AL5.A.01 - Techniques IA

Deep learning-based semantic segmentation

Minh-Tan Pham

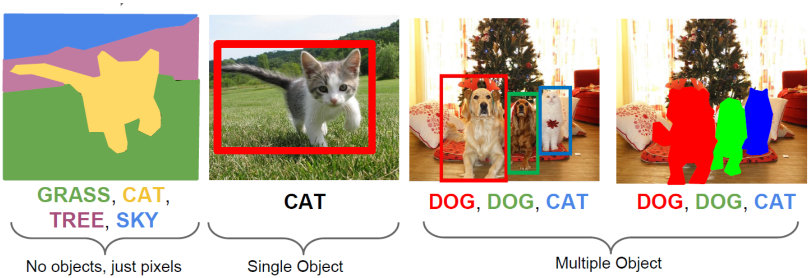
BUT3-INFO, IUT Vannes, Université Bretagne-Sud

minh-tan.pham@univ-ubs.fr

Table of contents

Table of contents

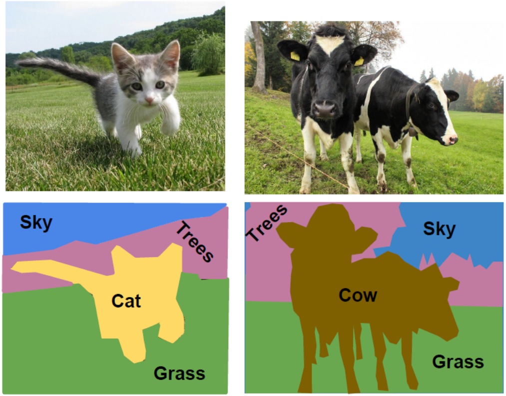
Reminder of main vision tasks: classification/recognition, classification+localization, object detection, semantic segmentation and instance segmentation [1]



[1] Image credit: `https://cs231n.github.io/`, Lecture 11, 2017

Today's focus: Semantic segmentation

- Predict a label for each pixel from the image (= pixel-wise classification)
- Not differentiate the instances, only the semantic [2]



[2] Image credit: `https://cs231n.github.io/`, Lecture 11, 2017

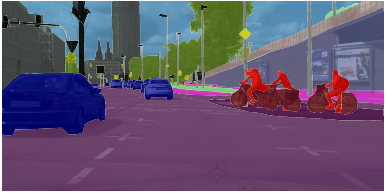
Today's focus: Semantic segmentation

- Input: image of size $H \times W \times 3$ (for RGB images as usual)
- Output: segmentation map of size $H \times W \times 1$ (with class label at each pixel position) [2]

Input — Semantic Labels

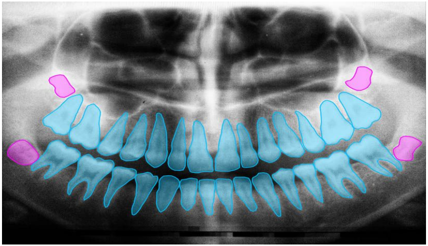
[2] Image credit: `https://www.jeremyjordan.me/semantic-segmentation/#loss`

- **Autonomous driving.** Popular datasets: Cityscapes (30 classes, 5k images), KITTI (11, 400), Mapillary Vistas (66, 25k) , CamVid (32, 700), etc.



[3]Image credit: `https://keymakr.com/blog/semantic-segmentation-uses-and-applications/`

- **Medical imaging.** Many sub-tasks: tumor segmentation, retinal vessel extraction, MRI-based various disease scanning, etc.



[3] Image credit: `https://keymakr.com/blog/semantic-segmentation-uses-and-applications/`

Semantic segmentation Applications[3]

- **Scene understanding (for image captioning/description).** Indoor and outdoor; street-view, panorama, aerial views. Popular datasets: ADE20K, SceneNet, UAVid, TorontoCity, etc.

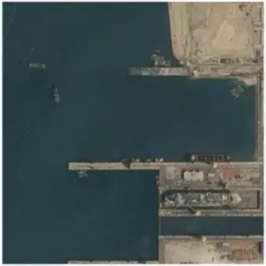
[3] Image credit: `https://keymakr.com/blog/semantic-segmentation-uses-and-applications/`

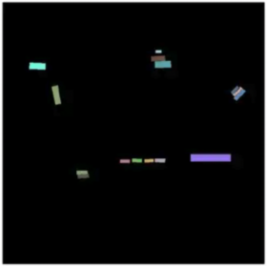
Semantic segmentation Evaluation metrics

Metrics for evaluation

- Pixel accuracy: overall accuracy, average (class) accuracy, confusion matrix
- Mean IOU (Intersection-over-Union)
- For binary maps: Recall, Precision, F1-score

Remarks: using pixel accuracy is sometime dangerous: class imbalance, high portion of background, etc. [4]

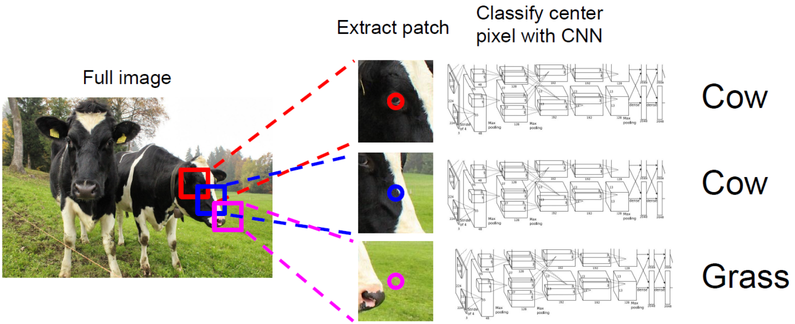




[4] Image credit: `https://medium.com/towards-data-science/image-segmentation-kaggle-experience-9a41cb8924f0`

Table of contents

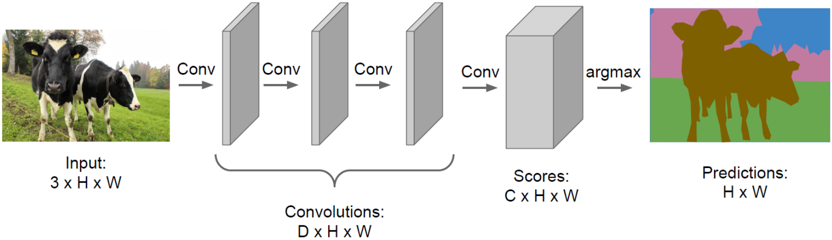
Sliding window: very very slow (inefficient) without reusing shared features from the ConvNet



[5] Image credit: `https://cs231n.github.io/`, Lecture 11, 2017

Segmentation networks Very first idea[5]

Fully convolutional: to fully share ConvNet features, design a network with only convolutional layers (no pooling, no fully-connected)

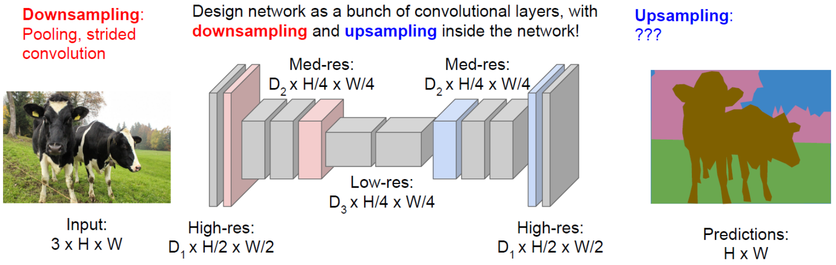


Problem: convolutions at the original image resolution are very expensive !

[5]Image credit: `https://cs231n.github.io/`, Lecture 11, 2017

Fully convolutional with downsampling and upsampling:

- allow to share features at multiple scales (low-level to high level)
- allow to learn deeper features (with more layers) without exploding the computational cost



[5] Image credit: `https://cs231n.github.io/`, Lecture 11, 2017

Segmentation networks

Read & discuss:

- An overview of semantic image segmentation, Jeremy Jordan
  `https://www.jeremyjordan.me/semantic-segmentation/`

Questions for discussion:

- How to upsample ?
- What are the main architectures ?
- Which loss functions that could be used ?

Nearest neighbor or “Bed of nails"

- very simple method
- no parameters
- not efficient

Nearest Neighbor

Input: 2 x 2

1	2
3	4

Output: 4 x 4

1	1	2	2
1	1	2	2
3	3	4	4
3	3	4	4

“Bed of Nails”

Input: 2 x 2

1	2
3	4

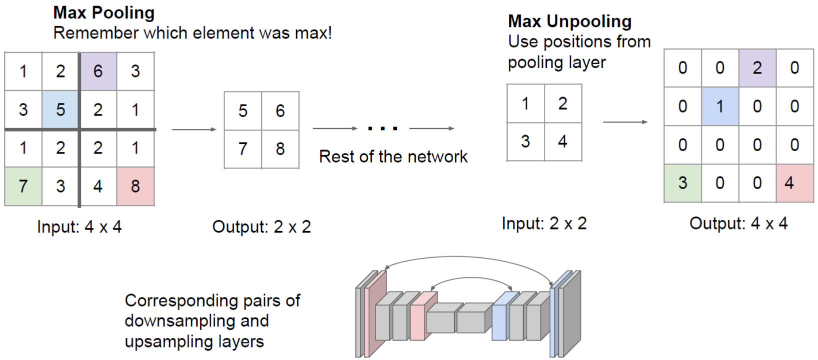
Output: 4 x 4

1	0	2	0
0	0	0	0
3	0	4	0
0	0	0	0

[6] Image credit: `https://cs231n.github.io/`, Lecture 11, 2017

Max unpooling

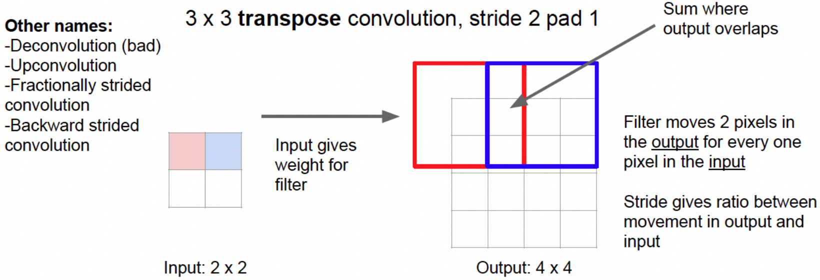
- remember the max element within max-pooling
- use the position in unpooling layers
- no parameter, better than Nearest neighbor or “Bed of nails"



[6] Image credit: `https://cs231n.github.io/`, Lecture 11, 2017

Transpose convolutions

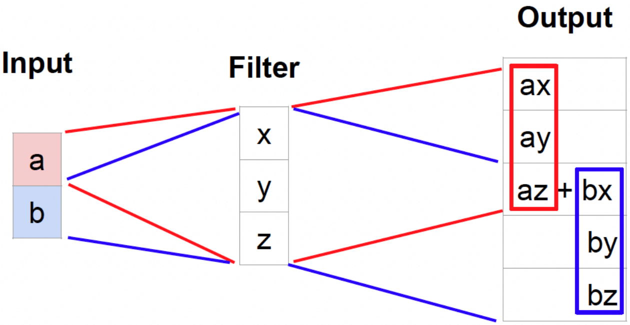
- popular approach
- trainable parameters
- multiply the single value from the input with the filter weights and project into the output with summing at overlapped positions



[6] Image credit: `https://cs231n.github.io/`, Lecture 11, 2017

Transpose convolutions

- 1D example for easy understanding



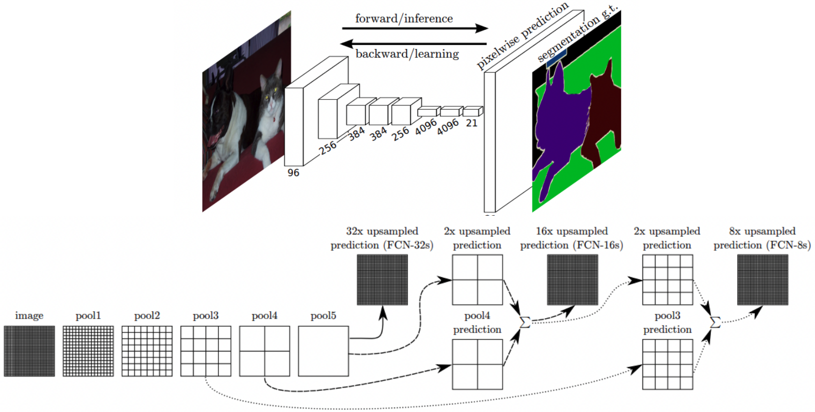
Output contains copies of the filter weighted by the input, summing at where at overlaps in the output

Need to crop one pixel from output to make output exactly 2x input

[6] Image credit: `https://cs231n.github.io/`, Lecture 11, 2017

Fully convolutional network [7]

- first every deep seg network
- **skip connections** to upsample the feature maps of the final layers and fuse with earlier maps



[7] Long, Shelhamer, and Darrell, “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015

Decovolutional network[7]

- combine unpooling and deconvolution layer (transpose convolution) for upsampling

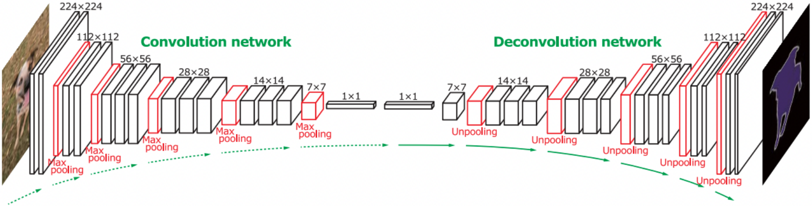
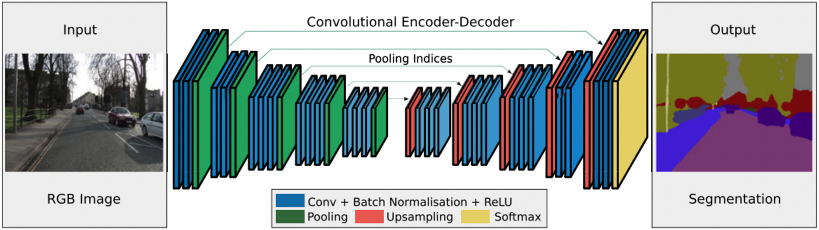


Figure 2. Overall architecture of the proposed network. On top of the convolution network based on VGG 16-layer net, we put a multi-layer deconvolution network to generate the accurate segmentation map of an input proposal. Given a feature representation obtained from the convolution network, dense pixel-wise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

[7] Noh et al, "Learning Deconvolution Network for Semantic Segmentation", ICCV 2015

SegNet[7]

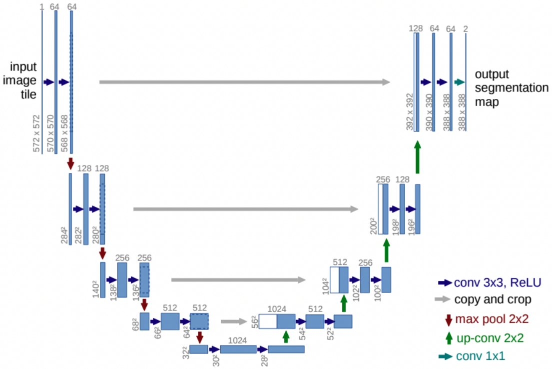
- popular approach in remote sensing
- max unpooling to remember the position of max element from pooling layer (pooling indices)



[7] SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation, IEEE PAMI 2017

U-net [7]

- proposed in the context of medical biomedical image segmentation
- skip connections for symmetric expanding path
- use transposed convolutions
- **very popular and efficient approach**



[7] U-Net: Convolutional Networks for Biomedical Image Segmentation, MICCAI 2015

Other architectures [7]

Pretrained models available on Pytorch: DeepLabV3, FCN

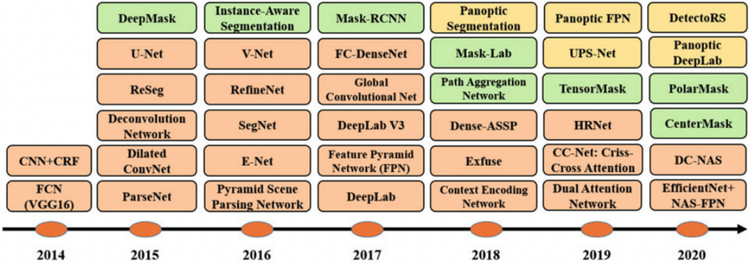
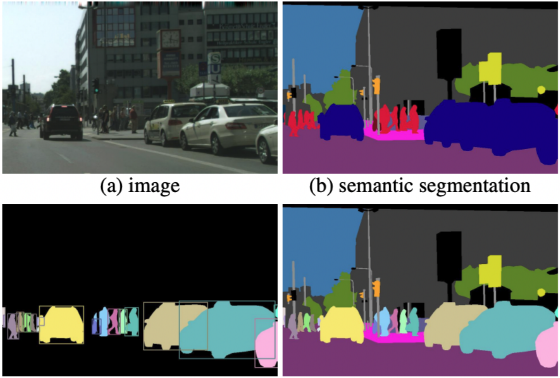


Fig. 33. Timeline of representative DL-based image segmentation algorithms. Orange, green, and yellow blocks indicate semantic, instance, and panoptic segmentation algorithms, respectively.

[7] Image Segmentation Using Deep Learning: A Survey, IEEE PAMI 2022

Remark Do no confuse tasks[8]

- Semantic segmentation
- Instance segmentation
- Panoptic segmentation (combination of the 2 others)



(a) image (b) semantic segmentation (c) instance segmentation (d) panoptic segmentation

[8] Panoptic Segmentation, CVPR 2019

References I

Books & articles:

1. Minaee, S., Boykov, Y.Y., Porikli, F., Plaza, A.J., Kehtarnavaz, N. and Terzopoulos, D., 2021. Image segmentation using deep learning: A survey. IEEE PAMI.

Courses and blogs:

1. Lecture 11: Segmentation and Detection
   `https://cs231n.github.io/transfer-learning/`
2. An overview of semantic image segmentation, 2018
   `https://www.jeremyjordan.me/semantic-segmentation/#loss`
3. Image segmentation with DeepLabV3
   `https://pytorch.org/tutorials/beginner/deeplabv3_on_ios.html`
4. Semantic segmentation benchmarks
   `https://paperswithcode.com/task/semantic-segmentation`