📖

[논문 리뷰] DeConvNet [2015]

🌟 해당 논문은 **Deconvolution Network**에 의하여, 학습되는 **Semantic Segmentation을 제안**합니다.

VGG16의 layer를 사용하였으며, deconvolution 과 unpooling layer로 구성되어 있습니다.
DeConvNet은 이전의 **Fully Convolutional networks에 기반한 이전의 방식의 한계를 완화**합니다.

Introduction

이전의 FCN에 기반한 Semantic Segmentataion에는 **2가지 문제**가 있습니다.

First Problem

- **Network**는 **고정된 수용영역으로 인하여,**
  크거나, 작은 물체의 경우 잘못된 레이블을 지정하는
  문제가 있습니다.
- **Label prediction**은 **큰 객체의 경우**만, 정보를 가져올 수 있습니다.
- **같은 Pixel**내의 **객체에서 inconsistent**가 발생하는 문제가 있습니다.
- Skip architecture를 시도해봤음에도 상당한 성능을 얻기에는 근본적인 해결책이 아니였습니다.

(a) Inconsistent labels due to large object size

(b) Missing labels due to small object size

Figure 1. Limitations of semantic segmentation algorithms based on fully convolutional network. (Left) original image. (Center) ground-truth annotation. (Right) segmentations by [17]

Second Problem

객체의 상세한 구조는 **label map**으로 인하여 **종종 손실되거나, 매끄럽지 않습니다.**

FCN에서는 종종 upsampling을 위해, bilinear interpolation[선형 보간법]을 사용합니다.
하지만, **Real Deconvolution**을 사용하지 않는 것은 **좋은 성능을 얻기에 상당히 어렵습니다.**

> 💡 **이러한 한계를 개선하기 위해 제안된 것이 DeepConvNet입니다.**

- Deconvolution, unpooling과 RELU로 구성된 deconvolution network를 구성하였습니다.
- Pre trained된 Network는 **instance-wise segmentatiaon을 얻기 위해**
  individual object proposals에 적용됩니다.

Architecture

✨ 모델은 **Convolution 과 Deconvolution Networks**으로 **2부분**으로 이루어져 있습니다.

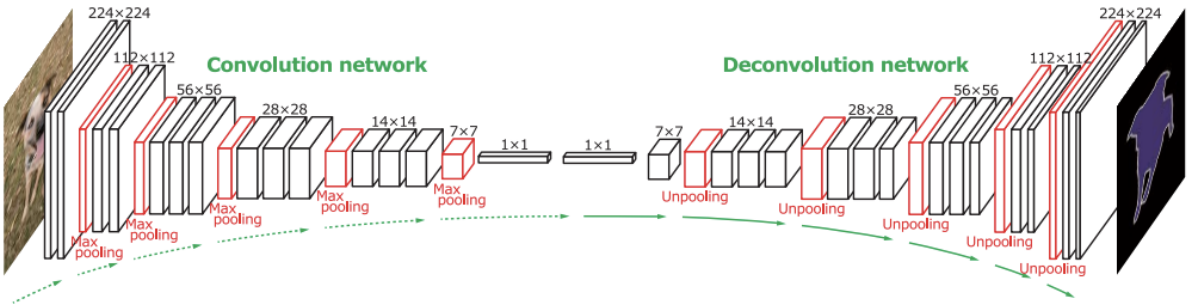


Figure 2. Overall architecture of the proposed network. On top of the convolution network based on VGG 16-layer net, we put a multi-layer deconvolution network to generate the accurate segmentation map of an input proposal. Given a feature representation obtained from the convolution network, dense pixel-wise class prediction map is constructed through multiple series of unpooling, deconvolution and rectification operations.

Convolution Network와 Input Image를 multidimensional feature representataion으로 변환하는 **feature extractor는 유사**합니다.

반면, **Deconvolution Network**는 Convolution Network으로 부터 추출된 feature extractor로부터 **Object Segmentataion을 진행**합니다.

🌟 Network의 Final Output은
미리 정의된 클래스 중 하나에 속하는 확률을 나타내는 **Input image와 동일한 크기의 확률 맵입**니다.

Convolution은 VGG16에서 마지막 classifier layer를 제거하고 사용합니다. 그리고, class-specific projection을 사용하기 위해 2개의 fully connected layer를 추가합니다.

Deconvolution은 Convolution의 거울 버전이라고 보면 된다. Convolution Network와 다르게 feed forwarding을 통하여, 활성화된 크기를 감소시킵니다.

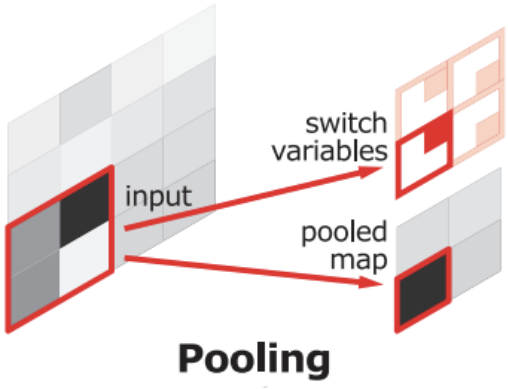
그리고, **deconvolution networks는 unpooling 과 deconvolution operations의 결합**을 통하여,
활성화를 확대시킵니다.

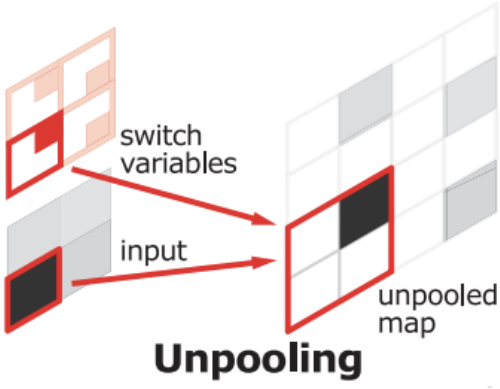
name	kernel size	stride	pad	output size
input	-	-	-	$224 \times 224 \times 3$
conv1-1	3×3	1	1	$224 \times 224 \times 64$
conv1-2	3×3	1	1	$224 \times 224 \times 64$
pool1	2×2	2	0	$112 \times 112 \times 64$
conv2-1	3×3	1	1	$112 \times 112 \times 128$
conv2-2	3×3	1	1	$112 \times 112 \times 128$
pool2	2×2	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	$224 \times 224 \times 21$

Unpooling & Deconvolution

Unpooling

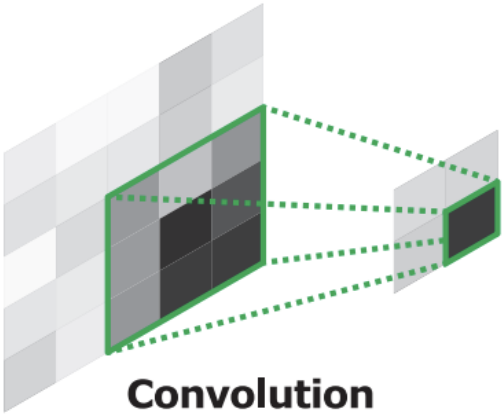
Unpooling strategy는 **input object의 구조로 재형성하는데 상당히 유용**합니다.

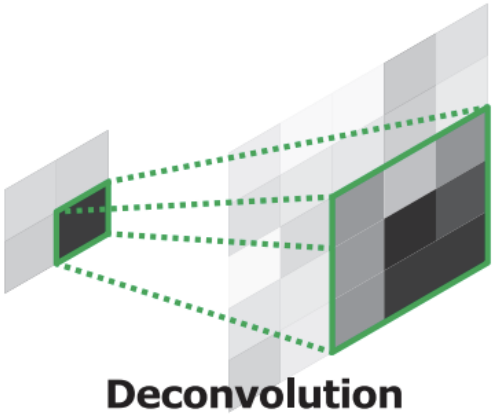




Deconvolution

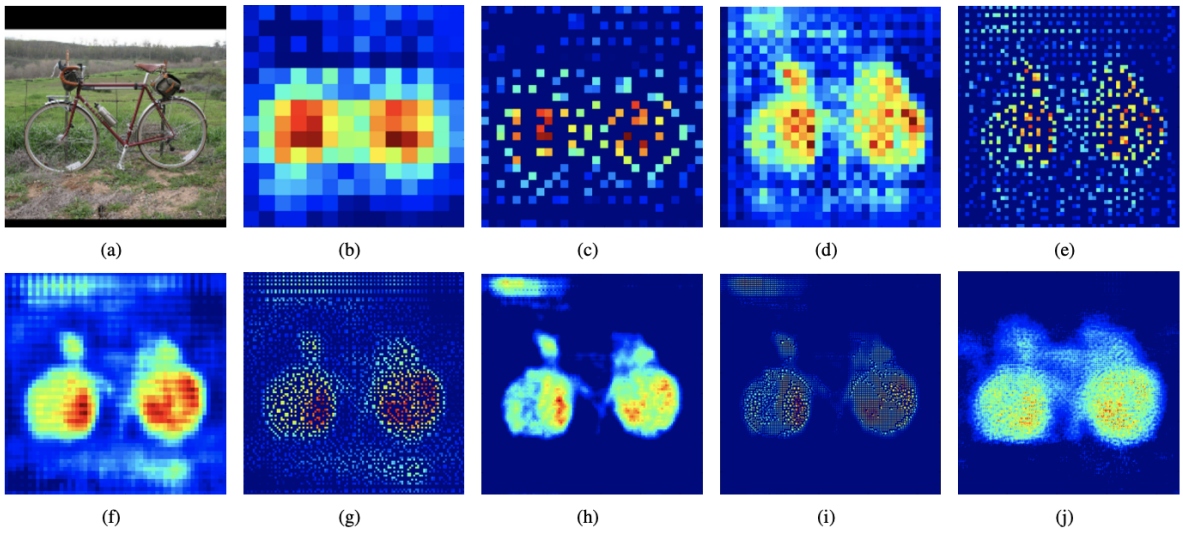
Deconvolution layer들은 Sparse activatiaon을 조밀화합니다.
즉, **deconvolution 연산**을 통해 **dense featuremap**을 만듭니다.





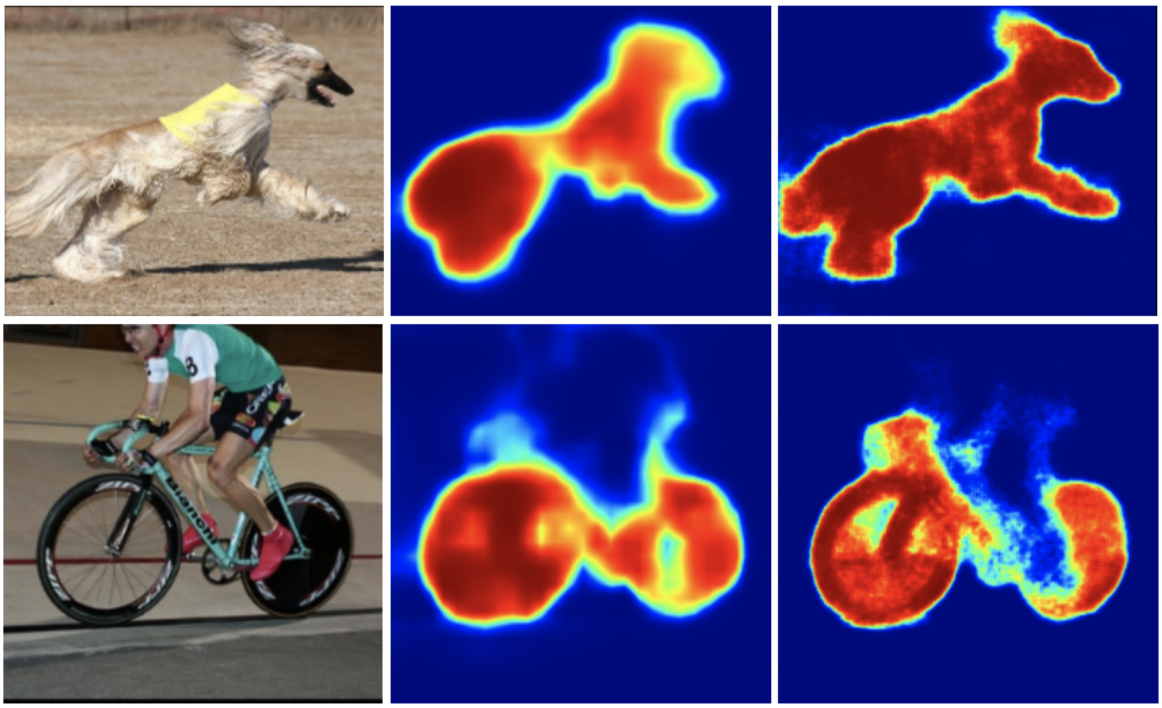
Analysis of Deconvolution Network

Deconvolution Network는 객체 분할의 핵심적인 요소이다. Deconvolution Algorithm은 deep deconvolution network를 사용하여 객체 분할을 시행합니다.



위 그림은 **deconvolution Network의 layer**로부터 **Output을 시각화한 것**입니다.

Unpooling은 원본이미지로부터 **higher activataion을 나타내는 것**들을 포착한 것들을,
다시 원상 복귀시켜놓습니다.



(a) Input image (b) FCN-8s (c) Ours

Figure 5. Comparison of class conditional probability maps from FCN and our network (top: dog, bottom: bicycle).

Deconvolution layer내의 학습된 filter들은 객체들의 형태를 포착하는것이 가능합니다.
또한 background의 propagation이 점차 사라질수록, **target의 activation이 점차 증폭**됨을 알 수 있습니다.

💡 **Unpooling 과 Deconvolution은 엄연히 다릅니다**

Unpooling : 이미지 공간으로의 강한 activataion을 통하여 실제 위치를 추적함으로써,
example과 관련된 예시를들을 알아내는 역할을 합니다.

Deconvolution : 원본 이미지 내의 noisy activataion을 잘 걸러내,
target 과 관련된 부분의 활성화를 증폭시킵니다.
그리하여, **Unpooling 과 Deconvolution이 결합하여, 높은 정확도의 segmentataion map을 생성**합니다.

System Overview

DeConvNet은 instance wise segmentataion으로부터, 제기되었습니다.

Instance-wise segmentation은 이미지를 예측하는데에서, **몇가지 장점을 가지고** 있습니다.

- **다양한 스케일의 Object**를 다룰 수 있습니다.
- **이전의 고정된 receptive fields**와 다르게, **객체의 세부사항을 다양하게 식별하는 것이 가능**합니다.
- 예측에 대한 **탐색 공간과 학습을 하는데 필요한 메모리 요구사항을 감소**시킴으로써,
  학습복잡도를 완화합니다.

Inference & Experiment

전체 class의 Score maps의 Pixel별 최댓값 혹은 평균값으로 대응하는 것은
좋은 결과를 초래합니다.
마지막으로, **최종적인 픽셀별 labeling을 하기 위해** **Fully Connected CRF를 Output map에 적용**합니다.

Table 1. Evaluation results on PASCAL VOC 2012 test set. (Asterisk (∗) denotes the algorithms trained with additional data.)

Method	bkg	areo	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbk	person	plant	sheep	sofa	train	tv	mean
Hypercolumn [10]	88.9	68.4	27.2	68.2	47.6	61.7	76.9	72.1	71.1	24.3	59.3	44.8	62.7	59.4	73.5	70.6	52.0	63.0	38.1	60.0	54.1	59.2
MSRA-CFM [3]	87.7	75.7	26.7	69.5	48.8	65.6	81.0	69.2	73.3	30.0	68.7	51.5	69.1	68.1	71.7	67.5	50.4	66.5	44.4	58.9	53.5	61.8
FCN8s [17]	91.2	76.8	34.2	68.9	49.4	60.3	75.3	74.7	77.6	21.4	62.5	46.8	71.8	63.9	76.5	73.9	45.2	72.4	37.4	70.9	55.1	62.2
TTI-Zoomout-16 [18]	89.8	81.9	35.1	78.2	57.4	56.5	80.5	74.0	79.8	22.4	69.6	53.7	74.0	76.0	76.6	68.8	44.3	70.2	40.2	68.9	55.3	64.4
DeepLab-CRF [1]	93.1	84.4	**54.5**	81.5	63.6	65.9	85.1	79.1	83.4	30.7	74.1	59.8	79.0	76.1	83.2	80.8	**59.7**	82.2	50.4	73.1	63.7	71.6
DeconvNet	92.7	85.9	42.6	78.9	62.5	66.6	87.4	77.8	79.5	26.3	73.4	60.2	70.8	76.5	79.6	77.7	58.2	77.4	52.9	75.2	59.8	69.6
DeconvNet+CRF	92.9	87.8	41.9	80.6	63.9	67.3	**88.1**	78.4	81.3	25.9	73.7	61.2	72.0	77.0	79.9	78.7	59.5	78.3	**55.0**	75.2	61.5	70.5
EDeconvNet	92.9	88.4	39.7	79.0	63.0	67.7	87.1	**81.5**	84.4	27.8	76.1	61.2	78.0	79.3	83.1	79.3	58.0	82.5	52.3	80.1	64.0	71.7
EDeconvNet+CRF	93.1	**89.9**	39.3	79.7	63.9	**68.2**	87.4	81.2	**86.1**	28.5	**77.0**	62.0	79.0	**80.3**	**83.6**	80.2	58.8	**83.4**	54.3	**80.7**	65.0	**72.5**
* WSSL [19]	93.2	85.3	36.2	**84.8**	61.2	67.5	84.7	81.4	81.0	**30.8**	73.8	53.8	77.5	76.5	82.3	**81.6**	56.3	78.9	52.3	76.6	63.3	70.4
* BoxSup [2]	**93.6**	86.4	35.5	79.7	**65.2**	65.2	84.3	78.5	83.7	30.5	76.2	**62.6**	**79.3**	76.1	82.1	81.3	57.0	78.2	**55.0**	72.5	**68.1**	71.0

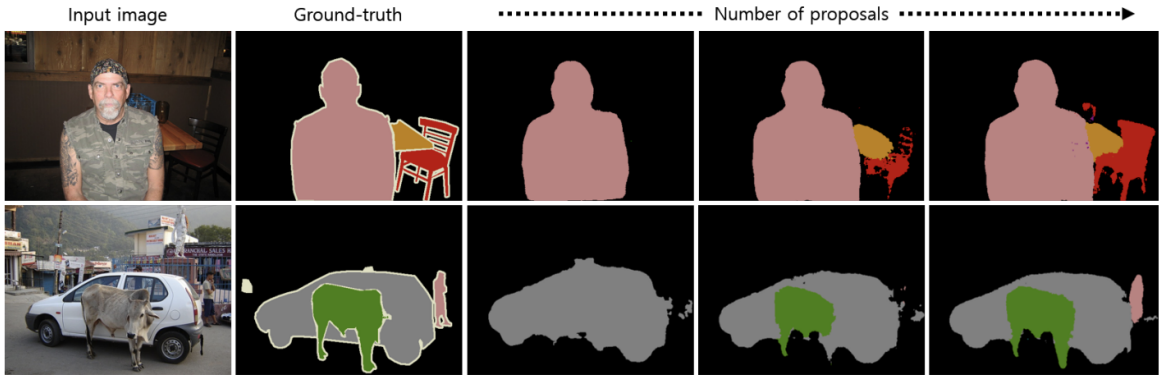


Figure 6. Benefit of instance-wise prediction. We aggregate the proposals in a decreasing order of their sizes. The algorithm identifies finer object structures through iterations by handling multi-scale objects effectively.

Conclusion

⭐ 해당 논문은 **Deconvolution network**로 학습되는 새로운 **Segmentataion Algorithm**을 **제안**합니다.

Instance-wise-prediction에 기반을 하였기에, **고정된 수용 영역의 크기를 제안을 제거**함으로써, **다양한 형태의 객체를 다루는 것이 가능**합니다.

FCN에 기반을 방법과 Deconvolution의 출력과 앙상블을 한다면, 더 좋은 성능을 낼 것이라고 기대합니다.

Reference

Learning Deconvolution Network for Semantic Segmentation

We propose a novel semantic segmentation algorithm by learning a 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

https://arxiv.org/abs/1505.04366

[논문 읽기] DeConvNet(2015) 리뷰, Learning Deconvolution Network for Semantic Segmentation

이번에 읽어볼 논문은 DeepConvNet, 'Learning Deconvolution Network for Semantic Segmentation' 입니다. DeepConvNet은 Convolution network와 Deconvolution network, 두 파트로 구성되어 있습니다. Convolution network로 feature을 추출하고, Deconvolution network로 feature에 대한 object segmentation을 생성합니다. 이

https://deep-learning-study.tistory.com/565