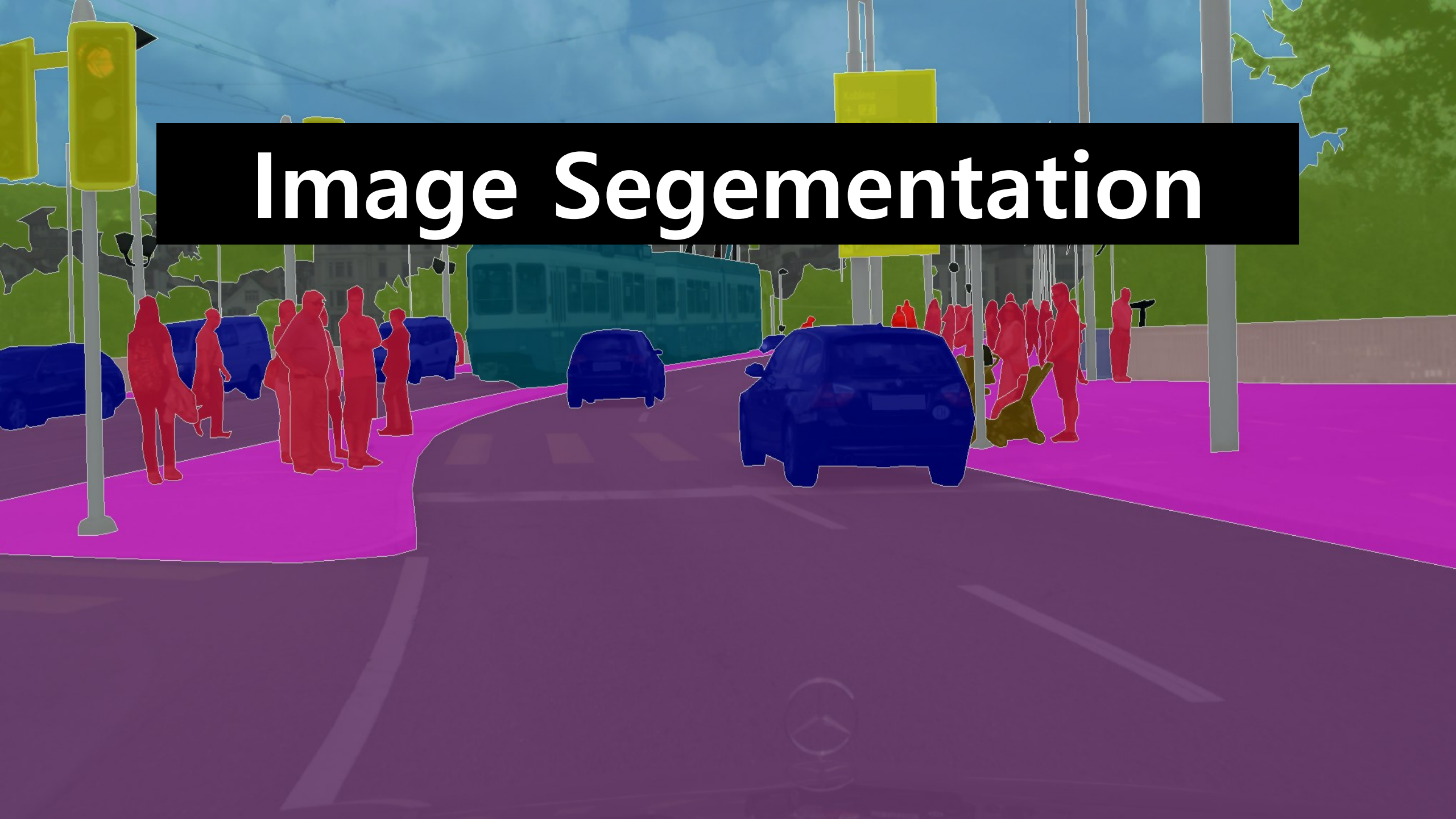
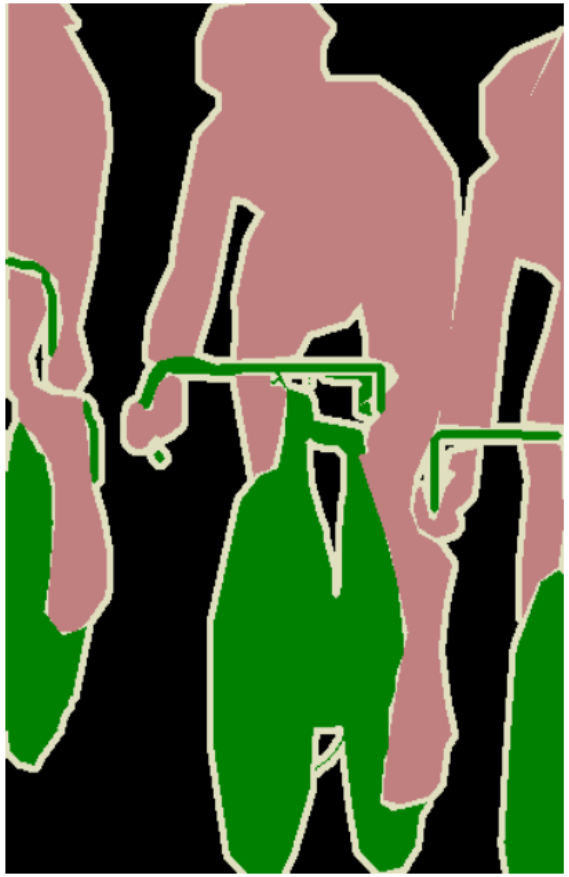


Image Segementation

Semantic Segmentation



- PASCAL VOC segmentation
  - 10K images, 20 classes + bgnd

PASCAL2
Pattern Analysis, Statistical Modelling and Computational Learning

- MS COCO
  - 100K images, 80 classes + bgnd

Microsoft COCO
Common Objects in Context

Fully Convolutional Networks for Semantic Segmentation

Fully Convolutional Networks for Semantic Segmentation

Jonathan Long* Evan Shelhamer* Trevor Darrell
UC Berkeley
{jonlong,shelhamer,trevor}@cs.berkeley.edu

Abstract

Convolutional networks are powerful visual models that yield hierarchies of features. We show that convolutional networks by themselves, trained end-to-end, pixels-to-pixels, exceed the state-of-the-art in semantic segmentation. Our key insight is to build "fully convolutional" networks that take input of arbitrary size and produce correspondingly-sized output with efficient inference and learning. We define and detail the space of fully convolutional networks, explain their application to spatially dense prediction tasks, and draw connections to prior models. We adapt contemporary classification networks (AlexNet [22],

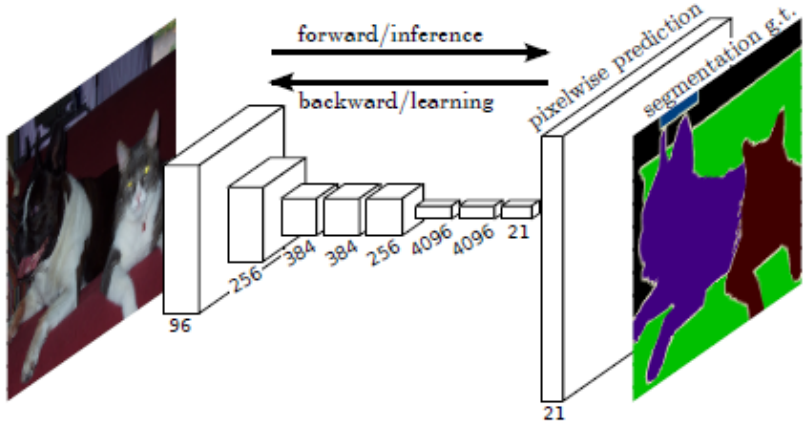
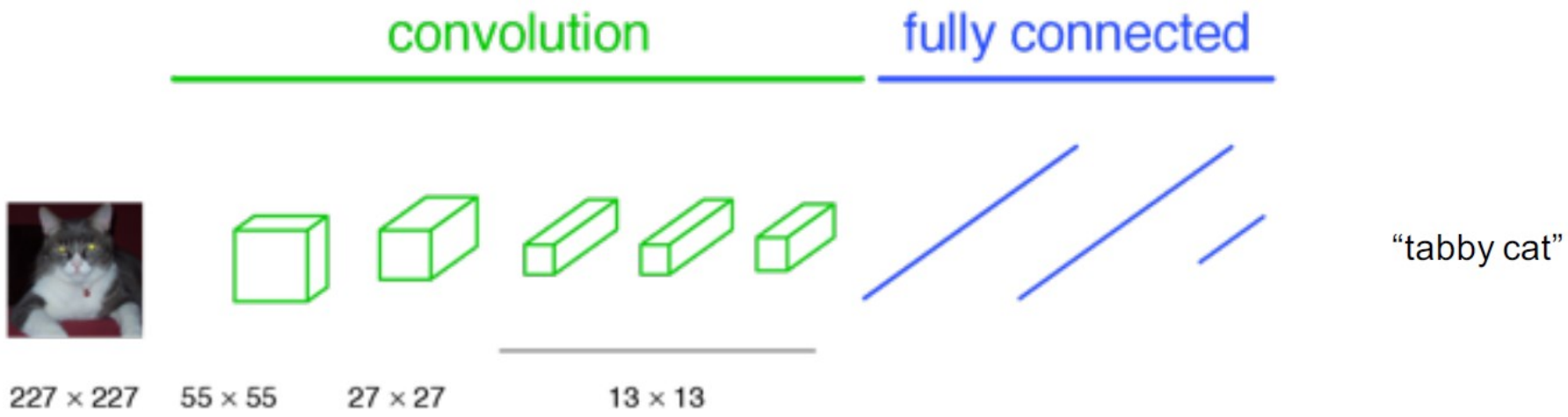


Figure 1. Fully convolutional networks can efficiently learn to make dense predictions for per-pixel tasks like semantic segmentation.

Fully Convolutional Networks for Semantic Segmentation

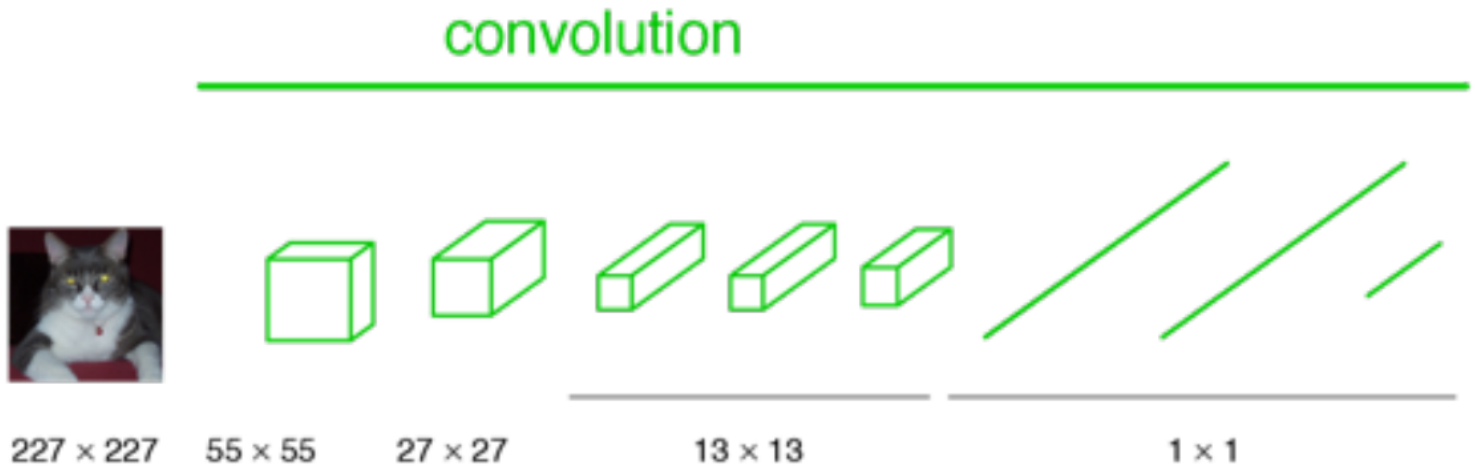
- Fully convolutional network(FCN)
  - Pixel-wise prediction with end to end learning
- Fully Convolutional versions of existing networks predict dense outputs from arbitrary-sized inputs

Classification Network



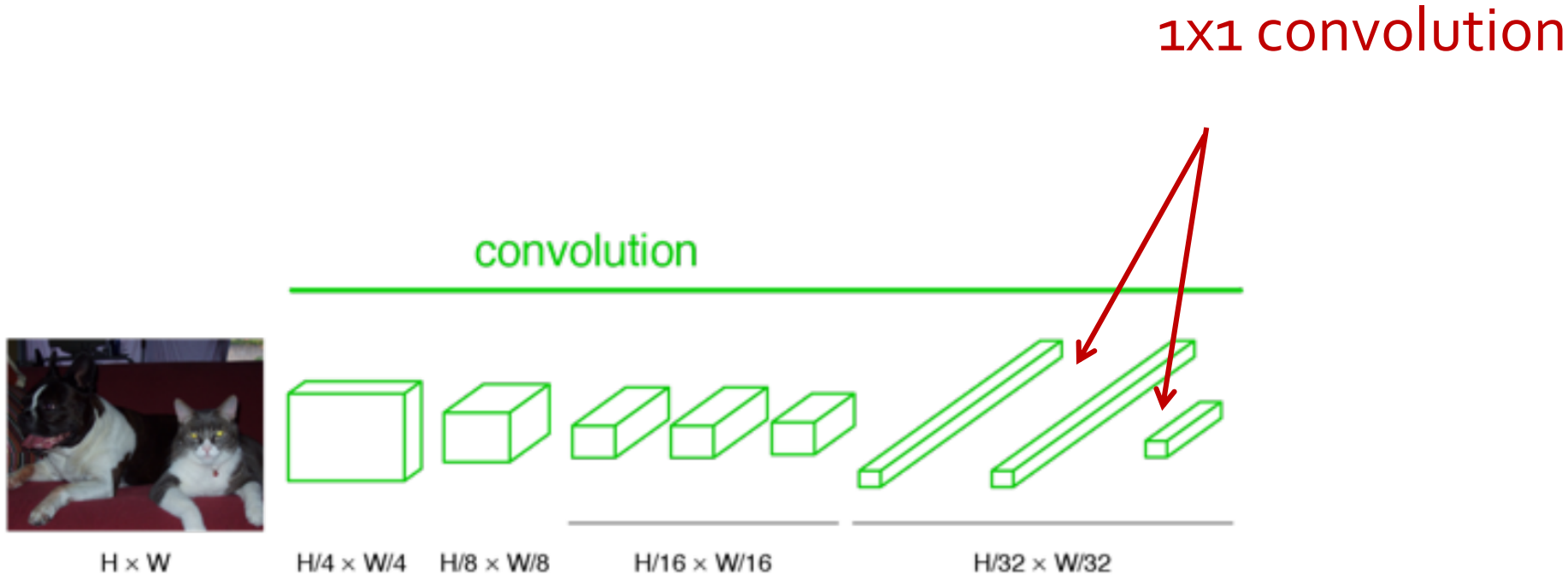
Slide from Jonathan Long

Becoming Fully Convolutional Network



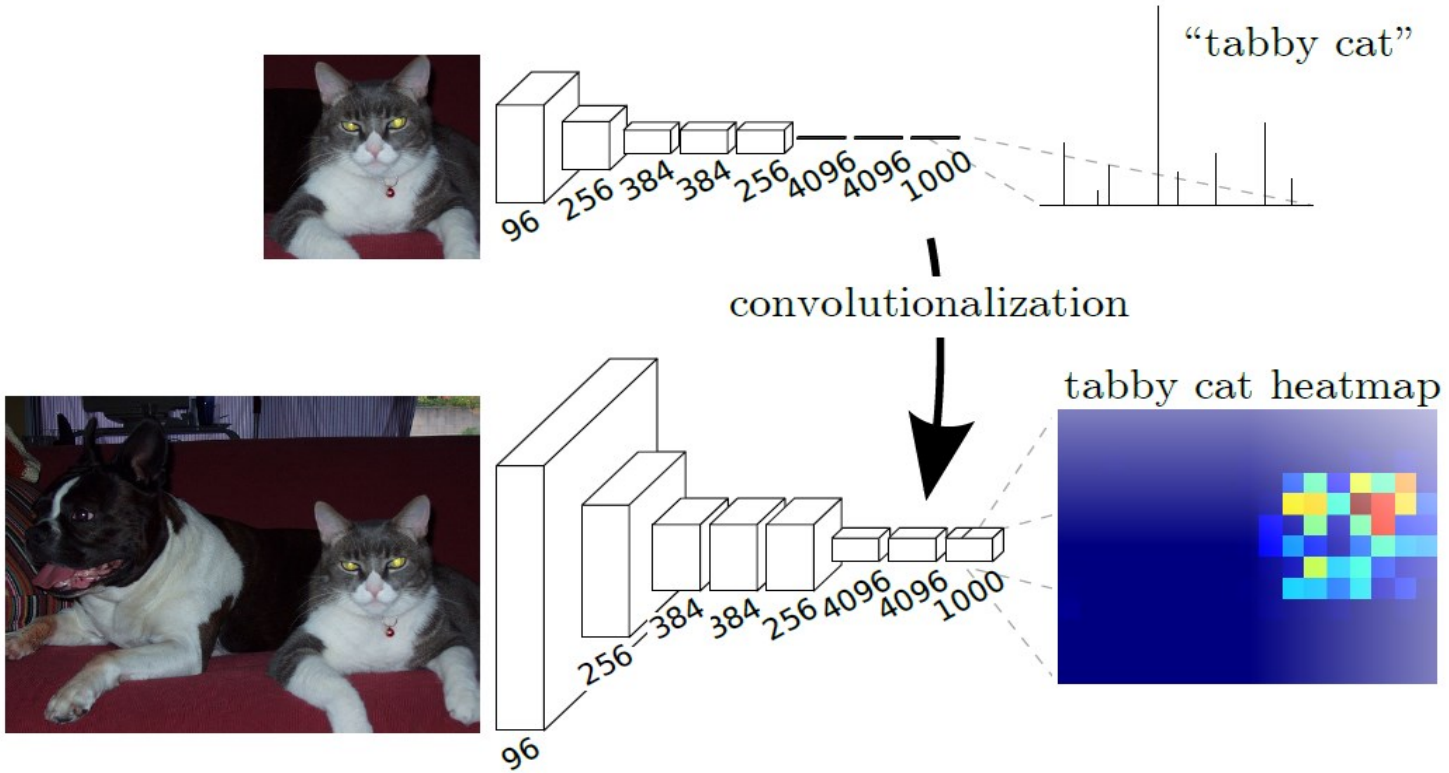
Slide from Jonathan Long

Becoming Fully Convolutional Network



Slide from Jonathan Long

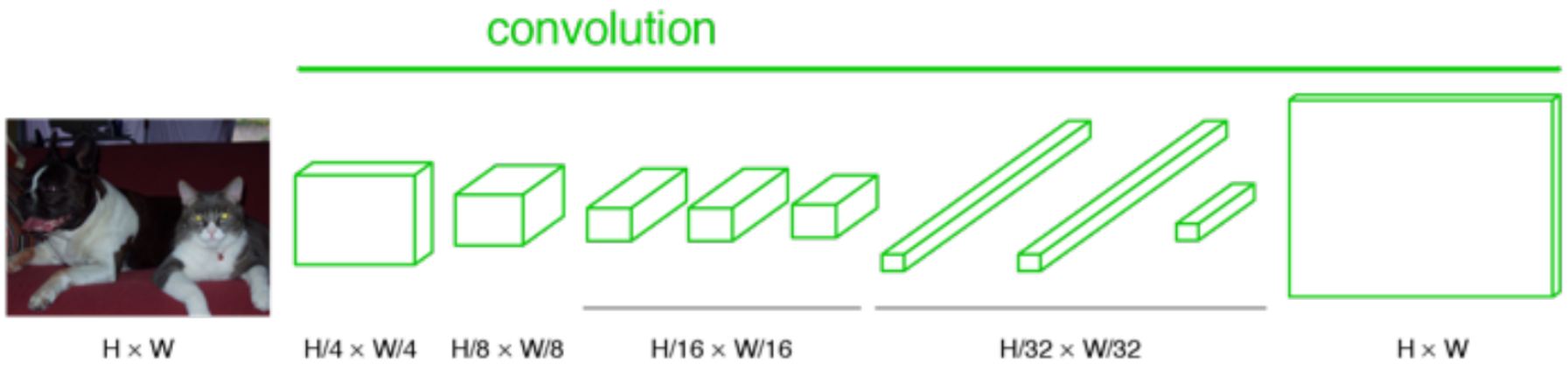
Heatmap from FCN



Slide from Jonathan Long

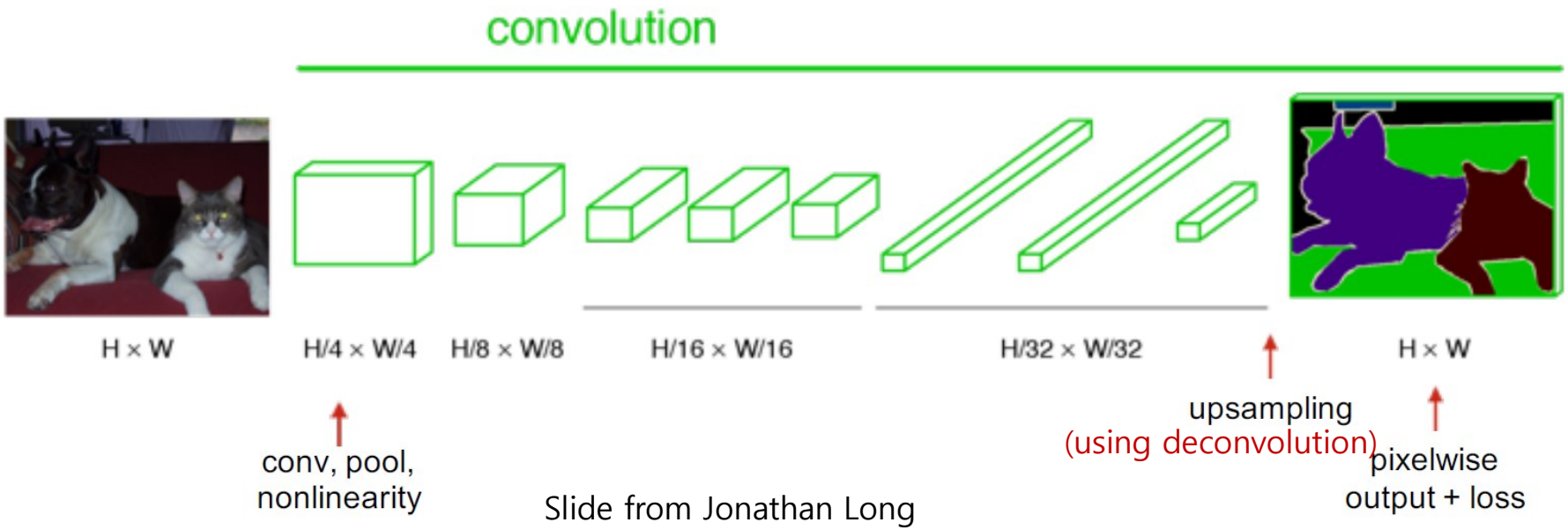
Upsampling Output

- They use deconvolution for upsampling



Slide from Jonathan Long

End to End, Pixels to Pixels Network



Slide from Jonathan Long

Slide from Jonathan Long

Classifier to Dense FCN

- Convolutionalize proven classification architectures:
  - AlexNet, VGG, and GoogLeNet (reimplementation)
- Remove classification layer and convert all fully connected layers to convolutions
- Append 1x1 convolution with channel dimensions and predict scores at each of the coarse output locations (21 categories for PASCAL)

Skip Layers

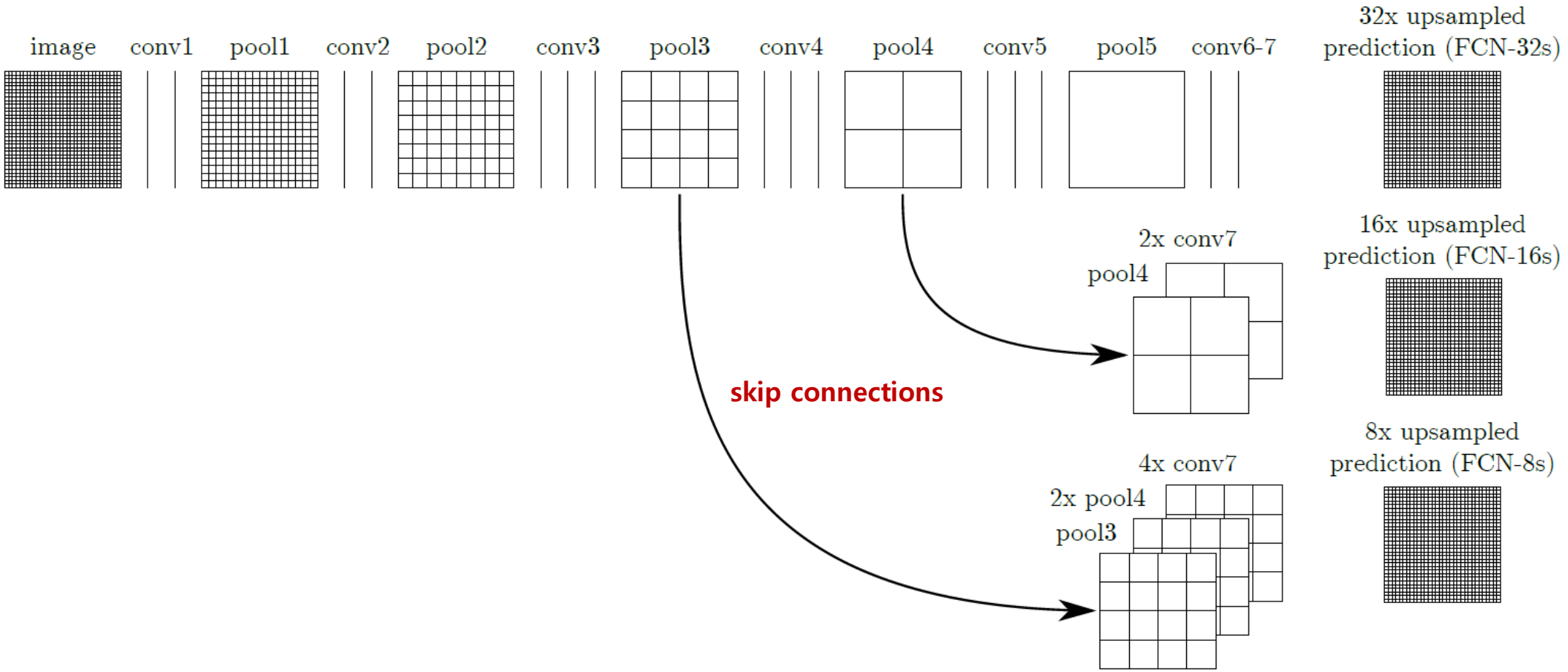
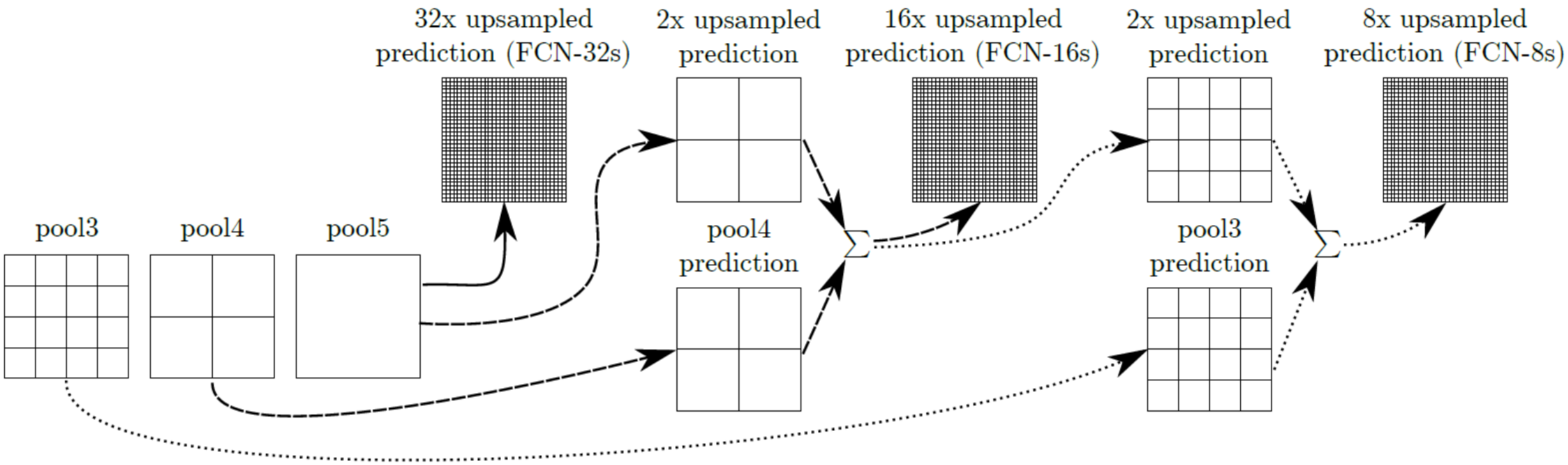


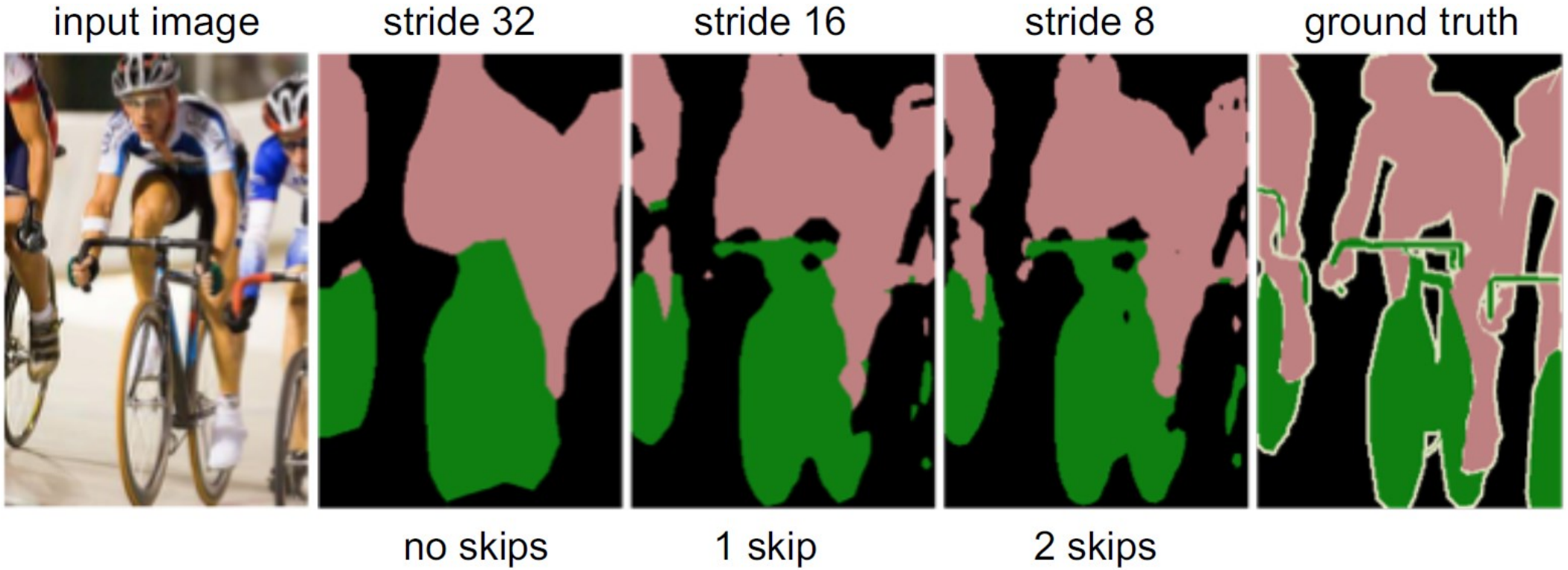
image
conv1
pool1
conv2
pool2
conv3
pool3
conv4
pool4
conv5
pool5
conv6-7
32x upsampled prediction (FCN-32s)
2x conv7
pool4
16x upsampled prediction (FCN-16s)
skip connections
4x conv7
2x pool4
pool3
8x upsampled prediction (FCN-8s)

Skip Layers

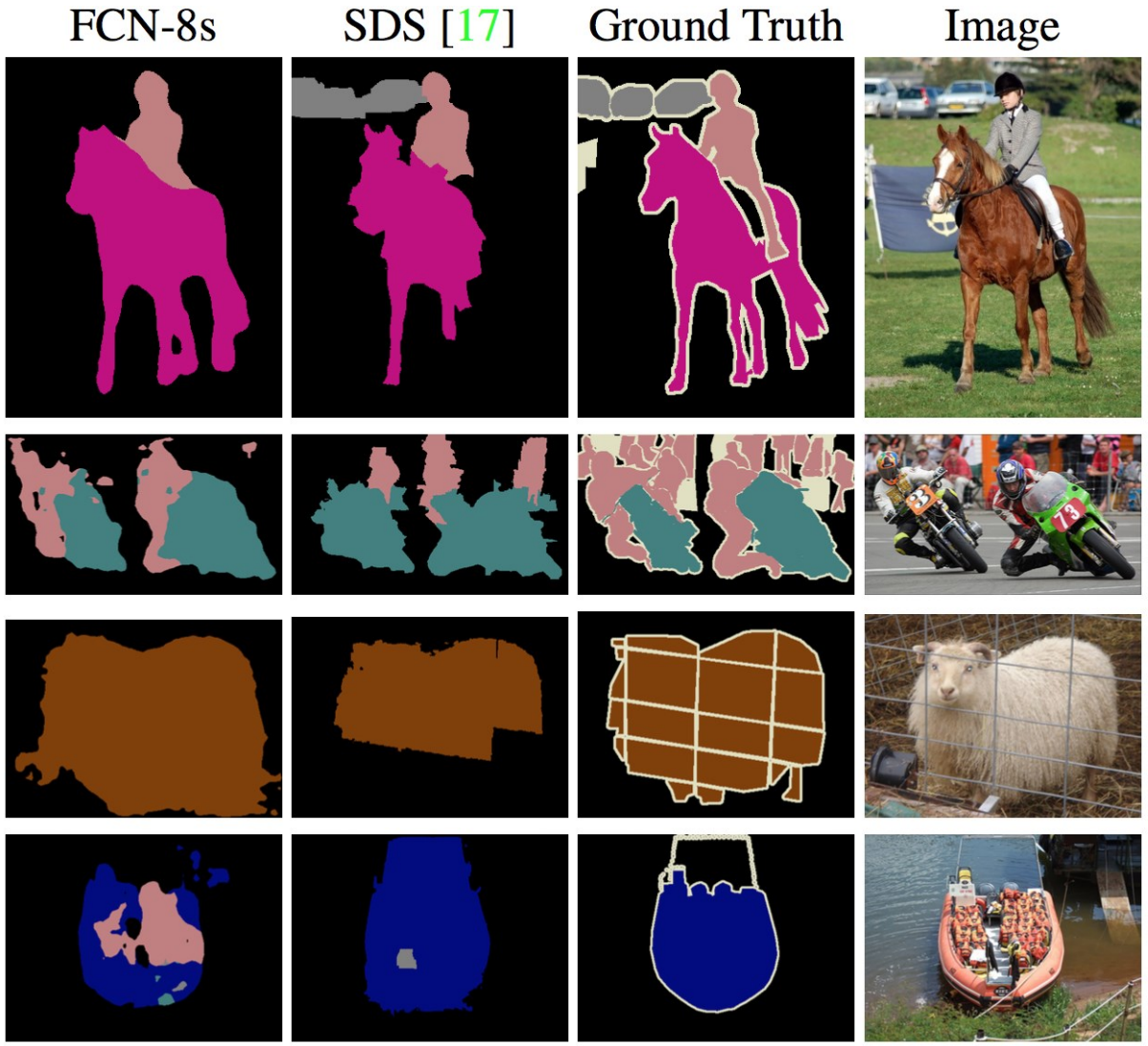


Slide from Jonathan Long

Results


input image
stride 32
stride 16
stride 8
ground truth
no skips
1 skip
2 skips

Results



Learning Deconvolution Network for Semantic Segmentation

Learning Deconvolution Network for Semantic Segmentation

Hyeonwoo Noh Seunghoon Hong Bohyung Han
Department of Computer Science and Engineering, POSTECH, Korea
{hyeonwoonoh_,maga33,bhhan}@postech.ac.kr

Abstract

We propose a novel semantic segmentation algorithm by learning a deep deconvolution network. We learn the network on top of the convolutional layers adopted from VGG 16-layer net. The deconvolution network is composed of deconvolution and unpooling layers, which identify pixel-wise class labels and predict segmentation masks. We apply the trained network to each proposal in an input image, and construct the final semantic segmentation map by combining the results from all proposals in a simple manner. The proposed algorithm mitigates the limitations of the existing methods based on fully convolutional networks by integrating deep deconvolution network and proposal-wise prediction; our segmentation method typically identifies detailed structures and handles objects in multiple scales naturally. Our network demonstrates outstanding performance

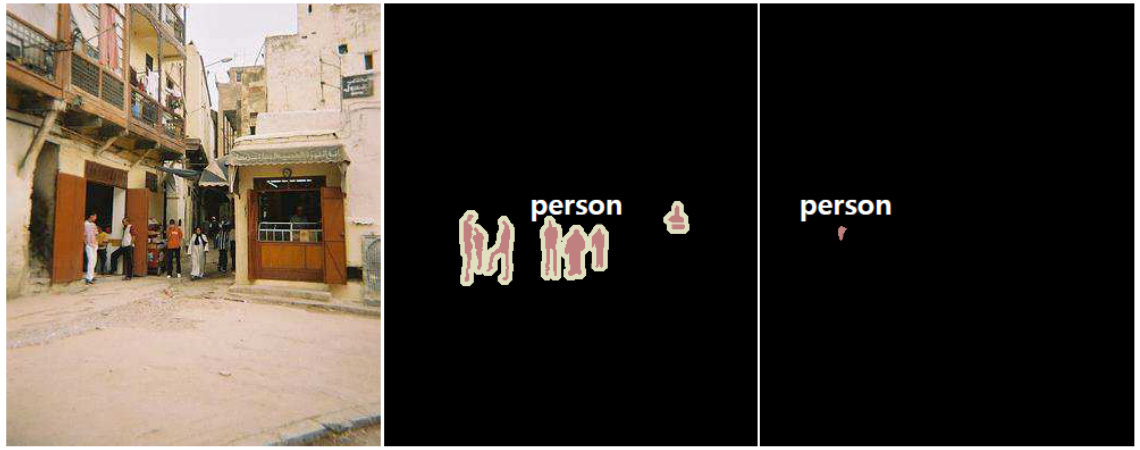
(a) Inconsistent labels due to large object size

(b) Missing labels due to small object size

FCN's Limitation

- FCN has a predefined fixed-size receptive field. Therefore, the object that is substantially larger or smaller than the receptive field may be fragmented or mislabeled
- The detailed structures of an object are often lost or smoothed because the label map is too coarse and deconvolution procedure is overly simple

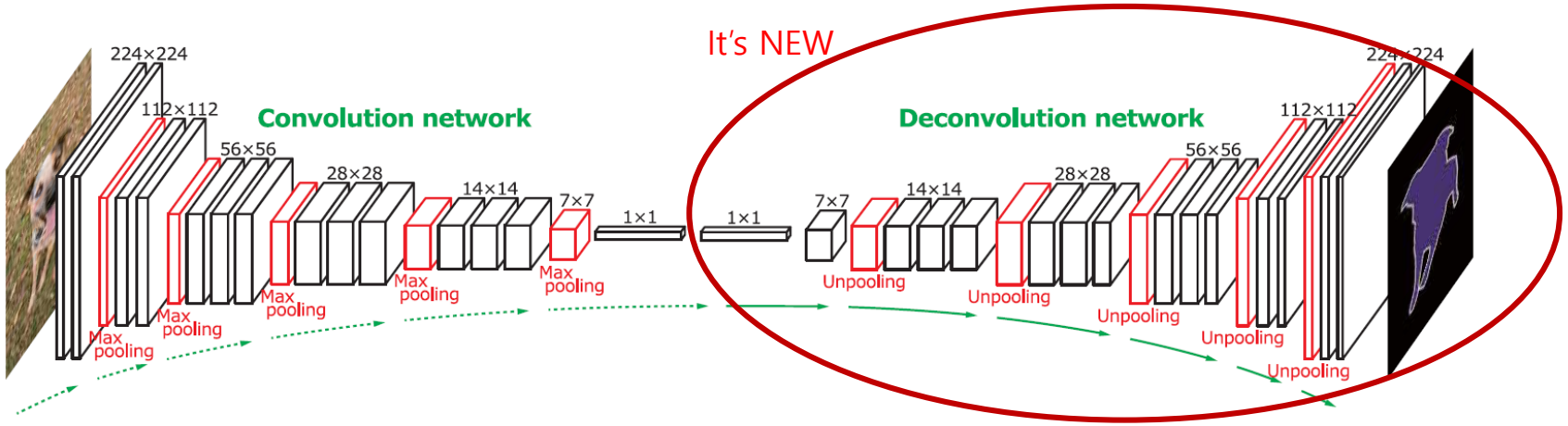
(a) Inconsistent labels due to large object size



(b) Missing labels due to small object size

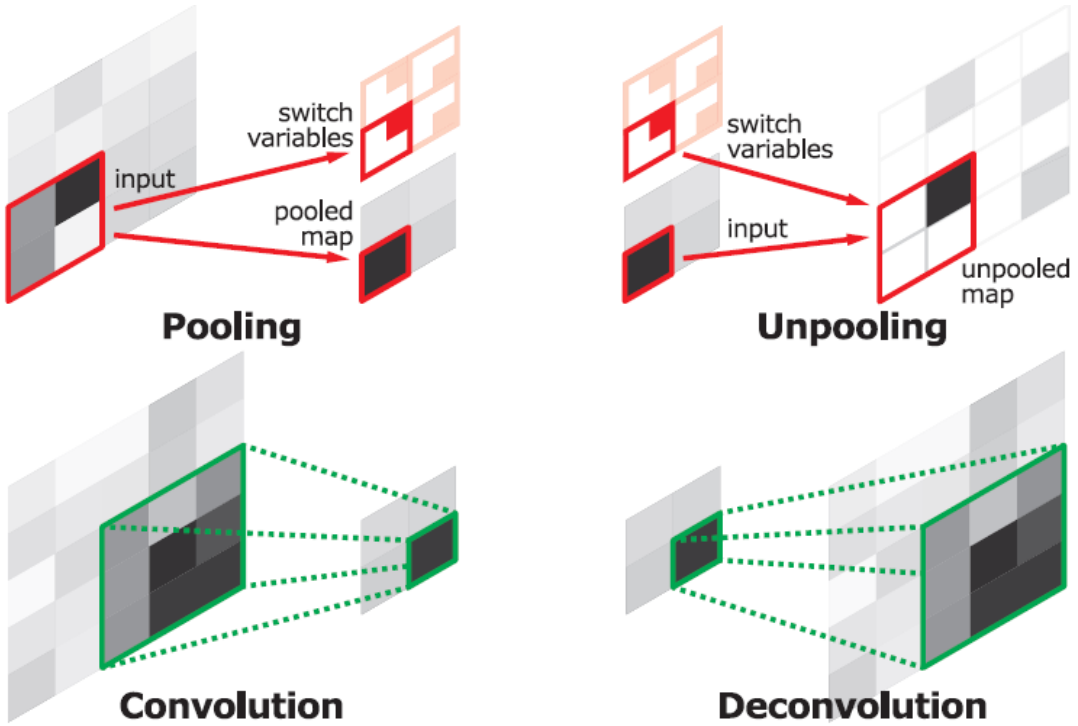
Deconvolution Network

- To address such issues, Use "Deconvolution"!
- Convolution network extract features(VGG-16)
- Deconvolution network generate probability map(same size to input image)
- Probability map indicates probability of each pixel belongs to one of class



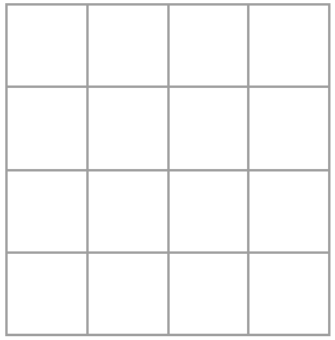
Unpooling and Deconvolution

- Unpooling
  - Reconstruct structure of original activation map
  - Activation size is preserved but still sparse
- Deconvolution
  - Densify sparse activation map

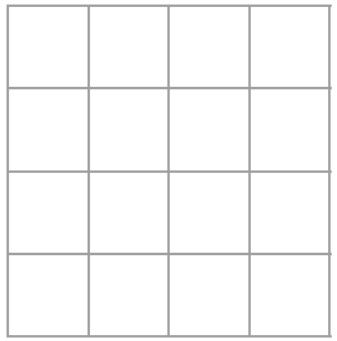


Learnable Upsampling: “Deconvolution”

Typical 3 x 3 convolution, stride 1 pad 1



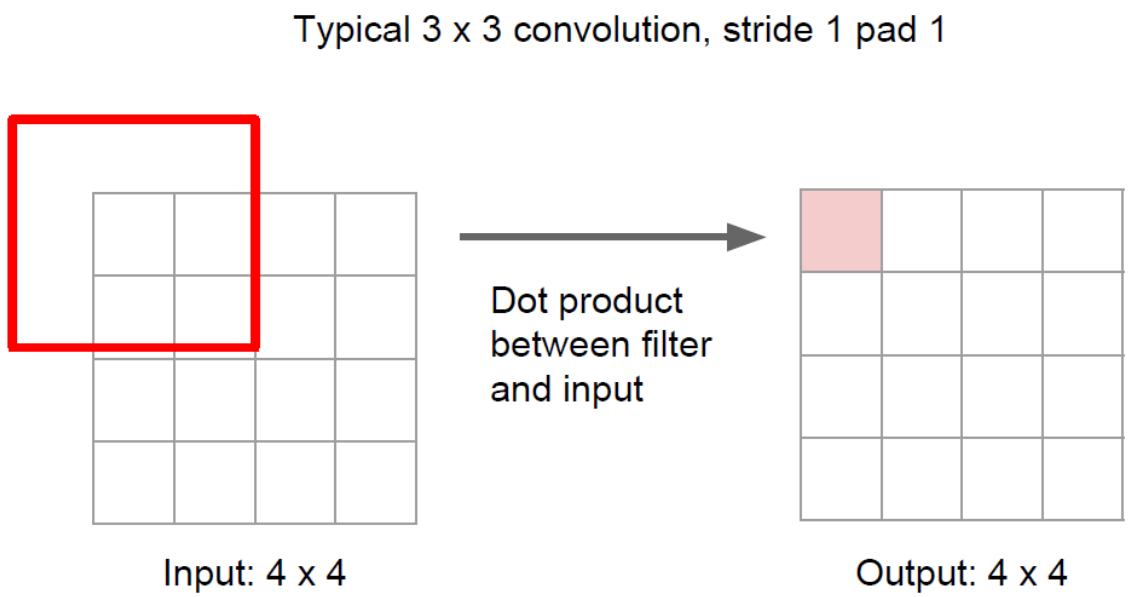
Input: 4 x 4



Output: 4 x 4

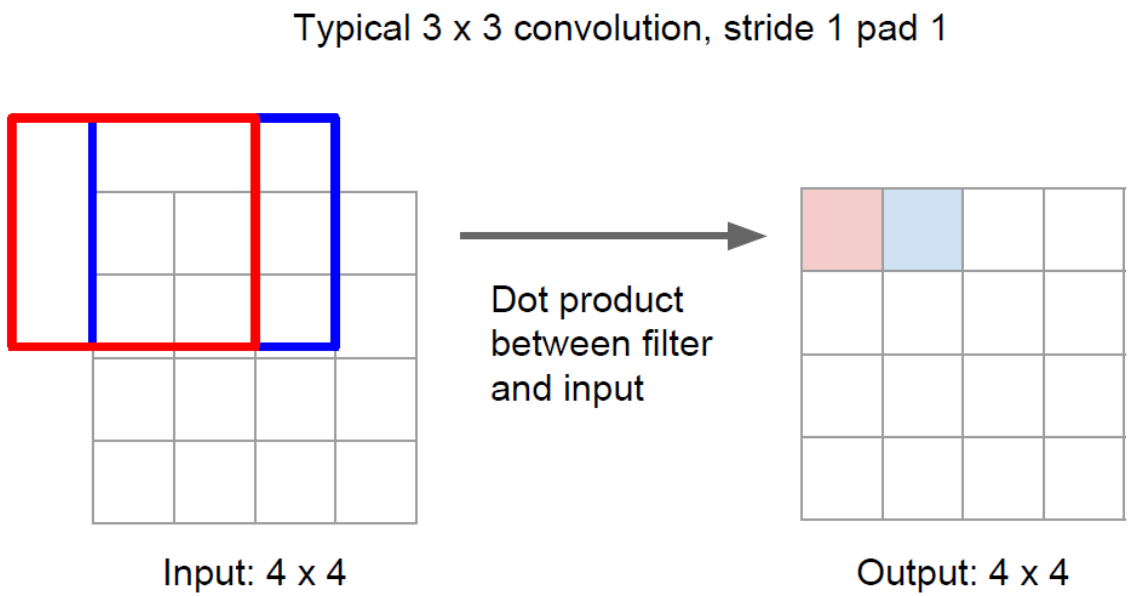
Slide from Stanford CS231n

Learnable Upsampling: "Deconvolution"



Slide from Stanford CS231n

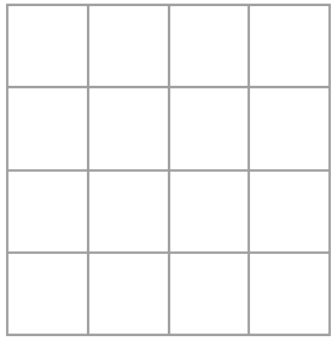
Learnable Upsampling: "Deconvolution"



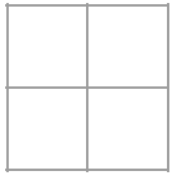
Slide from Stanford CS231n

Learnable Upsampling: “Deconvolution”

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



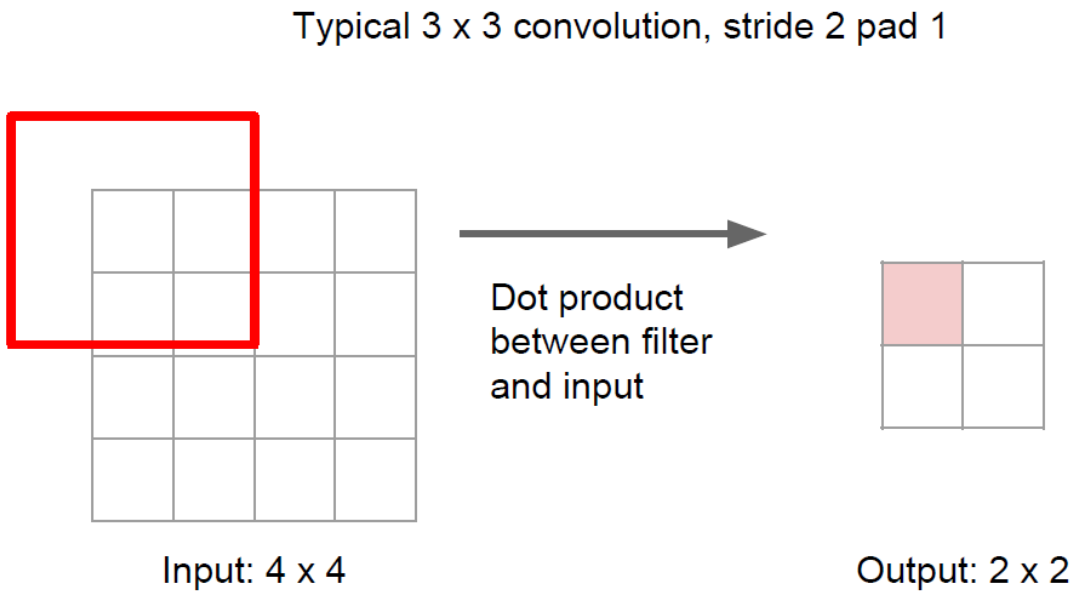
Input: 4 x 4



Output: 2 x 2

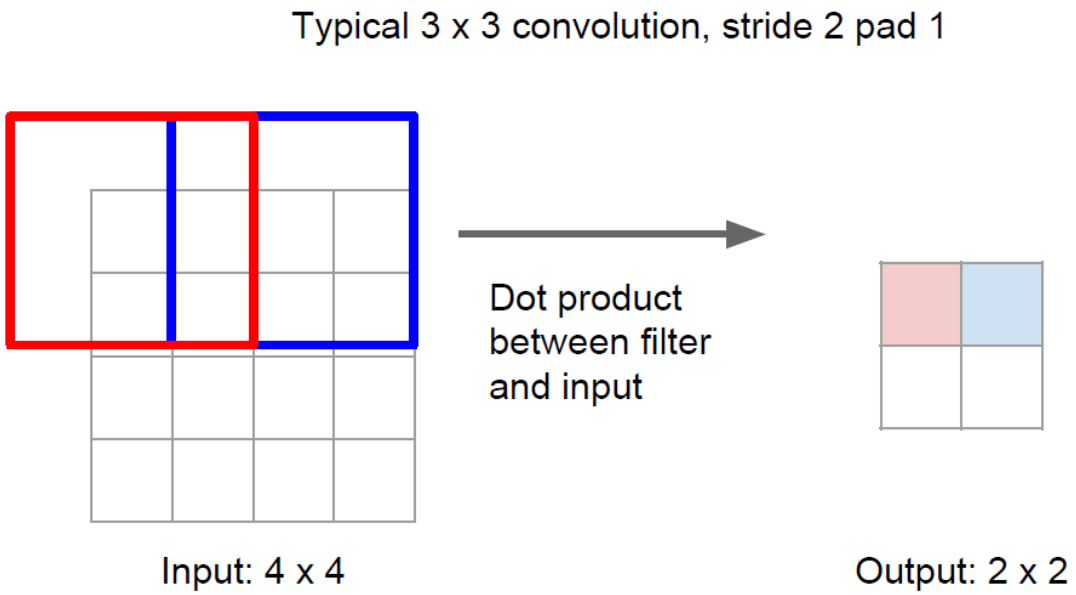
Slide from Stanford CS231n

Learnable Upsampling: "Deconvolution"



Slide from Stanford CS231n

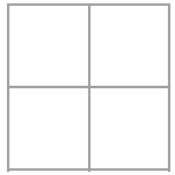
Learnable Upsampling: "Deconvolution"



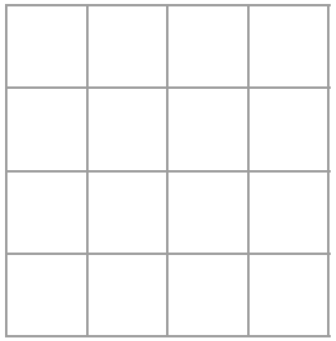
Slide from Stanford CS231n

Learnable Upsampling: “Deconvolution”

3 x 3 “deconvolution”, stride 2 pad 1



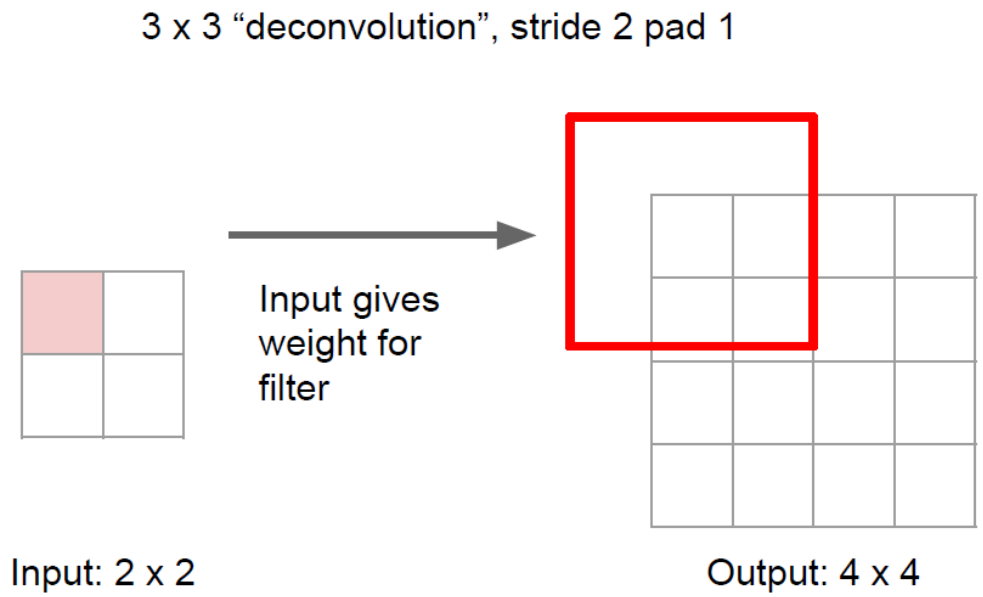
Input: 2 x 2



Output: 4 x 4

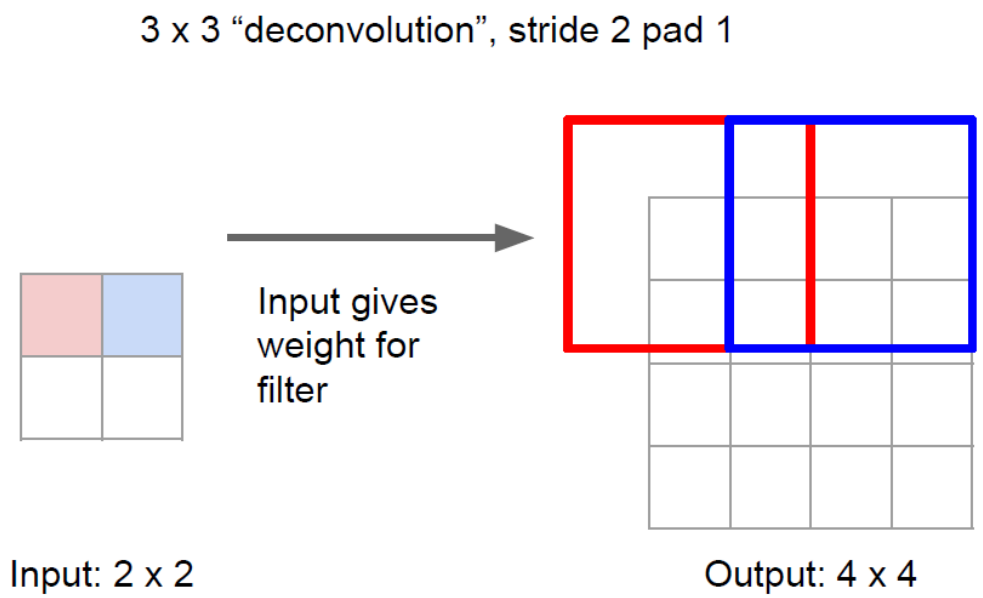
Slide from Stanford CS231n

Learnable Upsampling: "Deconvolution"



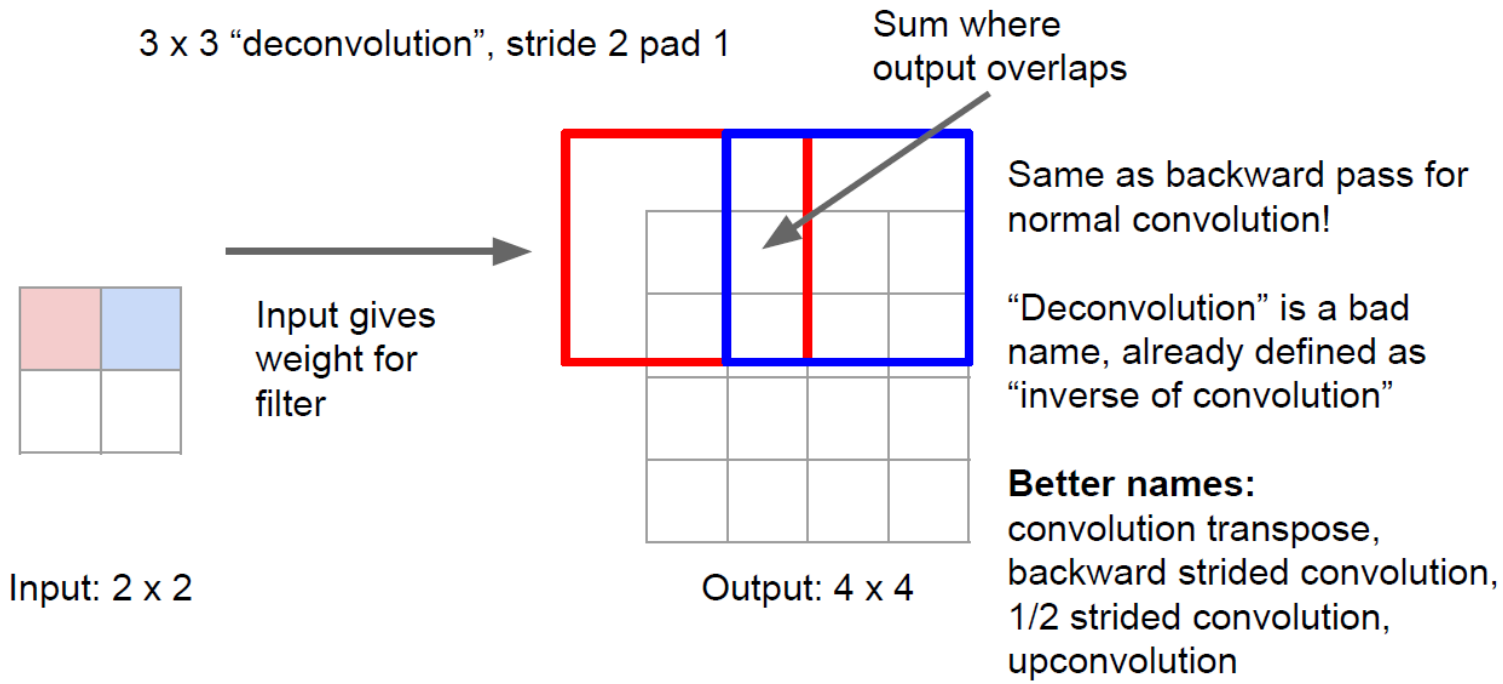
Slide from Stanford CS231n

Learnable Upsampling: “Deconvolution”



Slide from Stanford CS231n

Learnable Upsampling: "Deconvolution"



Slide from Stanford CS231n

Analysis of DeconvNet

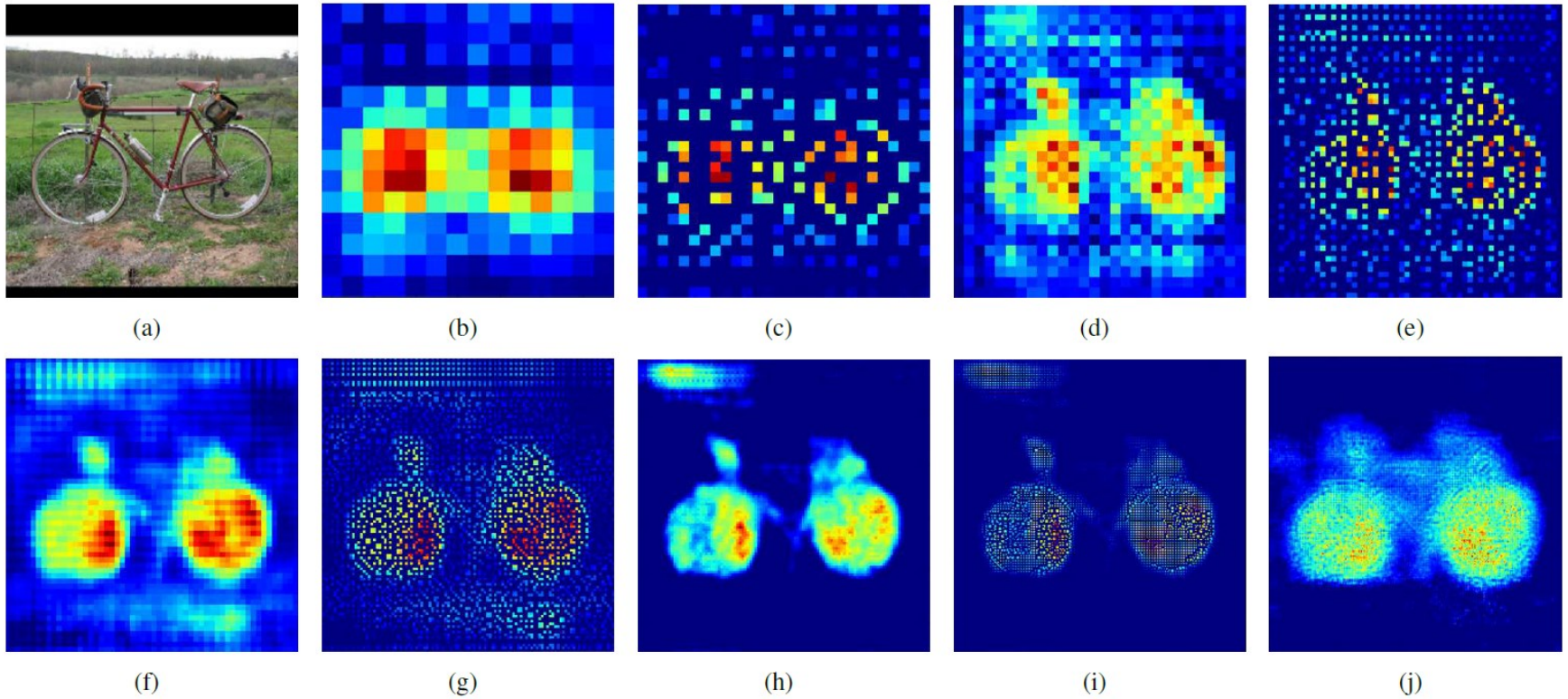
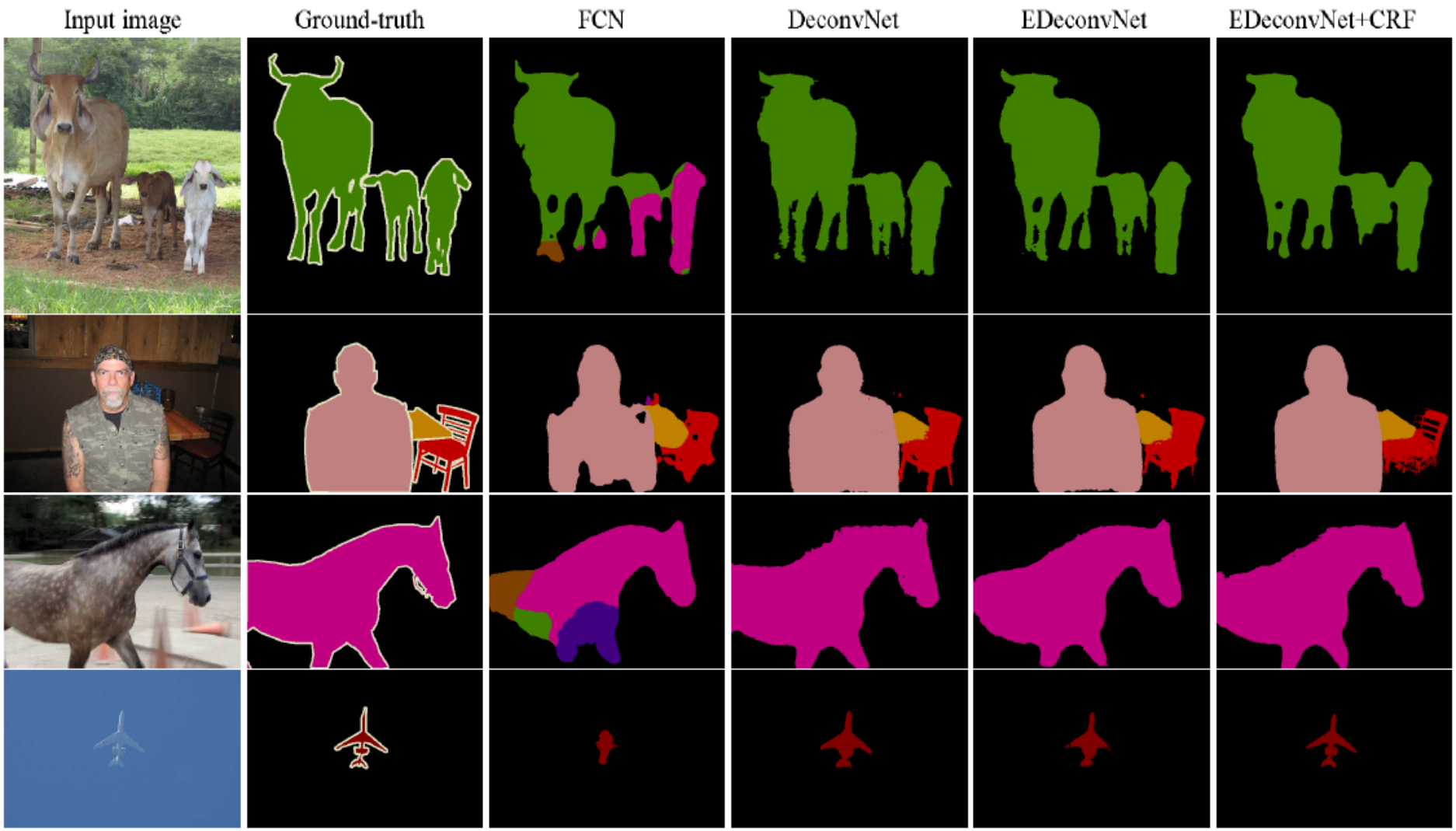


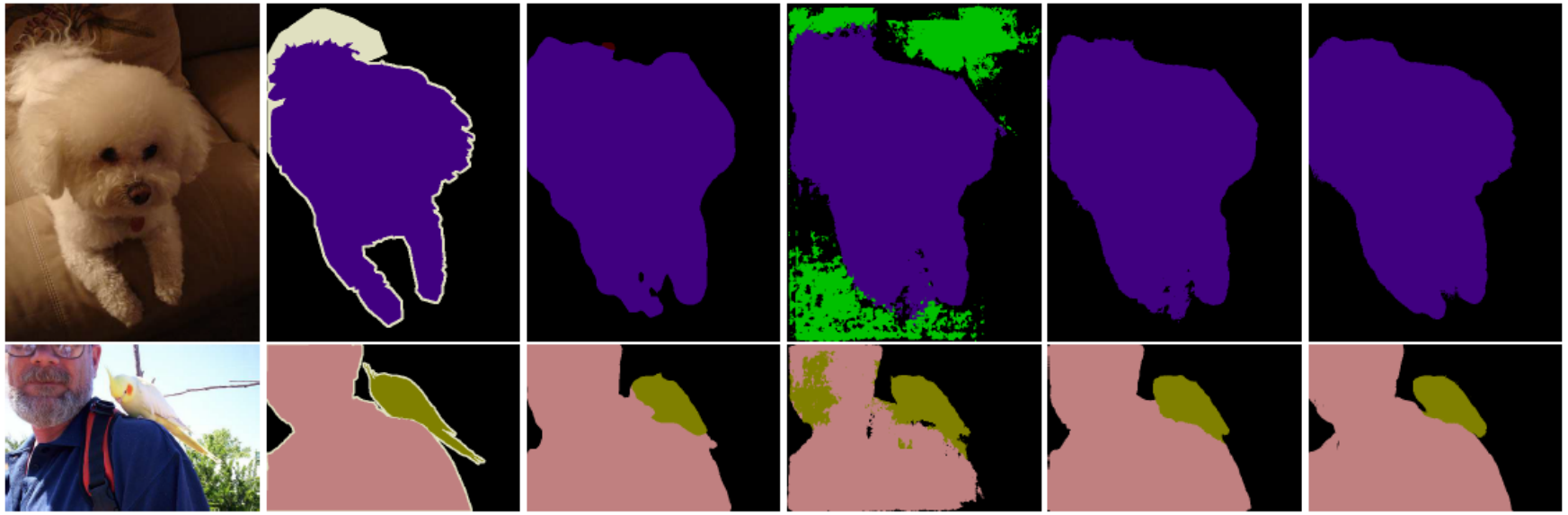
Figure 4. Visualization of activations in our deconvolution network. The activation maps from (b) to (j) correspond to the output maps from lower to higher layers in the deconvolution network. We select the most representative activation in each layer for effective visualization. The image in (a) is an input, and the rest are the outputs from (b) the last 14×14 deconvolutional layer, (c) the 28×28 unpooling layer, (d) the last 28×28 deconvolutional layer, (e) the 56×56 unpooling layer, (f) the last 56×56 deconvolutional layer, (g) the 112×112 unpooling layer, (h) the last 112×112 deconvolutional layer, (i) the 224×224 unpooling layer and (j) the last 224×224 deconvolutional layer. The finer details of the object are revealed, as the features are forward-propagated through the layers in the deconvolution network. Note that noisy activations from background are suppressed through propagation while the activations closely related to the target classes are amplified. It shows that the learned filters in higher deconvolutional layers tend to capture class-specific shape information.

Results

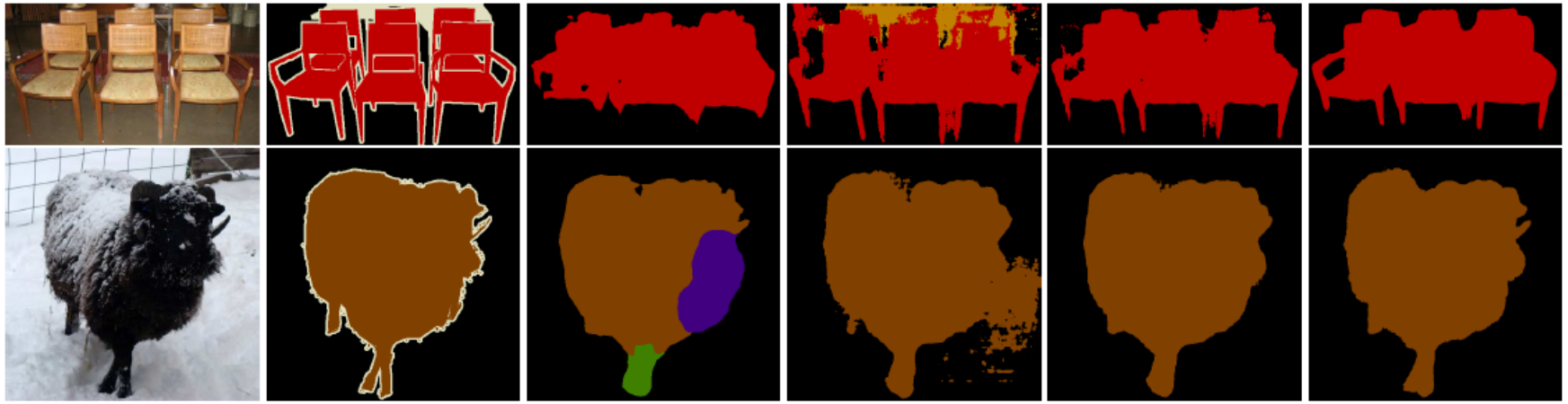


(a) Examples that our method produces better results than FCN [19].

Results



(b) Examples that FCN produces better results than our method.



(c) Examples that inaccurate predictions from our method and FCN are improved by ensemble.

Multi-Scale Context-Aggregation by Dilated Convolutions

MULTI-SCALE CONTEXT AGGREGATION BY DILATED CONVOLUTIONS

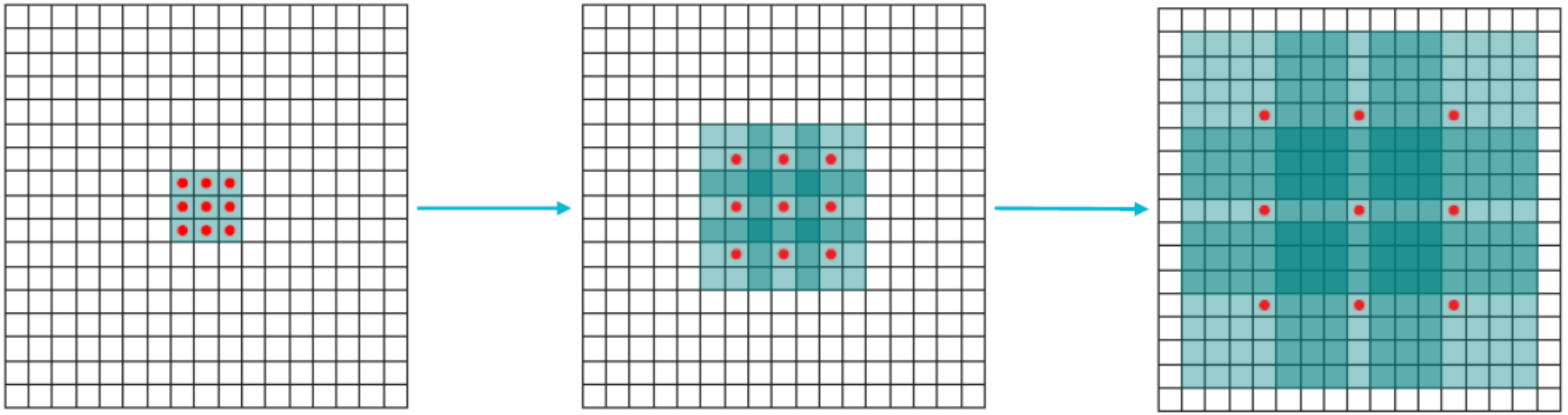
Fisher Yu
Princeton University

Vladlen Koltun
Intel Labs

ABSTRACT

State-of-the-art models for semantic segmentation are based on adaptations of convolutional networks that had originally been designed for image classification. However, dense prediction problems such as semantic segmentation are structurally different from image classification. In this work, we develop a new convolutional network module that is specifically designed for dense prediction. The presented module uses dilated convolutions to systematically aggregate multi-scale contextual information without losing resolution. The architecture is based on the fact that dilated convolutions support exponential expansion of the receptive field without loss of resolution or coverage. We show that the presented context module increases the accuracy of state-of-the-art semantic segmentation systems. In addition, we examine the adaptation of image classification networks to dense prediction and show that simplifying the adapted network can increase accuracy.

Dilated Convolution



3x3 Conv r=1

3x3 Conv r=1

3x3 Conv r=2

3x3 Conv r=1

3x3 Conv r=2

3x3 Conv r=4

3x3 Range

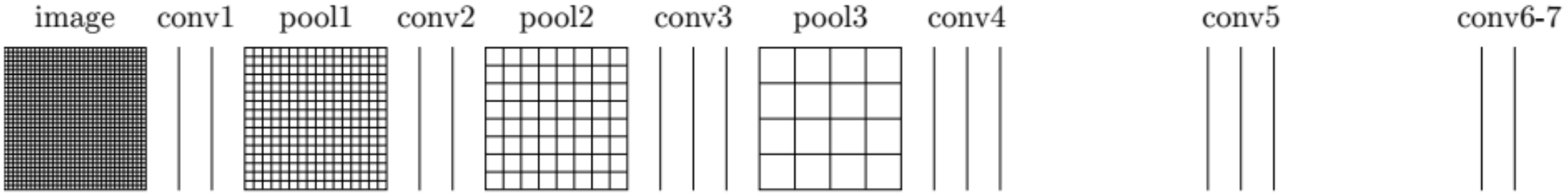
7x7 Range

15x15 Range

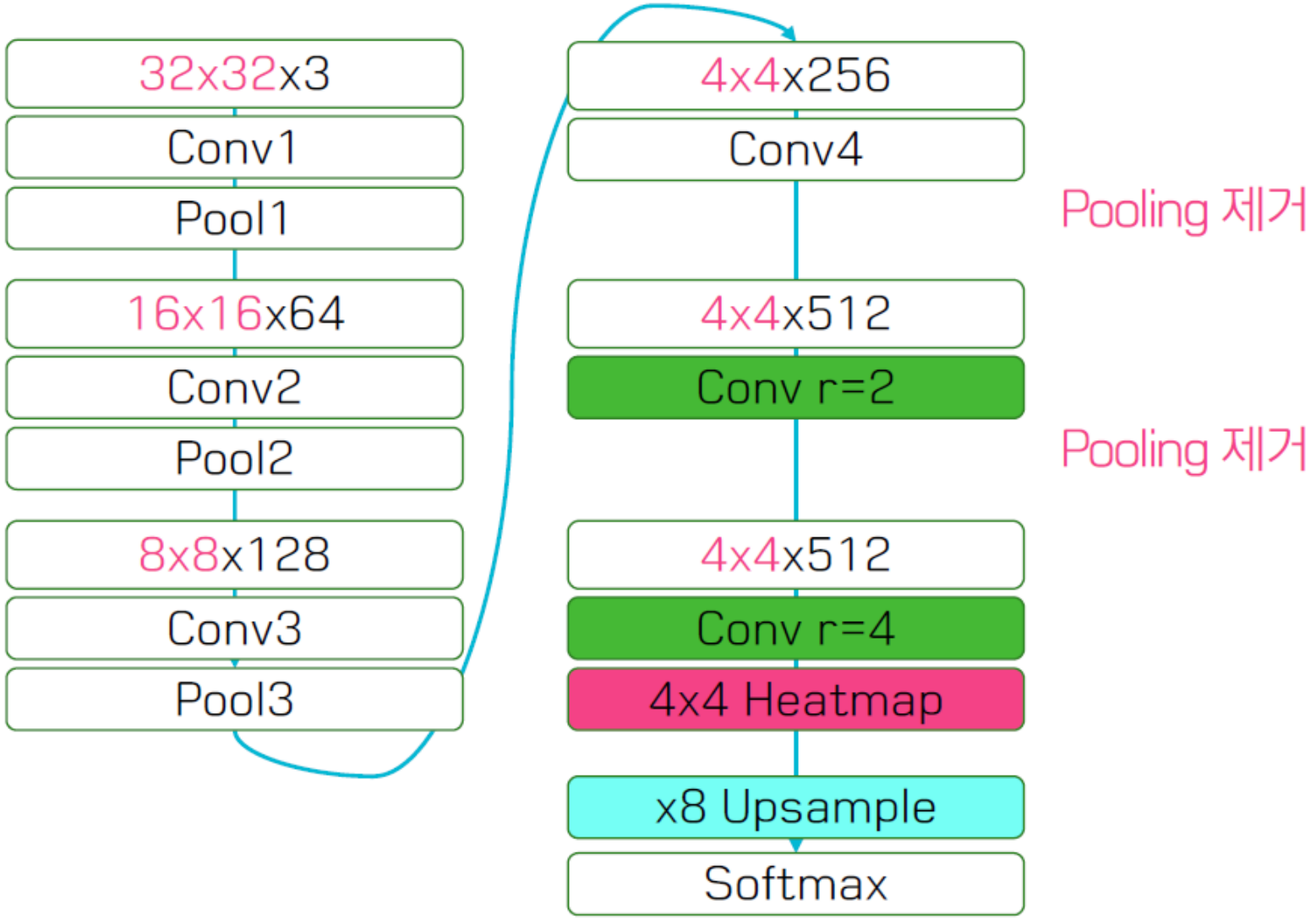
Slide from Taeoh Kim(PR12)

Front-End Module

- FCN에서 pool4, pool5 제거
- conv5 → 2-dilated convolution
- conv6 → 4-dilated convolution



Front-End Module



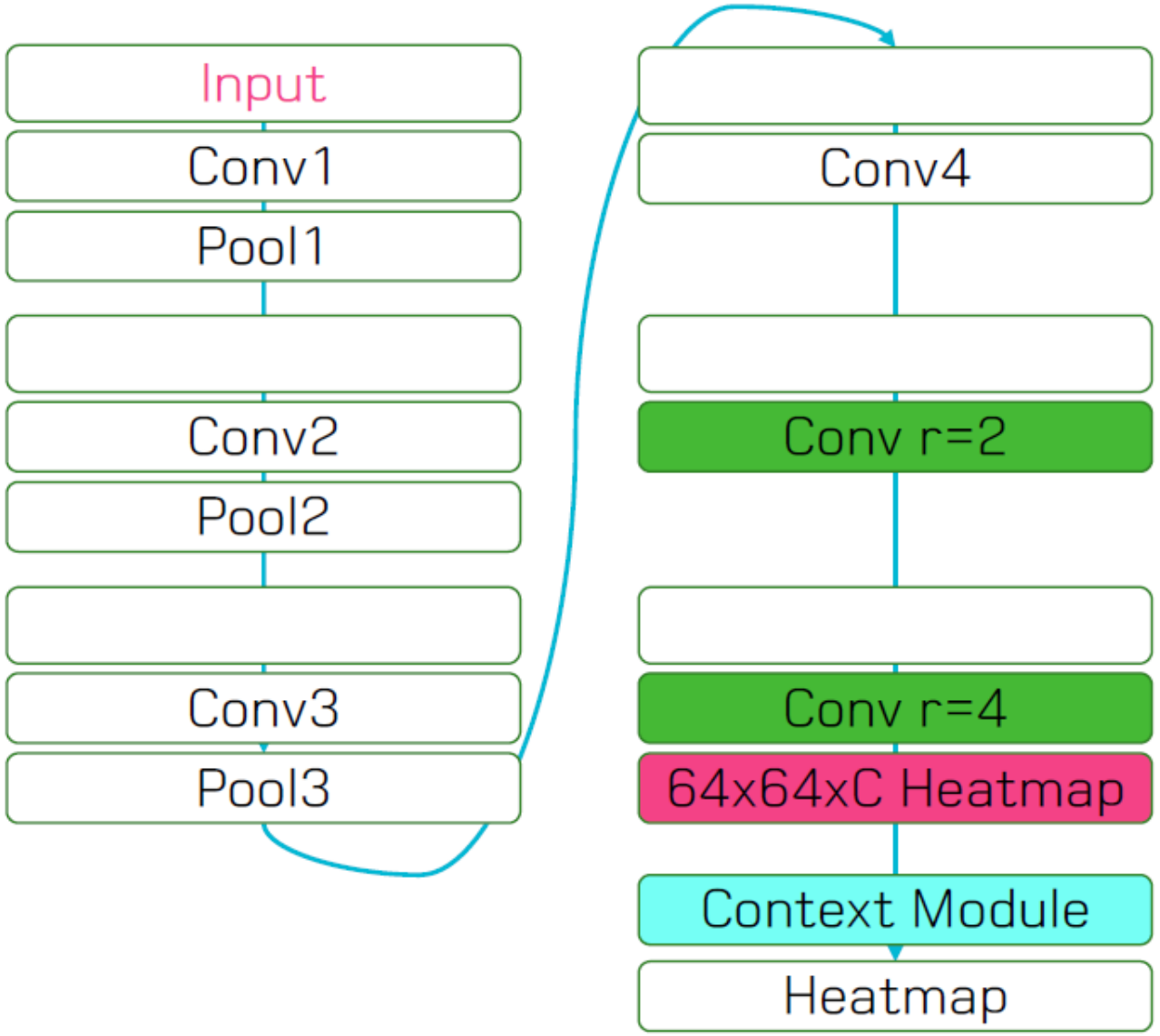
Slide from Taeoh Kim(PR12)

Context Network

Layer	1	2	3	4	5	6	7	8
Convolution	3×3	3×3	3×3	3×3	3×3	3×3	3×3	1×1
Dilation	1	1	2	4	8	16	1	1
Truncation	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Receptive field	3×3	5×5	9×9	17×17	33×33	65×65	67×67	67×67
Output channels								
Basic	C	C	C	C	C	C	C	C
Large	$2C$	$2C$	$4C$	$8C$	$16C$	$32C$	$32C$	C

Table 1: Context network architecture. The network processes C feature maps by aggregating contextual information at progressively increasing scales without losing resolution.

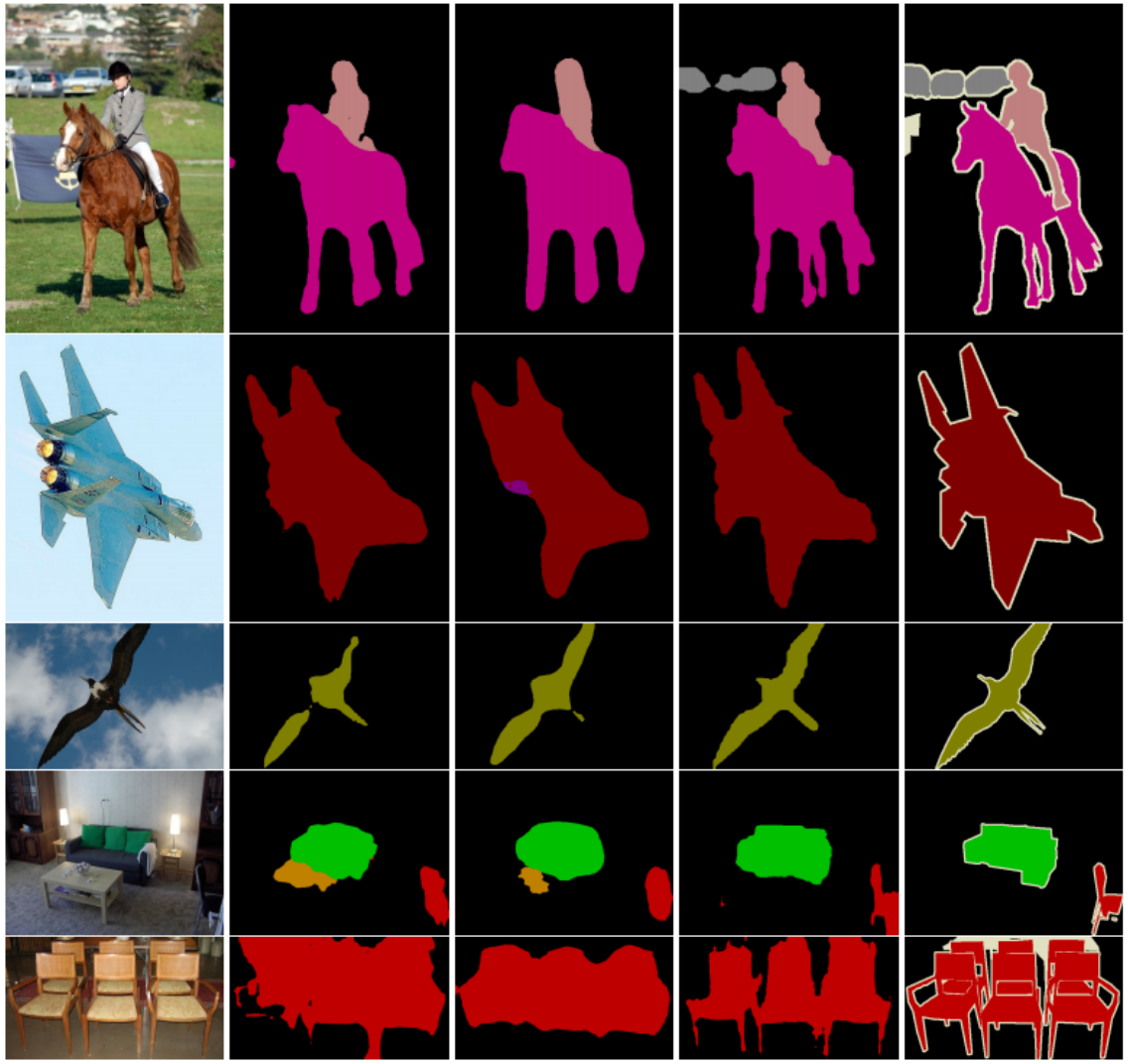
Front-End Module + Context Module



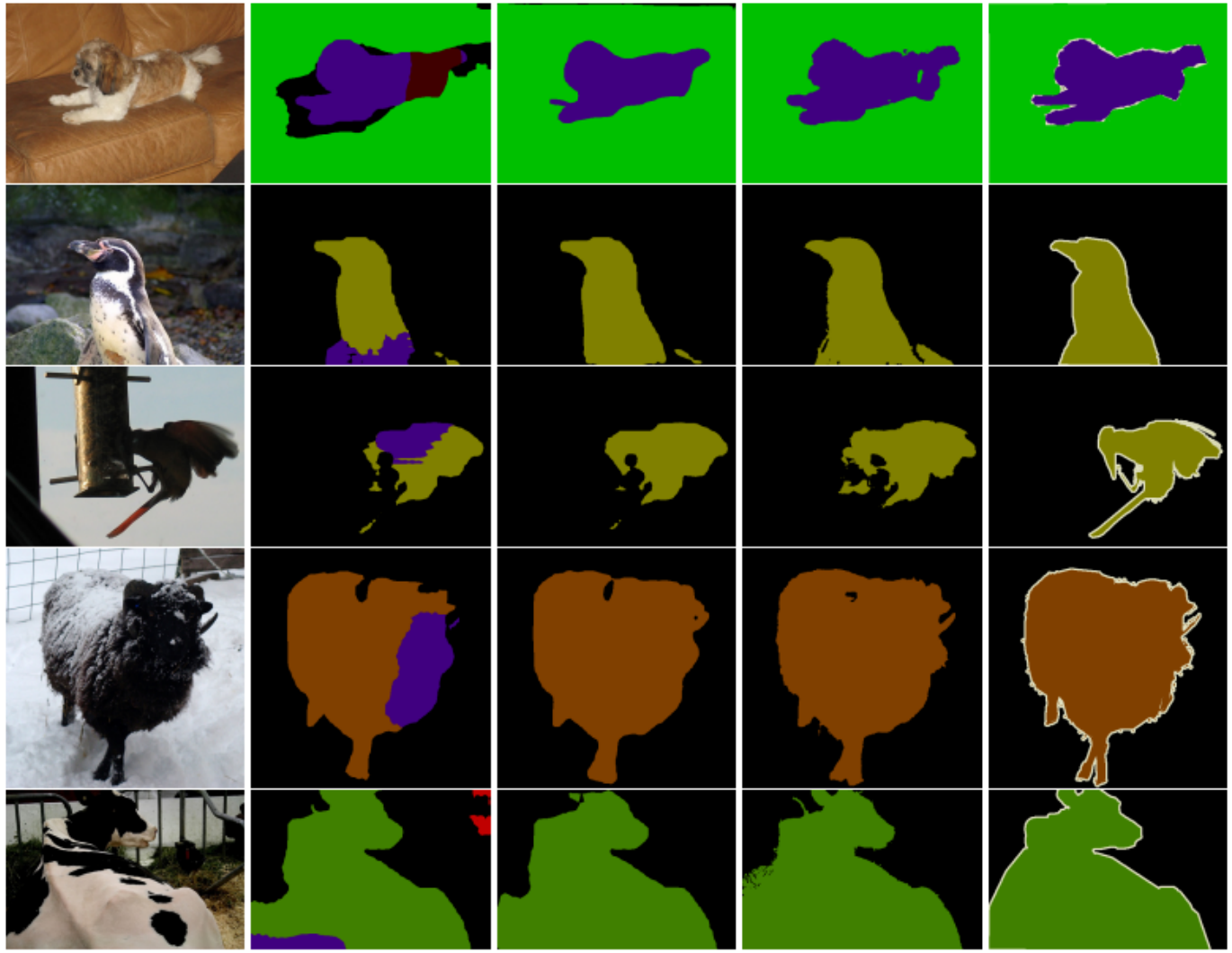
Size에 맞게 Input Padding
C = 21 (Class 개수)

Slide from Taeoh Kim(PR12)

Results



(a) Image (b) FCN-8s (c) DeepLab (d) Our front end (e) Ground truth



(a) Image (b) Front end (c) +Context (d) +CRF-RNN (e) Ground truth

DeepLab(v2)

DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs

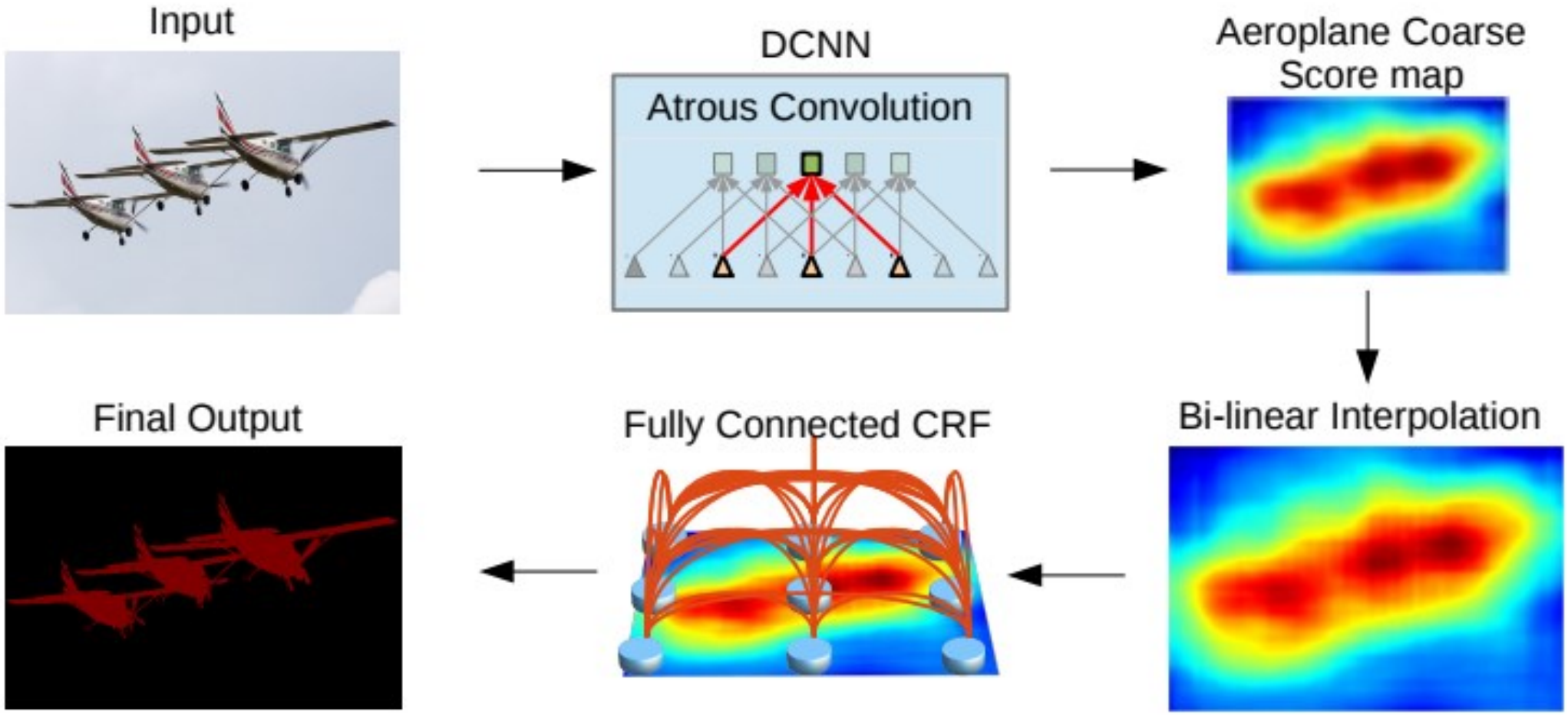
Liang-Chieh Chen, George Papandreou, *Senior Member, IEEE,* Iasonas Kokkinos, *Member, IEEE,* Kevin Murphy, and Alan L. Yuille, *Fellow, IEEE*

Abstract—In this work we address the task of semantic image segmentation with Deep Learning and make three main contributions that are experimentally shown to have substantial practical merit. *First*, we highlight convolution with upsampled filters, or 'atrous convolution', as a powerful tool in dense prediction tasks. Atrous convolution allows us to explicitly control the resolution at which feature responses are computed within Deep Convolutional Neural Networks. It also allows us to effectively enlarge the field of view of filters to incorporate larger context without increasing the number of parameters or the amount of computation. *Second*, we propose atrous spatial pyramid pooling (ASPP) to robustly segment objects at multiple scales. ASPP probes an incoming convolutional feature layer with filters at multiple sampling rates and effective fields-of-views, thus capturing objects as well as image context at multiple scales. *Third*, we improve the localization of object boundaries by combining methods from DCNNs and probabilistic graphical models. The commonly deployed combination of max-pooling and downsampling in DCNNs achieves invariance but has a toll on localization accuracy. We overcome this by combining the responses at the final DCNN layer with a fully connected Conditional Random Field (CRF), which is shown both qualitatively and quantitatively to improve localization performance. Our proposed "DeepLab" system sets the new state-of-art at the PASCAL VOC-2012 semantic image segmentation task, reaching 79.7% mIOU in the test set, and advances the results on three other datasets: PASCAL-Context, PASCAL-Person-Part, and Cityscapes. All of our code is made publicly available online.

Index Terms—Convolutional Neural Networks, Semantic Segmentation, Atrous Convolution, Conditional Random Fields.

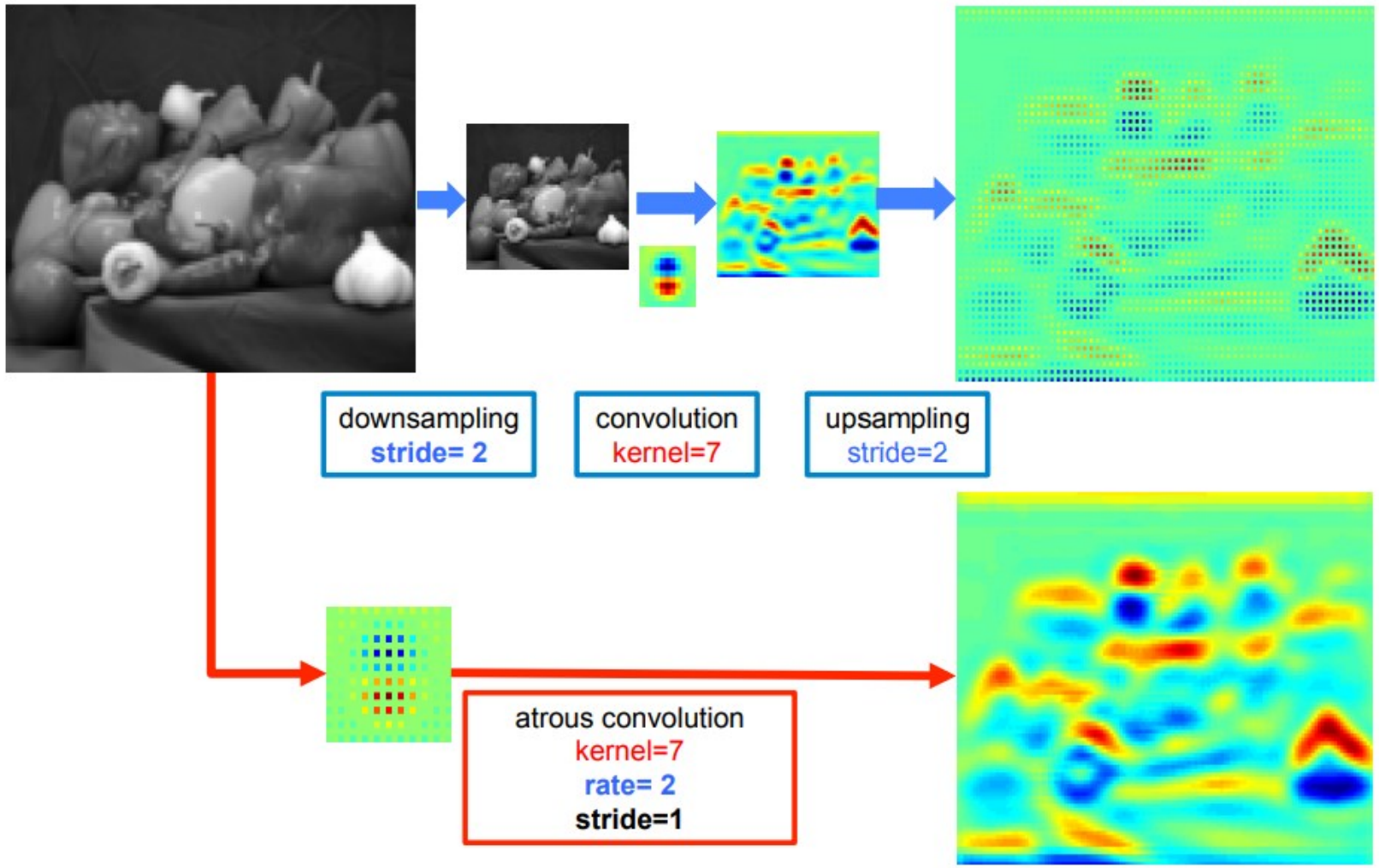
Model Illustartion

- Atrous Convolution + ASPP + Fully Connected CRF

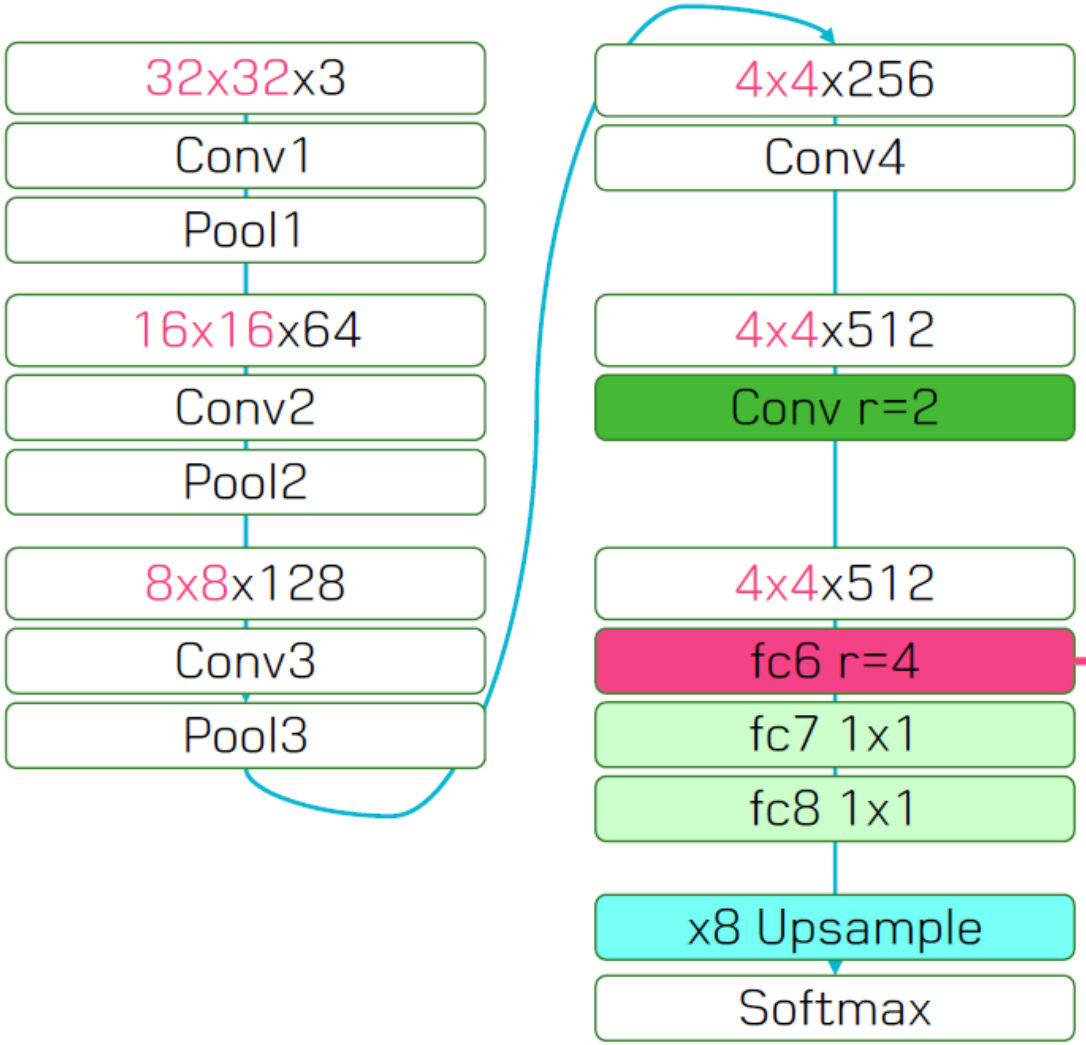


Atrous Convolution = Dilated Convolution

- Dense Feature Extraction



Atrous Convolution
(DeepLab v1 : VGG16, DeepLab v2 : ResNet)



DilatedConv와 비슷 (Front-End)
Fc6 Layer의 rate가 다름

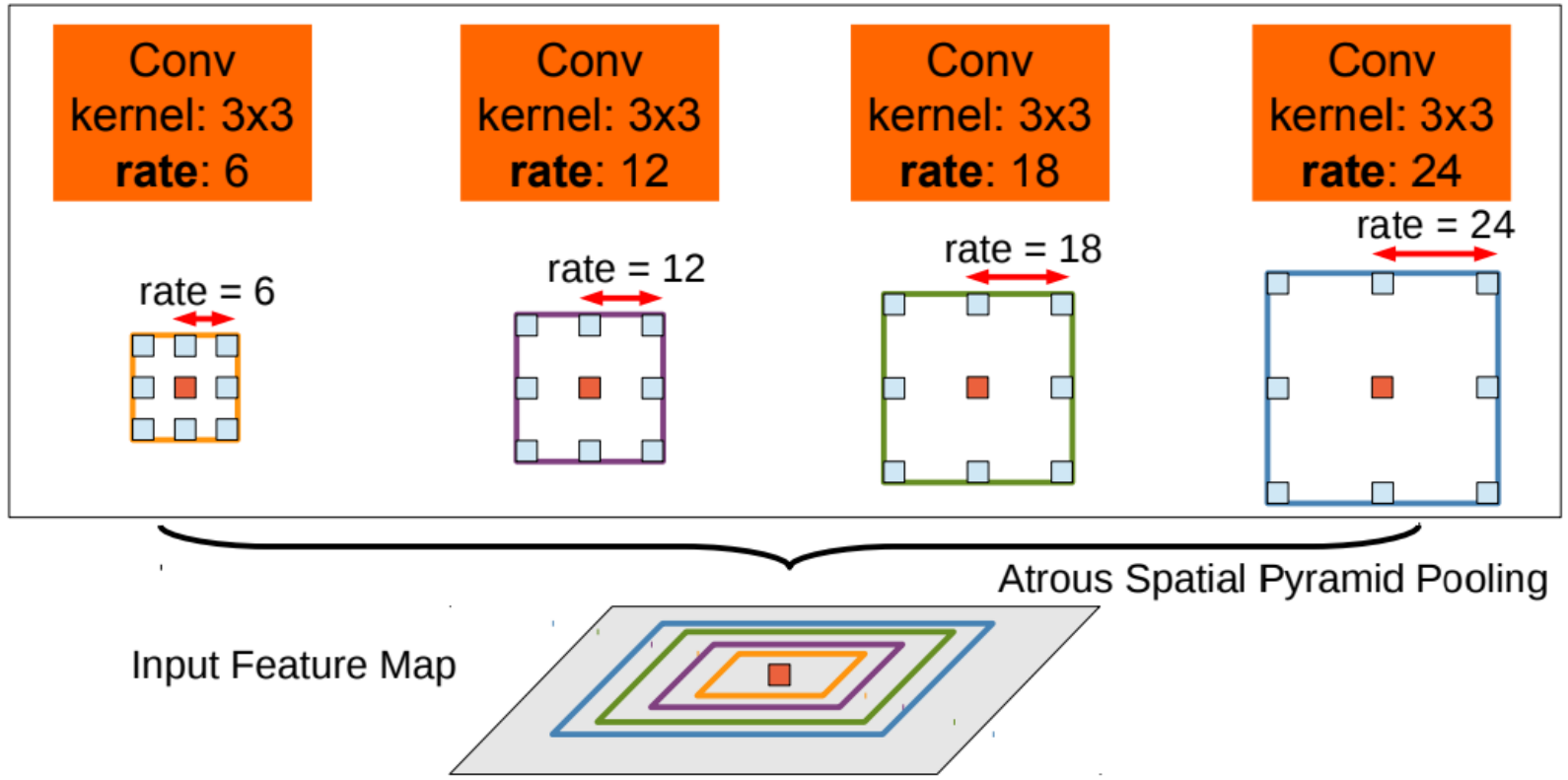
Kernel	Rate	FOV	Params	Speed	bef/aft CRF
7×7	4	224	134.3M	1.44	64.38 / 67.64
4×4	4	128	65.1M	2.90	59.80 / 63.74
4×4	8	224	65.1M	2.90	63.41 / 67.14
3×3	12	224	20.5M	4.84	62.25 / 67.64

TABLE 1: Effect of Field-Of-View by adjusting the kernel size and atrous sampling rate r at 'fc6' layer. We show number of model parameters, training speed (img/sec), and *val* set mean IOU before and after CRF. DeepLab-LargeFOV (kernel size 3×3, $r = 12$) strikes the best balance.

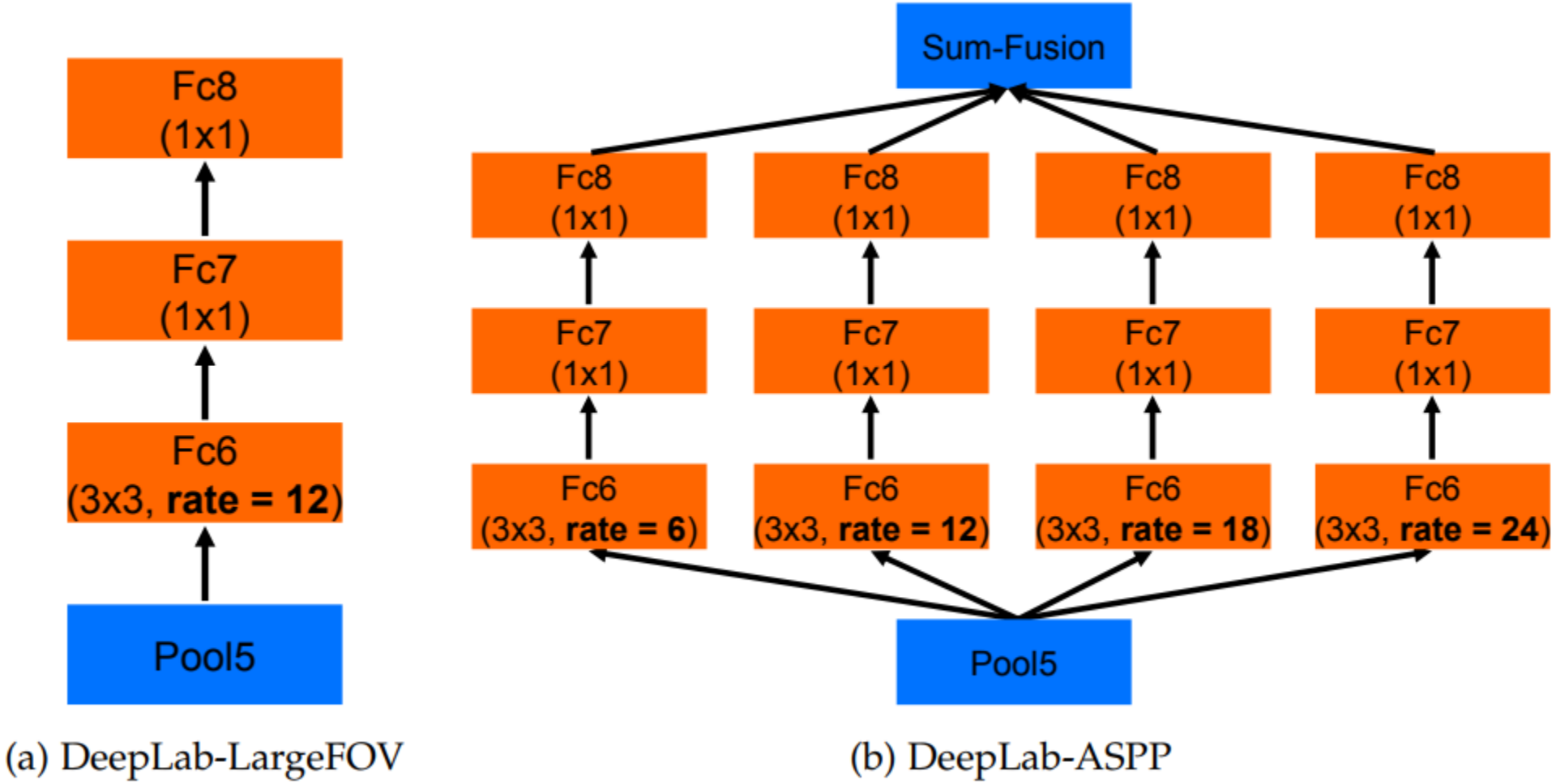
Slide from Taeoh Kim(PR12)

ASPP : Atrous Spatial Pyramid Pooling

- Dilated Conv is not enough for Multi-scale view
- To classify the center pixel (orange), ASPP exploits multi-scale features by employing multiple parallel filters with different rates.



ASPP



(a) DeepLab-LargeFOV

(b) DeepLab-ASPP

Fully Connected Conditional Random Field

Maximize **Posterior**

$$P(X|I) = \frac{1}{Z(I)} \exp\left(-\sum \phi_c(X_C|I)\right)$$

Image

Label

Normalization

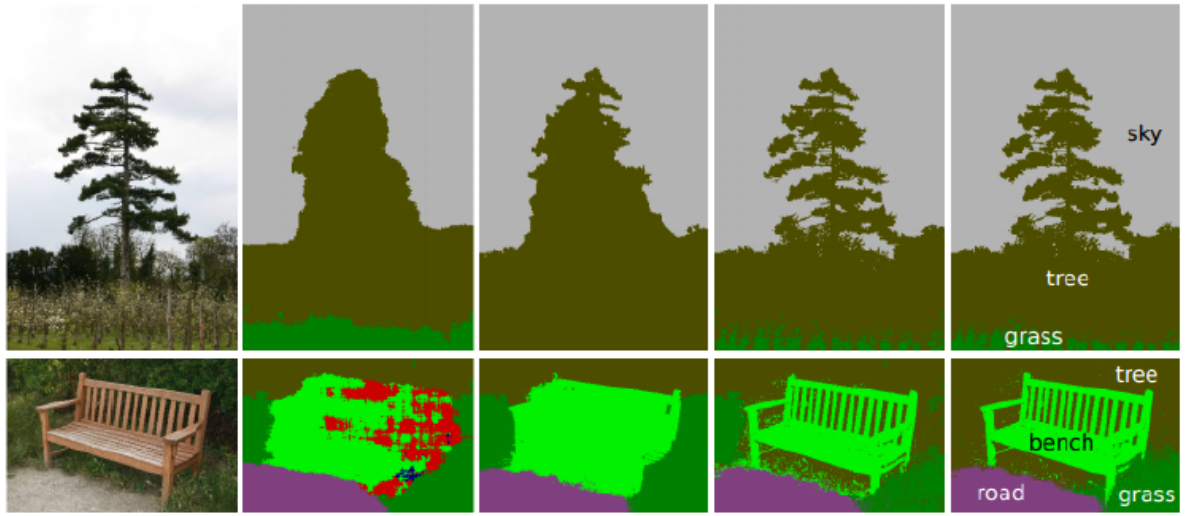
Minimize **Energy**

$$E(X|I) = \sum \phi_c(X_C|I)$$
$$E(X) = \sum \psi_C(X_C)$$

$$E(x) = \sum_i \psi_i(x_i) + \sum_{i,j} \psi_{i,j}(x_i, x_j)$$

Unary

Fully Connected



(a) Image (b) Unary classifiers (c) Robust P^n CRF (d) Fully connected CRF, MCMC inference, 36 hrs (e) Fully connected CRF, our approach, 0.2 seconds

Slide from Taeoh Kim(PR12)

Fully Connected Conditional Random Field

$$E(x) = \sum_i \psi_i(x_i) + \sum_{i,j} \psi_{i,j}(x_i, x_j)$$

Unary Term (from Classifier) $\psi_i(x_i) = -\log \mathrm{P}(x_i)$

Pairwise Term

$$\psi_{i,j}(x_i, x_j) = \mu(x_i, x_j)[w_1 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\alpha^2} - \frac{\|I_i - I_j\|^2}{2\sigma_\beta^2}\right) + w_2 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\gamma^2}\right)]$$

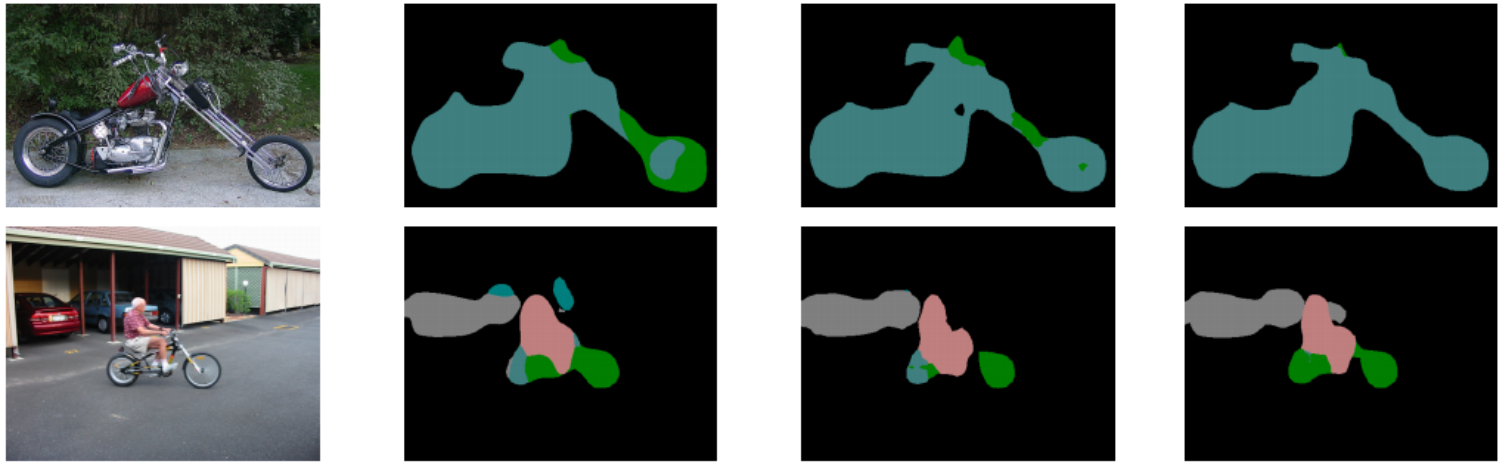
$$\mu(x_i, x_j) = \begin{cases} 1 & x_i \neq x_j \\ 0 & Otherwise \end{cases}$$

p_i, p_j: Pixel 위치
I_i, I_j: Pixel RGB값
w_1, w_2: Kernel Weights
$\sigma_\alpha, \sigma_\alpha, \sigma_\alpha$: Hyper-parameter

의미?
Pixel이 서로 비슷한데
(위치적으로, RGB상으로)
Label이 서로 다르면
Energy 증가하여 Penalty

Slide from Taeoh Kim(PR12)

Results



(a) Image (b) LargeFOV (c) ASPP-S (d) ASPP-L

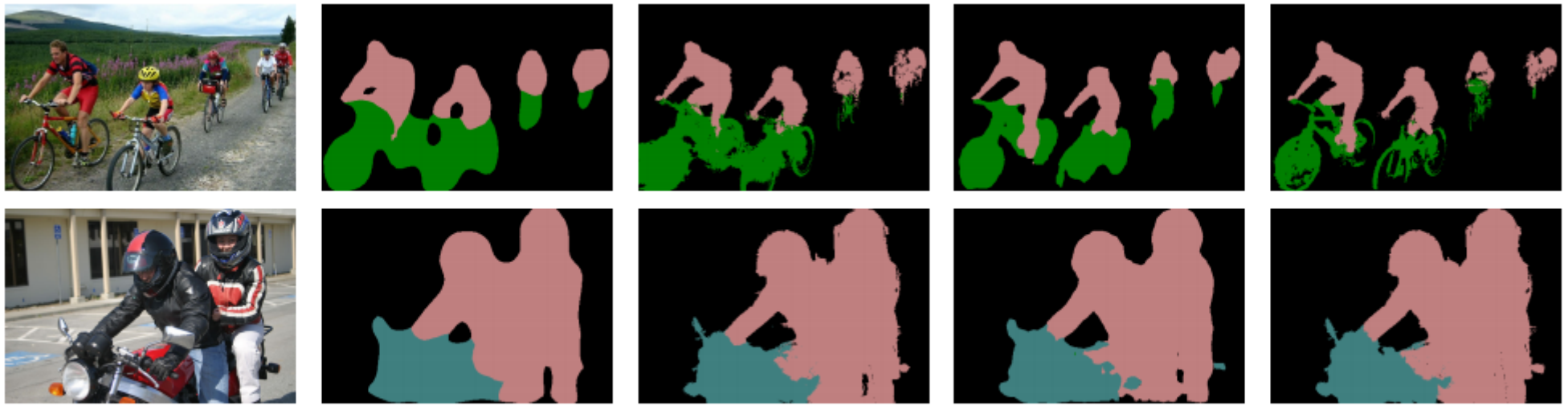


Image VGG-16 Bef. VGG-16 Aft. ResNet Bef. ResNet Aft.

U-Net: Convolutional Networks for Biomedical Image Segmentation

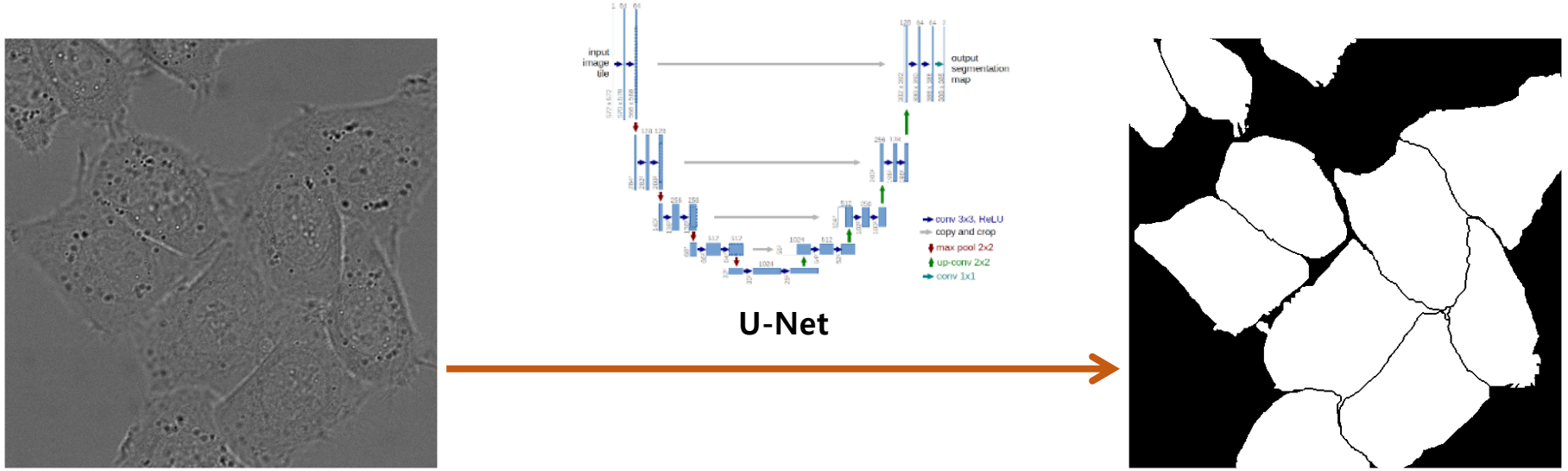
U-Net: Convolutional Networks for Biomedical Image Segmentation

Olaf Ronneberger, Philipp Fischer, and Thomas Brox

Computer Science Department and BIOSS Centre for Biological Signalling Studies,
University of Freiburg, Germany
ronneber@informatik.uni-freiburg.de,
WWW home page: http://lmb.informatik.uni-freiburg.de/

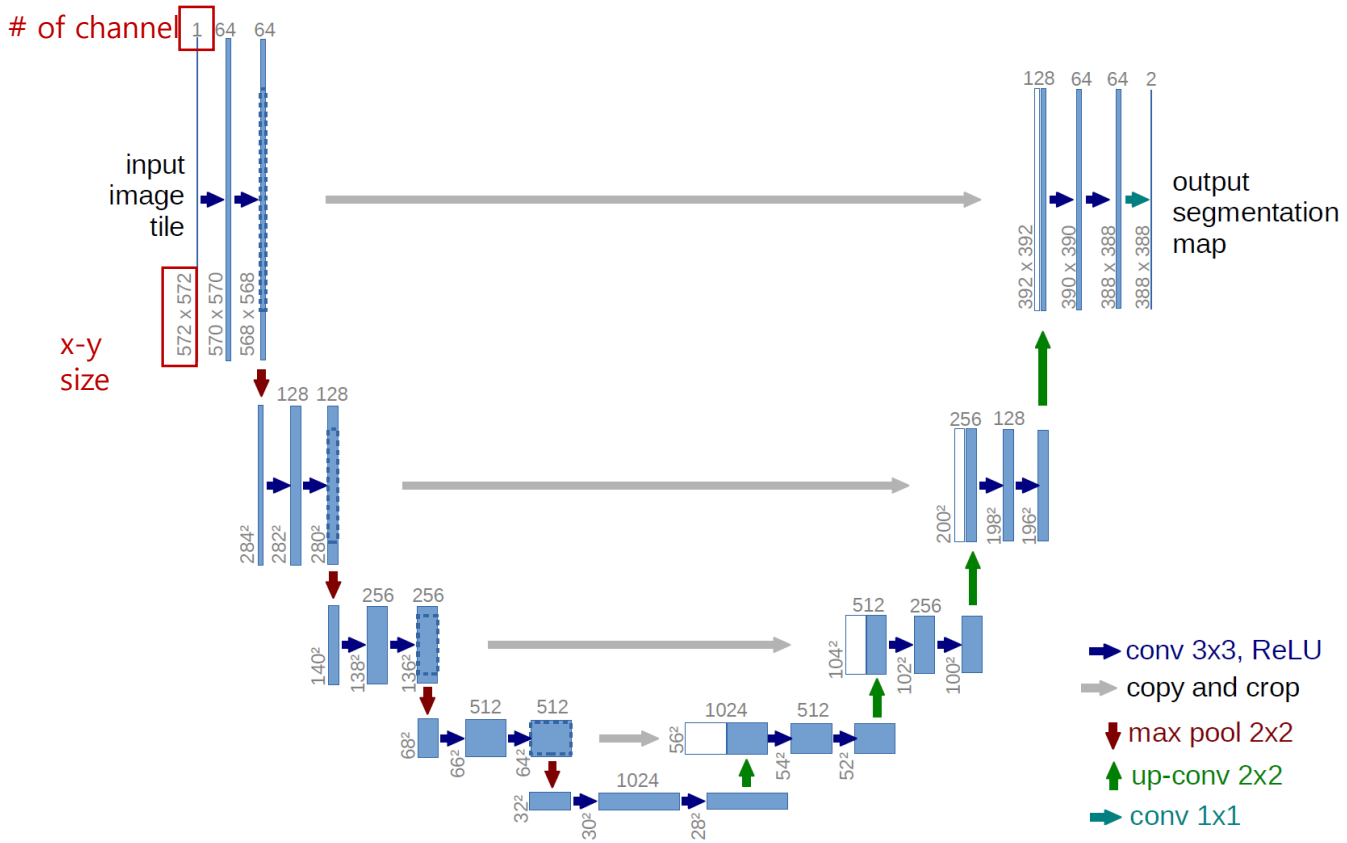
Abstract. There is large consent that successful training of deep networks requires many thousand annotated training samples. In this paper, we present a network and training strategy that relies on the strong use of data augmentation to use the available annotated samples more efficiently. The architecture consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. We show that such a network can be trained end-to-end from very few images and outperforms the prior best method (a sliding-window

Biomedical Image Segmentation with U-net

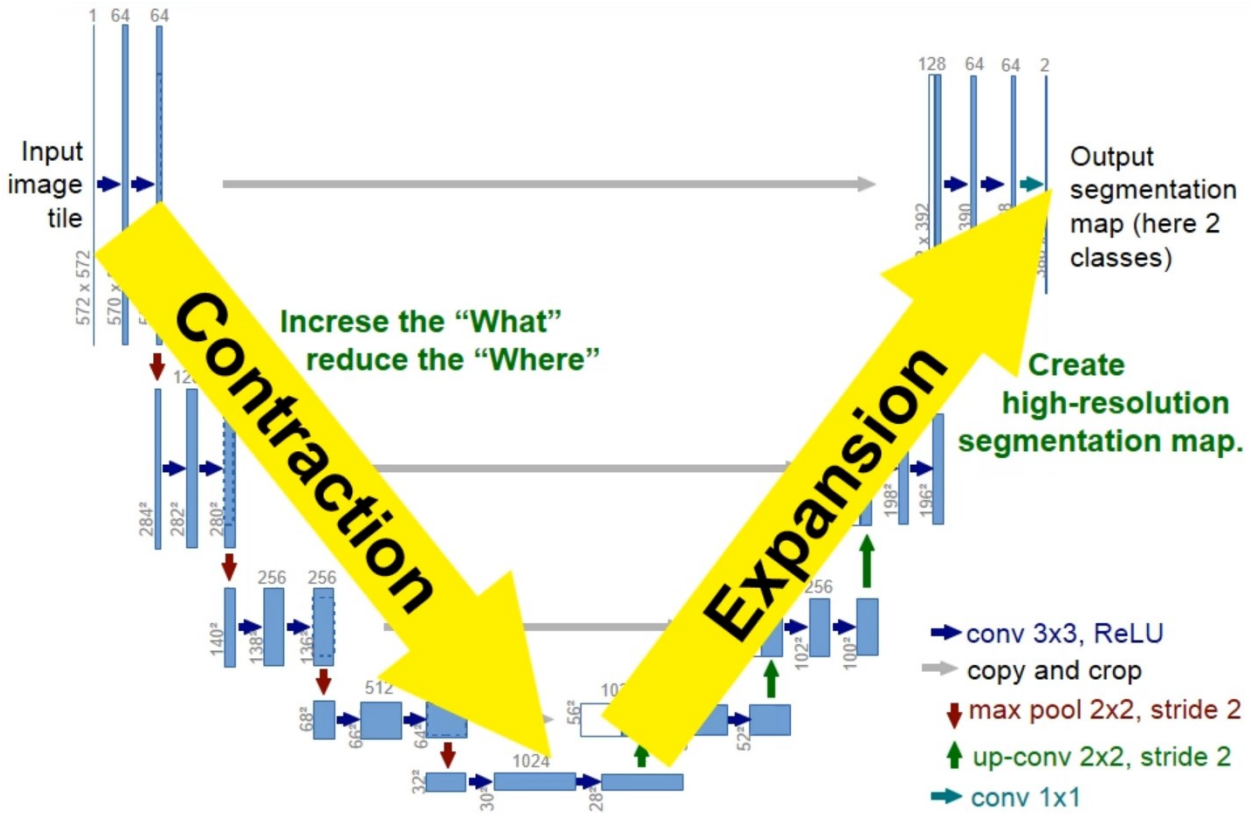


- U-Net learns segmentation in an end-to-end setting
- Very few annotated images (approx. 30 per application)

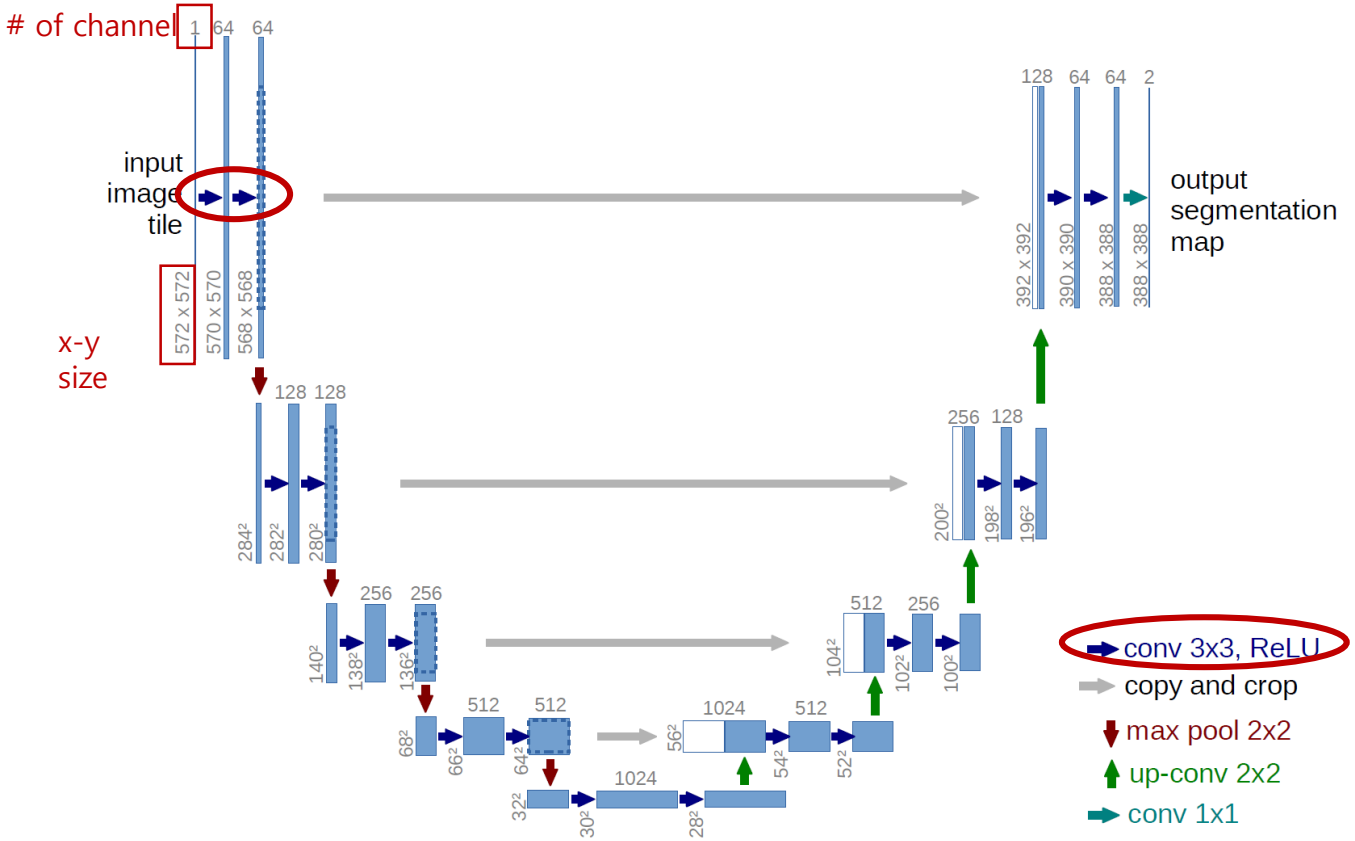
U-Net Architecture



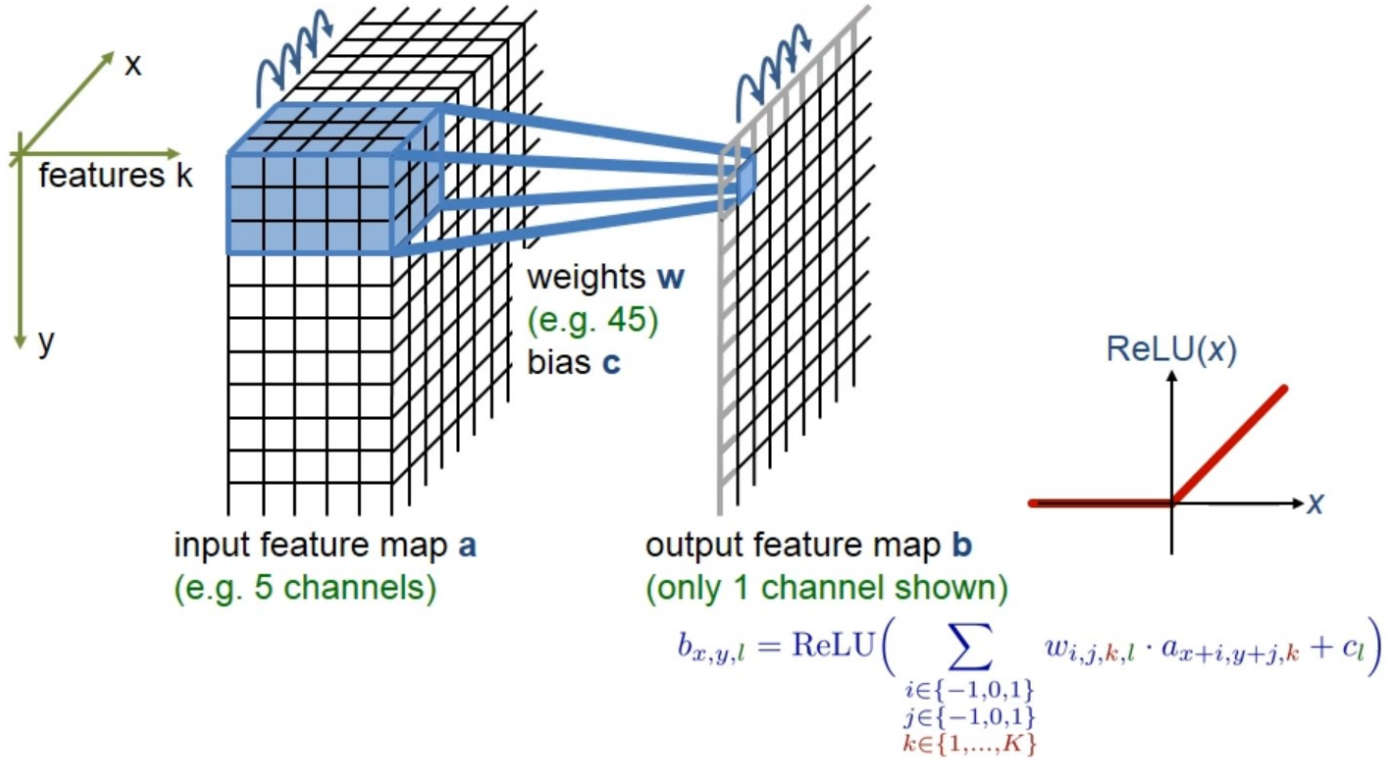
U-Net Architecture



U-Net Architecture

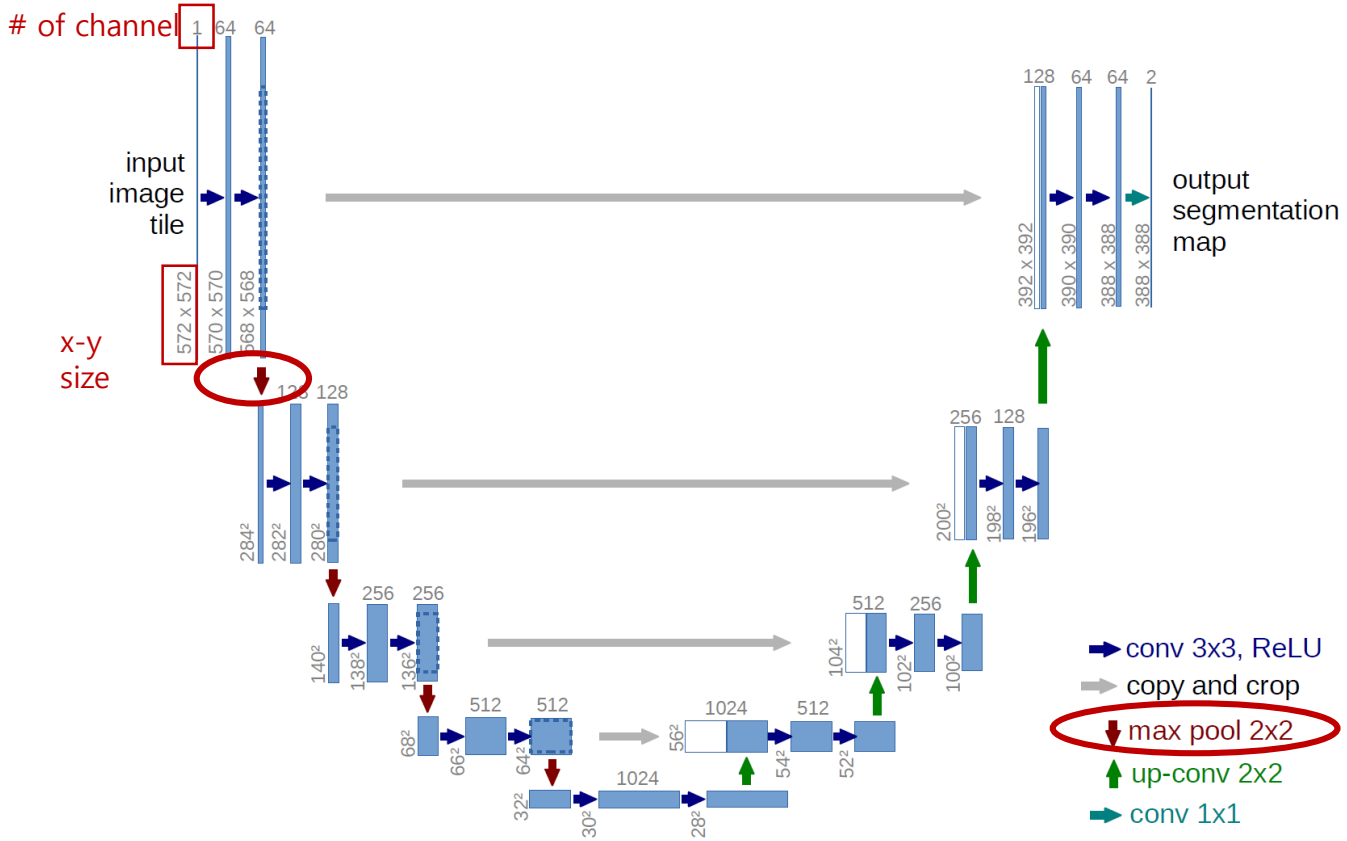


3x3 Convolution + ReLU

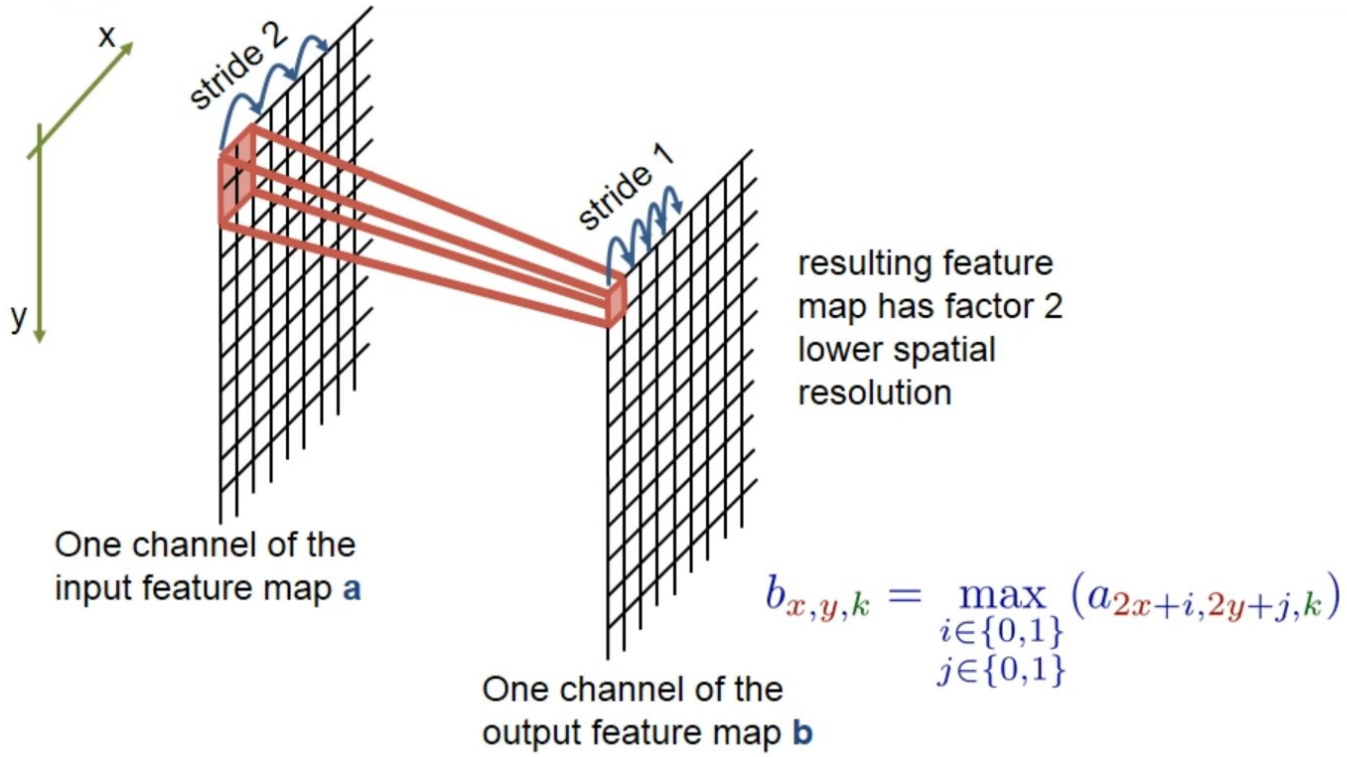


$$b_{x,y,l} = \mathrm{ReLU}\Big(\sum_{\substack{i\in\{-1,0,1\}\\ j\in\{-1,0,1\}\\ k\in\{1,\ldots,K\}}} w_{i,j,k,l} \cdot a_{x+i,y+j,k} + c_l \Big)$$

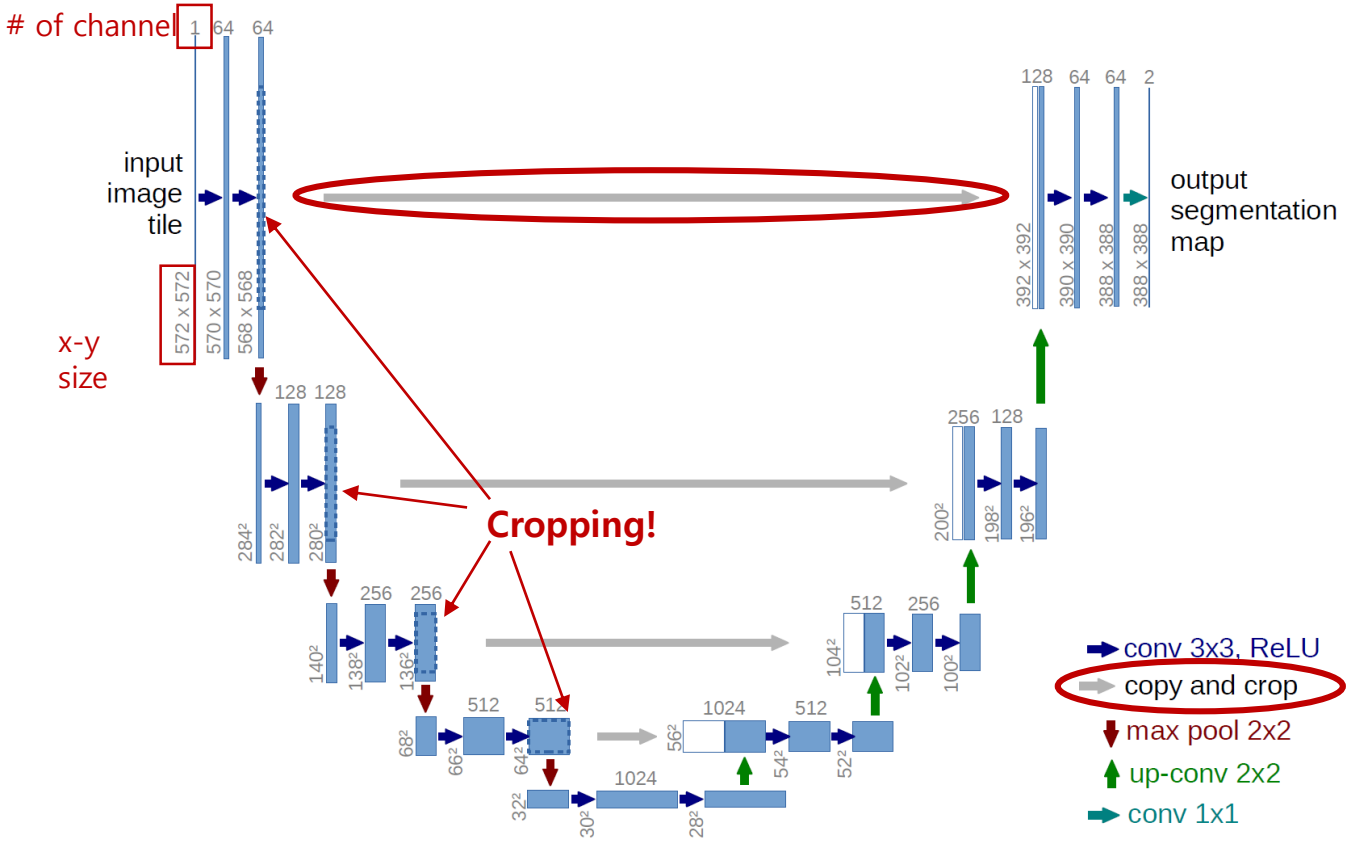
U-Net Architecture



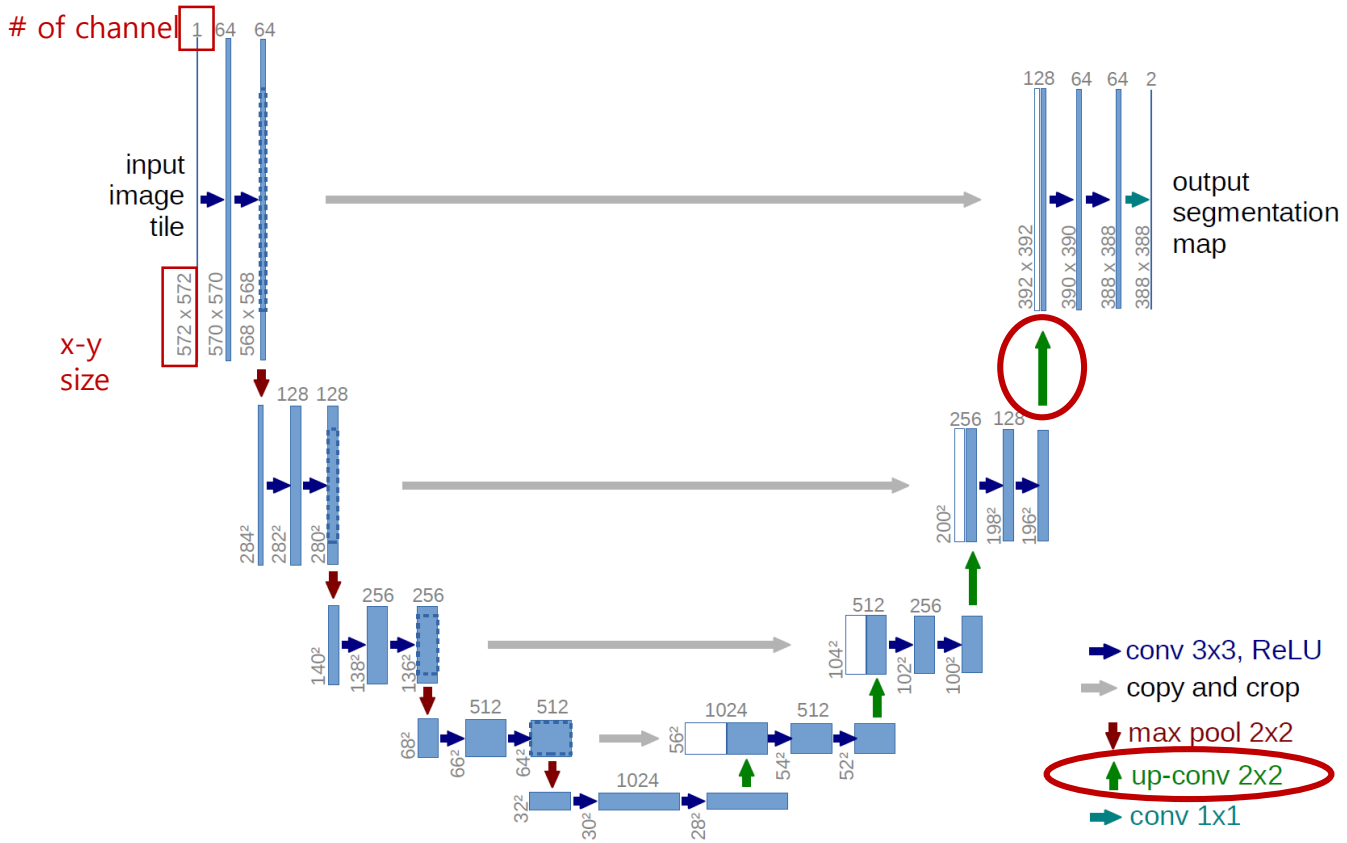
2x2 Max-pooling



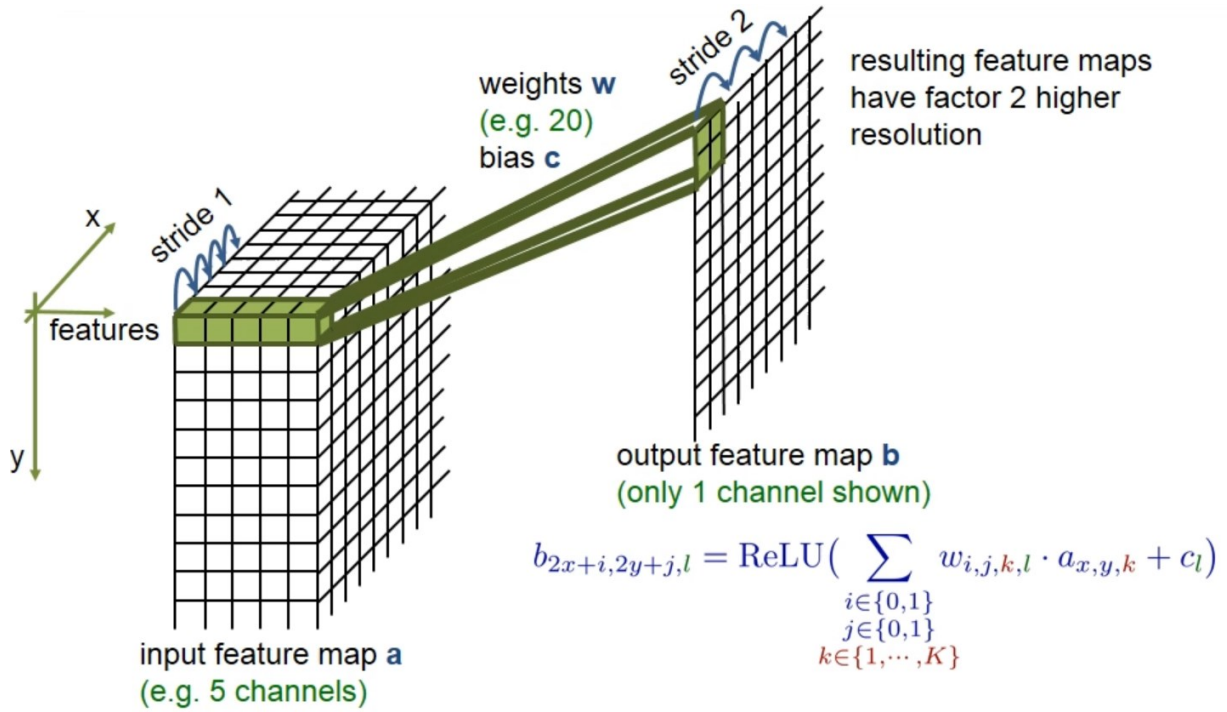
U-Net Architecture



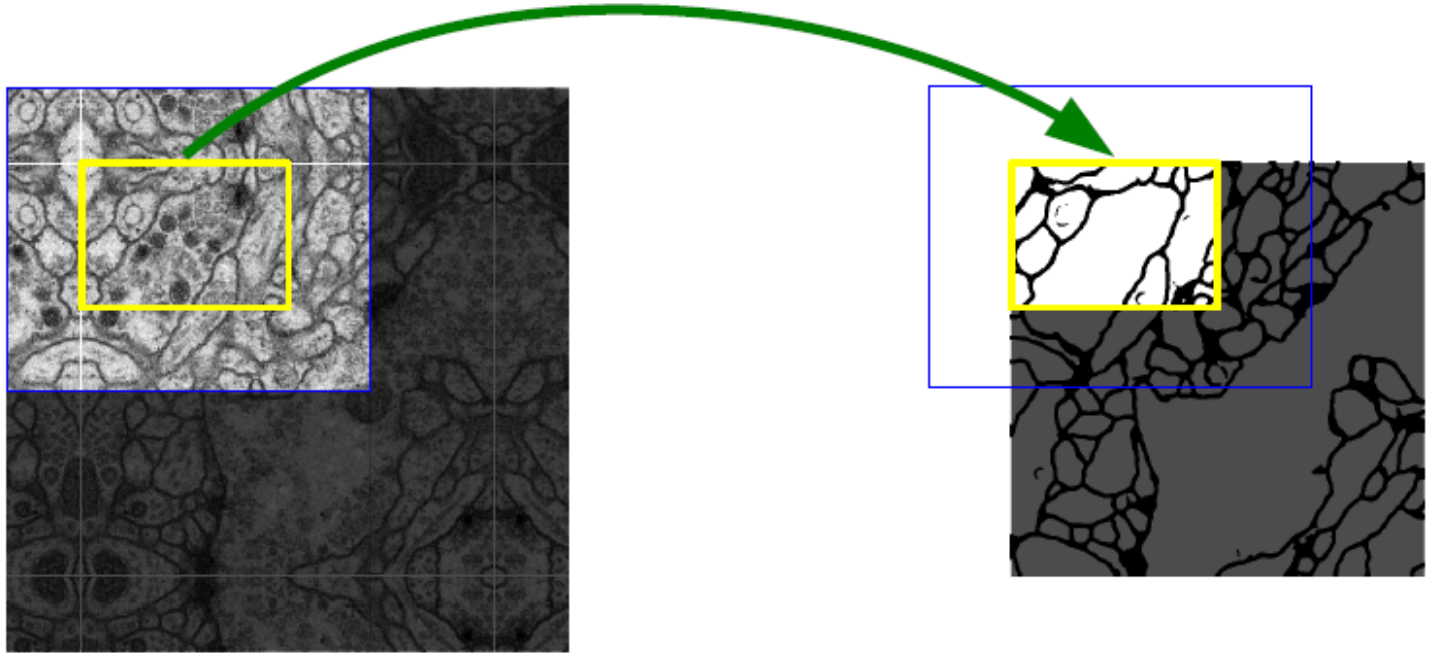
U-Net Architecture



2x2 Up-convolution



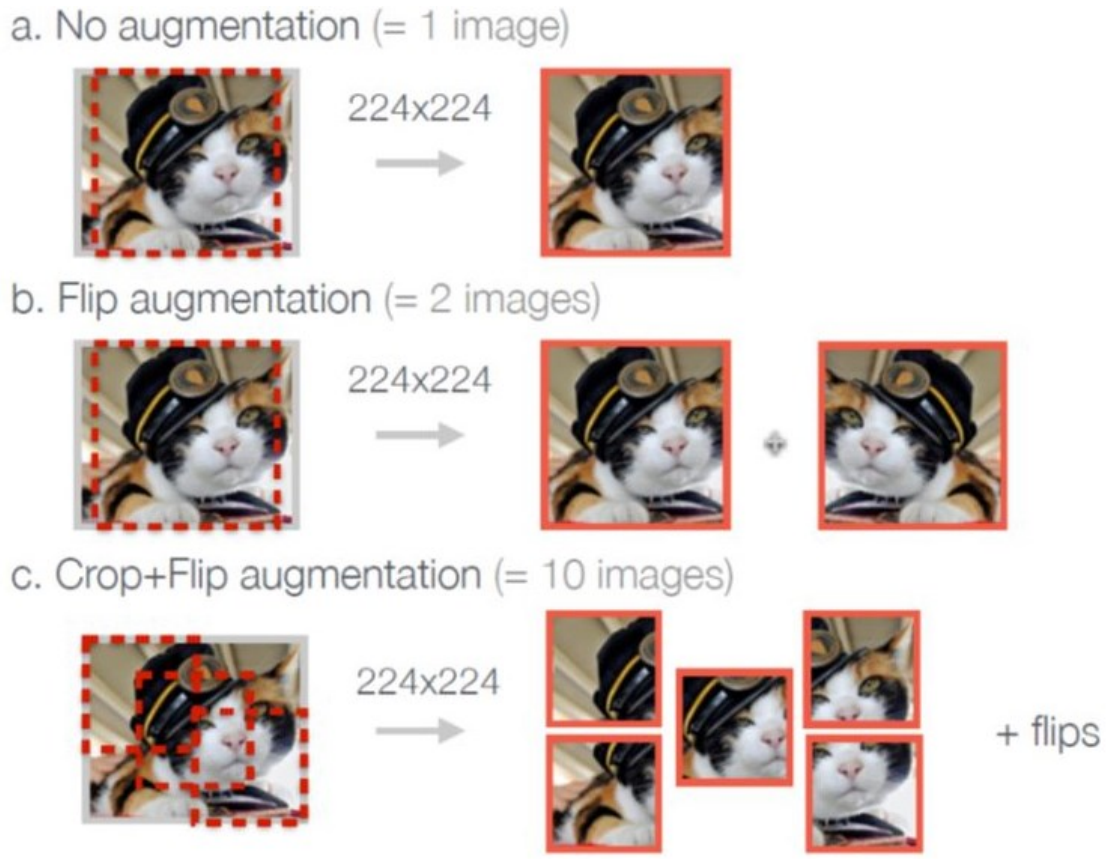
Overlap-tile strategy for arbitrary large images



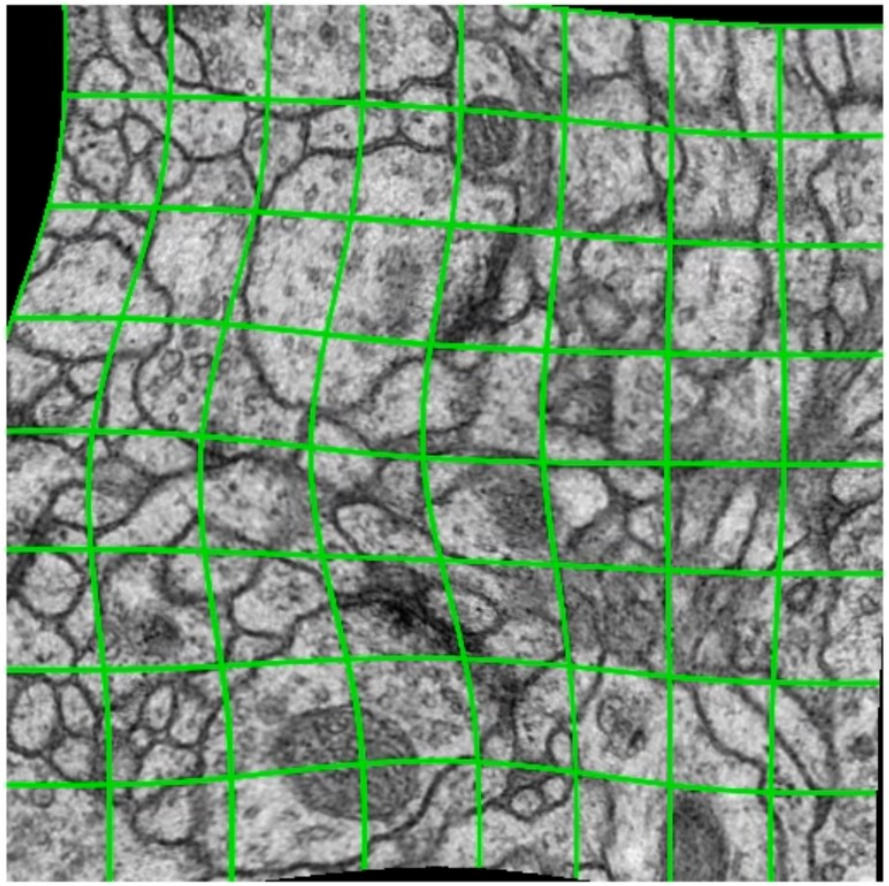
- Segmentation of the yellow area uses input data of the blue area
- Raw data extrapolation by mirroring

Data Augmentation

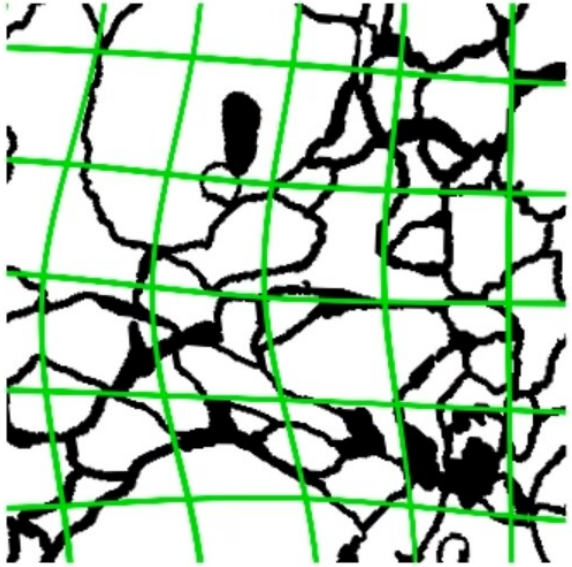
- Data 수가 적을 때 일반적으로 사용
  - Random scale, random crop, changing aspect ratio, flip, rotation, adding noise or distortion 등이 많이 사용됨



Augment Training Data using Deformations

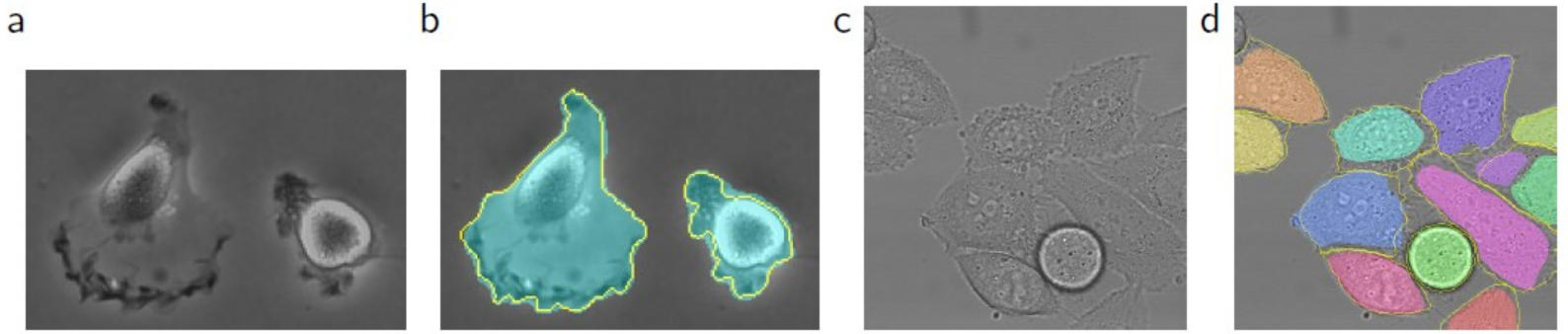


resulting deformed image
(for visualization: no rotation, no shift, no extrapolation)



correspondingly deformed
manual labels

ISBI 2015 Result



Name	PhC-U373	DIC-HeLa
IMCB-SG (2014)	0.2669	0.2935
KTH-SE (2014)	0.7953	0.4607
HOUS-US (2014)	0.5323	-
second-best 2015	0.83	0.46
u-net (2015)	**0.9203**	**0.7756**

Pyramid Scene Parsing Network(PSPNet)

Pyramid Scene Parsing Network

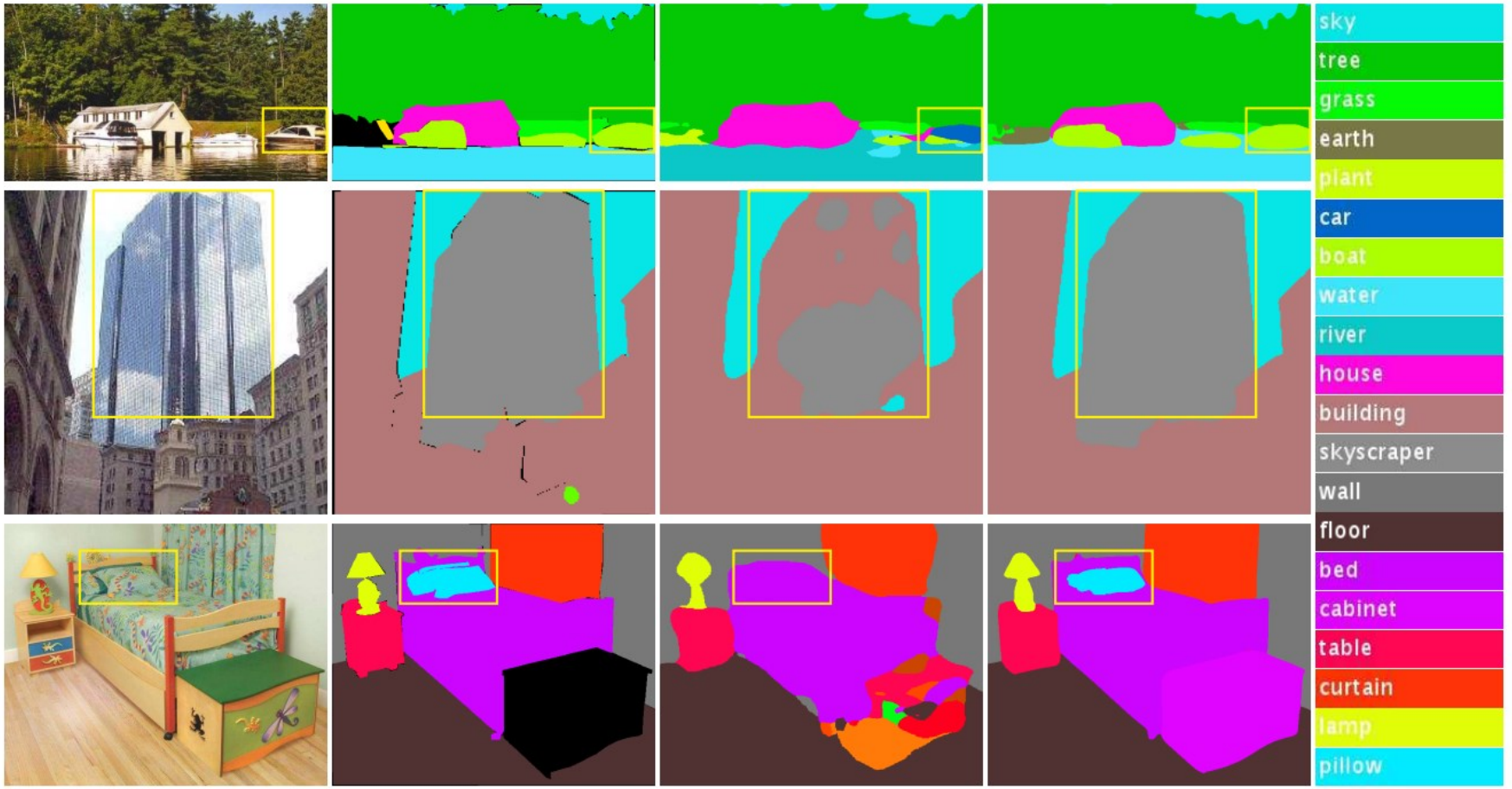
Hengshuang Zhao[1] Jianping Shi[2] Xiaojuan Qi[1] Xiaogang Wang[1] Jiaya Jia[1]
[1]The Chinese University of Hong Kong [2]SenseTime Group Limited
{hszhao, xjqi, leojia}@cse.cuhk.edu.hk, xgwang@ee.cuhk.edu.hk, shijianping@sensetime.com

Abstract

Scene parsing is challenging for unrestricted open vocabulary and diverse scenes. In this paper, we exploit the capability of global context information by different-region-based context aggregation through our pyramid pooling module together with the proposed pyramid scene parsing network (PSPNet). Our global prior representation is effective to produce good quality results on the scene parsing task, while PSPNet provides a superior framework for pixel-level prediction. The proposed approach achieves state-of-the-art performance on various datasets. It came first in ImageNet scene parsing challenge 2016, PASCAL VOC 2012 benchmark and Cityscapes benchmark. A single PSPNet yields the new record of mIoU accuracy 85.4% on PASCAL

Pyramid Scene Parsing Network(PSPNet)

- Deep Network with a suitable global-scene-level prior can much improve the performance of scene parsing



PSPNet Architecture

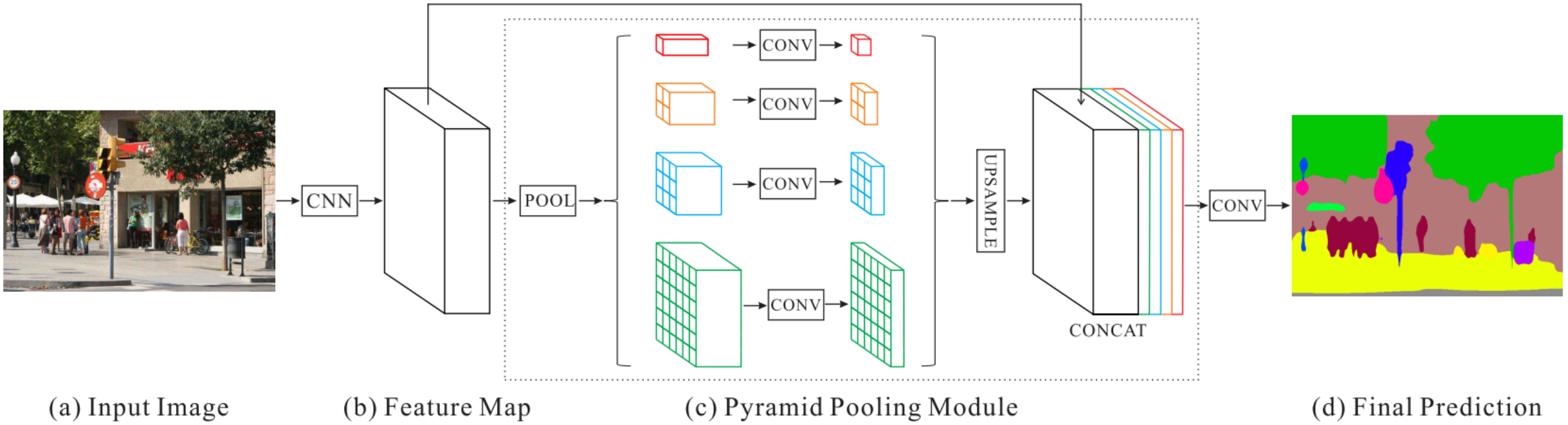
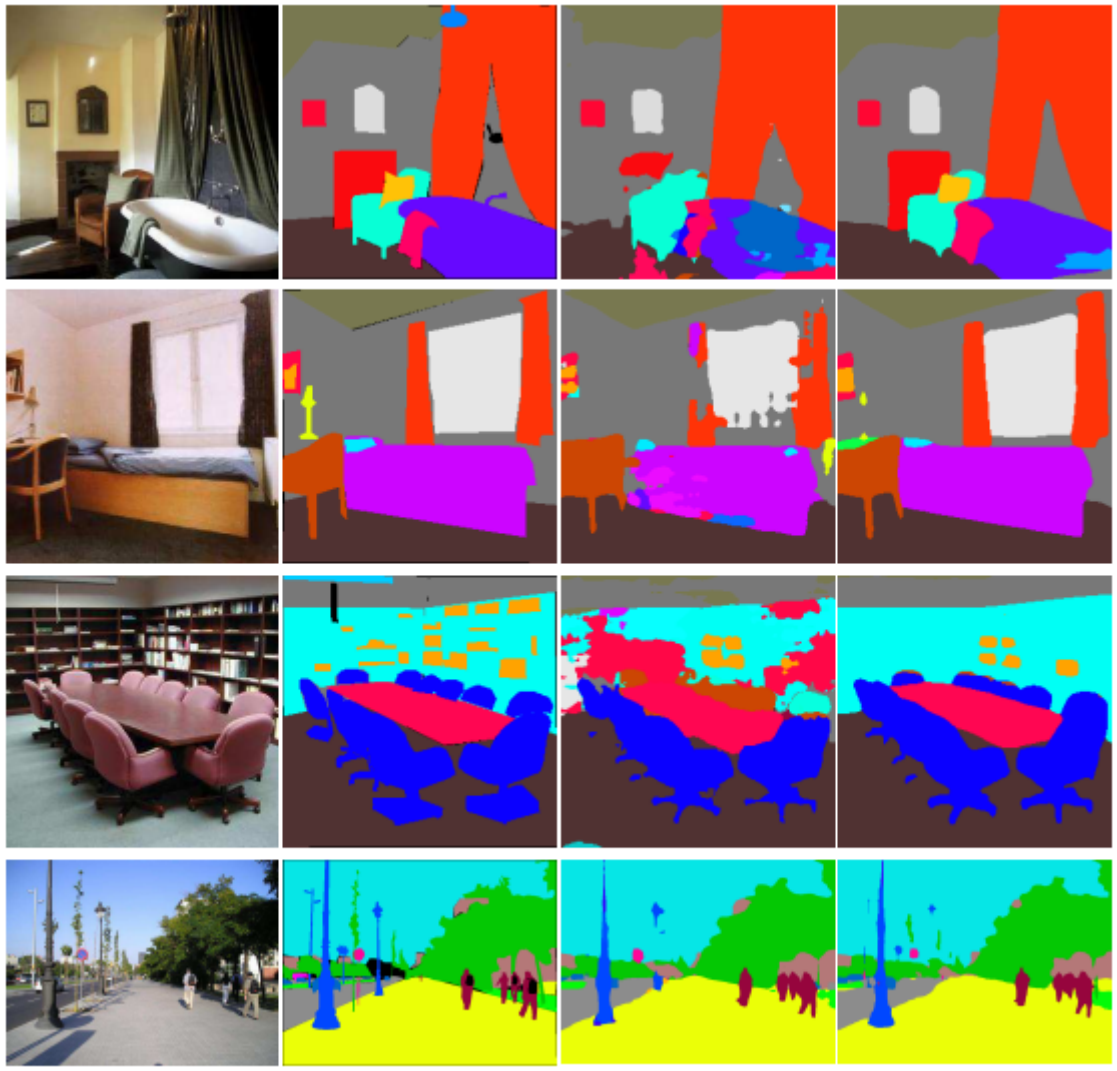


Figure 3. Overview of our proposed PSPNet. Given an input image (a), we first use CNN to get the feature map of the last convolutional layer (b), then a pyramid parsing module is applied to harvest different sub-region representations, followed by upsampling and concatenation layers to form the final feature representation, which carries both local and global context information in (c). Finally, the representation is fed into a convolution layer to get the final per-pixel prediction (d).

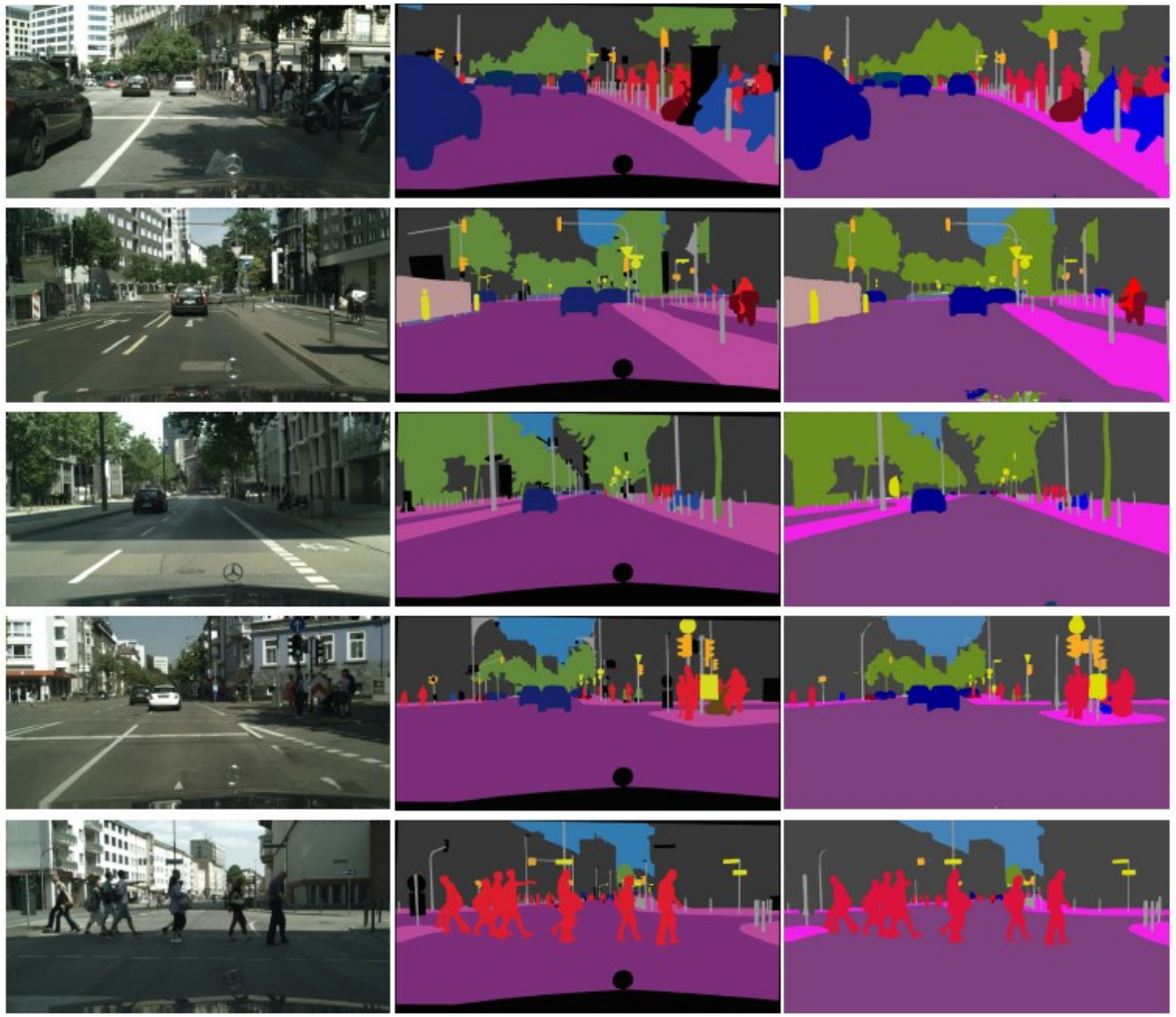
Results



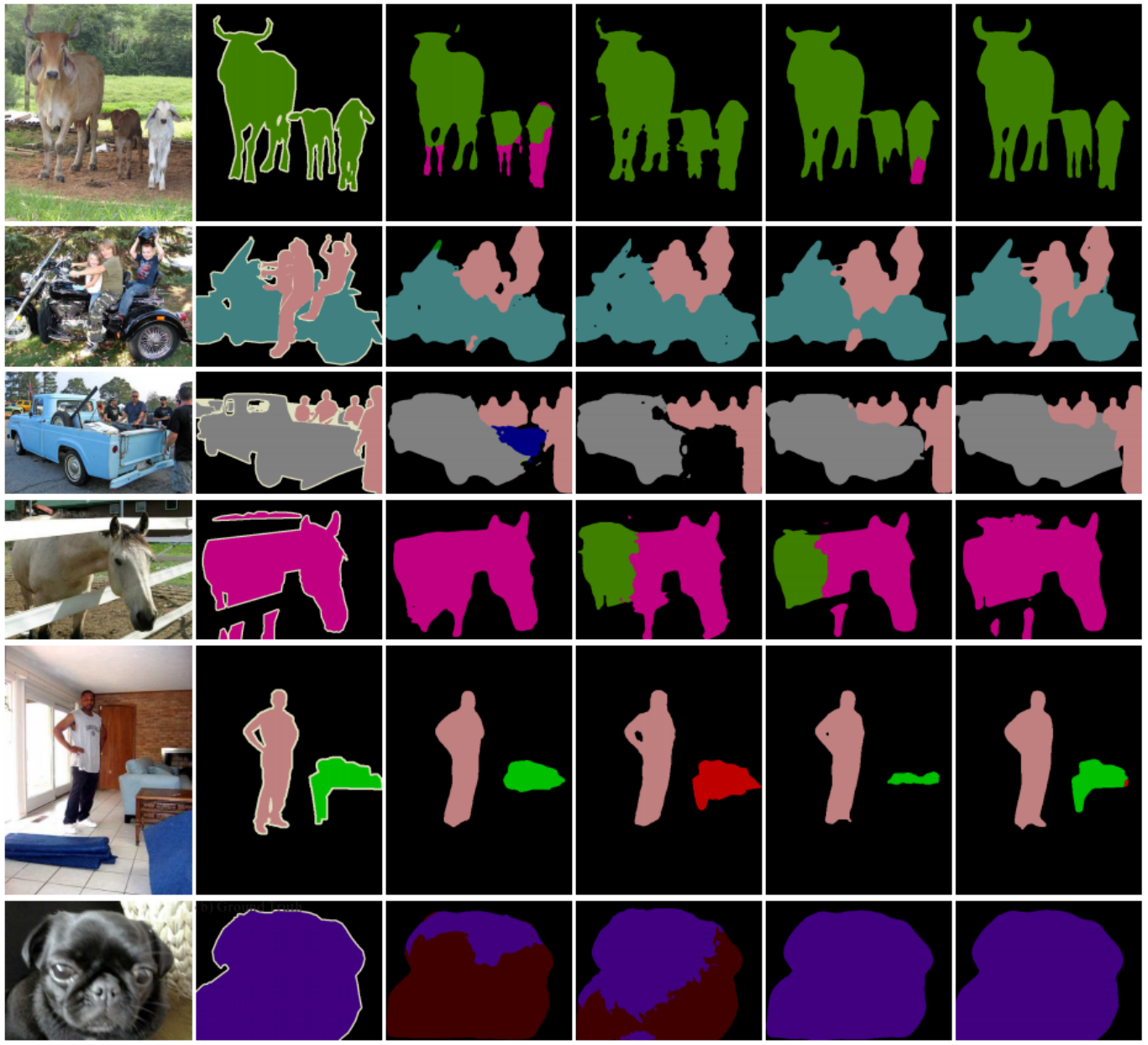
(a) Image (b) Ground Truth (c) Baseline (d) PSPNet

(a) Image (b) Ground Truth (c) Baseline (d) PSPNet

Results



(a) Image (b) Ground Truth (c) PSPNet



(a) Image (b) Ground Truth (c) FCN (d) DPN (e) DeepLab (f) PSPNet

Mask R-CNN

Mask R-CNN

Kaiming He Georgia Gkioxari Piotr Dollár Ross Girshick

Facebook AI Research (FAIR)

Abstract

We present a conceptually simple, flexible, and general framework for object instance segmentation. Our approach efficiently detects objects in an image while simultaneously generating a high-quality segmentation mask for each instance. The method, called Mask R-CNN, extends Faster R-CNN by adding a branch for predicting an object mask in parallel *with the existing branch for bounding box recognition. Mask R-CNN is simple to train and adds only a small overhead to Faster R-CNN, running at 5 fps. Moreover, Mask R-CNN is easy to generalize to other tasks,* e.g., *allowing us to estimate human poses in the same framework. We show top results in all three tracks of the COCO suite of challenges, including instance segmentation, bounding*

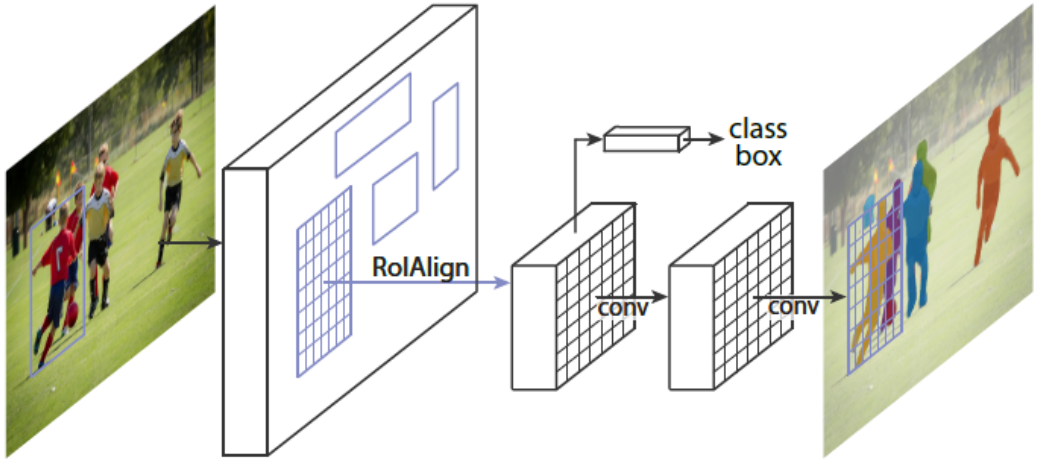


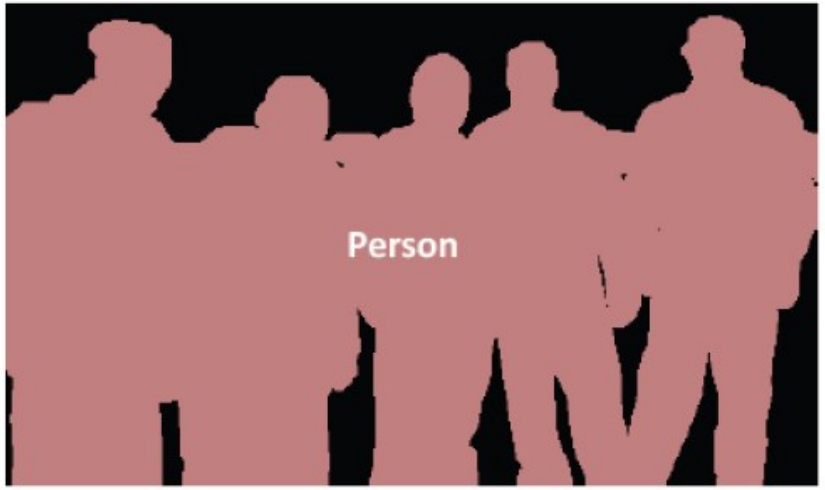
Figure 1. The **Mask R-CNN** framework for instance segmentation.

segmentation, where the goal is to classify each pixel into a fixed set of categories without differentiating object instances.[1] Given this, one might expect a complex method

Instance Segmentation

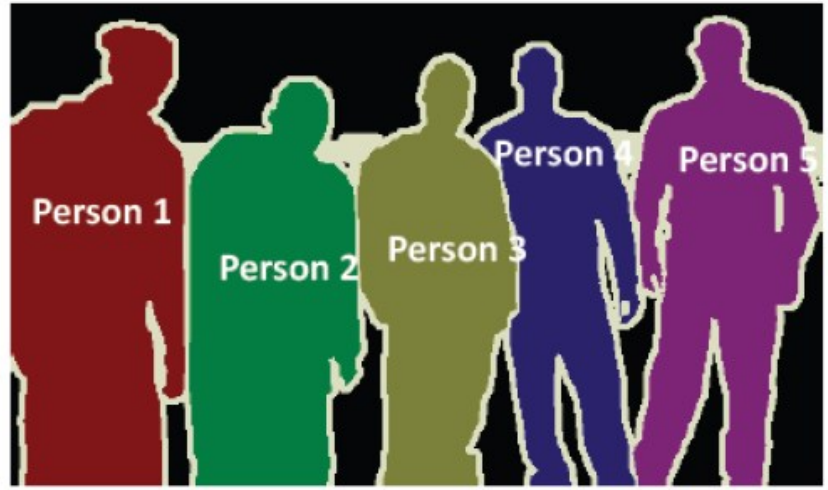
Object Detection

✓



Semantic Segmentation

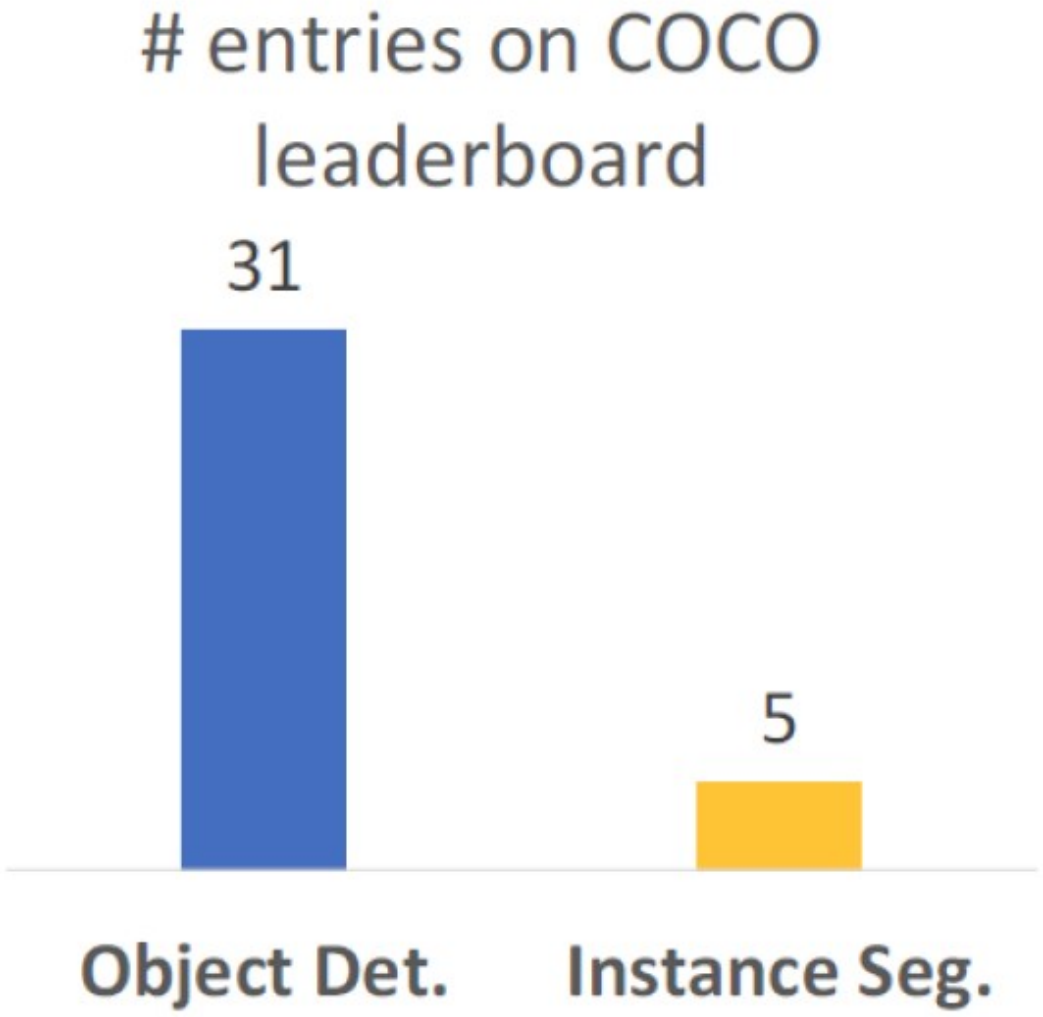
✓

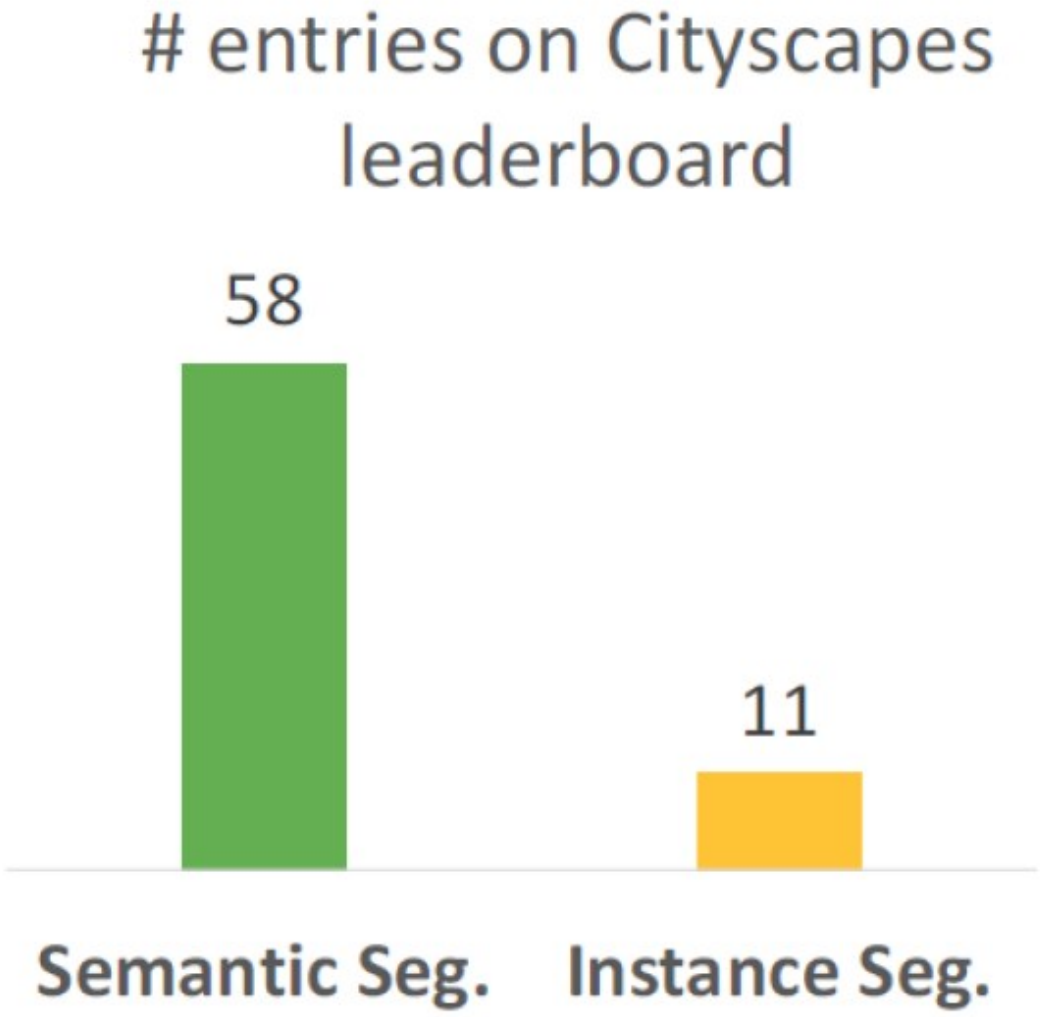


Instance Segmentation

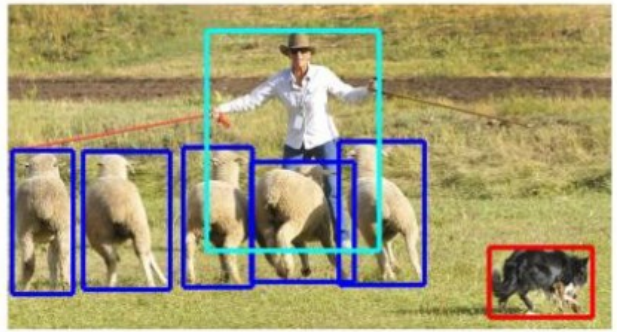
?

Instance Segmentation





Instance Segmentation



BBox
Classification

+

Segmentation
Classification

=

Segmentation
in BBox
Classification

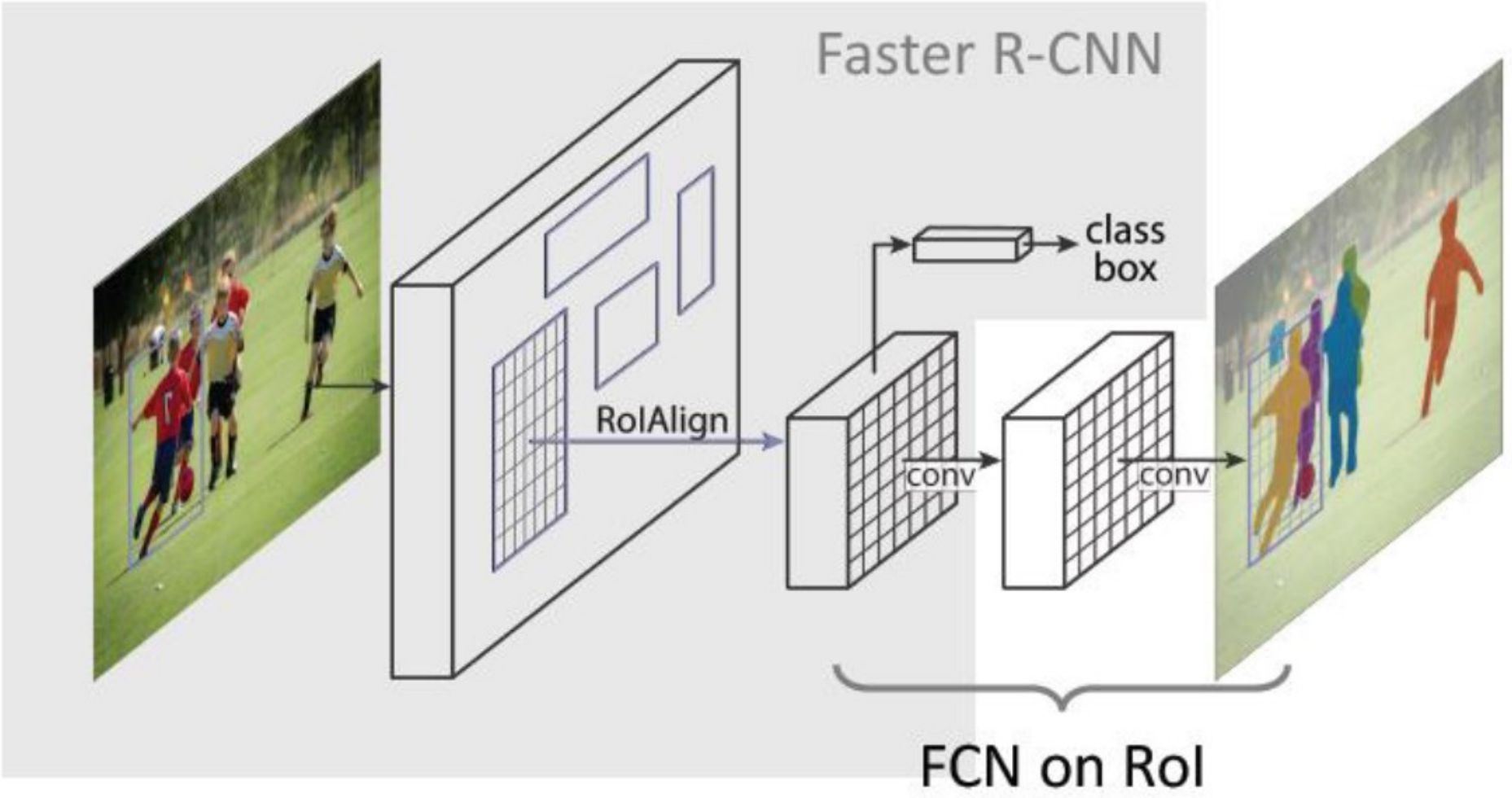
Can Separate
Cannot Segment

Cannot Separate
Can Segment

Faster R-CNN + FCN = FCN
on BBOX !

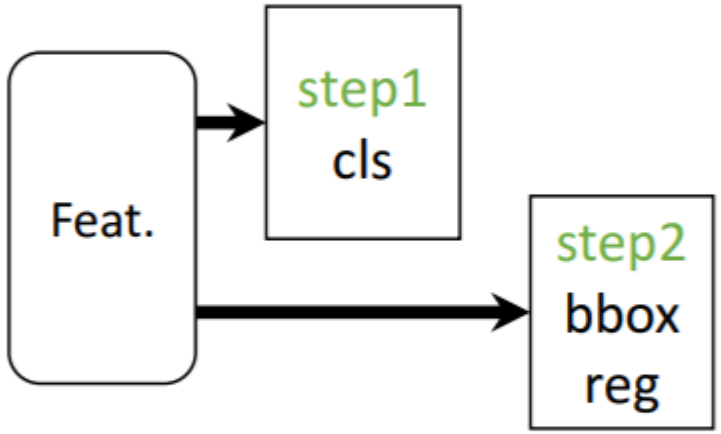
Mask R-CNN

- Mask R-CNN = **Faster R-CNN** with **FCN** on RoIs

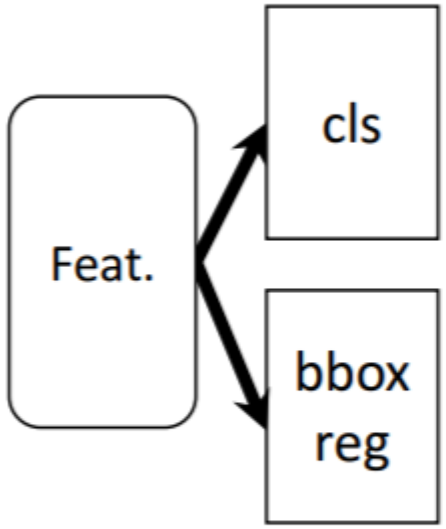


Parallel Heads

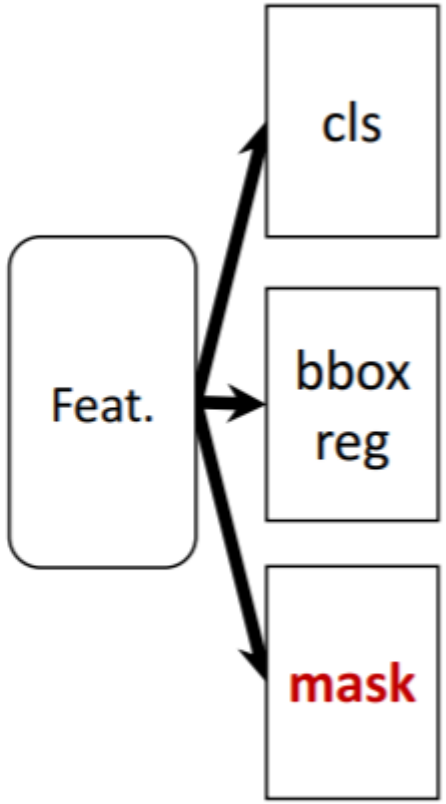
- Easy, fast to implement and train



(slow) R-CNN

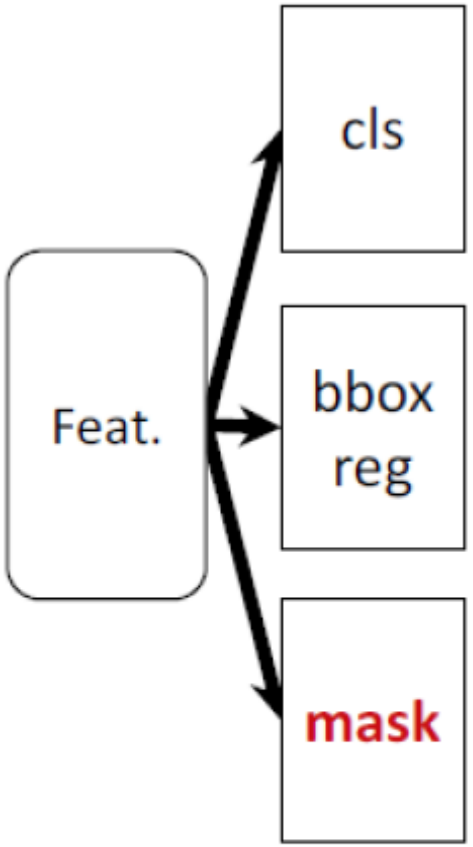


Fast/er R-CNN



Mask R-CNN

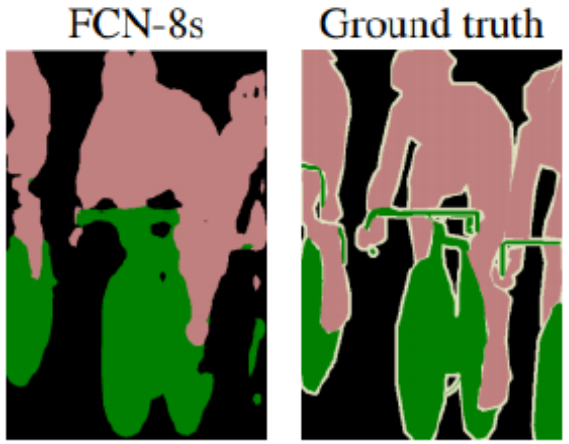
Parallel Heads



Mask R-CNN

Faster R-CNN

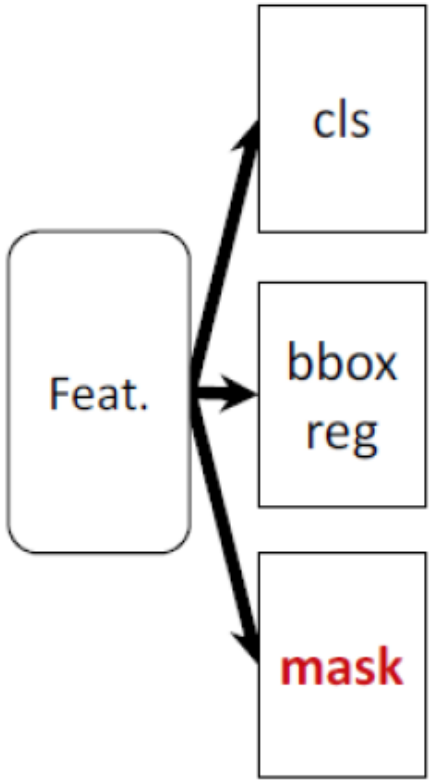
- Classification
- Instance Level RoI



FCN

- Pixel-level Classification
- Per Pixel Softmax (Multinomial)
- Multi Instance

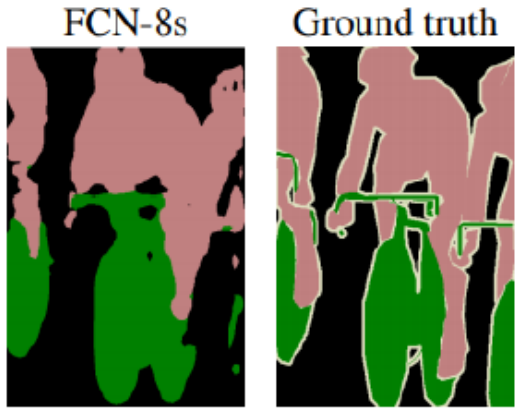
Parallel Heads



Mask R-CNN

Faster R-CNN

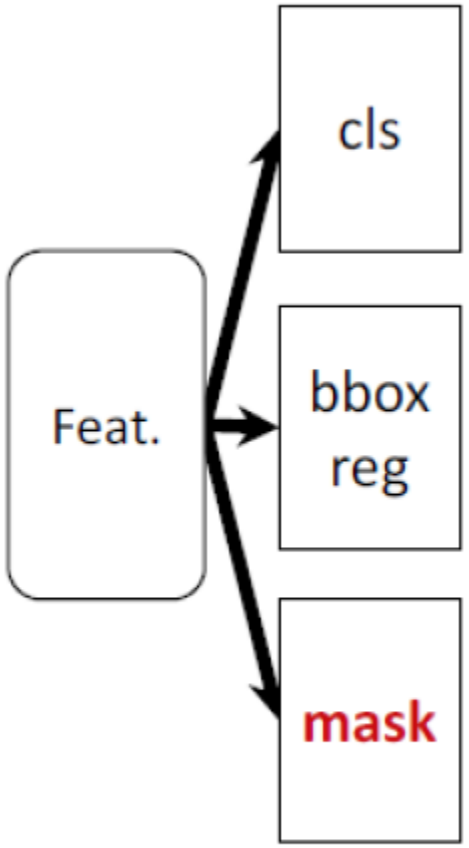
- Classification
- Instance Level RoI



FCN

- ~~Pixel-level Classification~~
- Per Pixel ~~Softmax~~ → Sigmoid (Binary)
- ~~Multi-Instance~~

Parallel Heads



Mask R-CNN

DB

BBox + Class + Mask

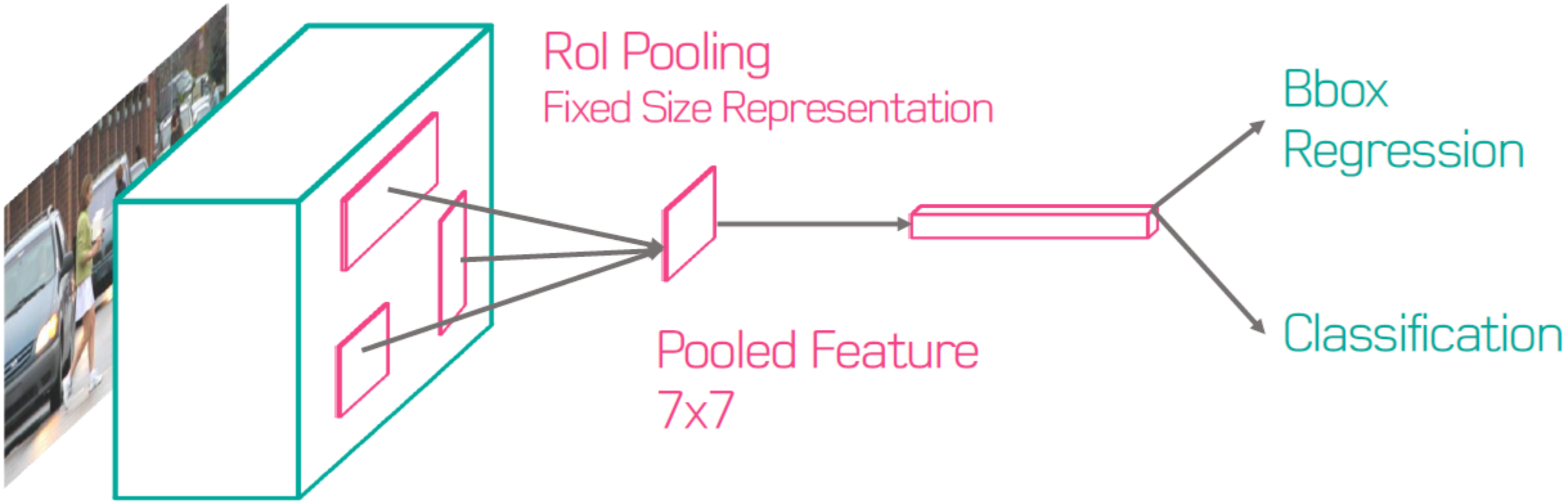
$$L = L_{cls} + L_{box} + L_{mask}$$

L_{cls}: Softmax Cross Entropy

L_{box}: Regression

L_{mask}: Binary Cross Entropy

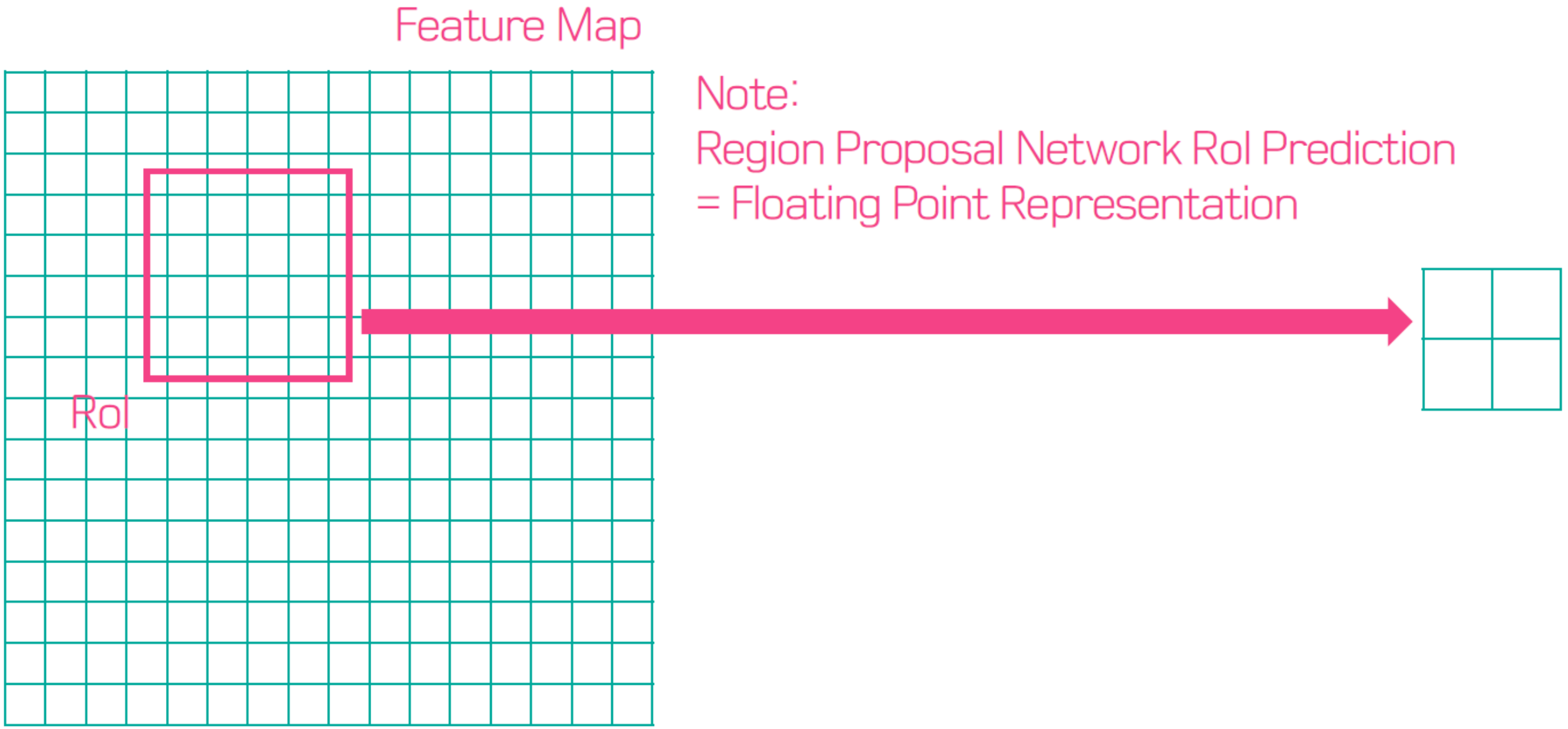
Recap : Faster R-CNN



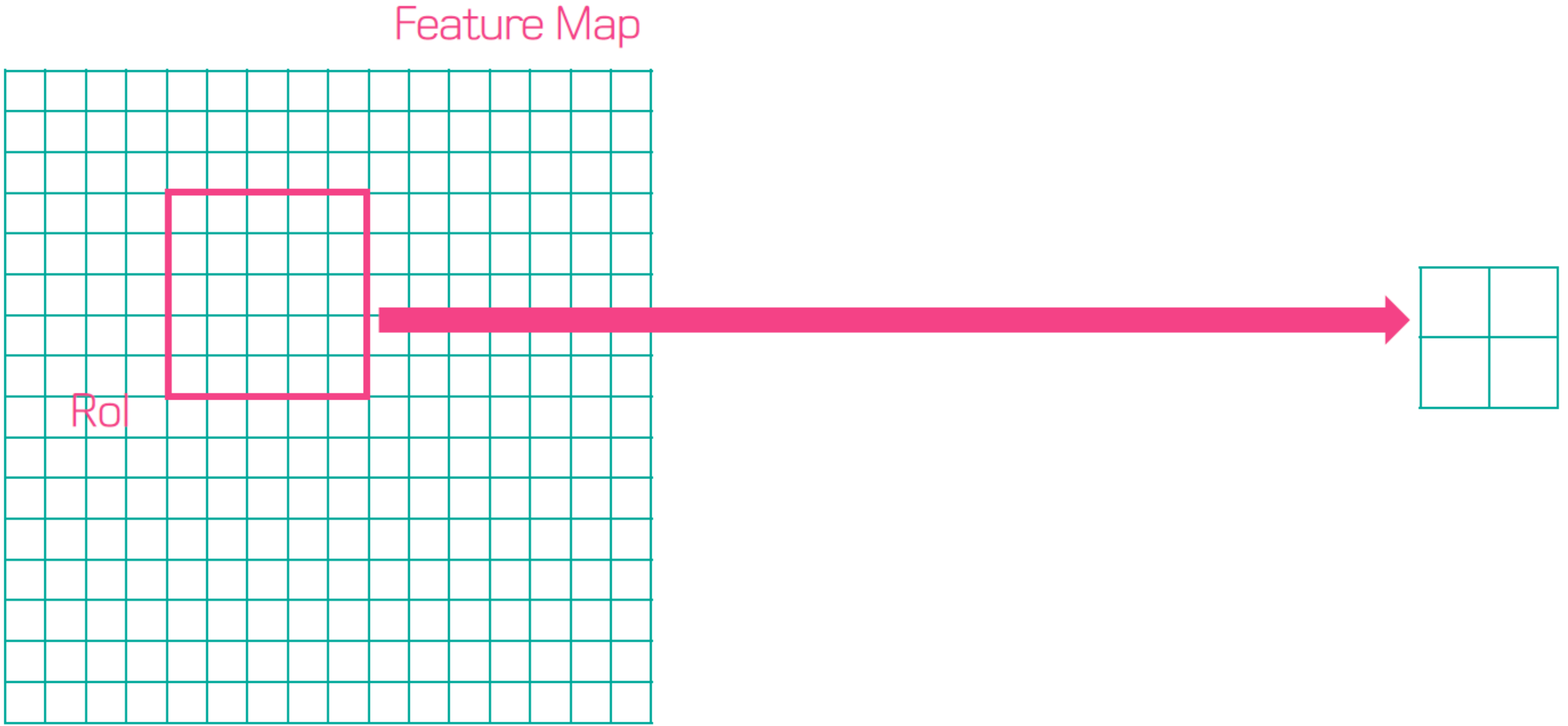
Problems of Faster R-CNN

- RoI Projection and Pooling has some quantization operations
- These quantizations introduce misalignments between the RoI and the extracted features
- While this may not impact classification, it can make a negative effect on predicting bbox

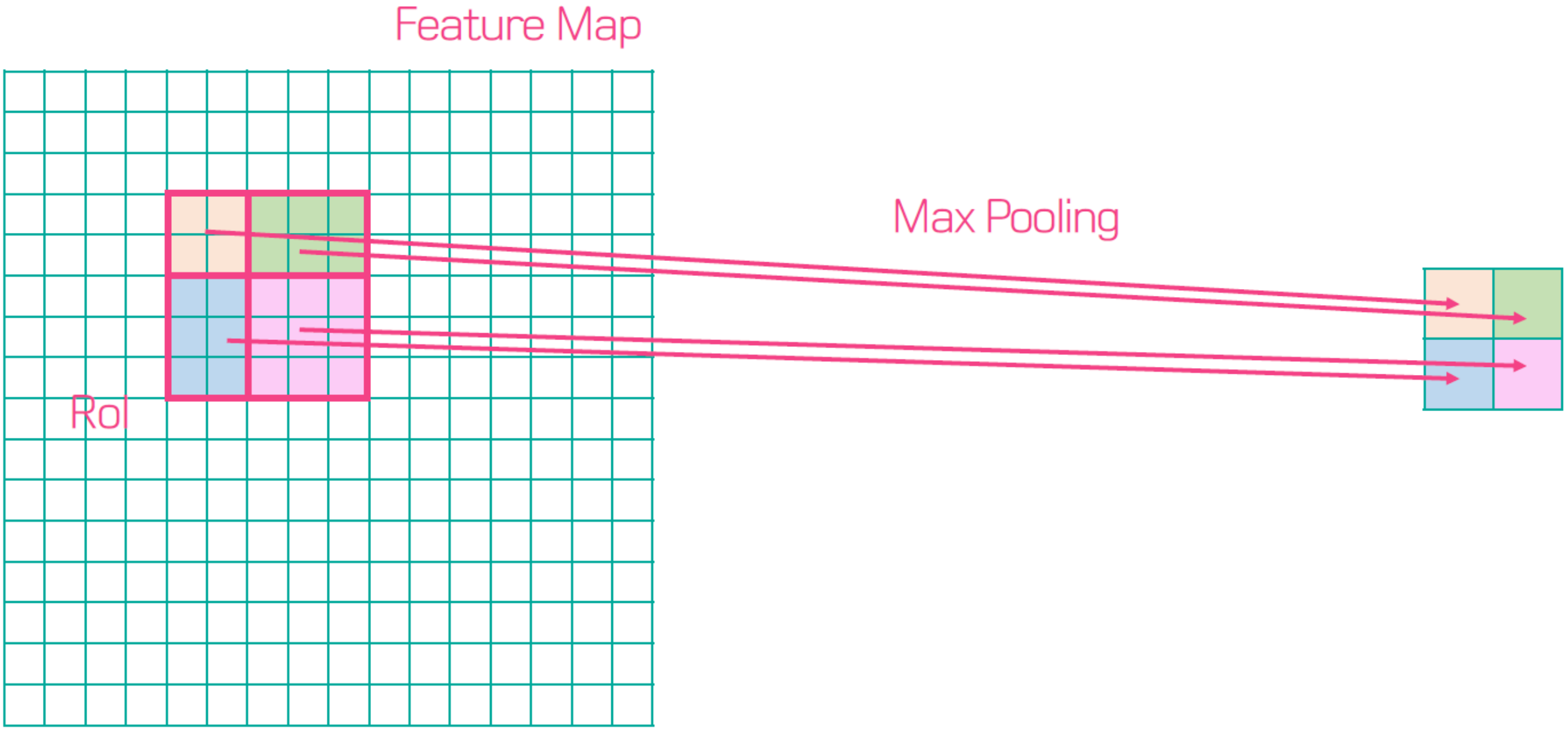
RoI Pooling


Feature Map
RoI
Note:
Region Proposal Network RoI Prediction
= Floating Point Representation

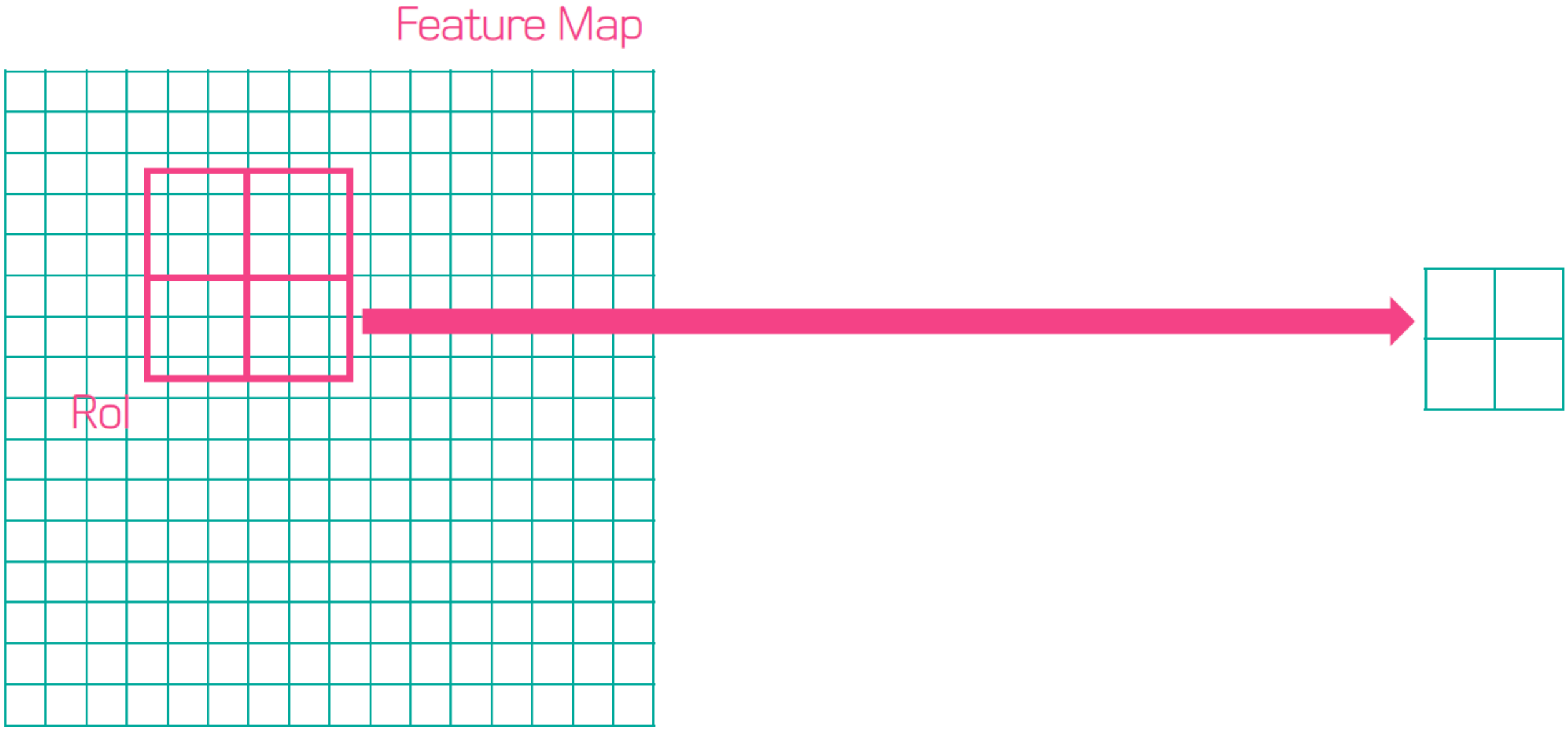
RoI Pooling



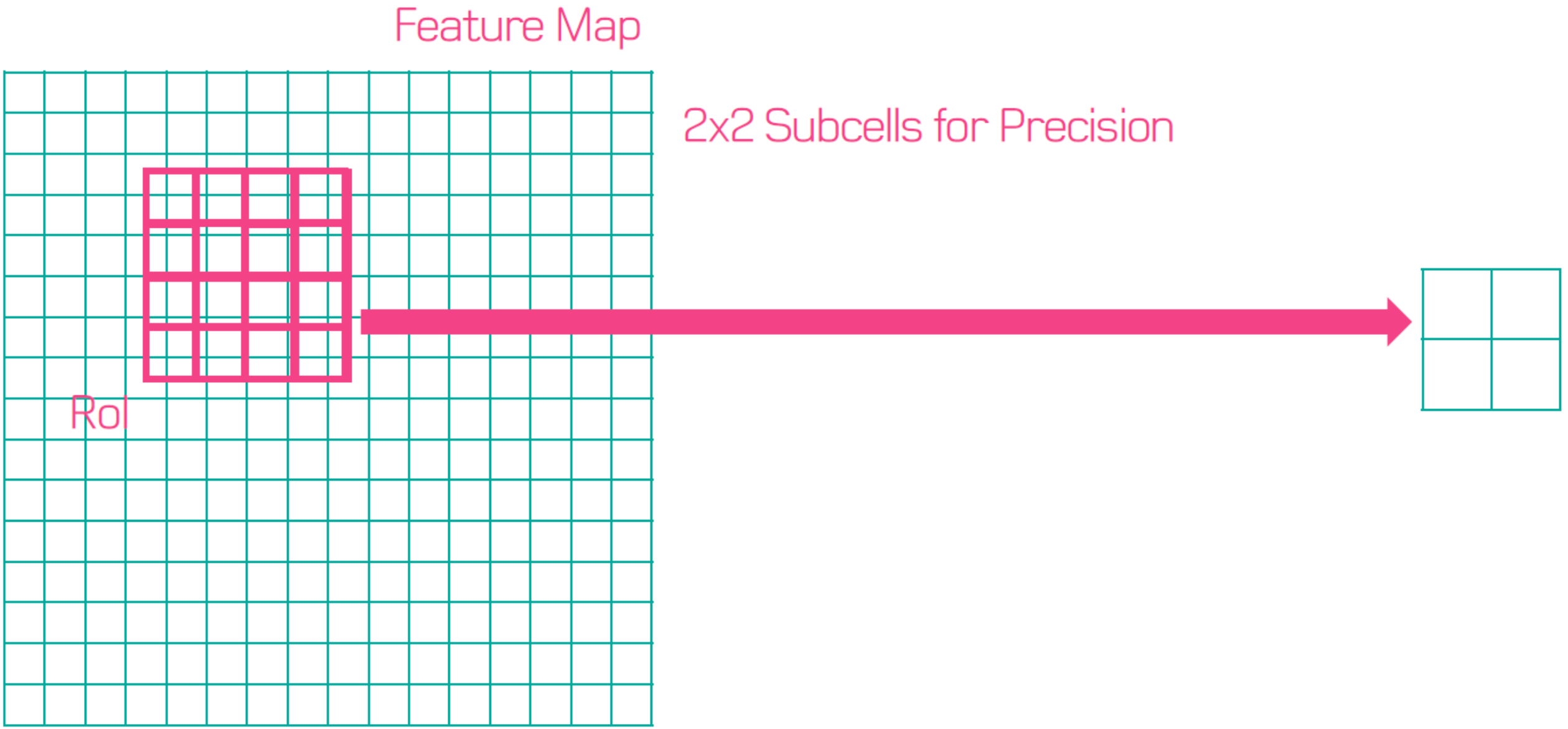
RoI Pooling



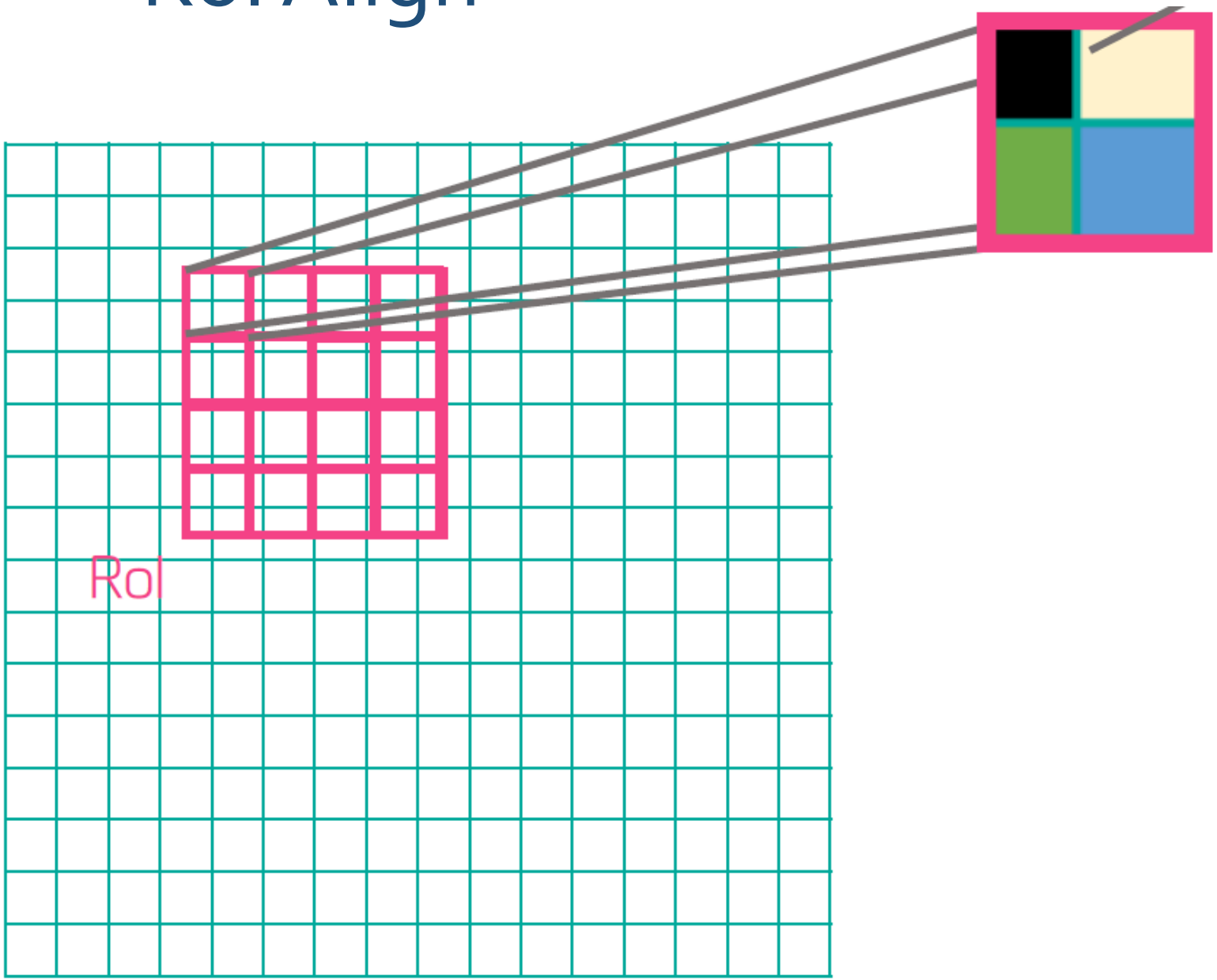
RoI Align



RoI Align



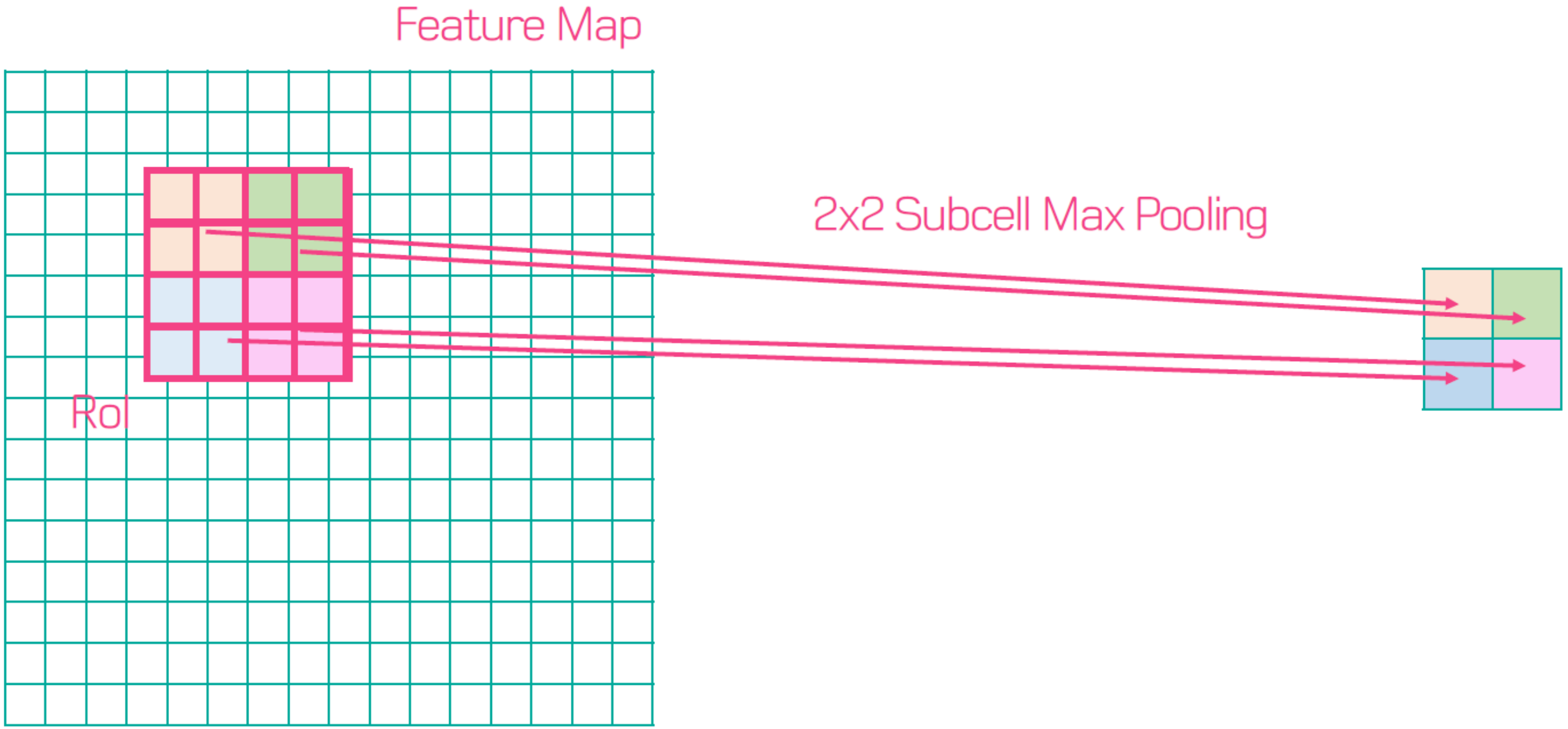
RoI Align



= 0.15 + 0.25

+ 0.25 + 0.35

RoI Align



Result