Week 6: Semantic image segmentation

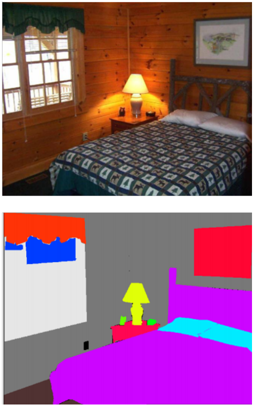
Instructor: Ruixuan Wang
wangruix5@mail.sysu.edu.cn

School of Data and Computer Science
Sun Yat-Sen University

4 April, 2019

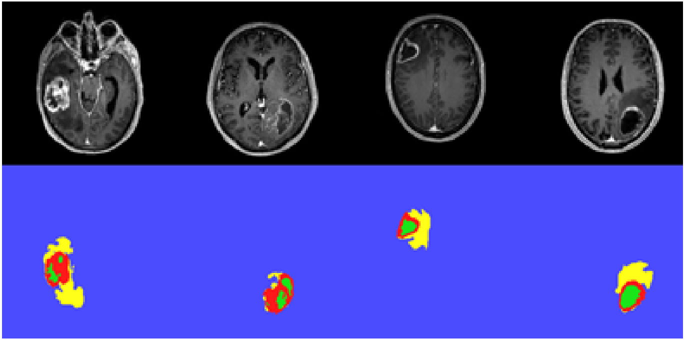
What and why

- Semantic segmentation: classifying each pixel
- Applications: self-driving, smart home



What and why

- More applications: intelligent medical analysis, etc.

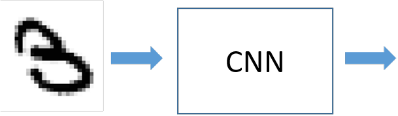


CNNs for semantic segmentation

How are CNNs applied to semantic segmentation?

Traditional CNN-based classifiers

- Classification of the *whole image*



Probability	Class
0.005	0
0.002	1
0.001	2
0.93	3
0.001	4
0.005	5
0.003	6
0.001	7
0.05	8
0.002	9

- How can CNNs output classification result for each *pixel*?

Traditional sliding window idea

- Classify each pixel by classifying the surrounding patch

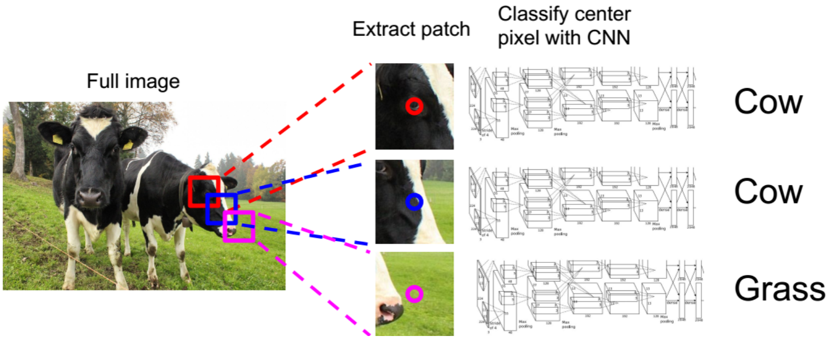
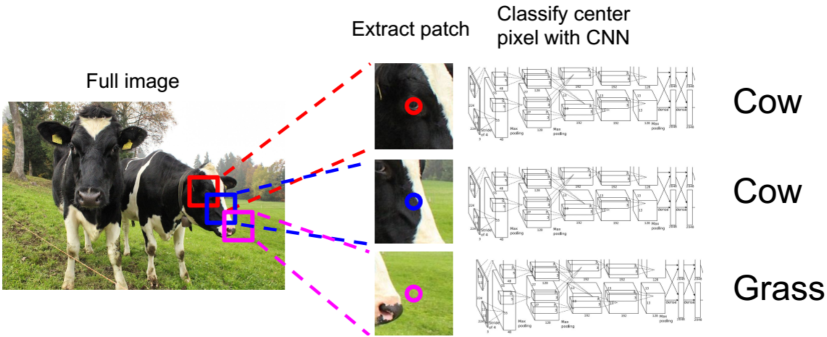


Figure from Stanford CS231n 2017 lecture 11

Traditional sliding window idea

- Classify each pixel by classifying the surrounding patch



- Issue: very inefficient! 10000 patches for a 100×100 image

Figure from Stanford CS231n 2017 lecture 11

Segmentation with fully convolutional idea

- Output has same size as input, but with C (# class) channels

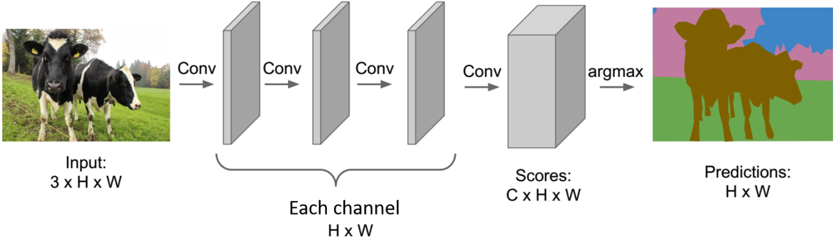
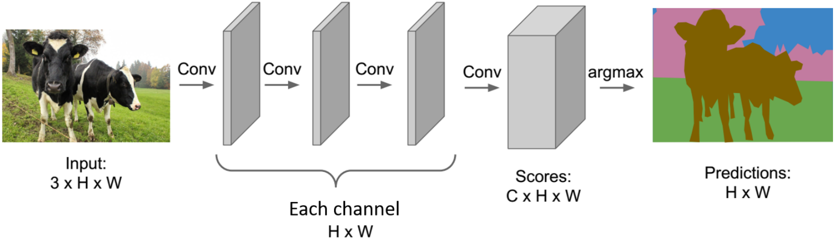


Figure from Stanford CS231n 2017 lecture 11

Segmentation with fully convolutional idea

- Output has same size as input, but with C (# class) channels

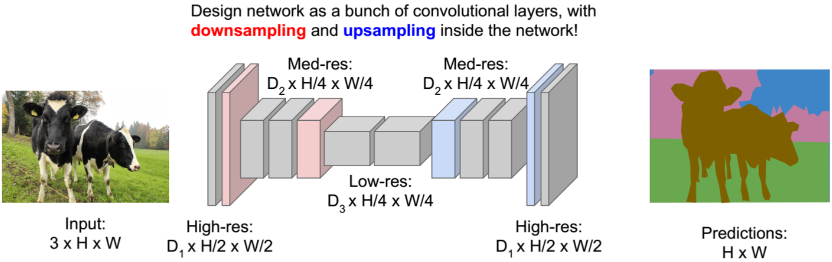


- Issue 1: expensive computation and large memory
- Issue 2: difficult to capture larger receptive field

Figure from Stanford CS231n 2017 lecture 11

Fully convolutional network (FCN)

- Encoder-decoder framework: reduce size of feature maps, then increase to original size

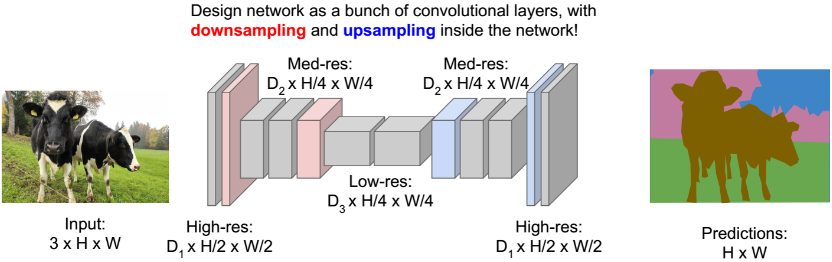


- But how to realize 'upsampling'?

Figure from Stanford CS231n 2017 lecture 11

Fully convolutional network (FCN)

- Encoder-decoder framework: reduce size of feature maps, then increase to original size



- But how to realize 'upsampling'?

Figure from Stanford CS231n 2017 lecture 11

SegNet: max unpooling as upsampling

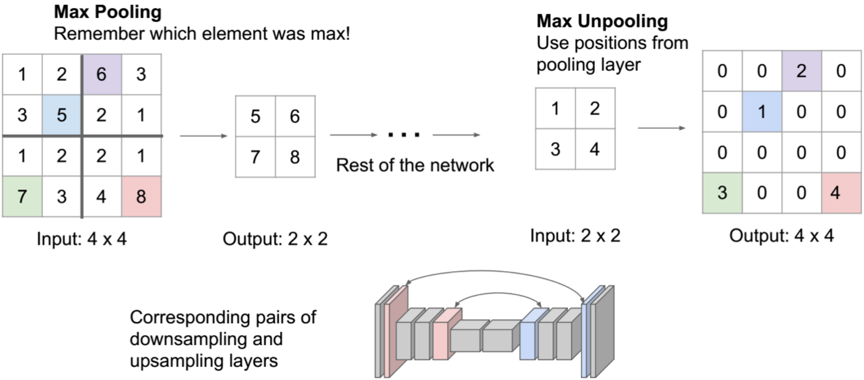


Figure from Stanford CS231n 2017 lecture 11;
Badrinarayanan, Kendall, Cipolla, "SegNet: a deep convolutional encoder-decoder architecture for image segmentation", arXiv, 2015

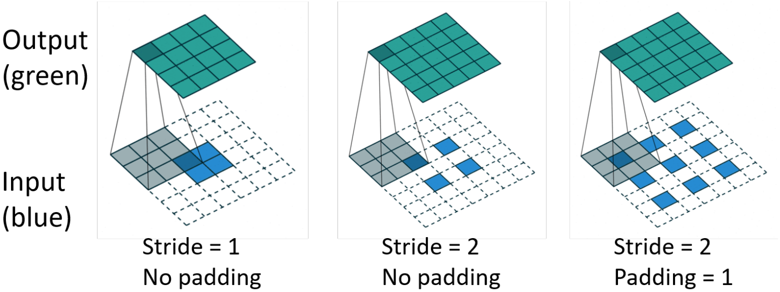
Transpose convolution: learnable upsampling

- Transpose convolution (‘deconvolution’): opposite convolution
- Stride results in upsampling rather than downsampling
- Padding means removing boundary from the output rather than addding to the input

Figures from https://github.com/vdumoulin/conv_arithmetic

Transpose convolution: learnable upsampling

- Transpose convolution ('deconvolution'): opposite convolution
- Stride results in upsampling rather than downsampling
- Padding means removing boundary from the output rather than addding to the input

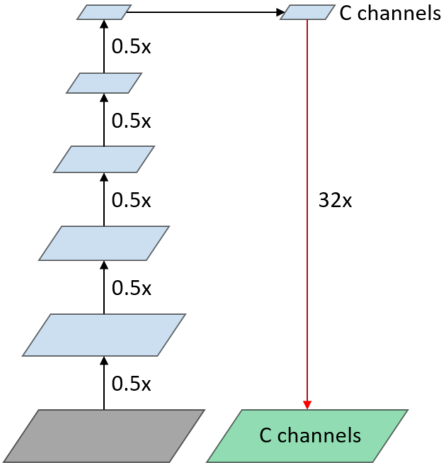


Figures from https://github.com/vdumoulin/conv_arithmetic

FCNs with transpose convolution

FCN-32s

- one transpose convolutional layer (line in red)
- stride=32
- kernel size 64×64
- C: number of segmentation classes

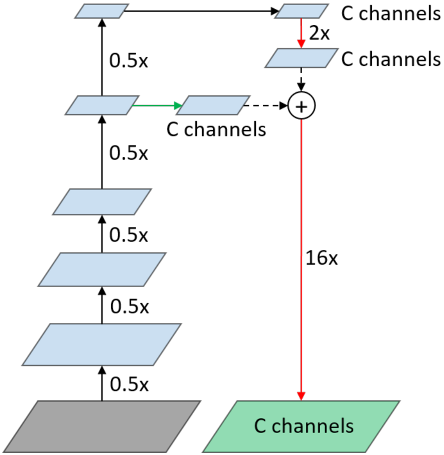


Long, Shelhamer, Darrell, "Fully convolutional networks for semantic segmentation", CVPR 2015

FCNs with transpose convolution (cont')

FCN-16s

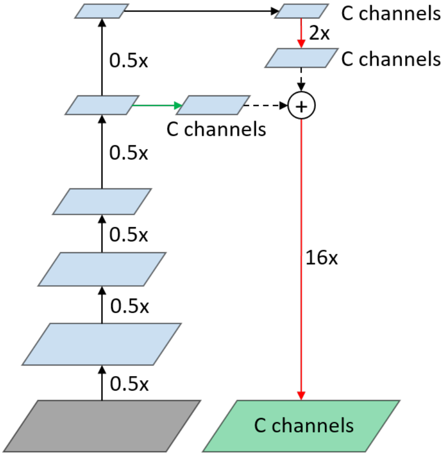
- two transpose conv layers (stride=2, size 4×4; stride=16, size 32×32)
- green line: 1×1 convolution
- upsampled feature map *plus* lower-layer feature map
- dashed lines: identity mapping



FCNs with transpose convolution (cont')

FCN-16s

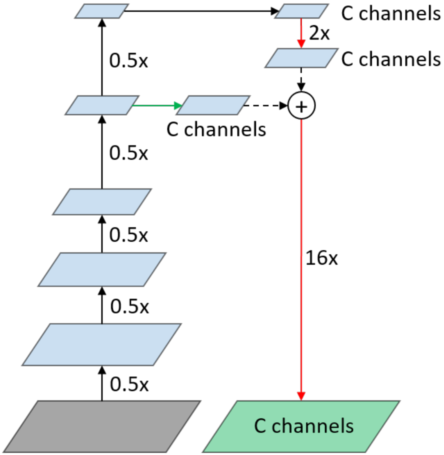
- two transpose conv layers (stride=2, size 4×4; stride=16, size 32×32)
- green line: 1×1 convolution
- upsampled feature map *plus* lower-layer feature map
- dashed lines: identity mapping



FCNs with transpose convolution (cont')

FCN-16s

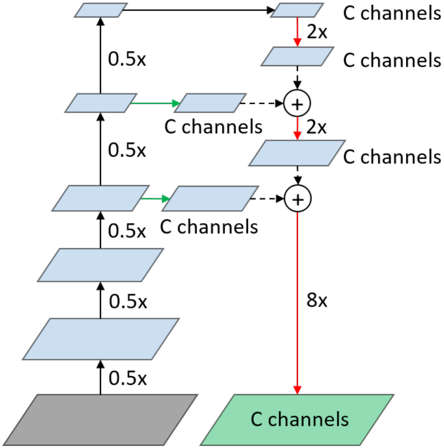
- two transpose conv layers (stride=2, size 4×4; stride=16, size 32×32)
- green line: 1×1 convolution
- upsampled feature map *plus* lower-layer feature map
- dashed lines: identity mapping



FCNs with transpose convolution (cont')

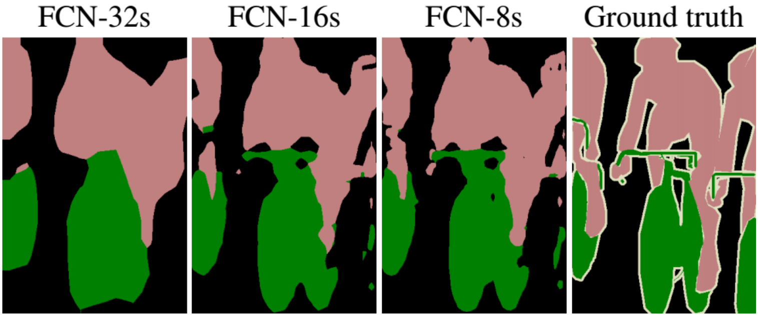
FCN-8s

- three transpose conv layers
- two with stride=2, kernel size 4×4
- one with stride=8, kernel size 16×16



FCNs with transpose convolution (cont')

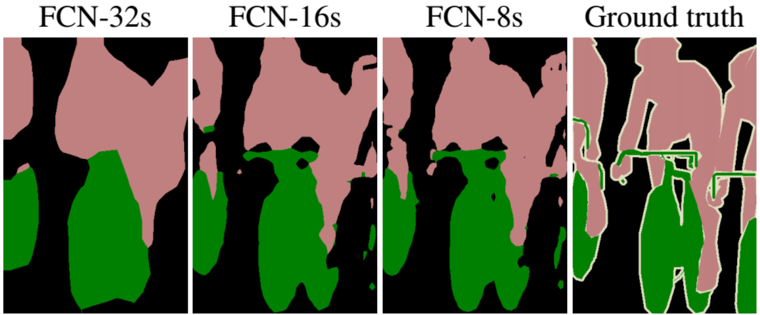
- Training FCN as for CNN classifier



- Better if more lower encoding layers involved in upsampling
- So, why not use further more encoding layers during upsampling?

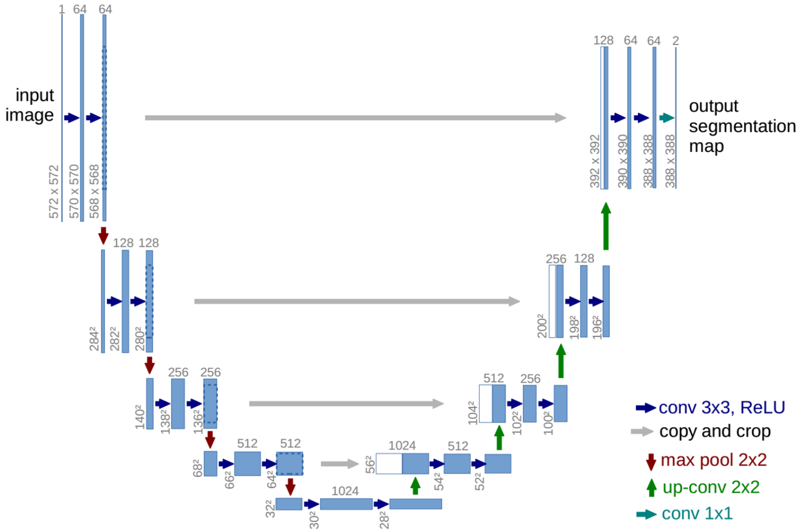
FCNs with transpose convolution (cont')

- Training FCN as for CNN classifier



- Better if more lower encoding layers involved in upsampling
- So, why not use further more encoding layers during upsampling?

U-net: using more encoding layers during upsampling



U-net (cont')

Different from above FCN model:

- Each upsampling layer contains more kernels (than class #)
- Encoding block connects to corresponding decoding block
- Upsampled feature maps *concatenate* with corresponding encoding feature maps

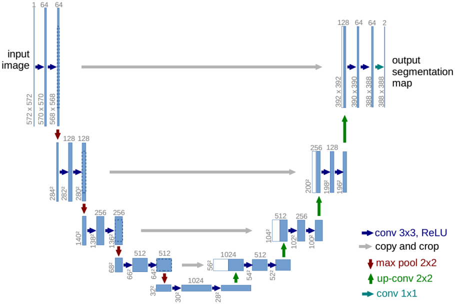


Figure from Ronneberger, Fischer, Brox, "U-Net: convolutional networks for biomedical image segmentation", MICCAI, 2015

U-net (cont')

Different from above FCN model:

- Each upsampling layer contains more kernels (than class #)
- Encoding block connects to corresponding decoding block
- Upsampled feature maps *concatenate* with corresponding encoding feature maps

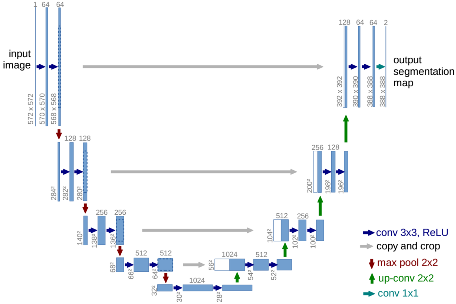


Figure from Ronneberger, Fischer, Brox, "U-Net: convolutional networks for biomedical image segmentation", MICCAI, 2015

U-net (cont')

Different from above FCN model:

- Each upsampling layer contains more kernels (than class #)
- Encoding block connects to corresponding decoding block
- Upsampled feature maps *concatenate* with corresponding encoding feature maps

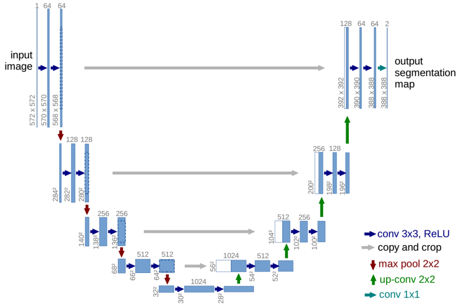
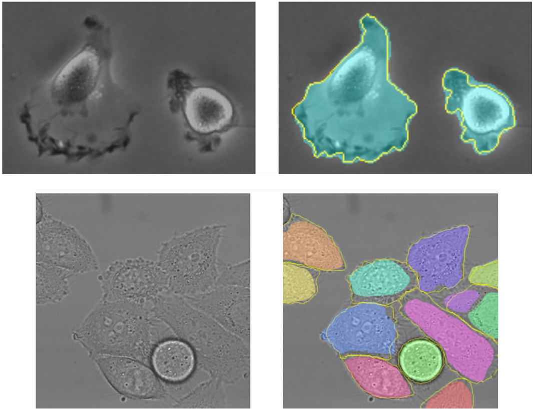


Figure from Ronneberger, Fischer, Brox, "U-Net: convolutional networks for biomedical image segmentation", MICCAI, 2015

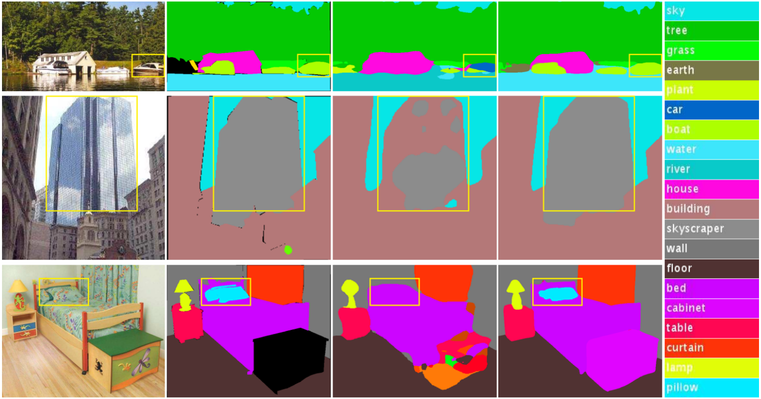
U-net (cont')

- U-net has been used for various medical image segmentations, with good performance; yellow borders for ground-truth



However ...

- FCNs did not effectively use global scene category clues



(a) Image (b) Ground Truth (c) FCN (d) PSPNet (e) ColorMap

- Errors relate to contextual relationship and global information

Pyramid scene parsing network (PSPNet)

PSPNet: use pyramid pooling to provide global context infor

- Encoder: dilated conv, feature map with 1/8 size of input
- Then pyramid pooling: pooling at multiple scales
  Spatial size of pooling ouput (not pooling window) is pre-set
- Concatenate upsampling result with original feature map

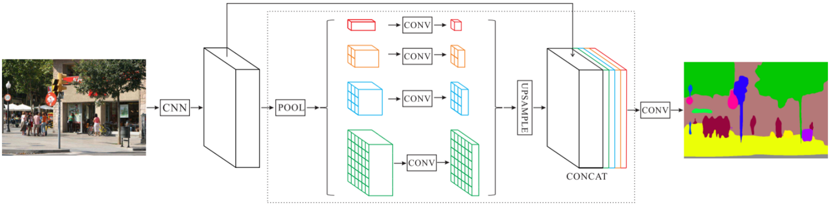
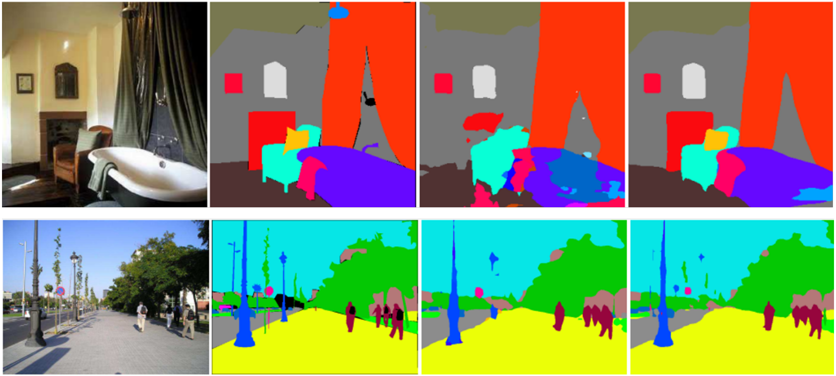


Figure from Zhao, Shi, Qi, Wang, Jia, "Pyramid scene parsing network", CVPR, 2017

PSPNet (cont')

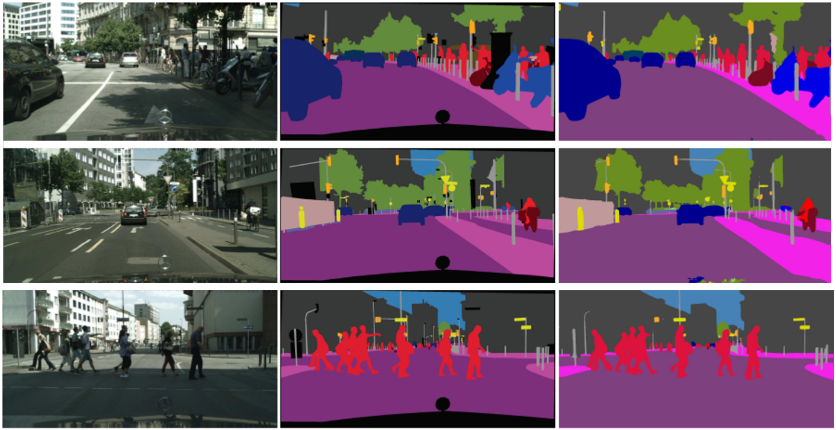
- Average segmentation results from multiple scales of input
- PSPNet produces more accurate and detailed results



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

PSPNet (cont')

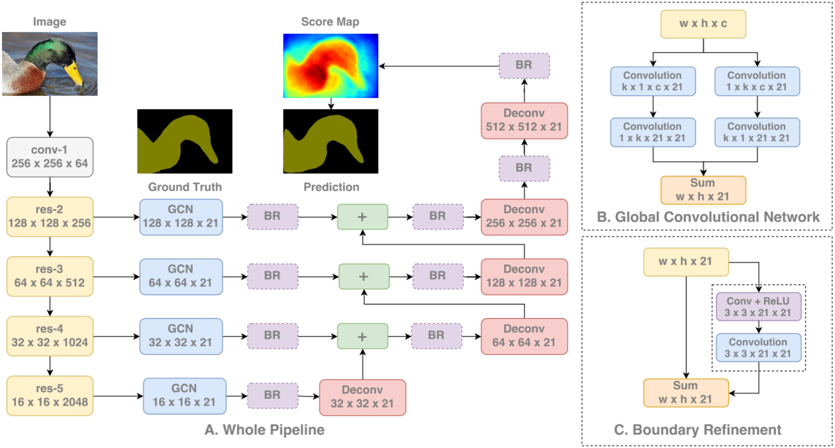
- PSPNet won multiple challenges/contests in 2016



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

Global convolutional network (GCN): large kernel matters

- Another way to capture global context: large kernels
- Kernel size in GCN: $1 \times K$ and $K \times 1$, e.g., $K = 17$



GCN (cont')

- Like FCN-8s: connect encoder and decoder layers
- Not like FCN-8s: add GCN and 'boundary refinement' (BR) layers between connections, and across de-conv layers
- BR layers help segment more accurate region boundaries
- Large kernels help handle large-scale object regions

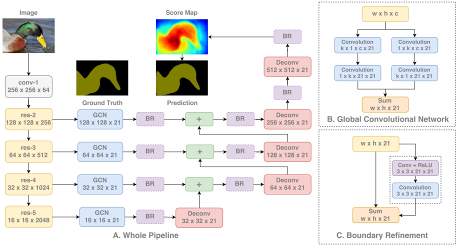


Figure from Peng, Zhang, Yu, Luo, Sun, "Large kernel matters - improve semantic segmentation by global convolutional network", CVPR, 2017

GCN (cont')

- Like FCN-8s: connect encoder and decoder layers
- Not like FCN-8s: add GCN and 'boundary refinement' (BR) layers between connections, and across de-conv layers
- BR layers help segment more accurate region boundaries
- Large kernels help handle large-scale object regions

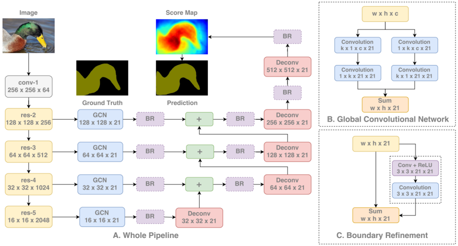


Figure from Peng, Zhang, Yu, Luo, Sun, "Large kernel matters - improve semantic segmentation by global convolutional network", CVPR, 2017

GCN (cont')

- Intersection-over-union (IoU) on PASCAL VOC 2012
- Performance is better with larger kernel size

k	base	3	5	7	9	11	13	15
Score	69.0	70.1	71.1	72.8	73.4	73.7	74.0	74.5

GCN (cont')

- Intersection-over-union (IoU) on PASCAL VOC 2012
- Performance is better with larger kernel size

k	base	3	5	7	9	11	13	15
Score	69.0	70.1	71.1	72.8	73.4	73.7	74.0	74.5

- GCN is better than equivalent stacked multi-layer small kernels

k	3	5	7	9	11
Score (GCN)	70.1	71.1	72.8	73.4	73.7
Score (Stack)	69.8	71.8	71.3	69.5	67.5

GCN (cont')

- Intersection-over-union (IoU) on PASCAL VOC 2012
- Performance is better with larger kernel size

k	base	3	5	7	9	11	13	15
Score	69.0	70.1	71.1	72.8	73.4	73.7	74.0	74.5

- GCN is better than equivalent stacked multi-layer small kernels

k	3	5	7	9	11
Score (GCN)	70.1	71.1	72.8	73.4	73.7
Score (Stack)	69.8	71.8	71.3	69.5	67.5

- BR improve boundary region segmentation (1st column)

Model	Boundary (acc.)	Internal (acc.)	Overall (IoU)
Baseline	71.3	93.9	69.0
GCN	71.5	95.0	74.5
GCN + BR	73.4	95.1	74.7

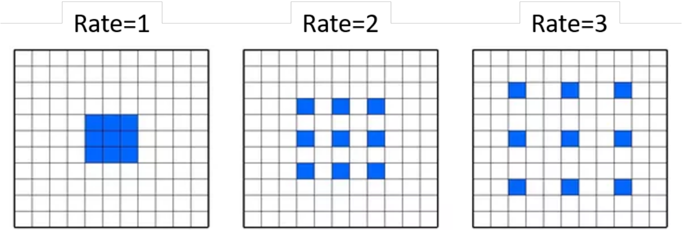
So far ...

We have seen several encoder-decoder models!

Is only encoder-decoder effective for segmentation?

Dilated (atrous) convolution: upsampling filters

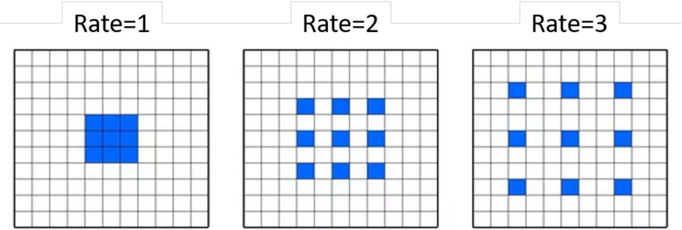
- Feature map size unchanged across layers
- Capture large-scale features on high-resolution maps



- But more memory assumption due to large-size feature maps

Dilated (atrous) convolution: upsampling filters

- Feature map size unchanged across layers
- Capture large-scale features on high-resolution maps



- But more memory assumption due to large-size feature maps

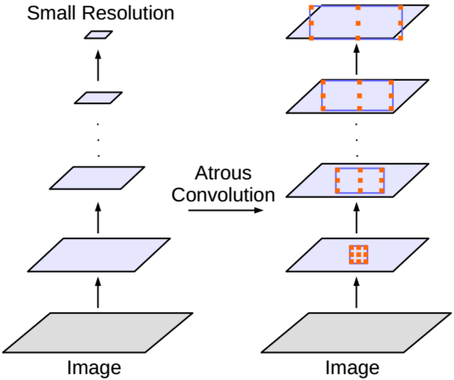
DeepLab

1: Conv+pooling to reduce feature map size to 1/16 of input

2: Followed by multilayer dilated convolutions

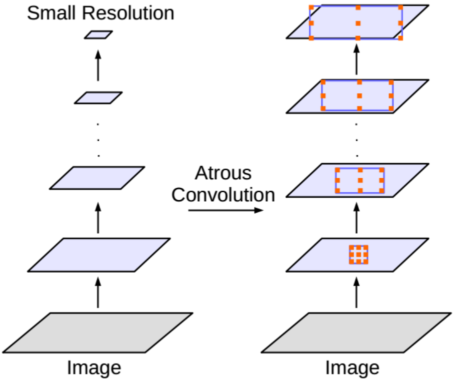
3: With bilinear interpolation as upsampling to input size

4: Post-processing with fully connected conditional random field



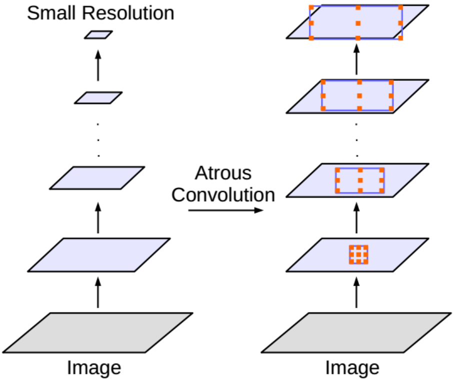
DeepLab

1: Conv+pooling to reduce feature map size to 1/16 of input
2: Followed by multilayer dilated convolutions
3: With bilinear interpolation as upsampling to input size
4: Post-processing with fully connected conditional random field



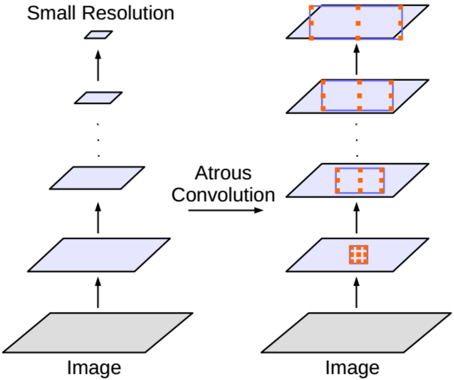
DeepLab

1: Conv+pooling to reduce feature map size to 1/16 of input
2: Followed by multilayer dilated convolutions
3: With bilinear interpolation as upsampling to input size
4: Post-processing with fully connected conditional random field



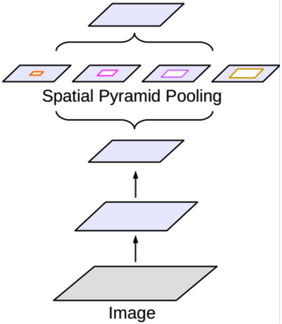
DeepLab

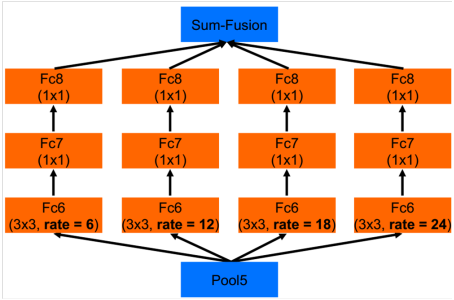
1: Conv+pooling to reduce feature map size to 1/16 of input
2: Followed by multilayer dilated convolutions
3: With bilinear interpolation as upsampling to input size
4: Post-processing with fully connected conditional random field



DeepLab V2

- Fuse multiple paths of dilated convolutions of different rates



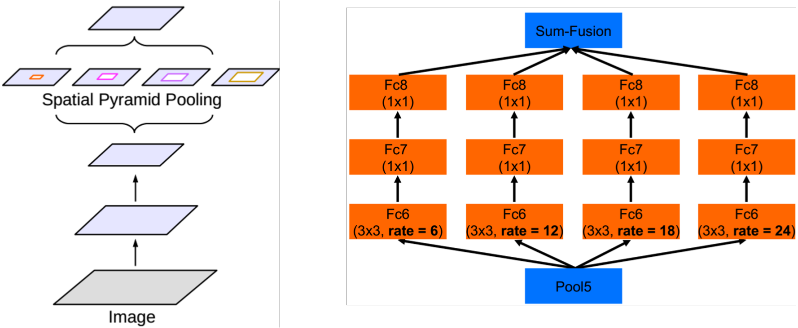


- It can capture objects and context at multiple scales
- Tricks: multi-scale inputs, ResNet, augmentaiton, pre-training

Figure from Chen, Papandreou, Kokkinos, Murphy, Yuille, "DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs", arXiv, 2017

DeepLab V2

- Fuse multiple paths of dilated convolutions of different rates

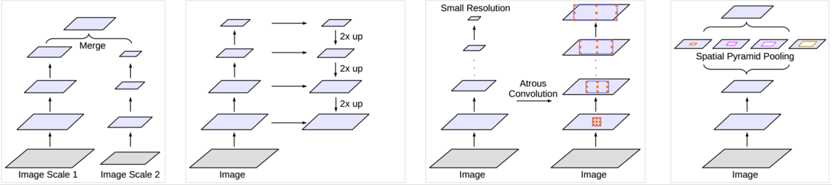


- It can capture objects and context at multiple scales
- Tricks: multi-scale inputs, ResNet, augmentaiton, pre-training

Figure from Chen, Papandreou, Kokkinos, Murphy, Yuille, "DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs", arXiv, 2017

So far ...

- Some of mentioned model structures above

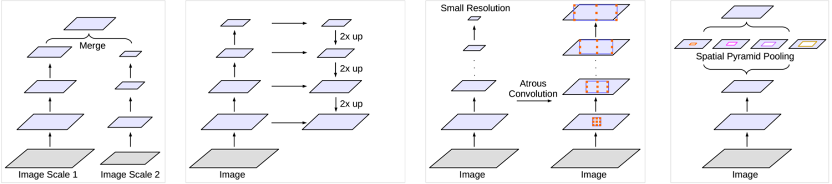


Multi-scale input | Encoder-decoder (e.g., U-net) | DeepLab V1 | DeepLab V2

Figures here and next slide from Chen, Papandreou, Schroff, Adam, "Rethinking atrous convolution for semantic image segmentation", arXiv, 2017

So far ...

- Some of mentioned model structures above



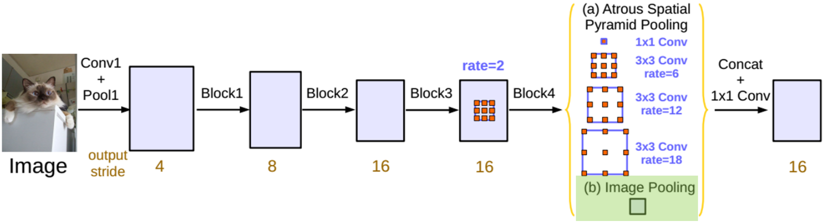
Multi-scale input | Encoder-decoder (e.g., U-net) | DeepLab V1 | DeepLab V2

- Issue of DeepLab V2: for large dilation, sampling at map boundary is not useful, i.e., image-scale features not well captured

Figures here and next slide from Chen, Papandreou, Schroff, Adam, "Rethinking atrous convolution for semantic image segmentation", arXiv, 2017

DeepLab V3

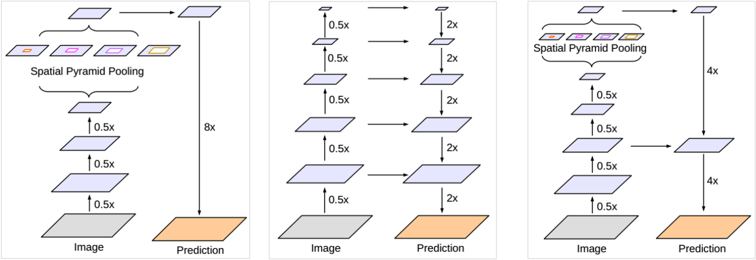
- Capture image-scale features with global pooling (green)



- More tricks: batch normalization, bootstrapping hard images

DeepLab V3+

- DeepLab V3 capture multi-scale context infor with dilated convolutions at multiple rates
- Encoder-decoder models capture sharper object boundaries by gradually recovering spatial information
- So why not combine the two worlds?

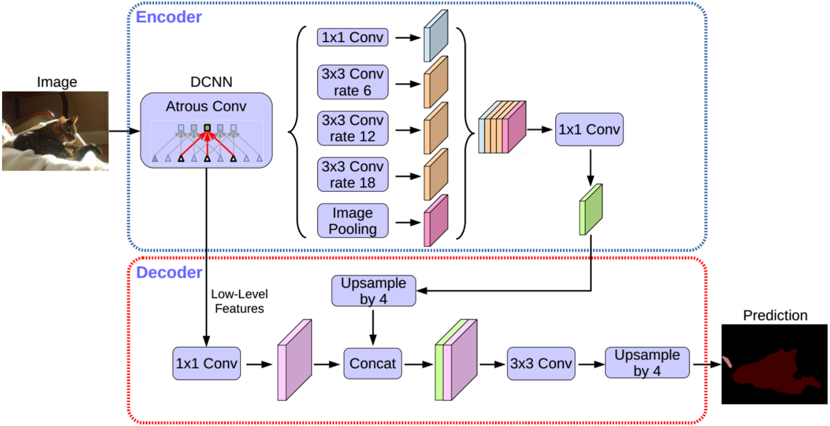


DeepLab V2/V3 Encoder-decoder DeepLab V3+

Figures here and next few slides from Chen, Zhu, Papandreou, Schroff, Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation", arXiv, 2018

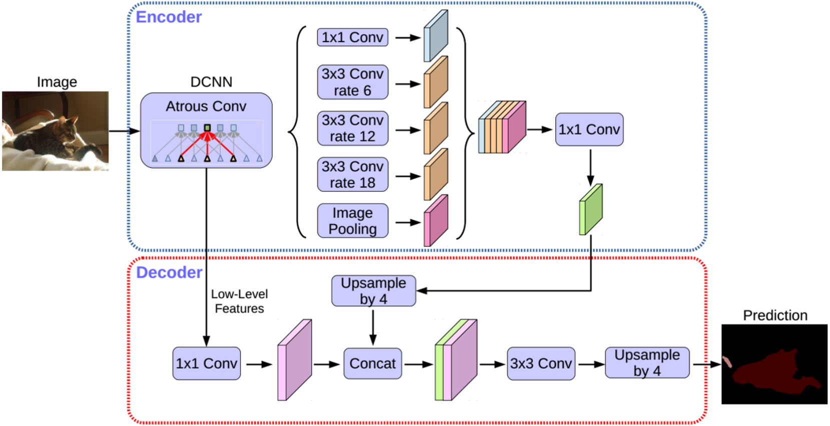
DeepLab V3+

- DeepLab V3+: adding decoder to DeepLab V3
- Feature maps are *concatenated* in decoder



DeepLab V3+

- DeepLab V3+: adding decoder to DeepLab V3
- Feature maps are *concatenated* in decoder



State-of-the-art segmentation results

- DeepLab V3+ is the best so far, on PASCAL VOC 2012

Method	mIOU
Deep Layer Cascade (LC) [42]	82.7
TuSimple [75]	83.1
Large_Kernel_Matters [57]	83.6
Multipath-RefineNet [43]	84.2
ResNet-38_MS_COCO [77]	84.9
PSPNet [81]	85.4
IDW-CNN [73]	86.3
CASIA_IVA_SDN [20]	86.6
DIS [50]	86.8
DeepLabv3 [10]	85.7
DeepLabv3-JFT [10]	86.9
DeepLabv3+ (Xception)	87.8
DeepLabv3+ (Xception-JFT)	89.0

Then...

What could be the next breakthrough?

MS-D: Mixed-scale dilate conv with dense connections

- Dense connection, concatenation of feature maps
- Unique dilation rate for each channel per layer
- Number of feature channels per layer can be just 2 or 1

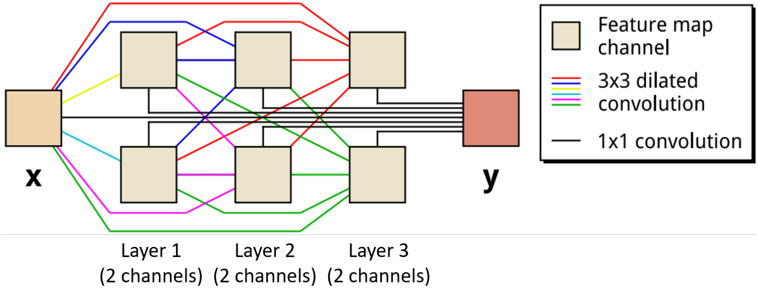


Figure from Pelt, Sethian, "A mixed-scale dense convolutional neural network for image analysis", PNAS, 2017

MS-D: Mixed-scale dilate conv with dense connections

- Dense connection, concatenation of feature maps
- Unique dilation rate for each channel per layer
- Number of feature channels per layer can be just 2 or 1

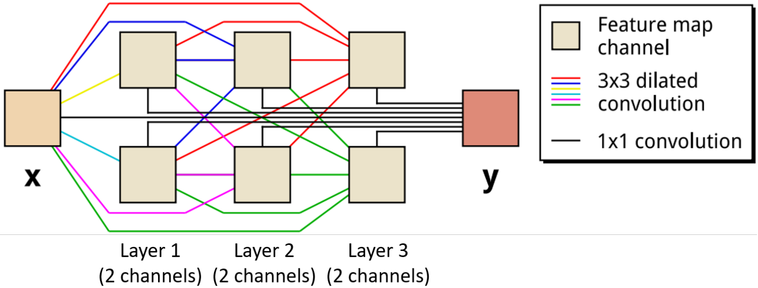


Figure from Pelt, Sethian, "A mixed-scale dense convolutional neural network for image analysis", PNAS, 2017

MS-D: Mixed-scale dilate conv with dense connections

- Dense connection, concatenation of feature maps
- Unique dilation rate for each channel per layer
- Number of feature channels per layer can be just 2 or 1

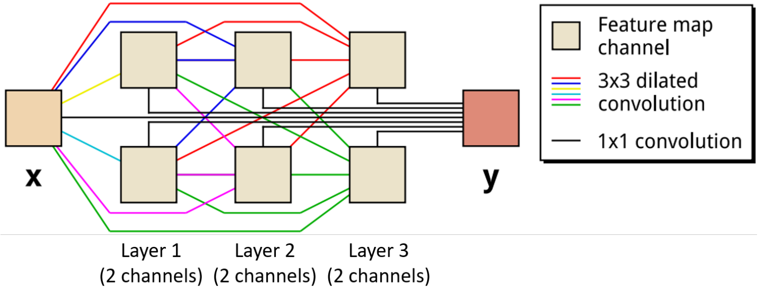


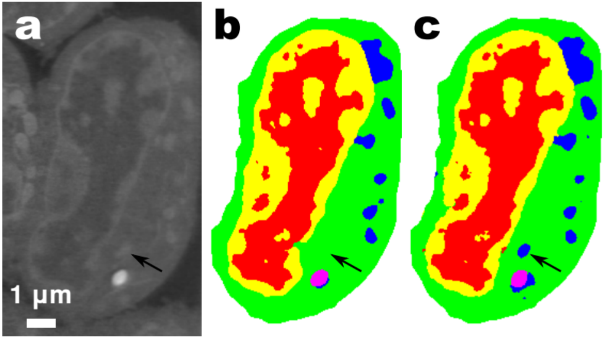
Figure from Pelt, Sethian, "A mixed-scale dense convolutional neural network for image analysis", PNAS, 2017

MS-D network (cont')

- Fewer parameters: few kernels + feature reuse
- Therefore, work well with small training dataset

MS-D network (cont')

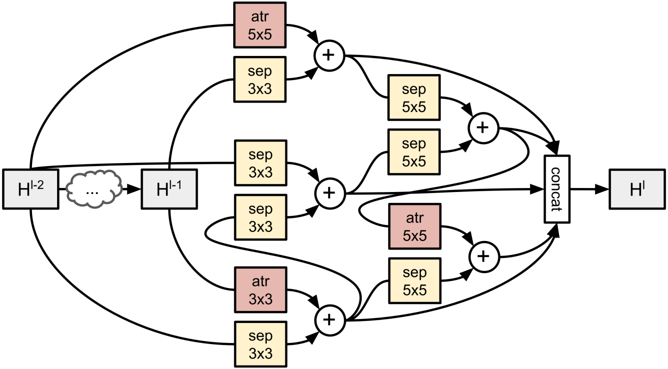
- Fewer parameters: few kernels + feature reuse
- Therefore, work well with small training dataset
- Cell segmentation, with 100 layers, 1 channel per layer



- figure (b): ground truth; (c) segmentation by model

AutoML for segmentation

- Automatically search for model architectures
- Same performance as DeepLab V3+, but faster



Note: 'atr': atrous conv; 'sep': depthwise-separable conv.

Figure from Liu, Chen, Schroff, Adam, Hua, Yuille, Fei-Fei, "Auto-DeepLab: Hierarchical Neural Architecture Search for Semantic Image Segmentation", arXiv, 2019

Summary

- FCNs are better than others for semantic segmentation
- Encoder-decoder FCN variations were developed
- DeepLab V3+ combine dilated conv with decoder
- New trends:dense connection, multi-scale, autoML, etc.

Further reading:

- Luo et al., Deep dual learning for semantic image segmentation, ICCV, 2017
- Liu et al., Auto-DeepLab: Hierarchical Neural Architecture Search for Semantic Image Segmentation', arXiv, 2019