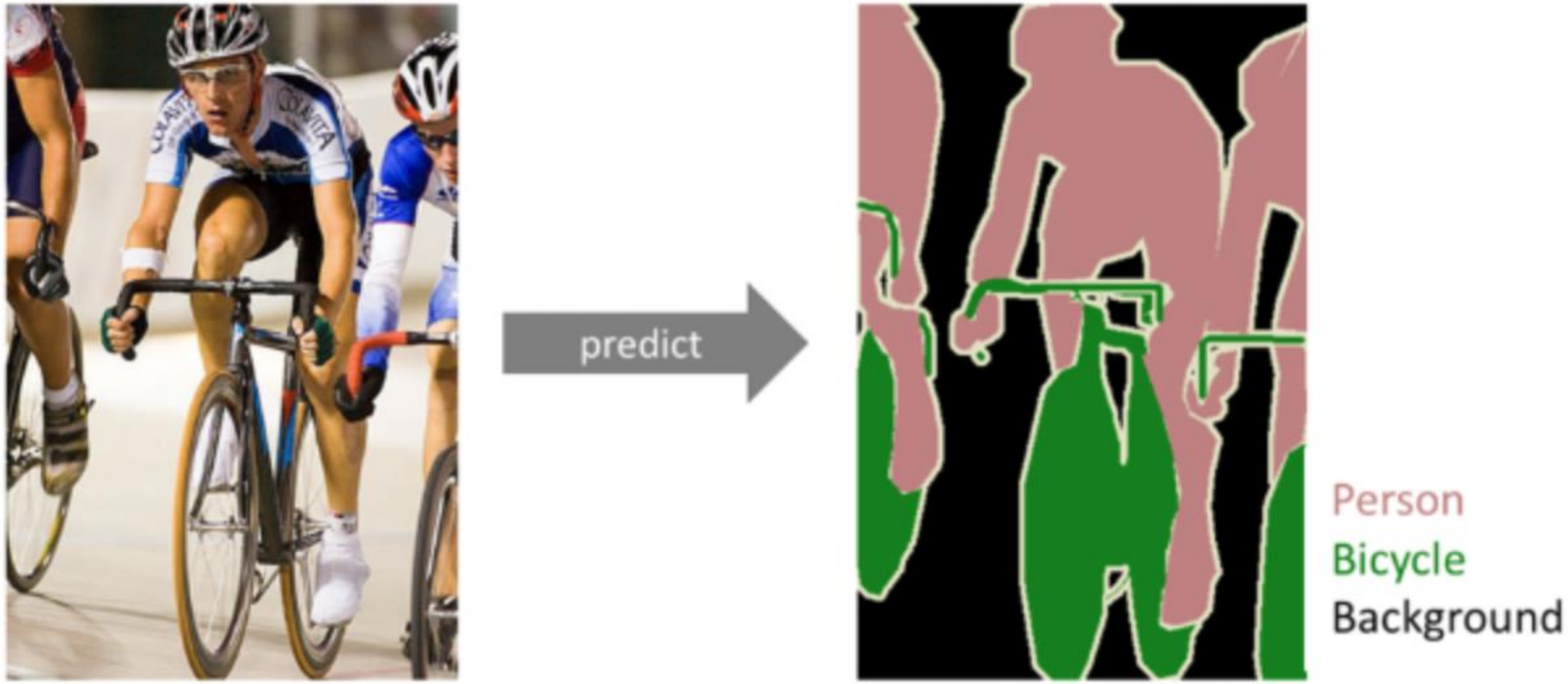
Semantic Segmentation

POSTECH MIP Lab.

TA: Joonhyuk Park, Seunghun Baek, Soojin Hwang

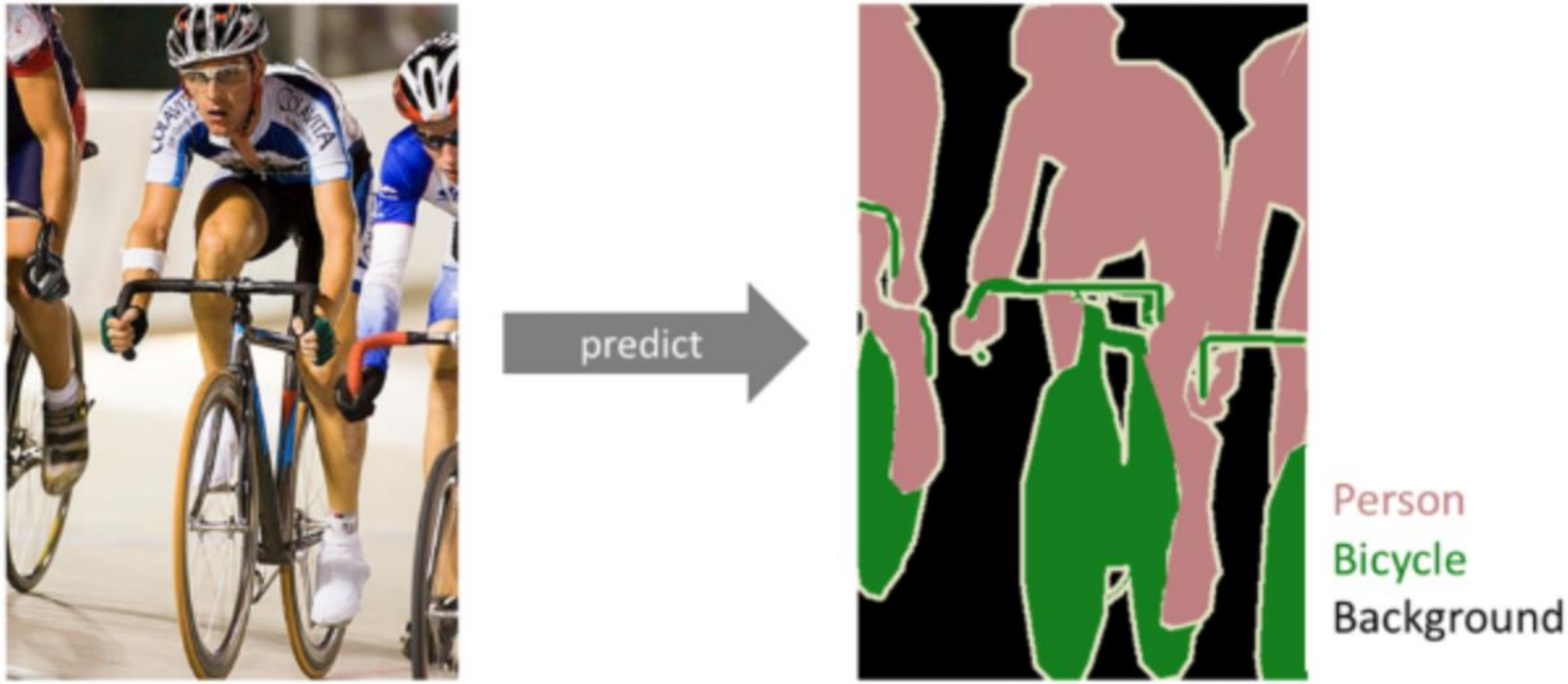
Overview

- Semantic Segmentation is a task to classify **segments with same semantic meanings/information**.

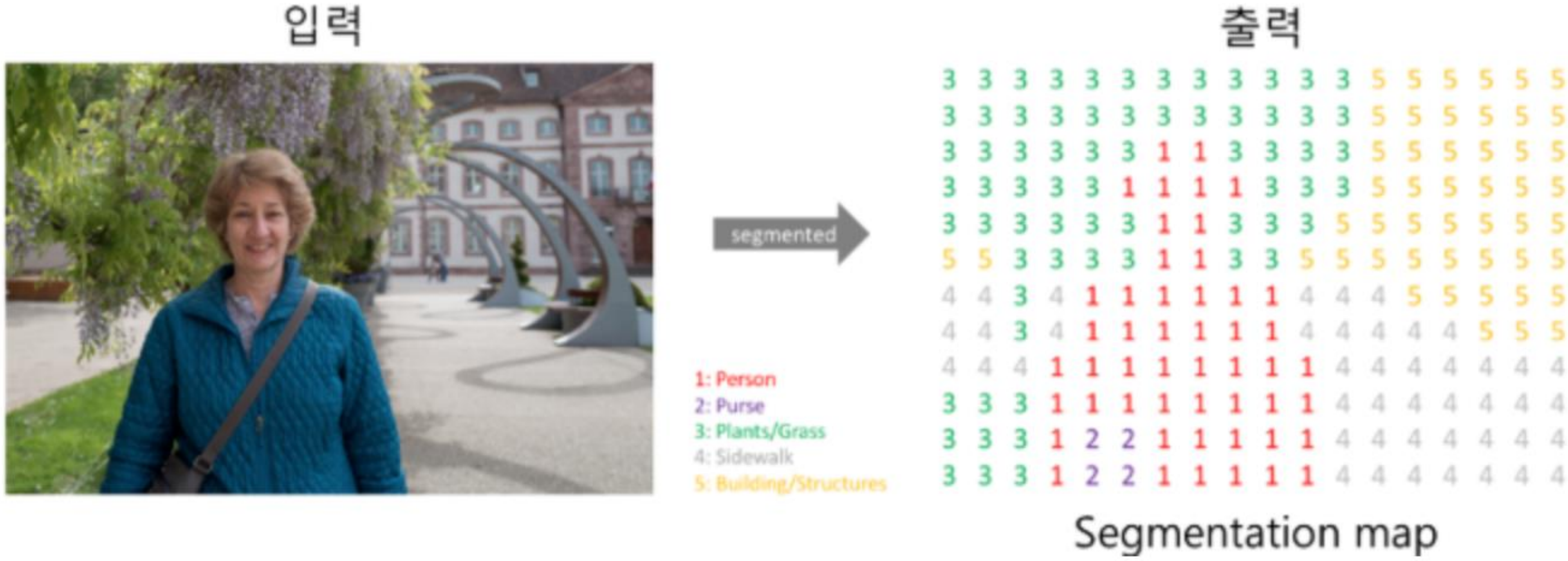


Overview

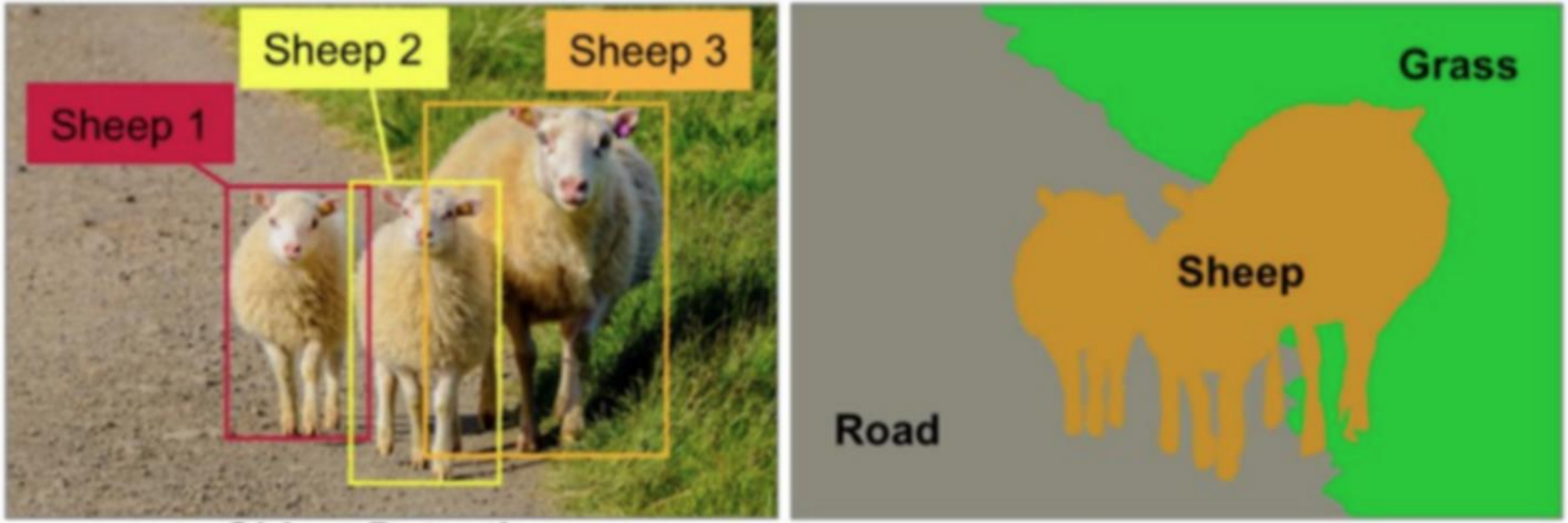
- Semantic Segmentation is a task to **classify each pixel** in the object



- Semantic Segmentation ➔ Pixel-level Classification



- Semantic Segmentation vs. Object Detection

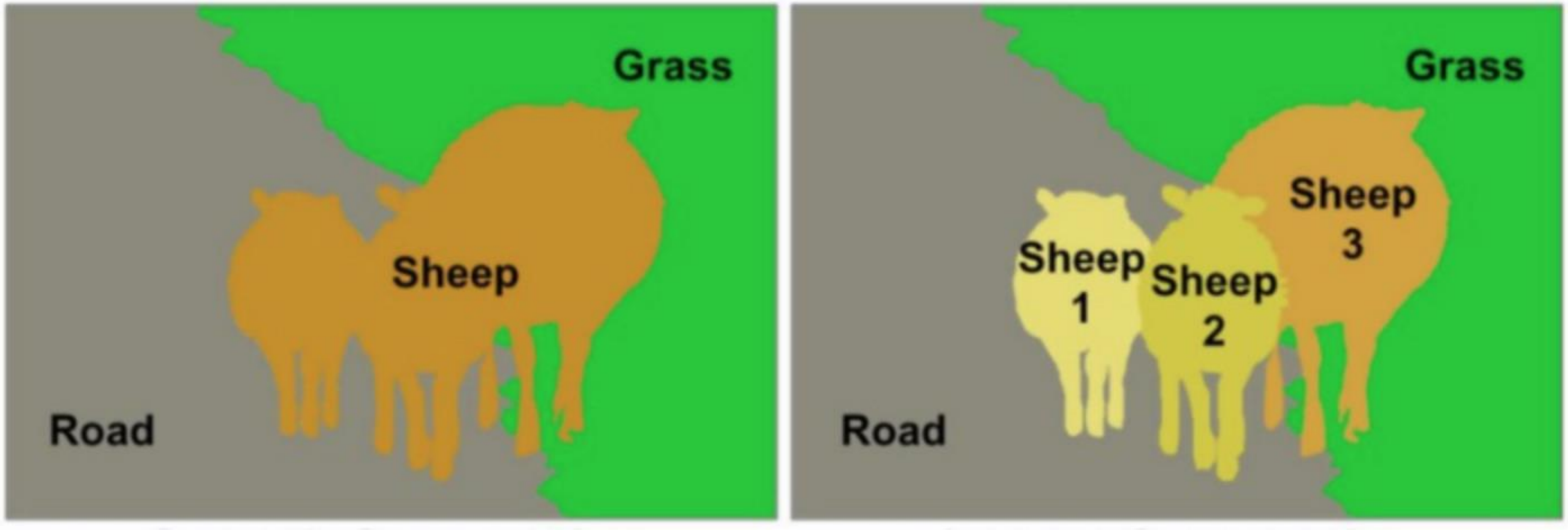


Object Detection

Semantic Segmentation

Overview

- Semantic Segmentation vs. Instance Segmentation



Overview

- Classification -> Detection -> Semantic segmentation

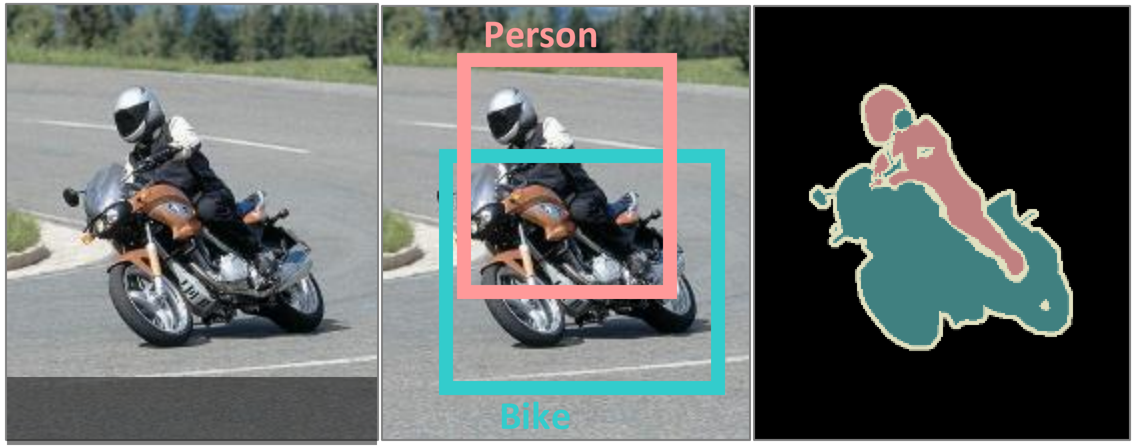


Image Classification | Object Detection | Semantic Segmentation

Higher supervision
Expensive labeling

- Semantic segmentation based on deep learning
  - FCN, DeepLab, DeconvNet, Pyramid Scene Parsing Network

Image vs Semantic

- Classification - determine label of image
  - Find function: Image -> number of label
  - e.g. 32x32x3 -> 10x1

- Semantic segmentation - determine label of each pixel
  - Find function: Image -> number of label x Image width x Image height
  - e.g. 32x32x3 -> 10x32x32, harder :<
  - But maybe not 32x32 times harder problem because locality :>

- What is the difference of two task?

Image vs Semantic

- Image classification

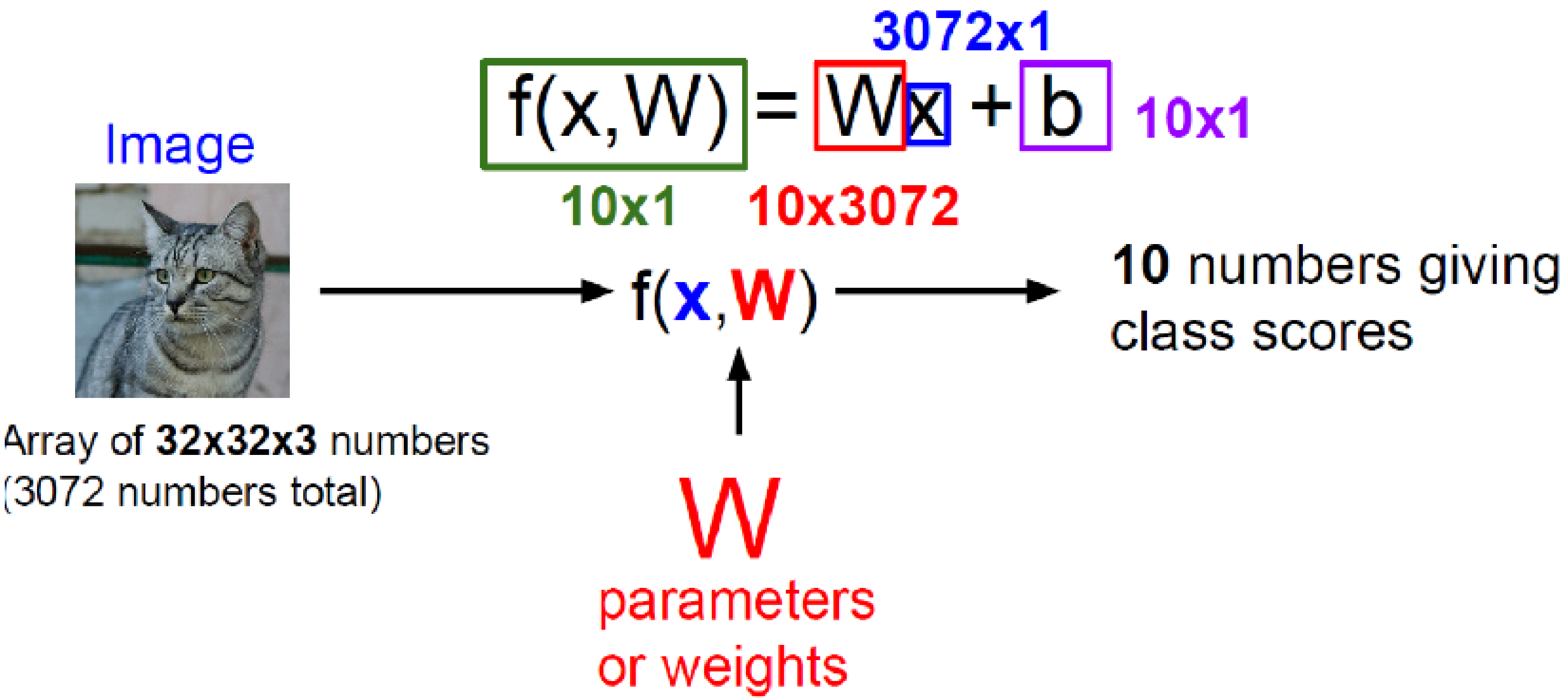


Image vs Semantic

- Semantic segmentation

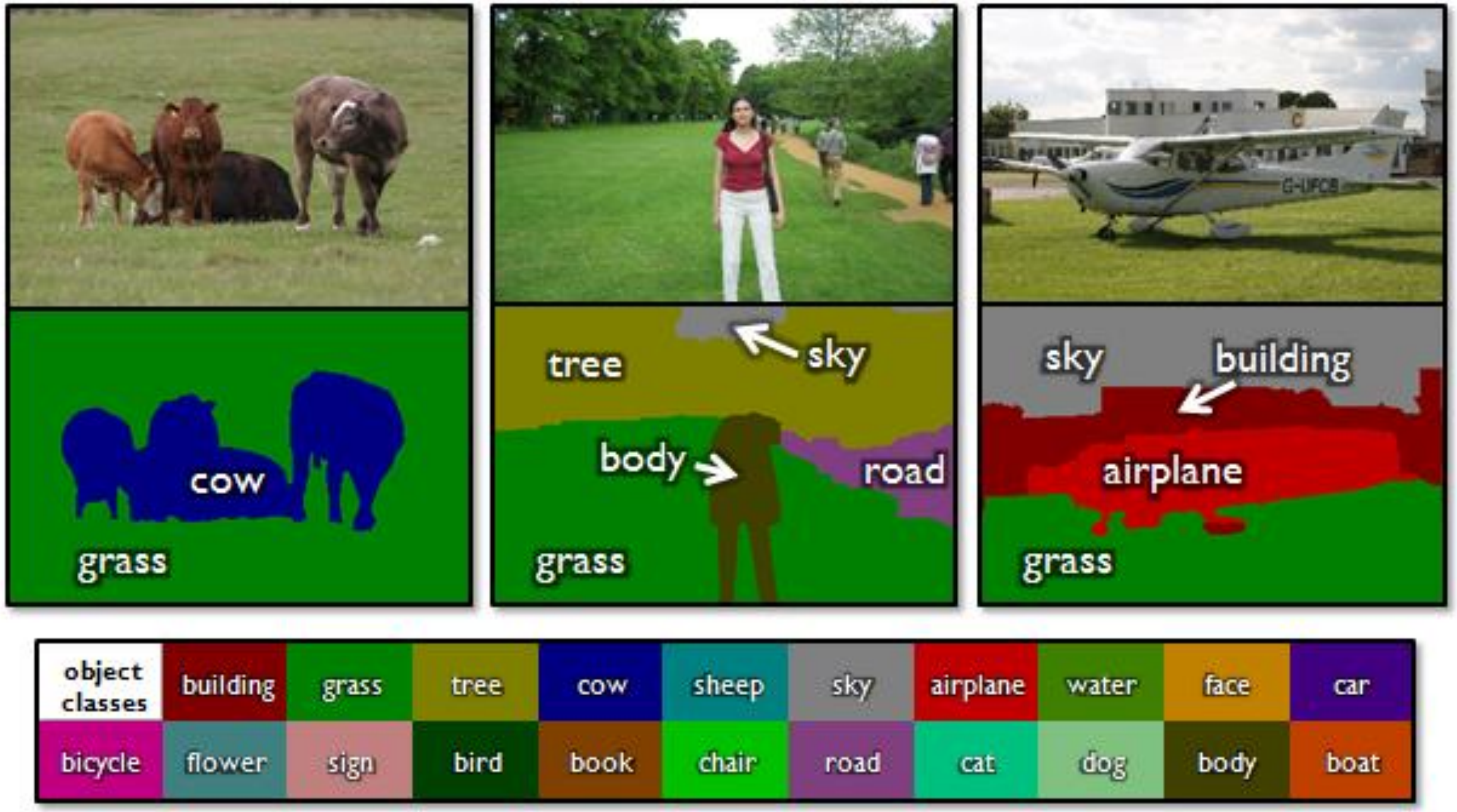
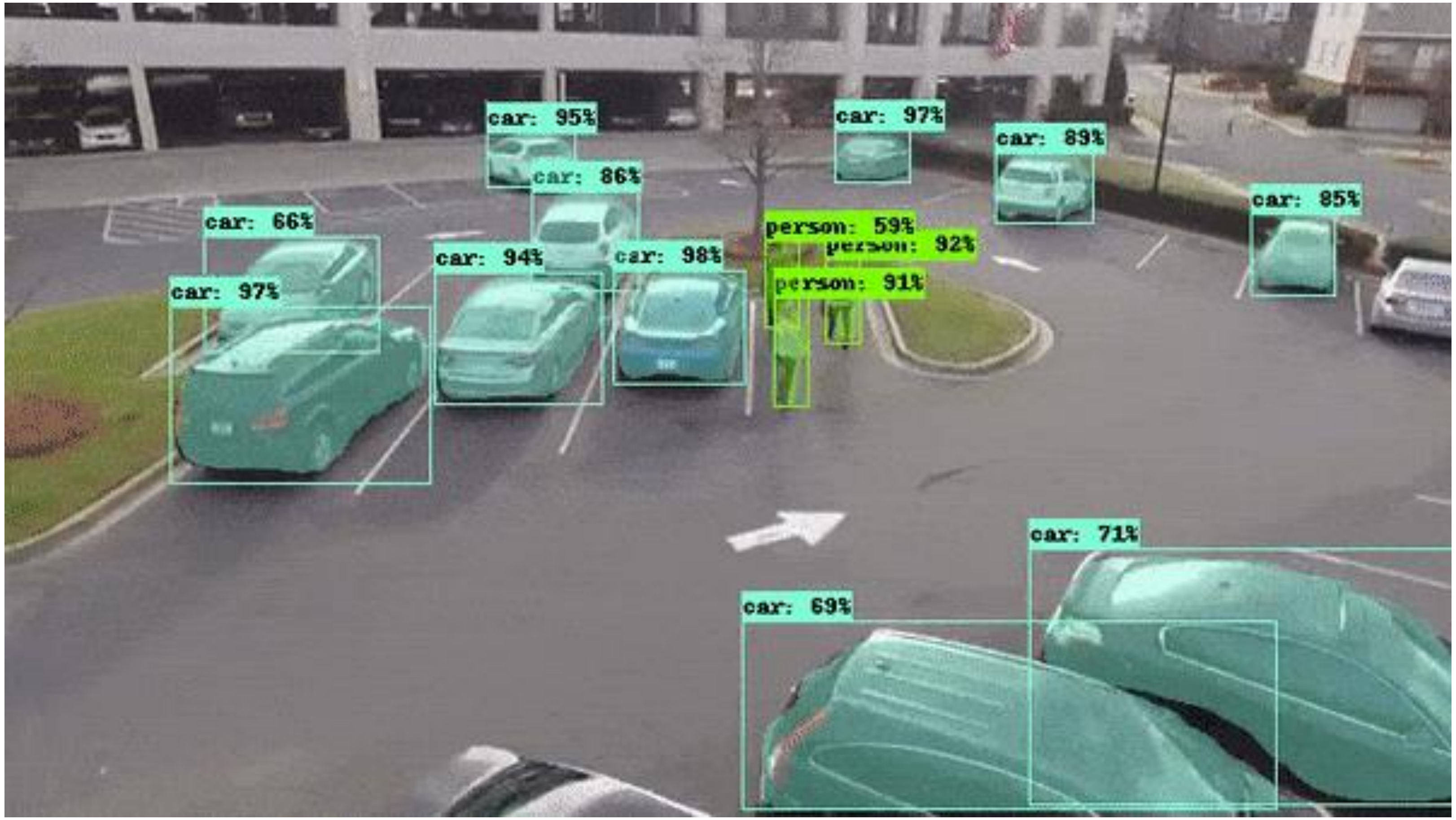


Image vs Semantic

- Semantic segmentation

- Instance Segmentation (Advanced)



Semantic segmentation based on deep learning

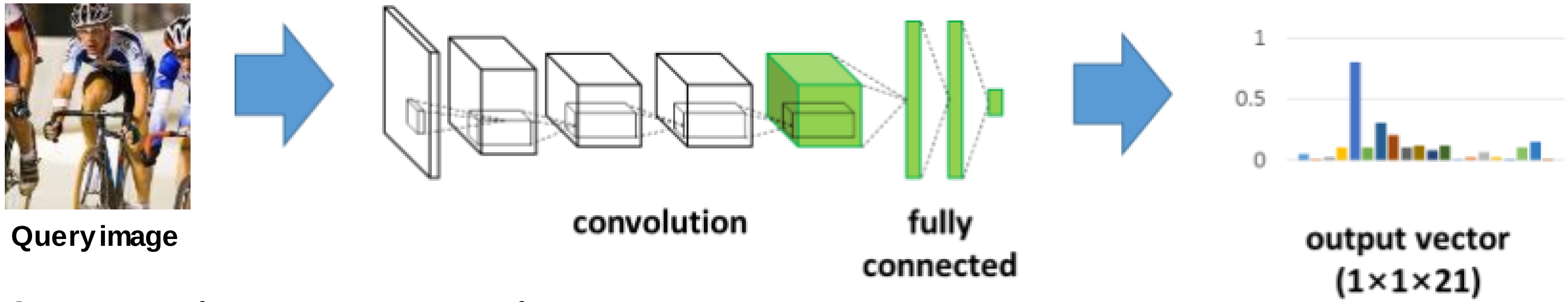
- CVPR
- NIPs → NeurIPs
- ICCV
- SIGGRAPH
- ICLR
- …

Fully Convolutional Network

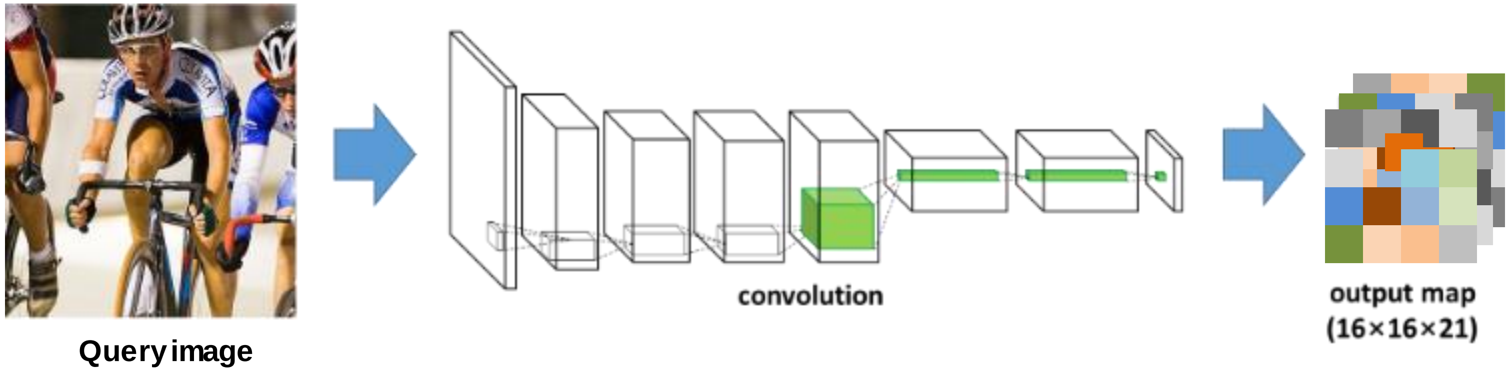
Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

- Image classification



- Semantic segmentation
  - Given an input image, obtain pixel-wise segmentation mask using a deep Convolutional Neural Network (CNN)



Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

- 기존 classification model은 분류를 위해서 마지막에 항상 FC layer를 붙인다.

- FC layer는 segmentation에 적합하지 않음
  고정된 사이즈의 image만 받을 수 있음.
  FC layer를 거치고 나면 2차원 위치 정보가 사라진다.
  pixel-wise classification을 하는 segmentation task에 치명적인 문제

Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

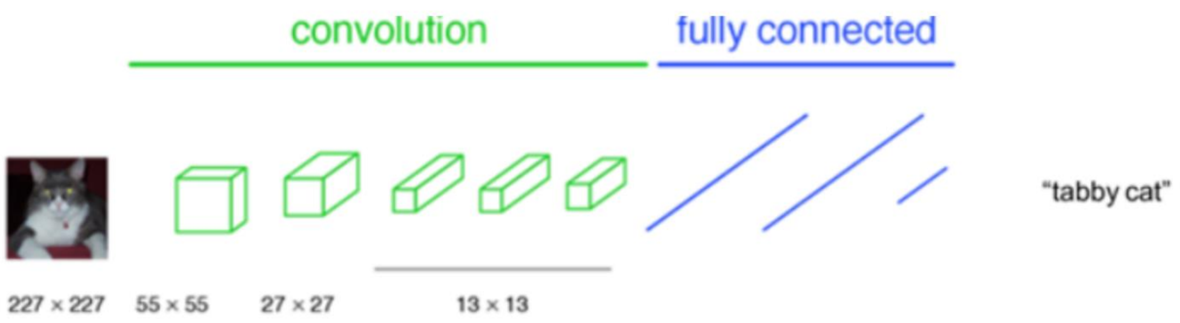
Fully Convolutional Network

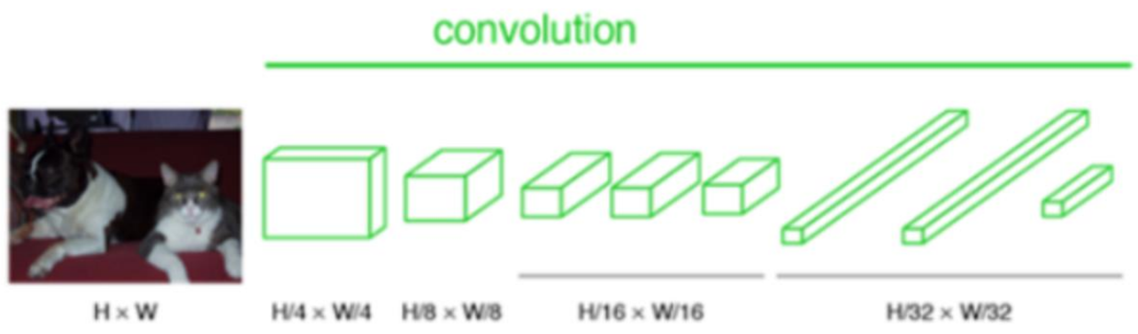
- Fully Convolution Network
- 마지막 FC layer들을 모두 convolutional layer로 대체

- 장점

  2차원 위치 정보를 유지

  FC layer를 쓰지 않기 때문에 어떠한 input이 오더라도 모델이 수용 가능.





Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

- Convolutionalization

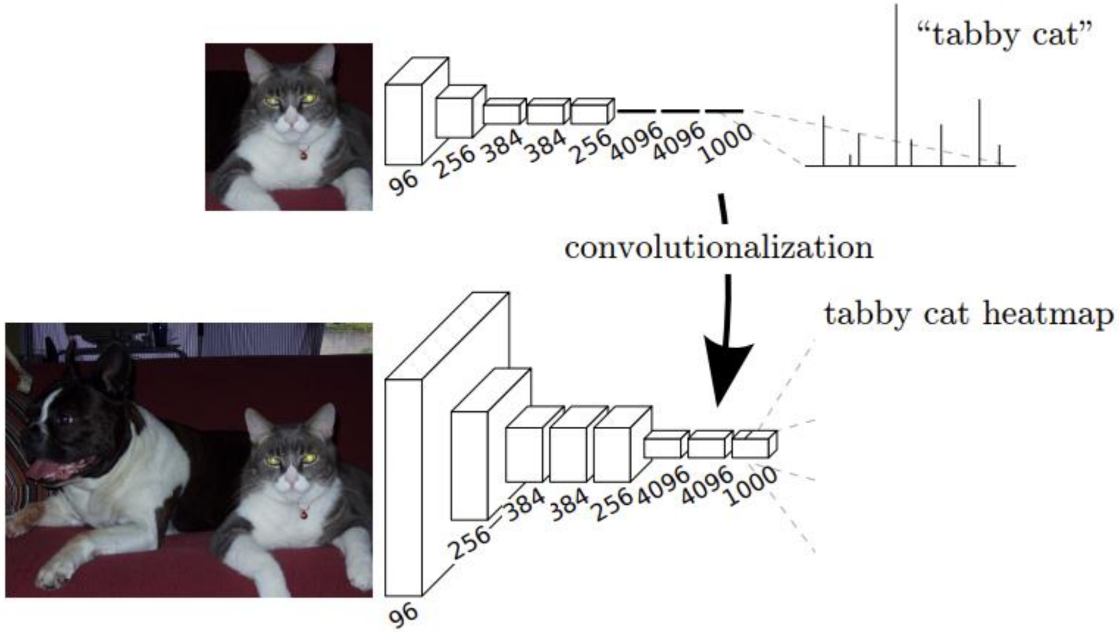
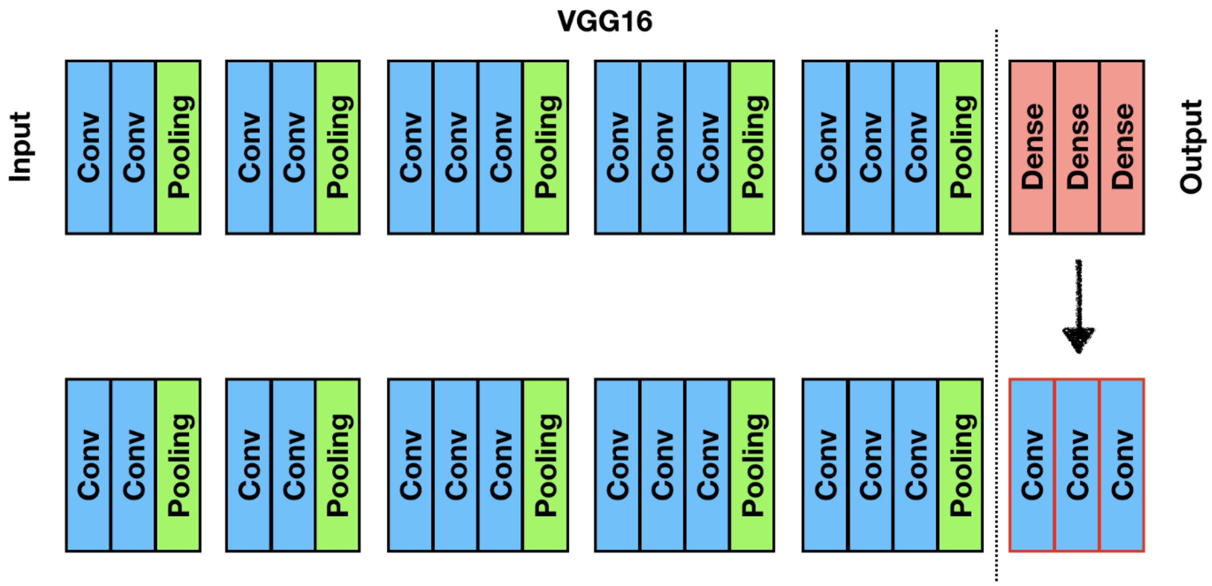


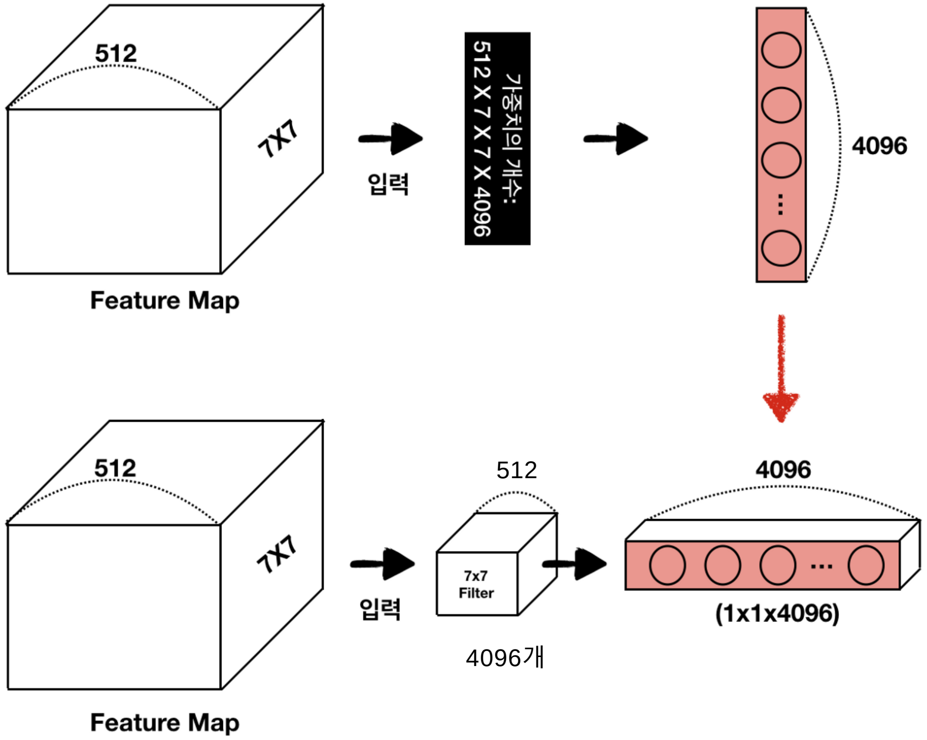
Figure 2. Transforming fully connected layers into convolution layers enables a classification net to output a heatmap. Adding layers and a spatial loss (as in Figure 1) produces an efficient machine for end-to-end dense learning.

Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

- FC layer -> Conv layer
- FC layer는 kernel_size가 input feature map의 spatial dimension인 convolution과 똑같다!

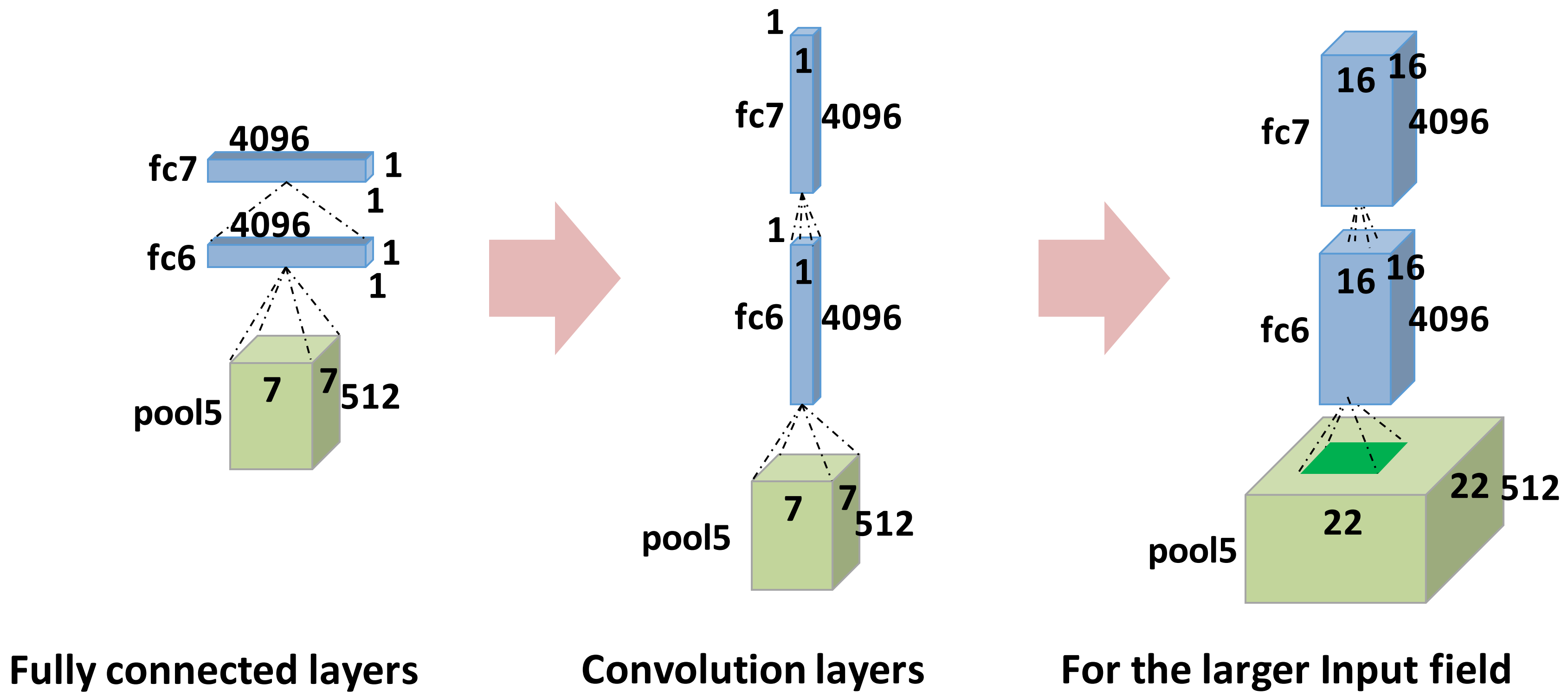




Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

- Converting fully connected layers to convolution layers
  - Each fully connected layer is interpreted as a convolution with a large spatial filter that covers entire input field



Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

- Convolution을 통과한 마지막 feature 맵은 H * W * Class size를 가지도록 한다.
- 즉, 각 channel이 하나의 클래스에 대한 정보를 가지고 있는 것.

Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

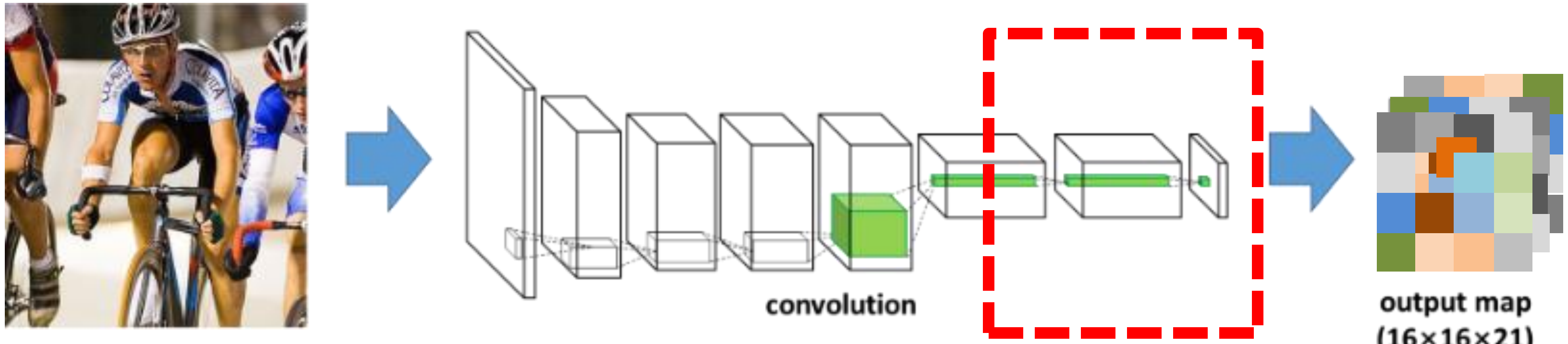
- Convolution을 통과한 마지막 feature 맵은 H * W * Class size를 가지도록 한다.
- 즉, 각 channel이 하나의 클래스에 대한 정보를 가지고 있는 것.

- 하지만 마지막 feature map은 conv와 pooling 연산을 거치면서 spatial dimension이 input에 비해 작아져 있음.
- 이것을 다시 input size에 맞게 키워주는 것이 필요.

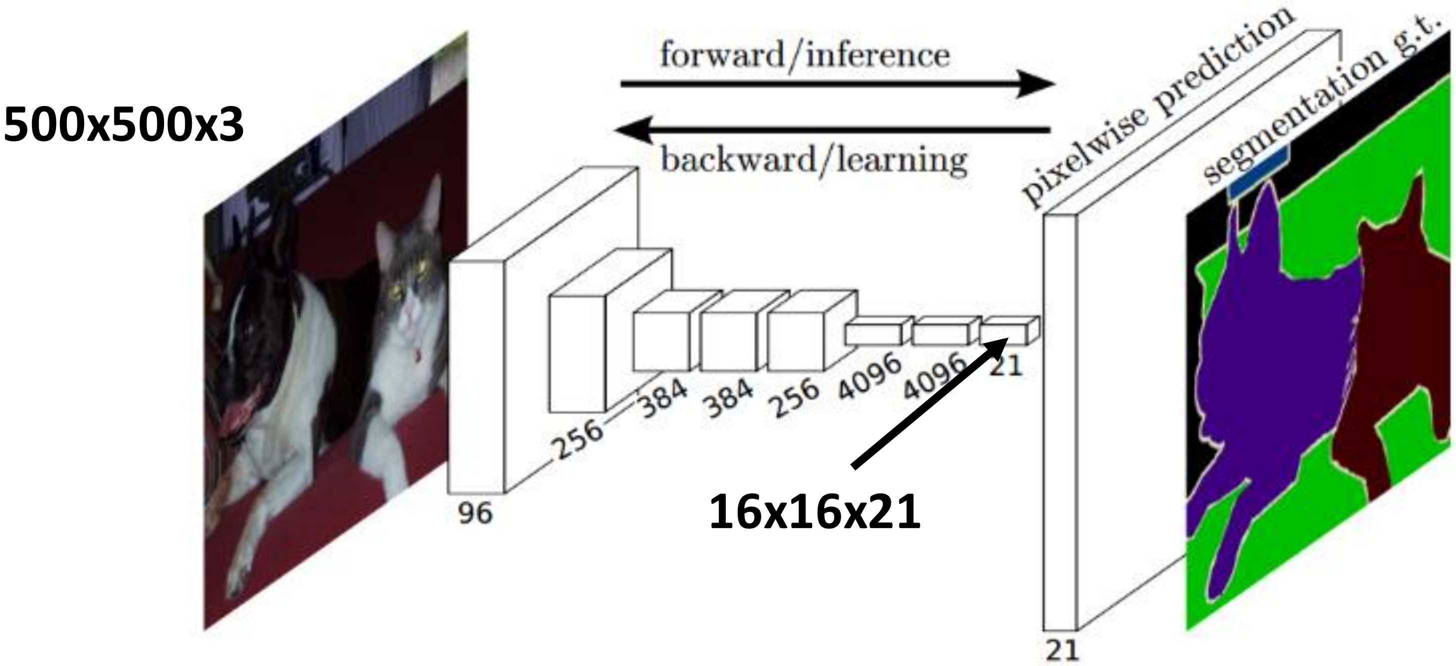
Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

- Recall:



Query image



ic segmentation, *CVPR* 2015.

Fully Convolutional Network

- 애초에 Encoding 부분에서 안 줄여주면 되지 않나요?
  ex) Apply padding, No pooling, …

- Pooling을 하지 않거나 pooling의 stride를 줄임으로써 Feature map의 크기가 작아지는 것을 처음부터 피할 수 있음.
  - 이 경우 receptive field가 줄어들어 이미지의 context를 놓치게 됨.
  - Pooling이 없으면 학습 파라미터 수가 급격히 증가, 연산이 많아짐, 메모리 사용량 증가

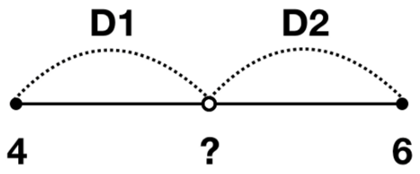
-> 따라서 coarse feature map을 dense map으로 upsampling하는 방법 고려!

Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

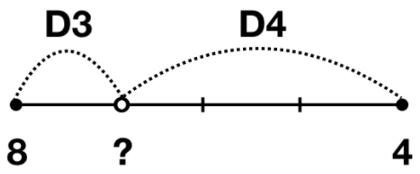
Fully Convolutional Network

10*10 이미지를 100*100으로 확대하려면 어떻게 할까?

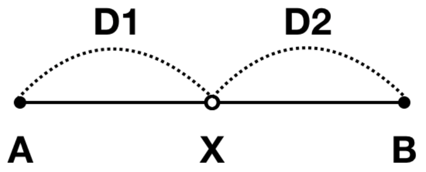
방법 1. Bilinear Interpolation



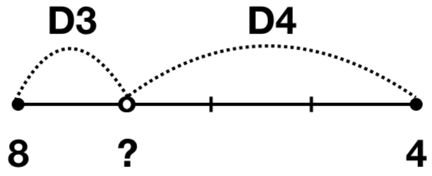
D1=D2



3xD3=D4



$$A\frac{D2}{D1+D2}+B\frac{D1}{D1+D2}$$



3xD3=D4

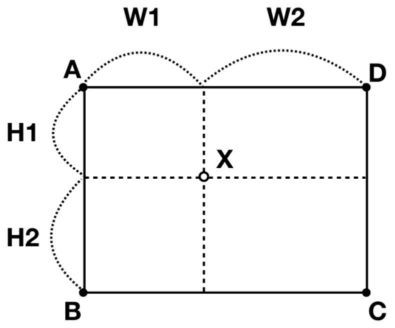
$$A\frac{D2}{D1+D2}+B\frac{D1}{D1+D2} \quad \textbf{=} \quad 8\frac{3}{4}+4\frac{1}{4}=7$$

Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

10*10 이미지를 100*100으로 확대하려면 어떻게 할까?

방법 1. Bilinear Interpolation on 2D



$$X = \left(A\frac{H2}{H1+H2} + B\frac{H1}{H1+H2}\right)\frac{W2}{W1+W2} + \left(D\frac{H2}{H1+H2} + C\frac{H1}{H1+H2}\right)\frac{W1}{W1+W2}$$

2	5
1	3

2x2

2	?	?	5
?	?	?	?
?	?	?	?
1	?	?	3

4x4

2	5	4
1	3	4
2	1	3

3x3

2				5				4
1				3				4
2				1				3

9x9

Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

- Feature map 사이즈를 키워주기 위한 구조 제안
  - Unpooling
  - Transposed Convolution
  - Skip Combining

Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

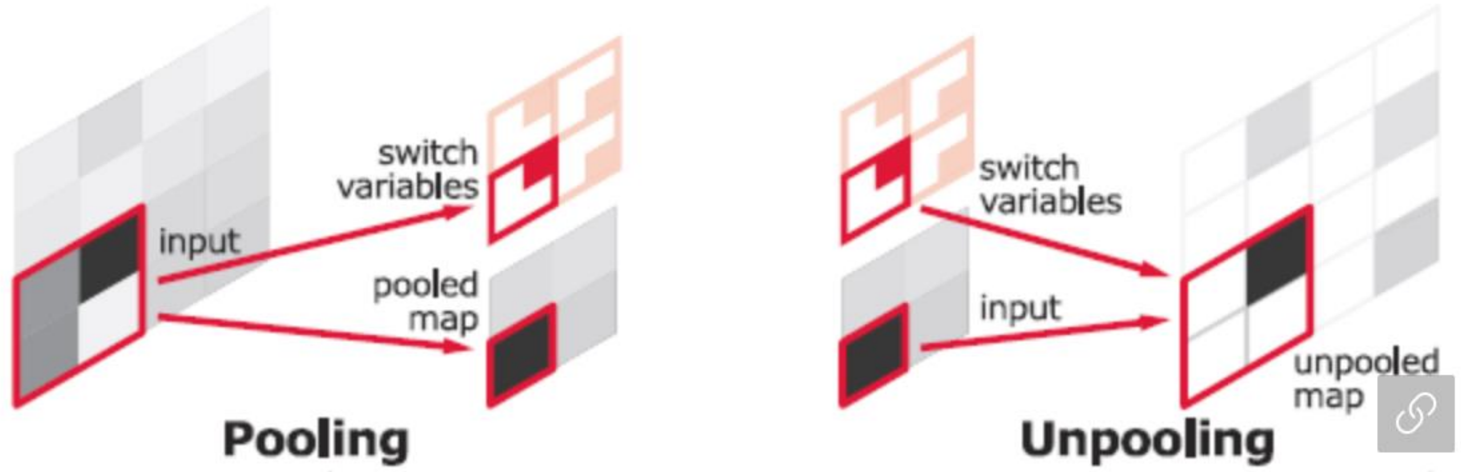
Fully Convolutional Network

- Feature map 사이즈를 키워주기 위한 구조 제안

  Unpooling

  Transposed Convolution

  Skip Combining



Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

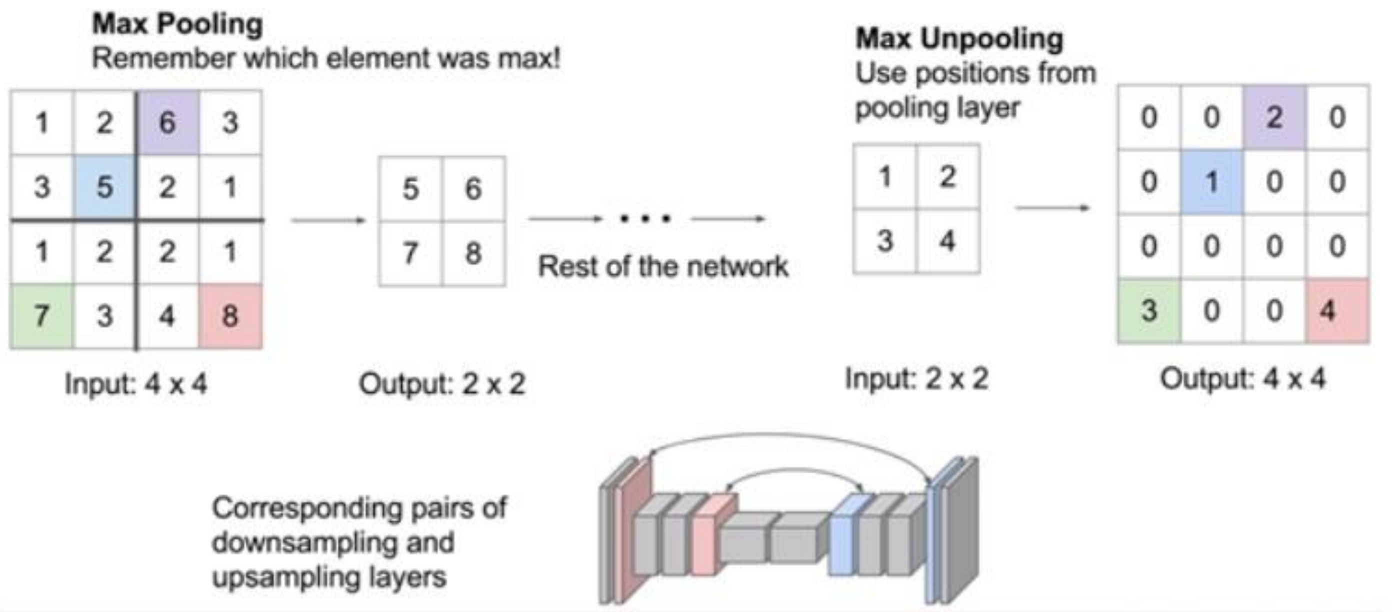
Fully Convolutional Network

- Feature map 사이즈를 키워주기 위한 구조 제안

    Unpooling

    Transposed Convolution

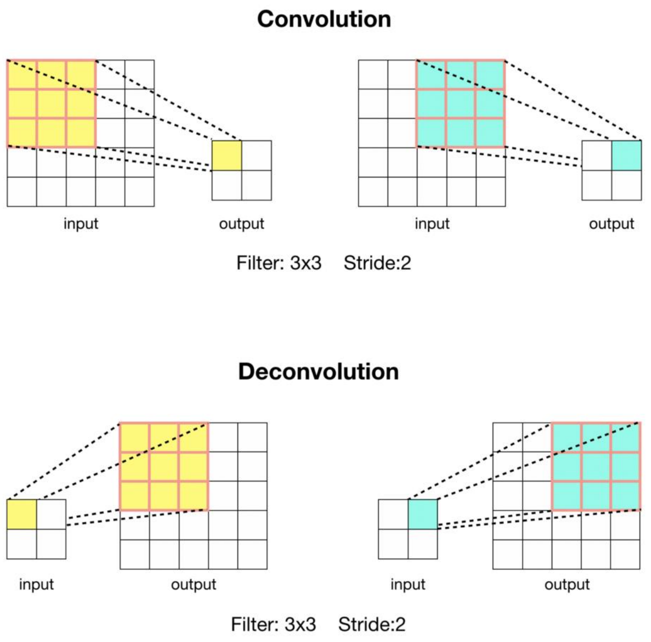
    Skip Combining

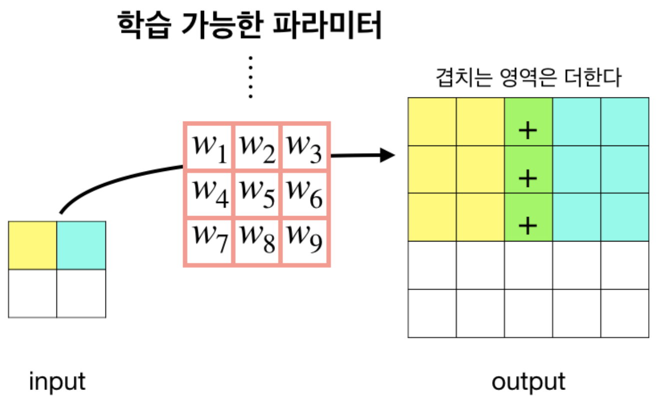


Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

- Feature map 사이즈를 키워주기 위한 구조 제안
  - Unpooling
  - **Transposed Convolution**
  - Skip Combining



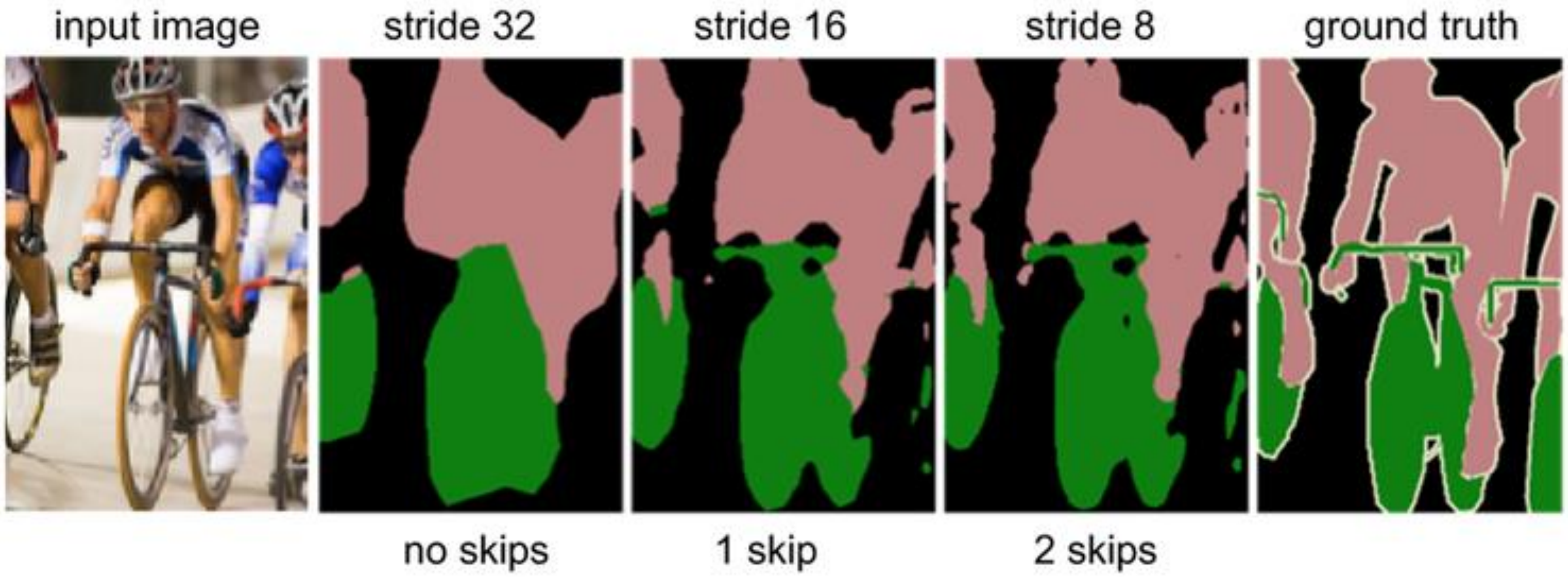


Backwards strided convolution
= Upsampling
= Deconvolution

Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

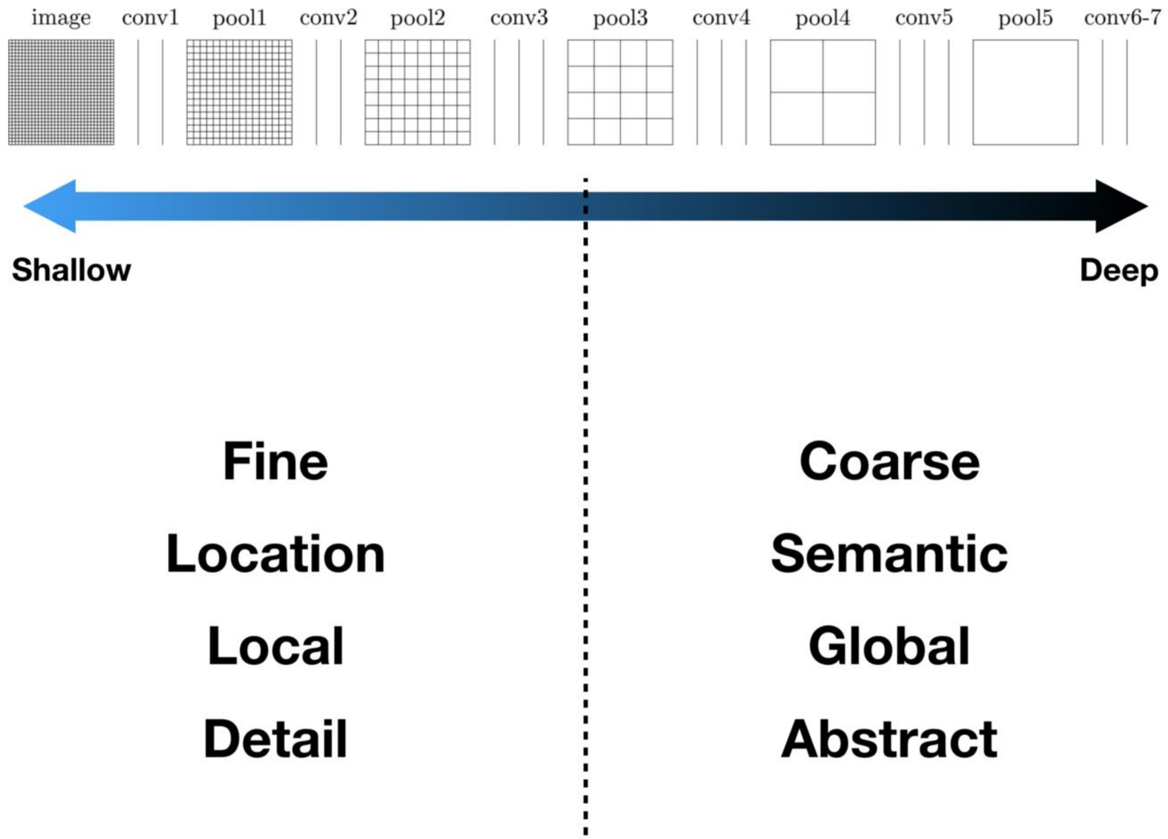
- Feature map 사이즈를 키워주기 위한 구조 제안
  - Unpooling
  - Transposed Convolution
  - **Skip Combining**



Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

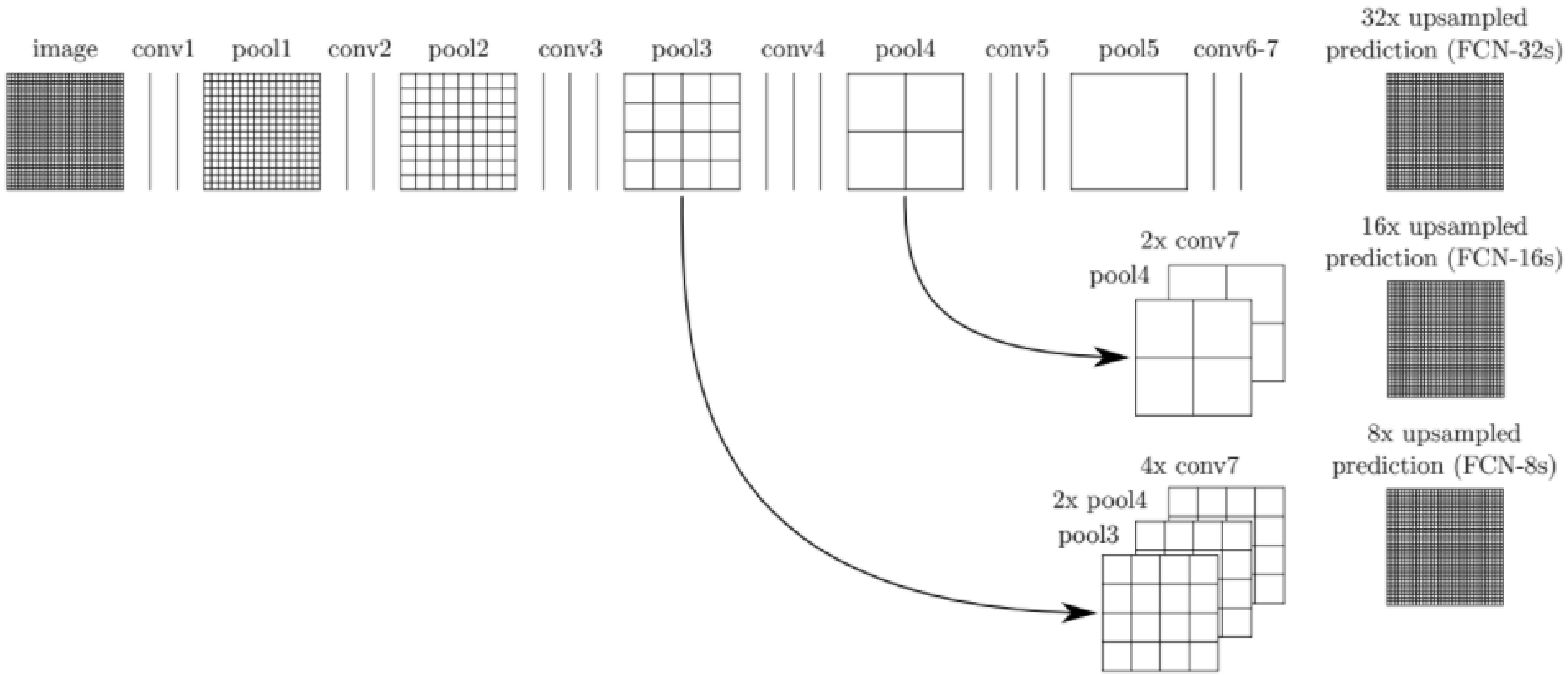
- Feature map 사이즈를 키워주기 위한 구조 제안
  - Unpooling
  - Transposed Convolution
  - **Skip Combining**



Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

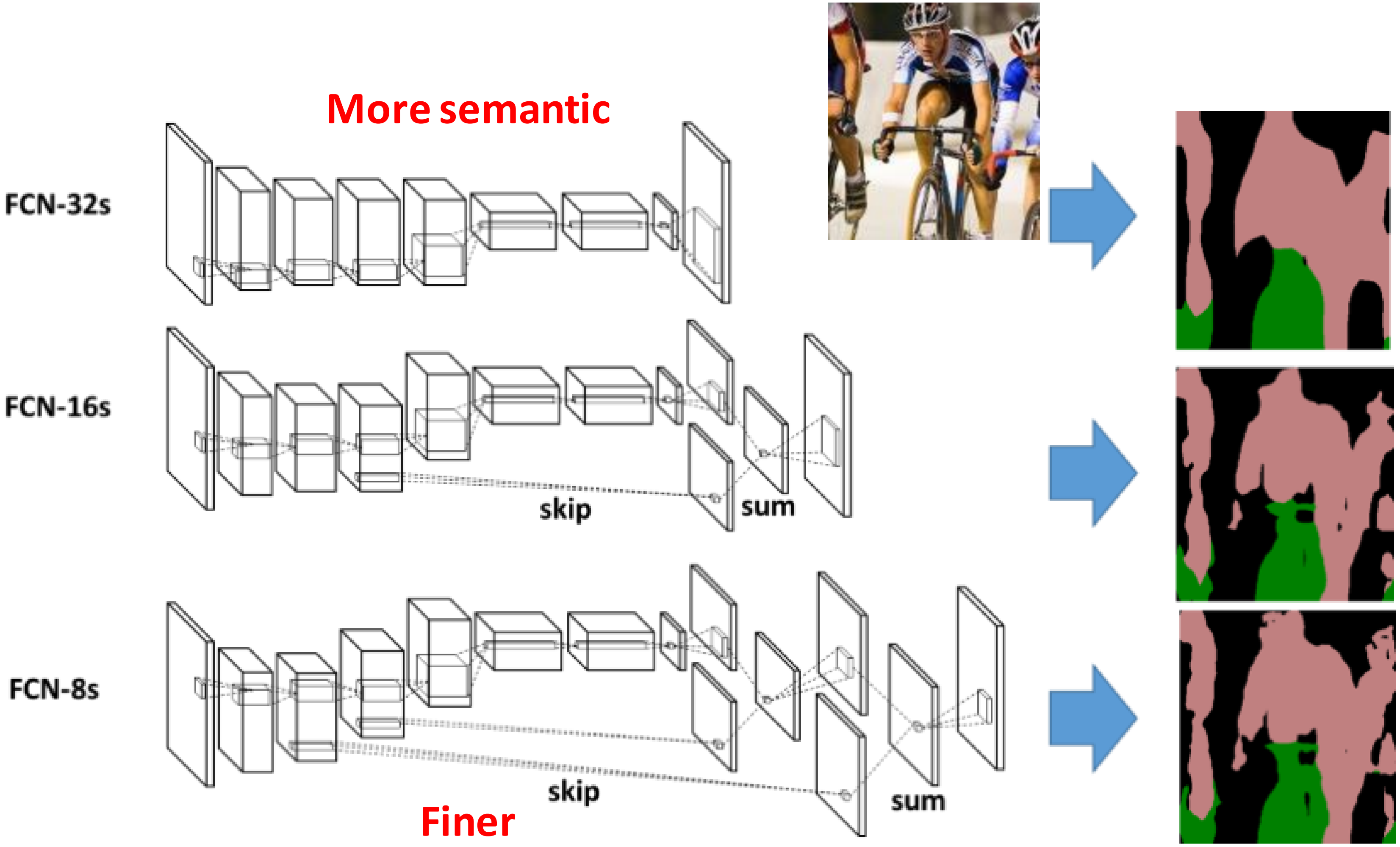
- Feature map 사이즈를 키워주기 위한 구조 제안
  - Unpooling
  - Transposed Convolution
  - **Skip Combining**



Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

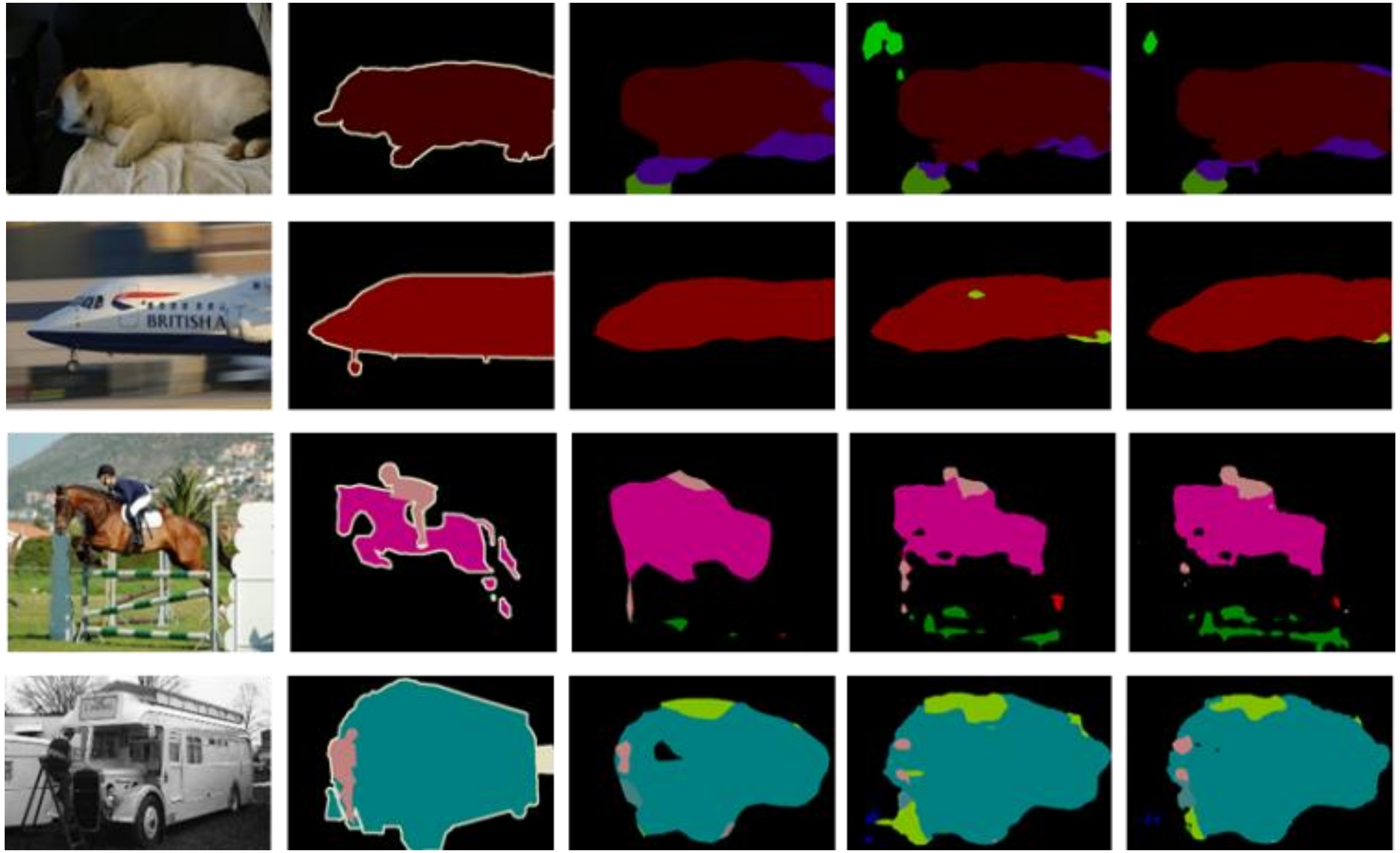
- Skip architecture - Ensemble of three different scales



Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

Input image | GT | FCN-32s | FCN-16s | FCN-8s



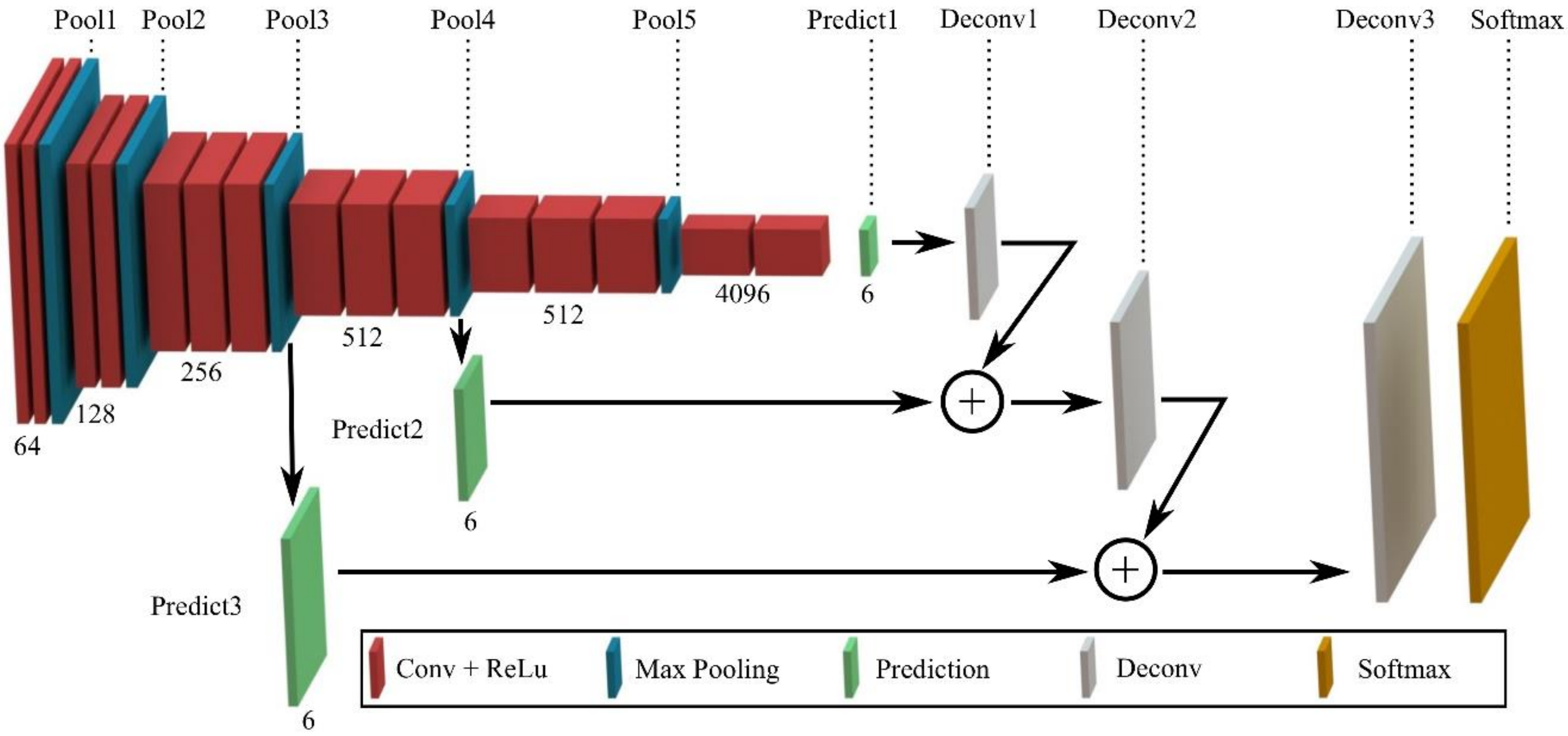
Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

- Limitation of FCN-based semantic segmentation
  - Coarse output score map
    - A single bilinear filter should handle the variations in all kinds of object classes.
    - Difficult to capture detailed structure of objects in image
  - Fixed size receptive field
    - Unable to handle multiple scales
    - Difficult to delineate too small or large objects compared to the size of receptive field
  - Noisy predictions due to skip architecture
    - Trade off between details and noises
    - Minor quantitative performance improvement

Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Fully Convolutional Network

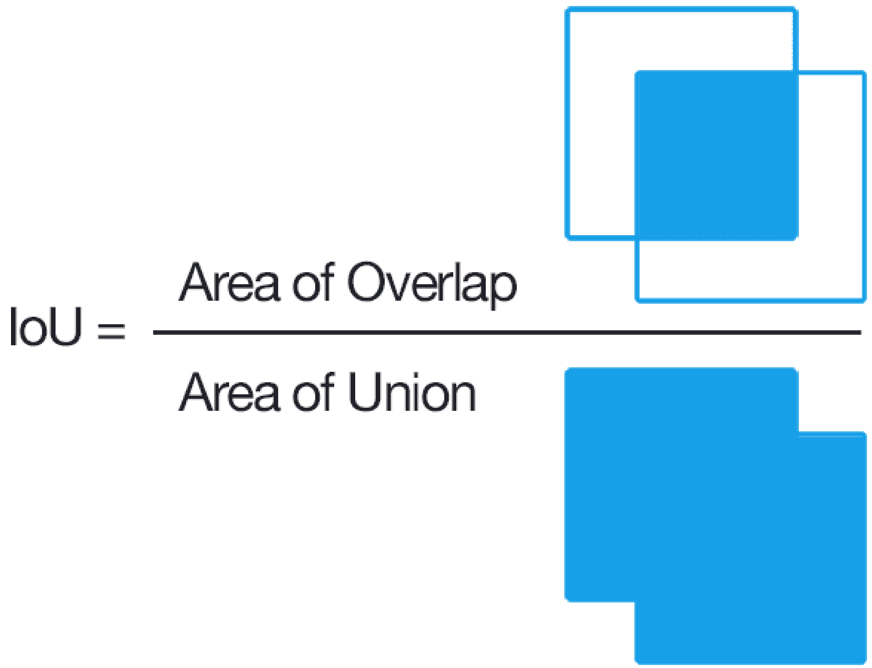


Jonathan et al., **Fully convolutional networks for semantic segmentation**, *CVPR* 2015.

Evaluation metric for segmentation

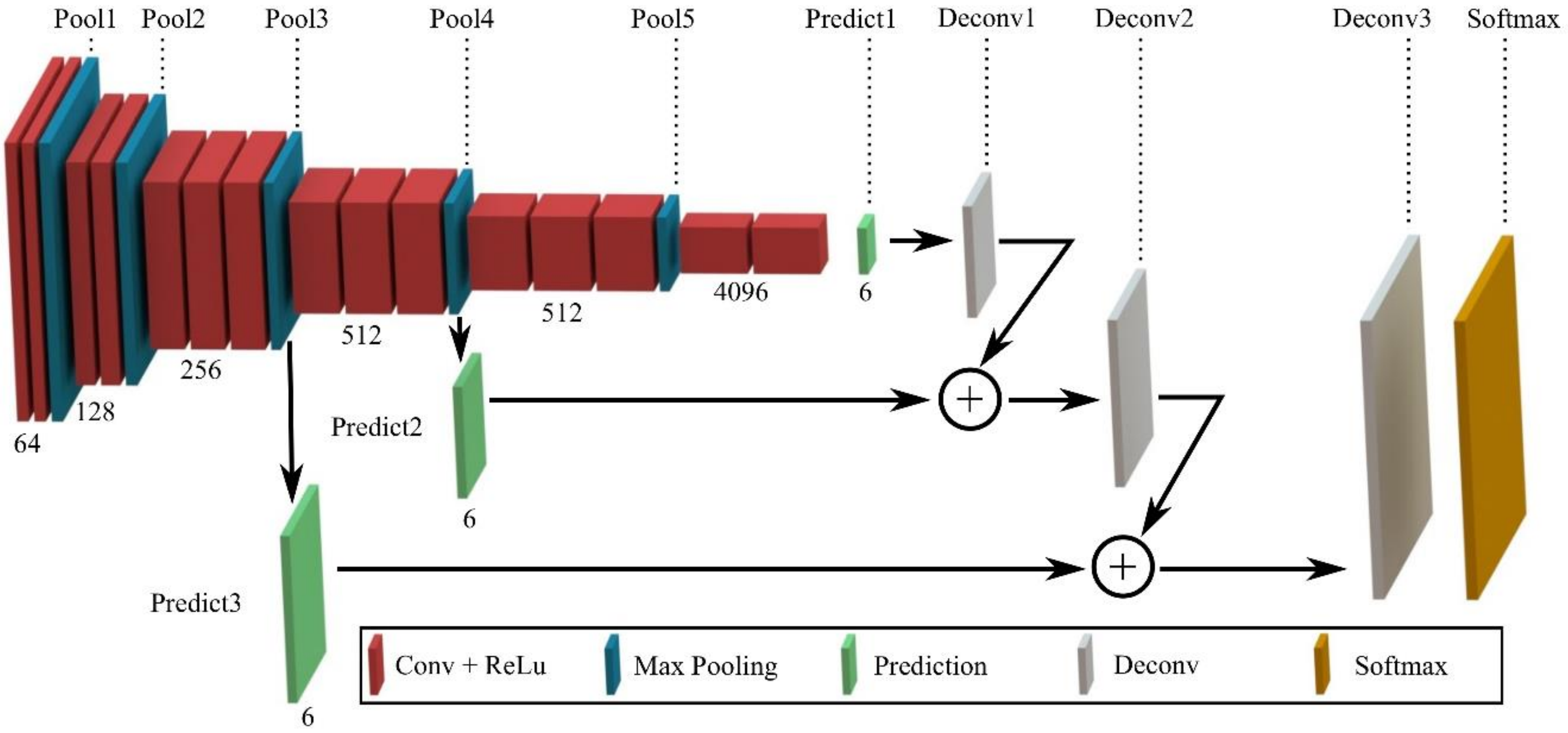
- There are several metrics for semantic segmentation
- Most popular one is Intersection over Union (IoU)
- IoU measures the number of pixels common between the target and prediction masks divided by the total number of pixels present across both masks
- Mean IoU (mIoU)

$$IoU = \frac{target \cap prediction}{target \cup prediction}$$



Exercise 1. FCN implementation

- Implement FCN with the following structure



Exercise 1. FCN implementation

- Exercise1. FCN implementation section in 230628_Segmentation.ipynb
- torch summary is shown on the right

- Things to consider
  - Batch size should be 1 ➔ Why?
  - How to combine feature maps into one feature map to calculate the IoU
  - How to set the number of output feature

```
   | Name        | Type             | Params
-------------------------------------------------
0  | loss        | CrossEntropyLoss | 0
1  | features1   | Sequential       | 38.7 K
2  | features2   | Sequential       | 221 K
3  | features3   | Sequential       | 1.5 M
4  | features4   | Sequential       | 5.9 M
5  | features5   | Sequential       | 7.1 M
6  | maxpool     | MaxPool2d        | 0
7  | classifier  | Sequential       | 119 M
8  | upscore2    | ConvTranspose2d  | 64
9  | upscore4    | ConvTranspose2d  | 64
10 | upscore8    | ConvTranspose2d  | 1.0 K
11 | score_pool4 | Conv2d           | 1.0 K
12 | score_pool3 | Conv2d           | 514
13 | softmax     | Softmax2d        | 0
-------------------------------------------------
134 M     Trainable params
0         Non-trainable params
134 M     Total params
537.086   Total estimated model params size (MB)
```

Exercise 1. FCN implementation

- **Why pad the input?**: The 100 pixel input padding guarantees that the network output can be aligned to the input for any input size in the given datasets, for instance PASCAL VOC. The alignment is handled automatically by net specification and the crop layer. It is possible, though less convenient, to calculate the exact offsets necessary and do away with this amount of padding.
- Why is the batch size 1?: The size of the images are different in the dataset. Although the network can be trained regardless of the input size, the images in the same batch should be the same.

Other Networks

1. DeconvNet
2. Deeplab
3. U-Net

DeconvNet

- Limitations of FCN
- 1. Fixed-size receptive field: 신경망이 오직 하나의 scale이미지만 다룰 수 있음

(a) Inconsistent labels due to large object size

(b) Missing labels due to small object size

*Noh et al., **Learning Deconvolution Network for Semantic Segmentation**, ICCV 2015

DeconvNet

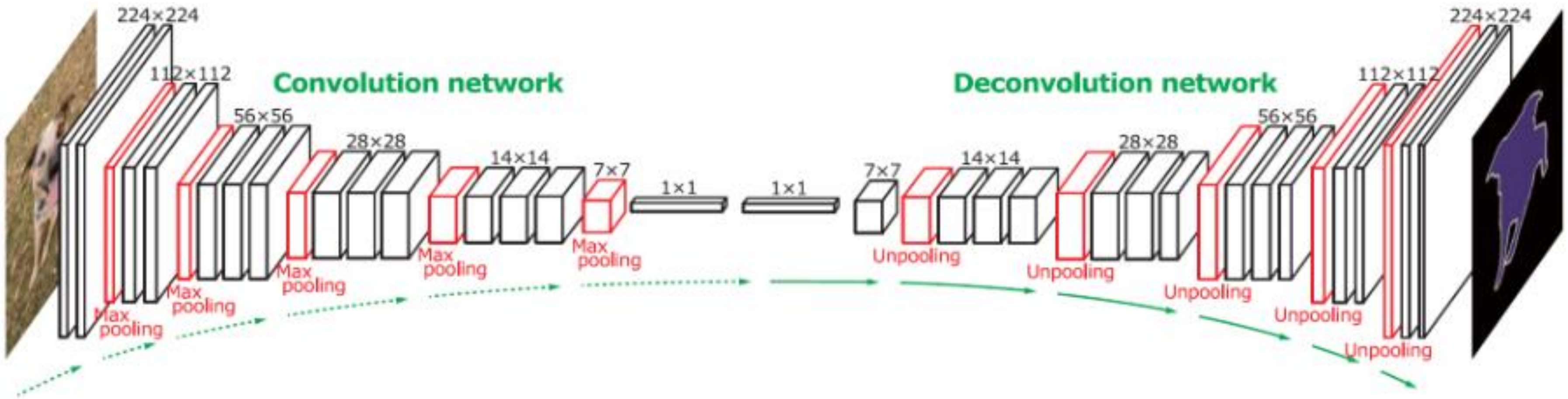
- Limitations of FCN
- 2. Deconvolution is too simple: bilinear interpolation is not good enough

*Noh et al., **Learning Deconvolution Network for Semantic Segmentation**, ICCV 2015

DeconvNet

- 직관적이고 intuition에 맞는 구조 (Auto Encoder)
- Convolution -> input image의 특징을 추출하기 위함
- Deconvolution -> convolution이 추출한 특징을 바탕으로 segmentation 하기 위함

- VGG16에서 마지막 classification layer를 제거하여 convolution으로 사용.
- Deconvolution: unpooling, deconvolution연산을 수행.

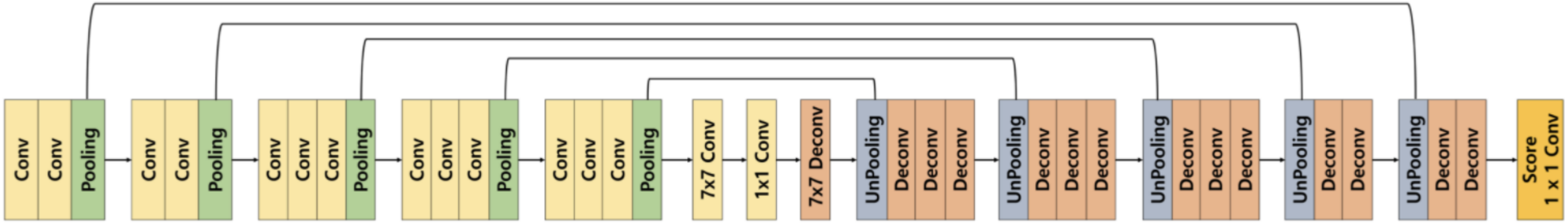


*Noh et al., **Learning Deconvolution Network for Semantic Segmentation**, ICCV 2015

DeconvNet

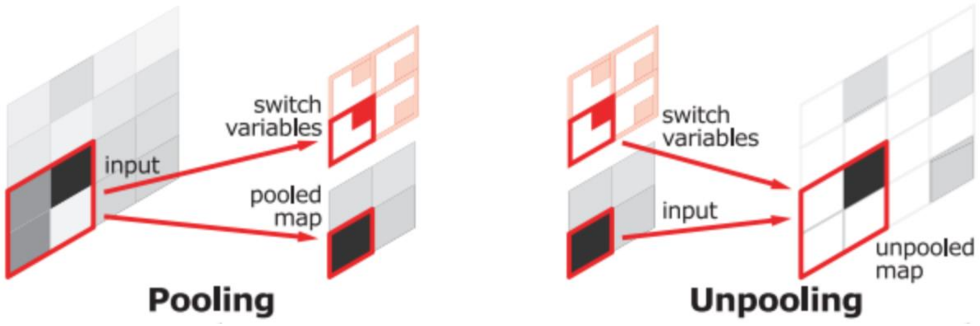
- 직관적이고 intuition에 맞는 구조 (Auto Encoder)
- Convolution -> input image의 특징을 추출하기 위함
- Deconvolution -> convolution이 추출한 특징을 바탕으로 segmentation 하기 위함

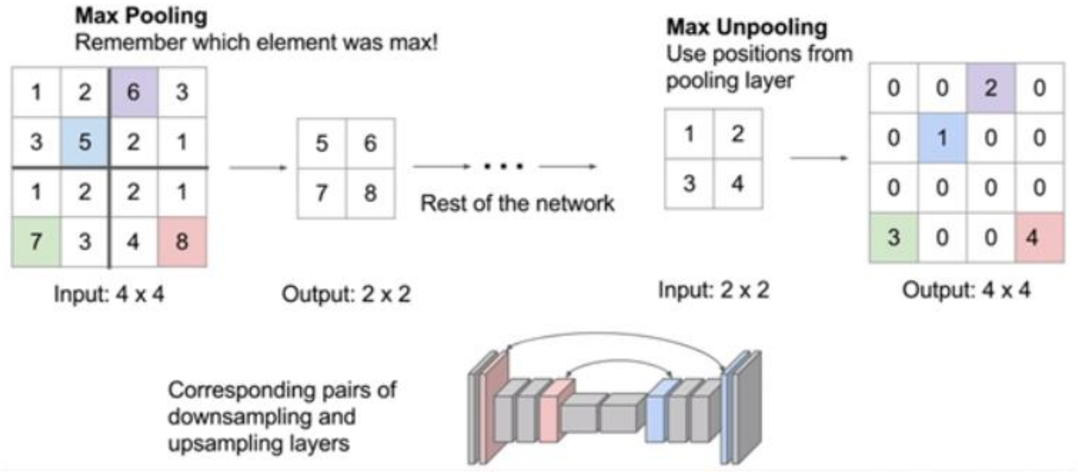
- VGG16에서 마지막 classification layer를 제거하여 convolution으로 사용.
- Deconvolution: unpooling, deconvolution연산을 수행.



*Noh et al., **Learning Deconvolution Network for Semantic Segmentation**, ICCV 2015

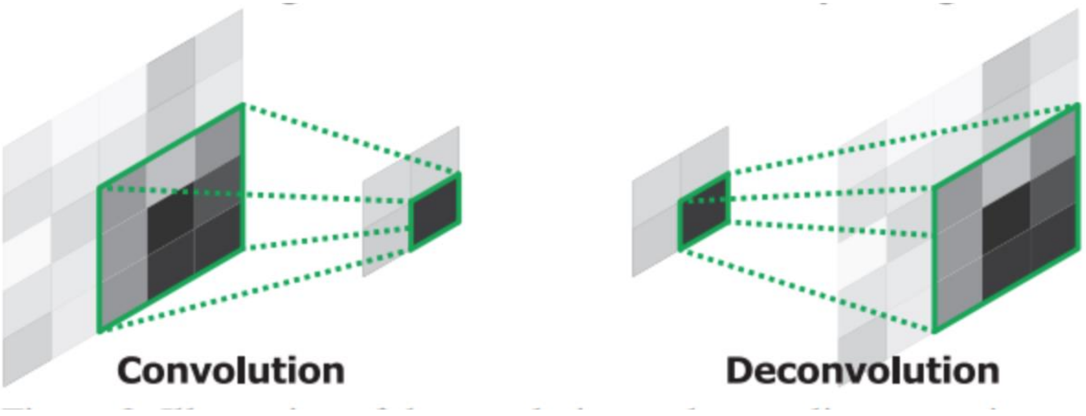
- **Unpooling**
- Deconvolution

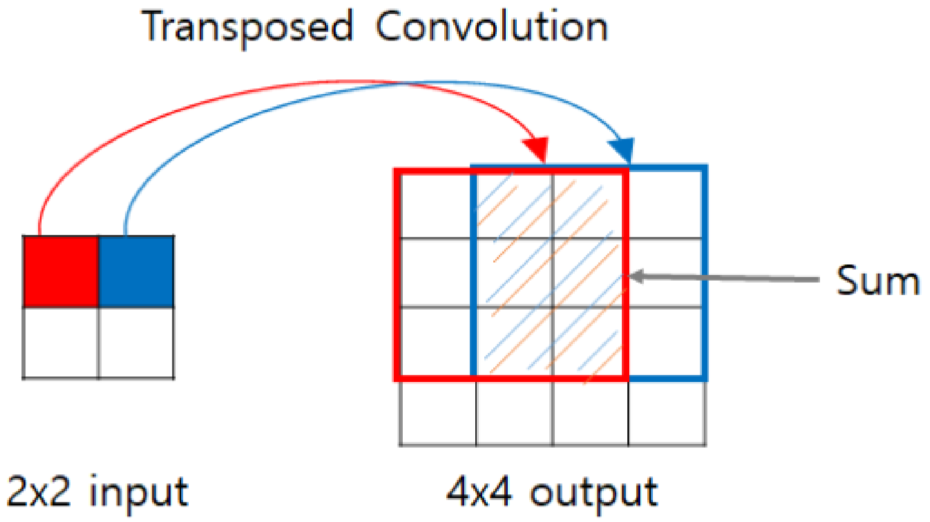




*Noh et al., **Learning Deconvolution Network for Semantic Segmentation**, ICCV 2015

- Unpooling -> using only unpooling, the size of feature map increased and its value is sparse (mostly 0)
- **Deconvolution**





*Noh et al., **Learning Deconvolution Network for Semantic Segmentation**, ICCV 2015

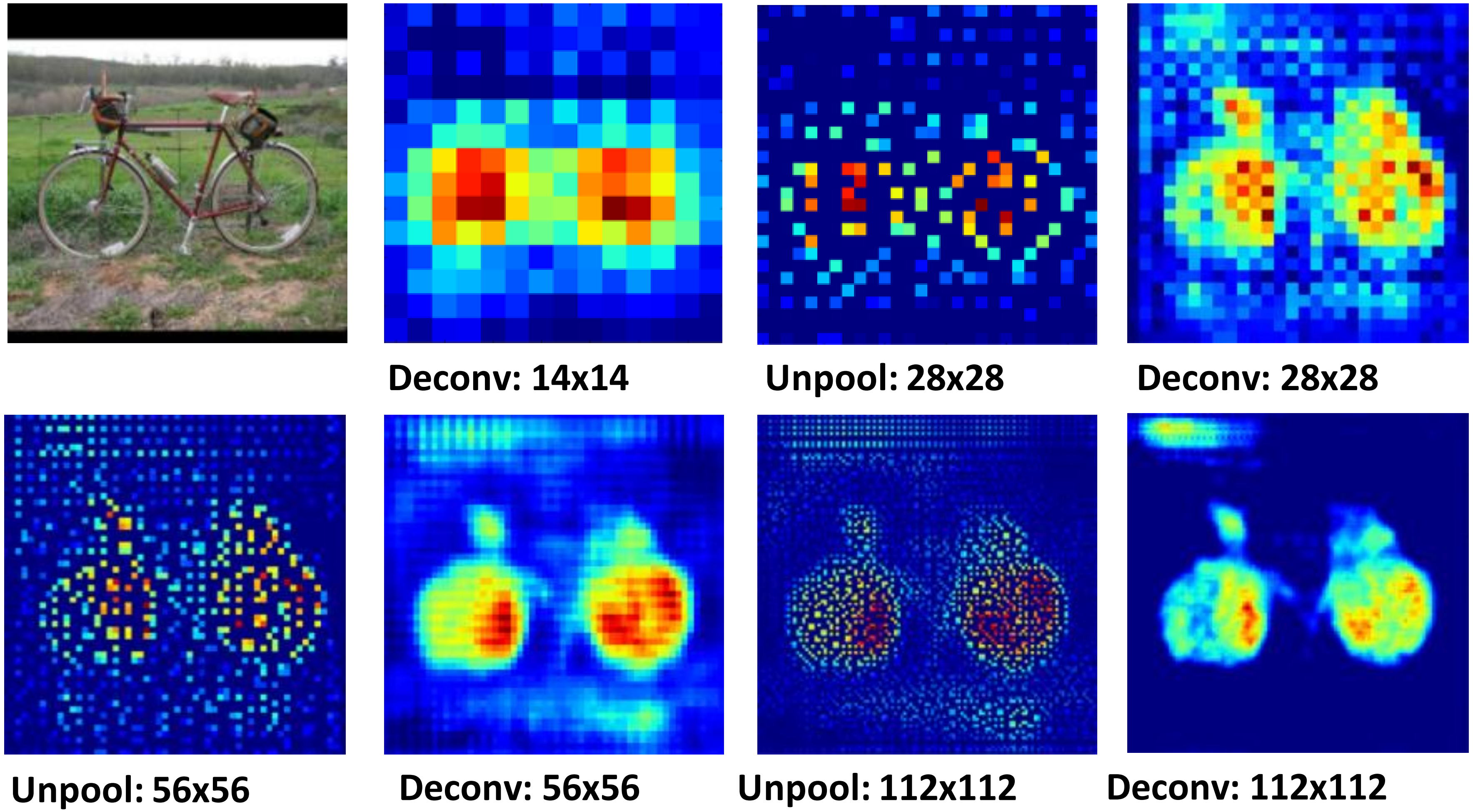
DeconvNet

Table 2. Detailed configuration of the proposed network. "conv" and "deconv" denote layers in convolution and deconvolution network, respectively, while numbers next to each layer name mean the order of the corresponding layer in the network. ReLU layers are omitted from the table for brevity.

name	kernel size	stride	pad	output size
input	-	-	-	$224 \times 224 \times 3$
conv1-1	3×3	1	1	$224 \times 224 \times 64$
conv1-2	3×3	1	1	$224 \times 224 \times 64$
pool1	2×2	2	0	$112 \times 112 \times 64$
conv2-1	3×3	1	1	$112 \times 112 \times 128$
conv2-2	3×3	1	1	$112 \times 112 \times 128$
pool2	2×2	2	0	$56 \times 56 \times 128$
conv3-1	3×3	1	1	$56 \times 56 \times 256$
conv3-2	3×3	1	1	$56 \times 56 \times 256$
conv3-3	3×3	1	1	$56 \times 56 \times 256$
pool3	2×2	2	0	$28 \times 28 \times 256$
conv4-1	3×3	1	1	$28 \times 28 \times 512$
conv4-2	3×3	1	1	$28 \times 28 \times 512$
conv4-3	3×3	1	1	$28 \times 28 \times 512$
pool4	2×2	2	0	$14 \times 14 \times 512$
conv5-1	3×3	1	1	$14 \times 14 \times 512$
conv5-2	3×3	1	1	$14 \times 14 \times 512$
conv5-3	3×3	1	1	$14 \times 14 \times 512$
pool5	2×2	2	0	$7 \times 7 \times 512$
fc6	7×7	1	0	$1 \times 1 \times 4096$
fc7	1×1	1	0	$1 \times 1 \times 4096$
deconv-fc6	7×7	1	0	$7 \times 7 \times 512$
unpool5	2×2	2	0	$14 \times 14 \times 512$
deconv5-1	3×3	1	1	$14 \times 14 \times 512$
deconv5-2	3×3	1	1	$14 \times 14 \times 512$
deconv5-3	3×3	1	1	$14 \times 14 \times 512$
unpool4	2×2	2	0	$28 \times 28 \times 512$
deconv4-1	3×3	1	1	$28 \times 28 \times 512$
deconv4-2	3×3	1	1	$28 \times 28 \times 512$
deconv4-3	3×3	1	1	$28 \times 28 \times 256$
unpool3	2×2	2	0	$56 \times 56 \times 256$
deconv3-1	3×3	1	1	$56 \times 56 \times 256$
deconv3-2	3×3	1	1	$56 \times 56 \times 256$
deconv3-3	3×3	1	1	$56 \times 56 \times 128$
unpool2	2×2	2	0	$112 \times 112 \times 128$
deconv2-1	3×3	1	1	$112 \times 112 \times 128$
deconv2-2	3×3	1	1	$112 \times 112 \times 64$
unpool1	2×2	2	0	$224 \times 224 \times 64$
deconv1-1	3×3	1	1	$224 \times 224 \times 64$
deconv1-2	3×3	1	1	$224 \times 224 \times 64$
output	1×1	1	1	$224 \times 224 \times 21$

on **Network for Semantic Segmentation**, ICCV 2015

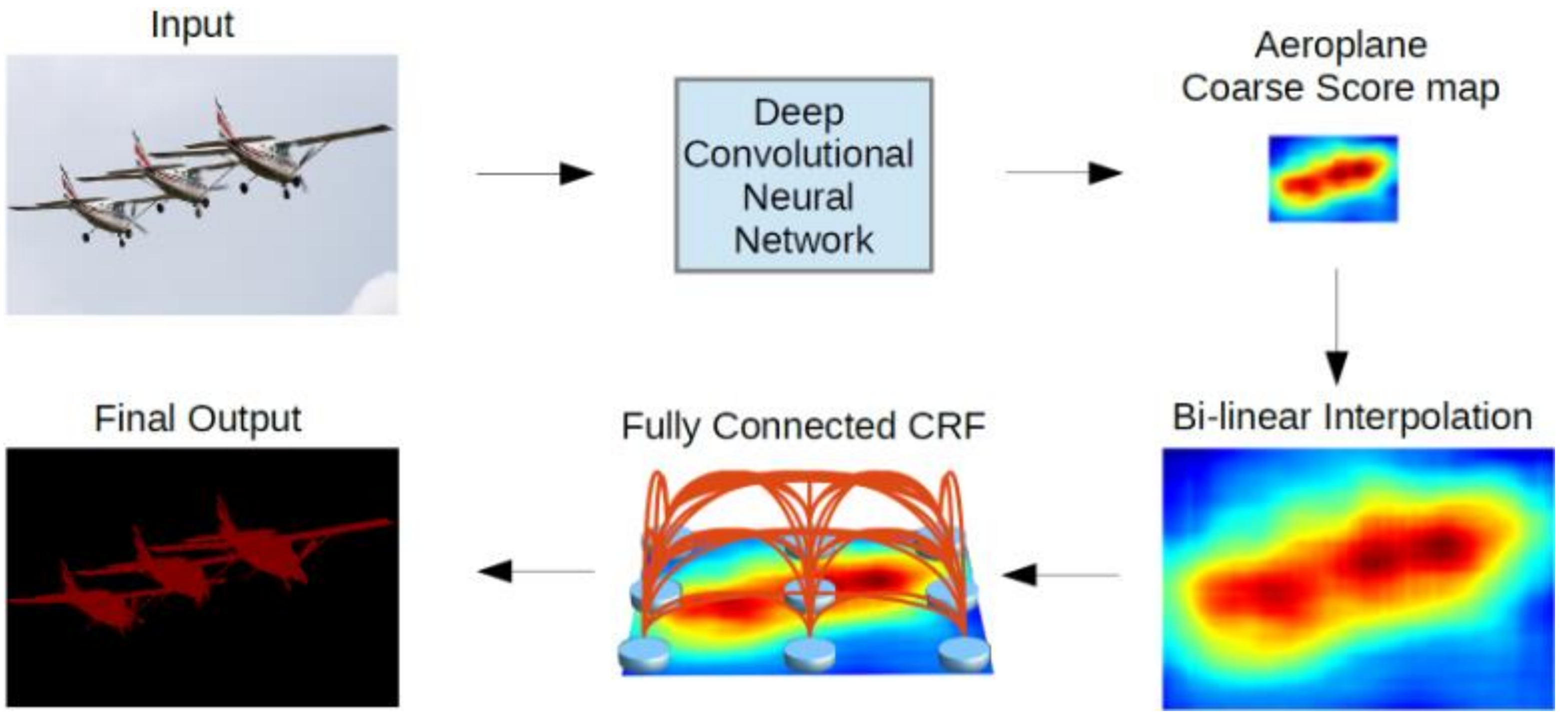
DeconvNet



*Noh et al., **Learning Deconvolution Network for Semantic Segmentation**, ICCV 2015

Deeplab

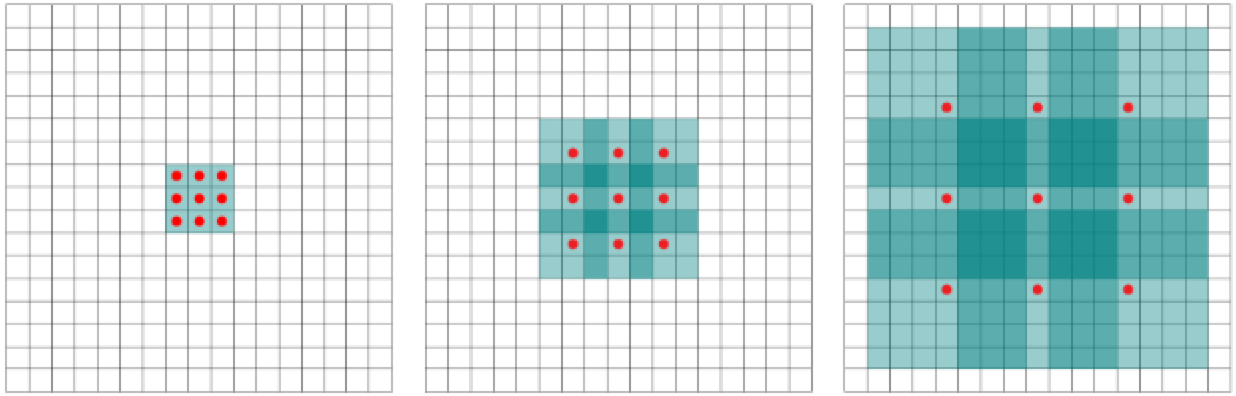
- An advanced version of FCN
  - *Atrous convolutions* that enlarge receptive fields
  - *Atrous Spatial Pyramid Pooling* (ASPP) on top of a convolutional feature map
  - *Fully-connected Conditional Random Field* (CRF), a post-processing technique
  - A more powerful backbone network: VGG16 -> ResNet101



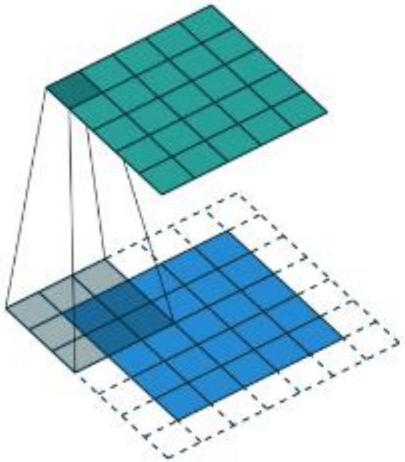
*Chen et al., **DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs**, TPAMI 2017

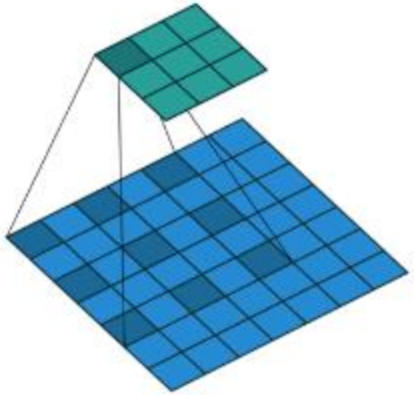
Deeplab

- Atrous convolution
  - Convolution kernel with “holes” (trous in French)
  - Another parameter to define the kernel: *dilation rate* r



$r = 1$ Ordinary convolution | $r = 2$ | $r = 4$



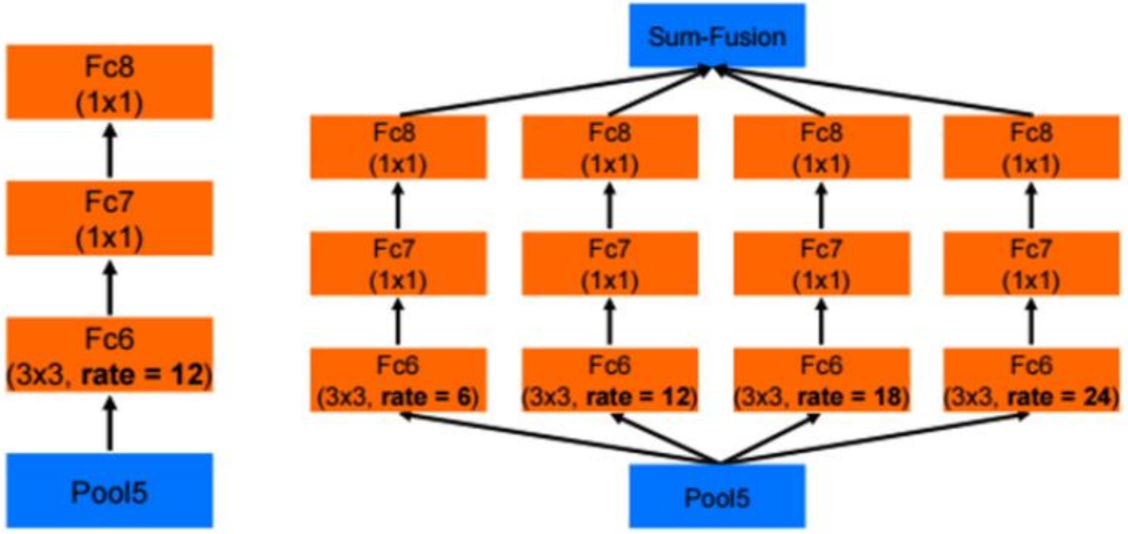


The receptive field grows exponentially while the number of parameters grows linearly!

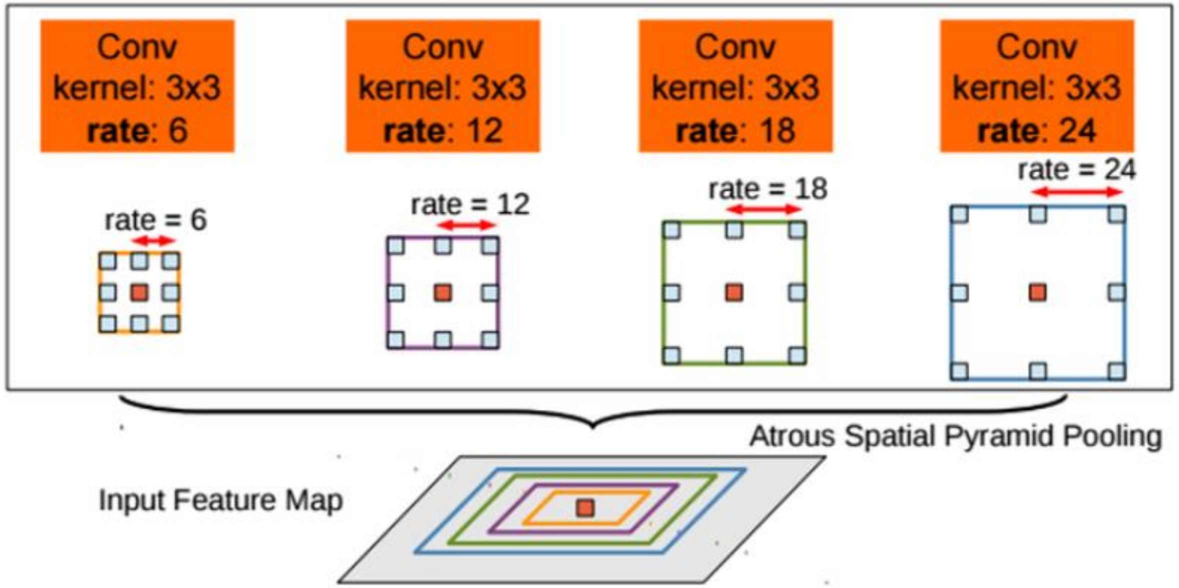
*Chen et al., **DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs** , TPAMI 2017

Deeplab

- Atrous Spatial Pyramid Pooling (ASPP)
  - Rich spatial information using various receptive fields



(a) DeepLab-LargeFOV (b) DeepLab-ASPP



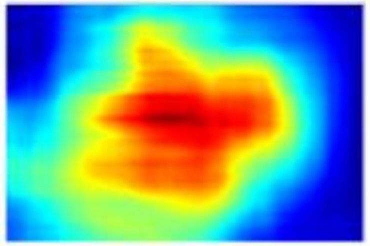
Deeplab

- Fully-connected CRF as a post-processing step

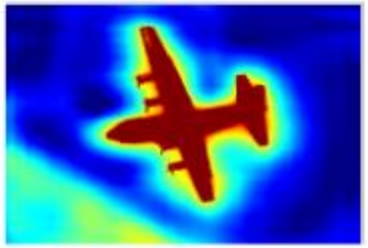
Basic intuition
If a pair of pixels are similar in the input image, their scores in the belief map should be also similar.

Input image

Score map (CNN output)



After applying CRF



- Energy minimization, where the energy is defined by unary and pairwise potentials

$$E(\boldsymbol{x}) = \sum_i \phi_i(x_i) + \sum_{ij} \psi_{ij}(x_i, x_j)$$

$\phi(x)$: $-\log P(x)$
$P(x)$: 해당 픽셀의 label이 x일 확률

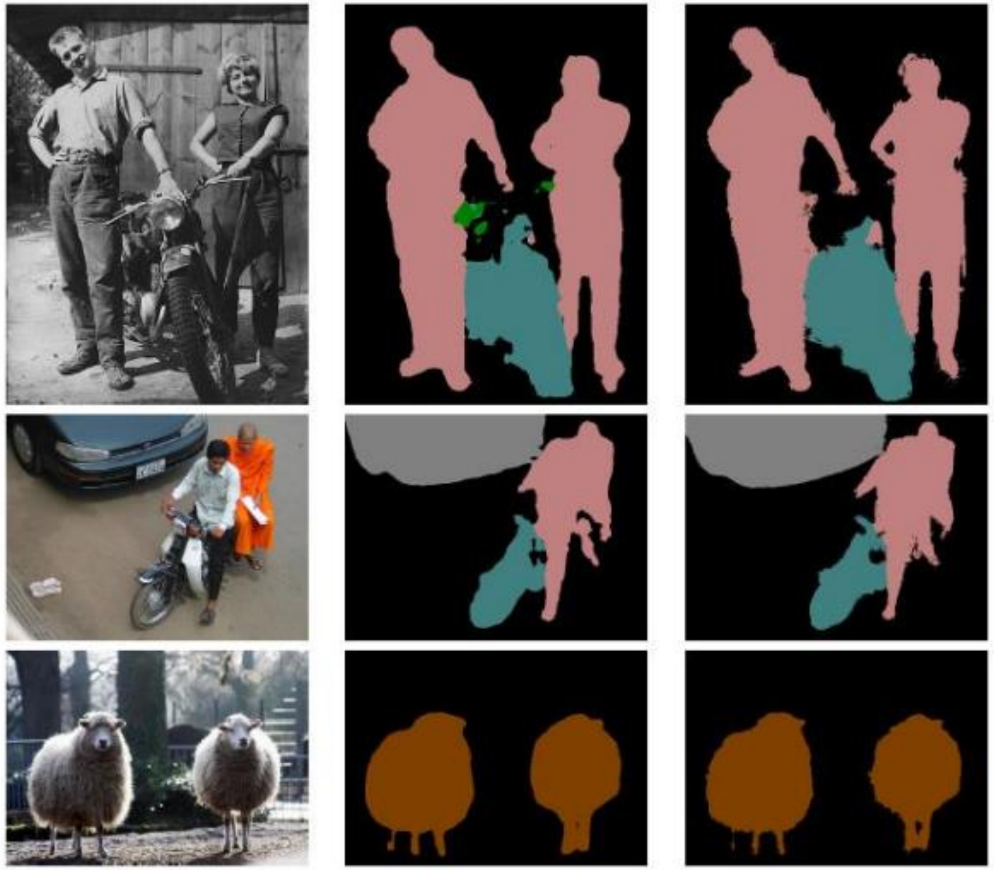
- Unary potential based on the output scores, and pairwise potential given by

$$\psi_{ij}(x_i, x_j) = I(x_i, x_j)\left[w_1 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\alpha^2} - \frac{\|c_i - c_j\|^2}{2\sigma_\beta^2}\right) + w_2 \exp\left(-\frac{\|c_i - c_j\|^2}{2\sigma_\gamma^2}\right)\right]$$

*Chen et al., **DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs**, TPAMI 2017

Deeplab

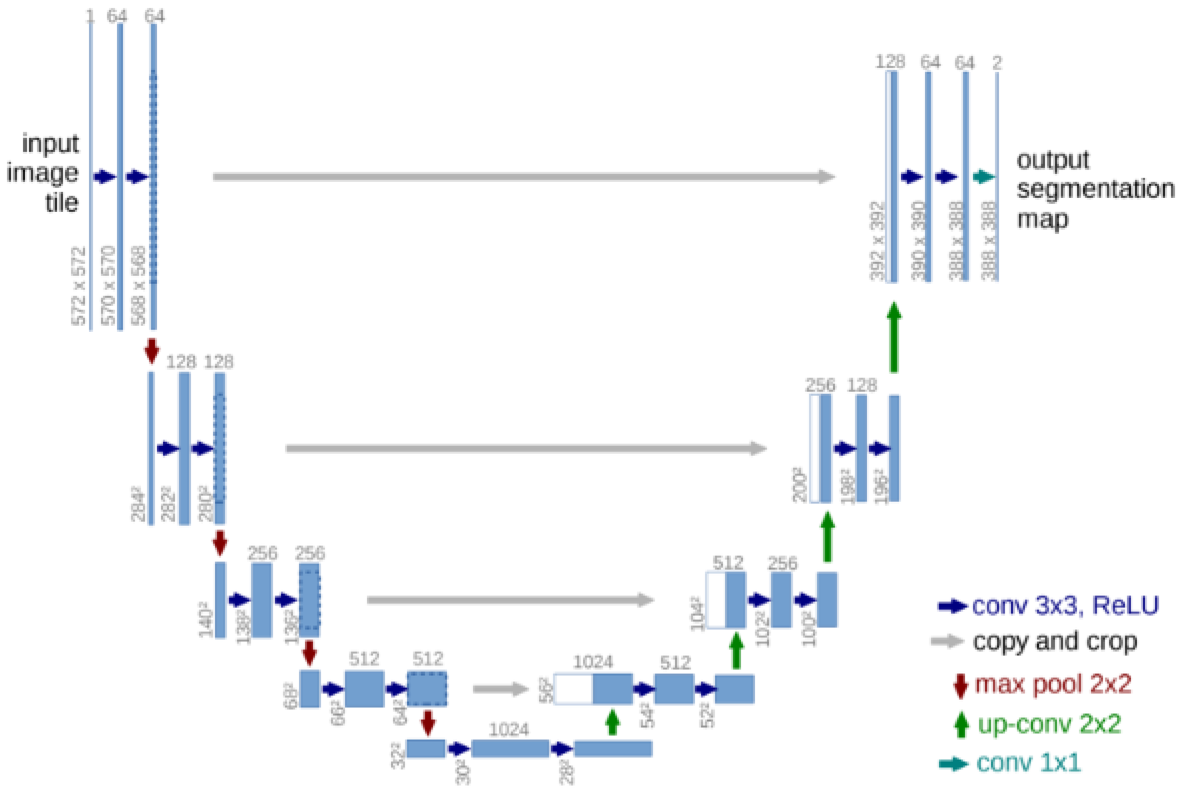
- Experimental results

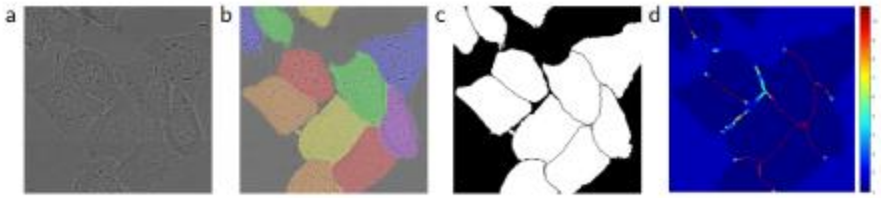


Input image Before CRF After CRF

Method	mIOU
DeepLab-CRF-LargeFOV-COCO [58]	72.7
MERL_DEEP_GCRF [89]	73.2
CRF-RNN [59]	74.7
POSTECH_DeconvNet_CRF_VOC [61]	74.8
BoxSup [60]	75.2
Context + CRF-RNN [76]	75.3
QO_4^{mres} [66]	75.5
DeepLab-CRF-Attention [17]	75.7
CentraleSuperBoundaries++ [18]	76.0
DeepLab-CRF-Attention-DT [63]	76.3
H-ReNet + DenseCRF [90]	76.8
LRR_4x_COCO [91]	76.8
DPN [62]	77.5
Adelaide_Context [40]	77.8
Oxford_TVG_HO_CRF [88]	77.9
Context CRF + Guidance CRF [92]	78.1
Adelaide_VeryDeep_FCN_VOC [93]	79.1
DeepLab-CRF (ResNet-101)	79.7

- U-Net: Convolutional Networks for Biomedical Image Segmentation





*Ronneberger et al, U-Net: Convolutional Networks for Biomedical Image Segmentation, MICCAI 2015

Exercise 2. U-Net implementation

- Exercise 2. U-Net implementation section in 230628_Segmentation.ipynb
- Think how to implement skip-connection