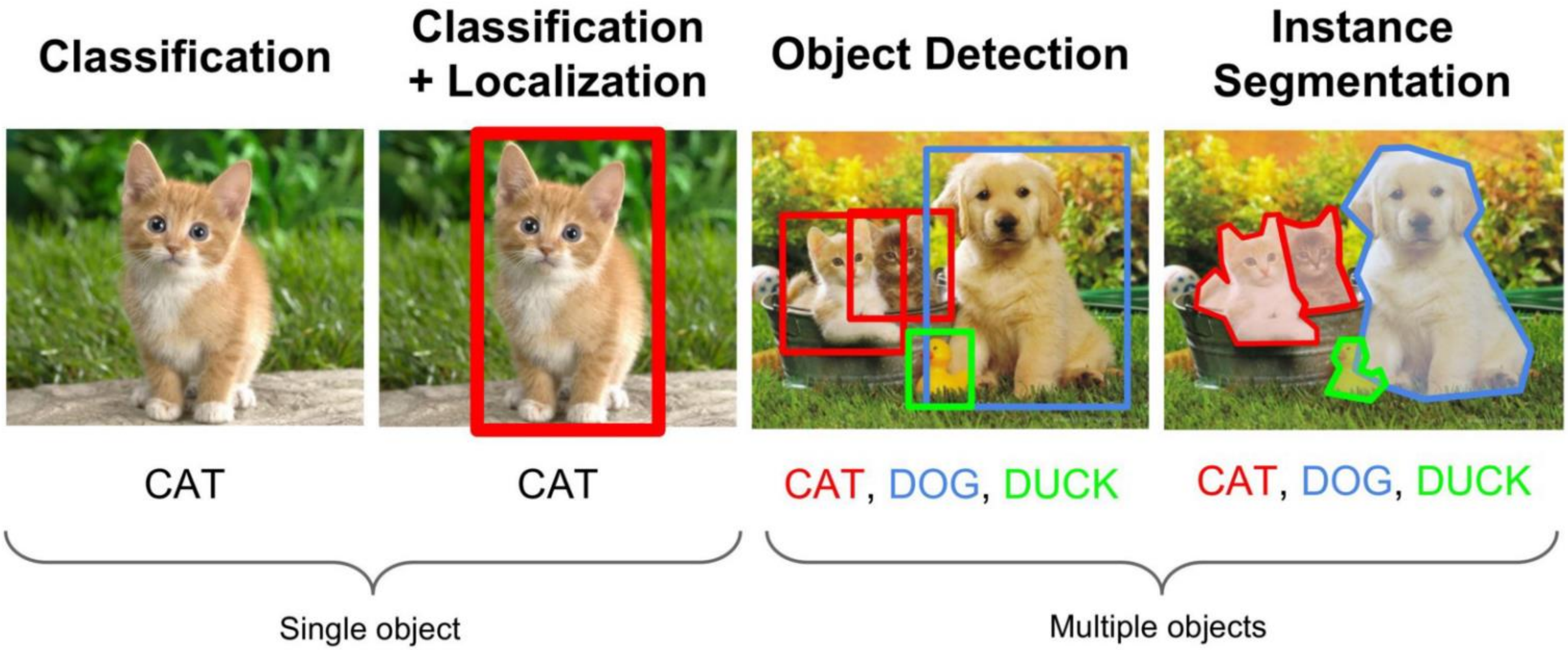
Image segmentation

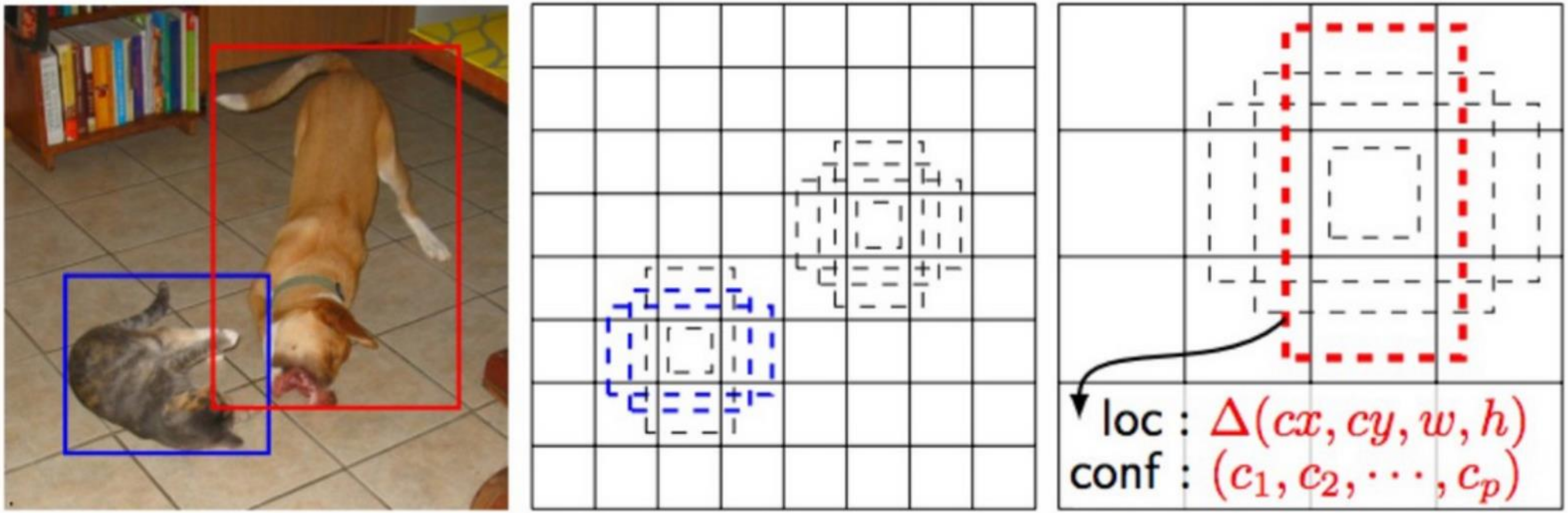
Ing. Marek Hrúz, PhD.

Segmentation vs. Detection

- **Segmentation** is the task of dividing the image into regions that represent abstract concepts, suitable for further processing
- **Detection** is the task of providing rectangular regions containing known objects

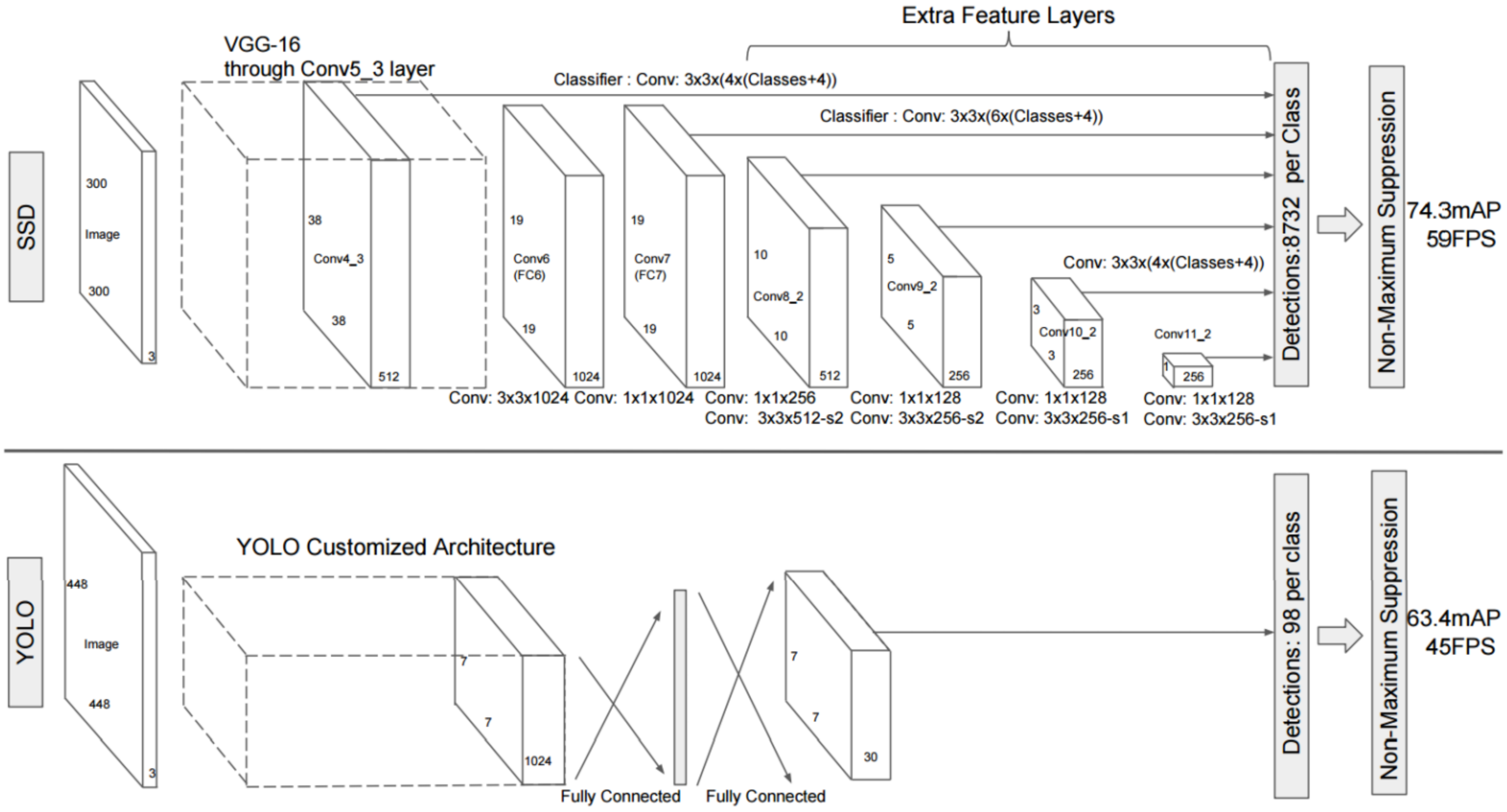


Detection using DNN - SSD



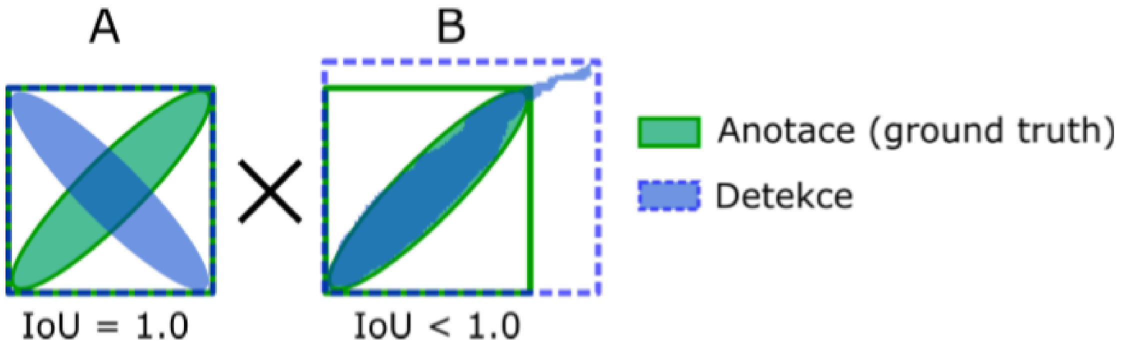
(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

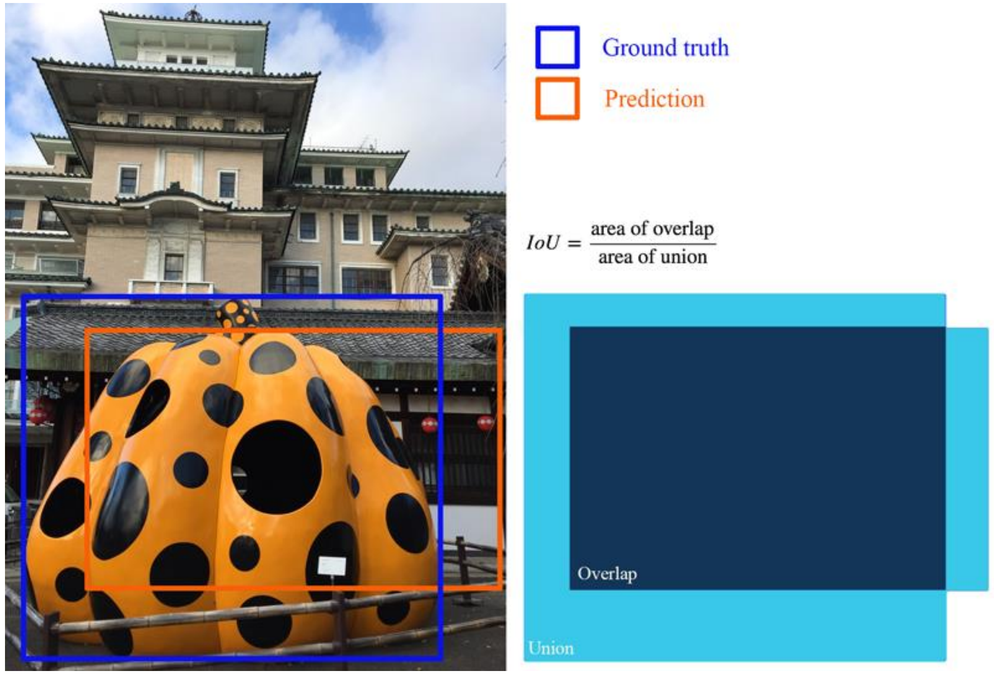
SSD vs YOLOv1



mAP – mean average precision

- Measures the quality of your detector
- Intersection over Union (IoU, or Jaccard Index)
  - Tells you whether you were able to detect an object (IoU > thresh)





mAP – mean average precision

- For a given class, compute the ranking of the ground-truth according to your predictions
- Two classes: DOG and CAT
- GT data




mAP - Rankings

- Query: DOG
- Ranking #1:

- Precision: **1/1** 1/2 **2/3** 2/4
- Query: CAT
- Ranking #2:

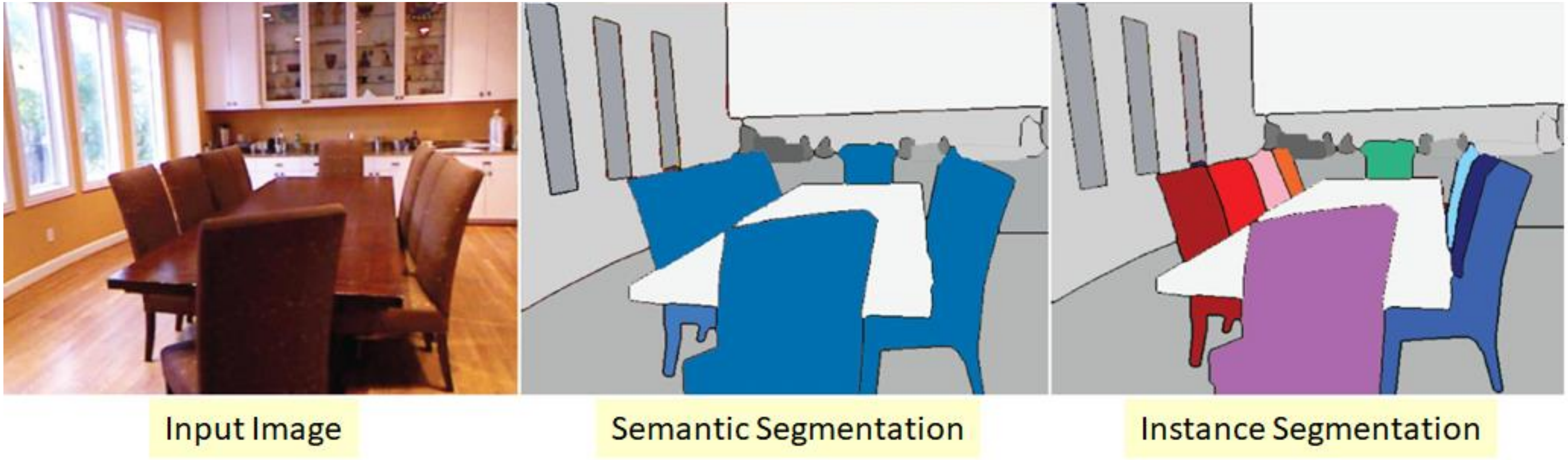
   

- Precision: **1/1** **2/2** 2/3 2/4
- Avg. Precisions:
  - Ranking #1 = (1/1 + 2/3) / 2 = 5/6 ~ 83%
  - Ranking #2 = (1/1 + 2/2) / 2 = 100%
- Mean average precision = (83% + 100%) / 2 = 91,5%

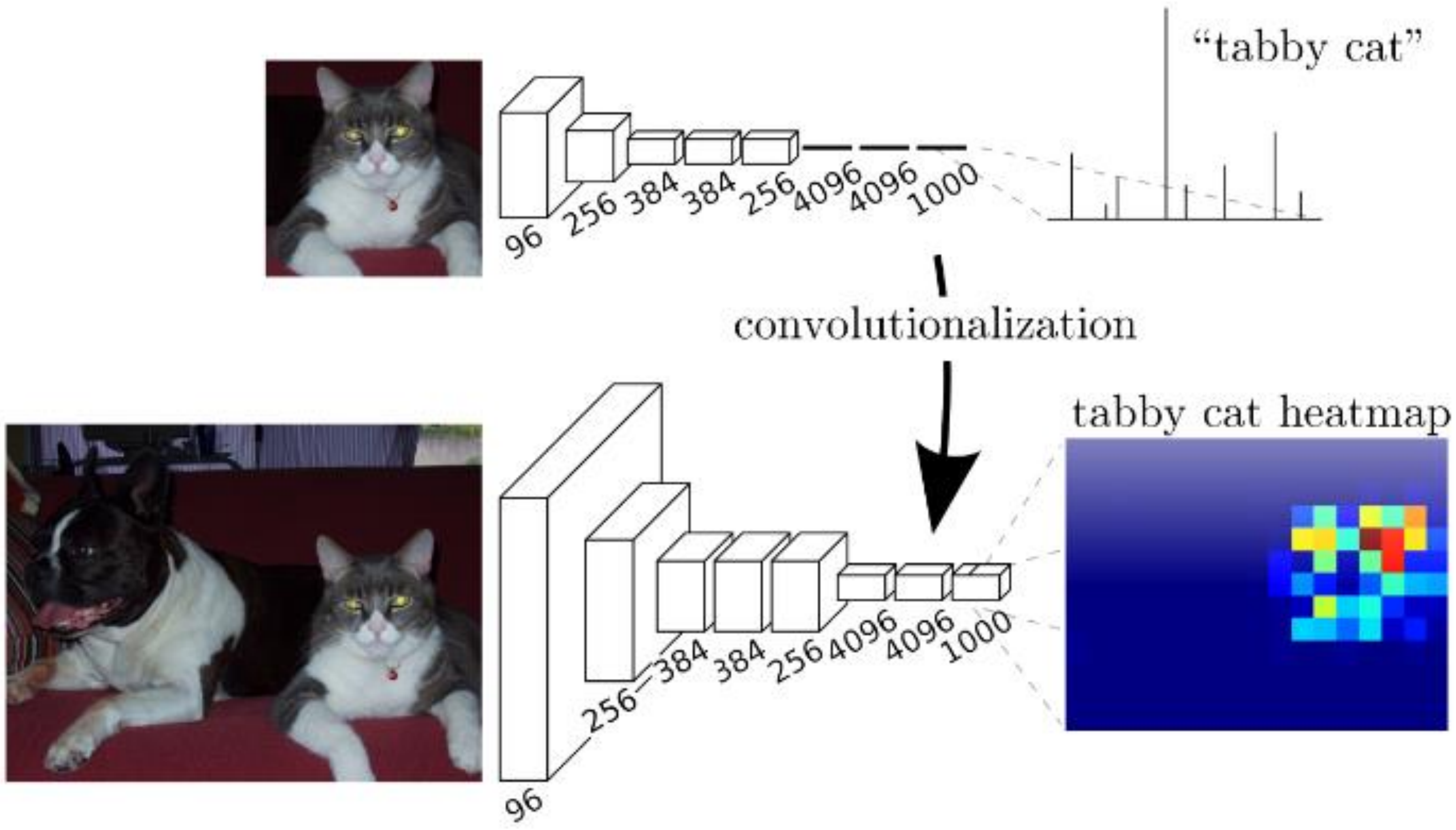
mAP – Cheat sheet

- Ranking #N
  - Rank your detections of class N according to objectiveness score (probability of your detection being N)
- Compute precision of your detections according to rank
  - P_rank = #True Positive@Rank/Rank
- Average precision for class N is average P_rank of ground truth data
- Mean Average Precision is the average of average precisions for all classes

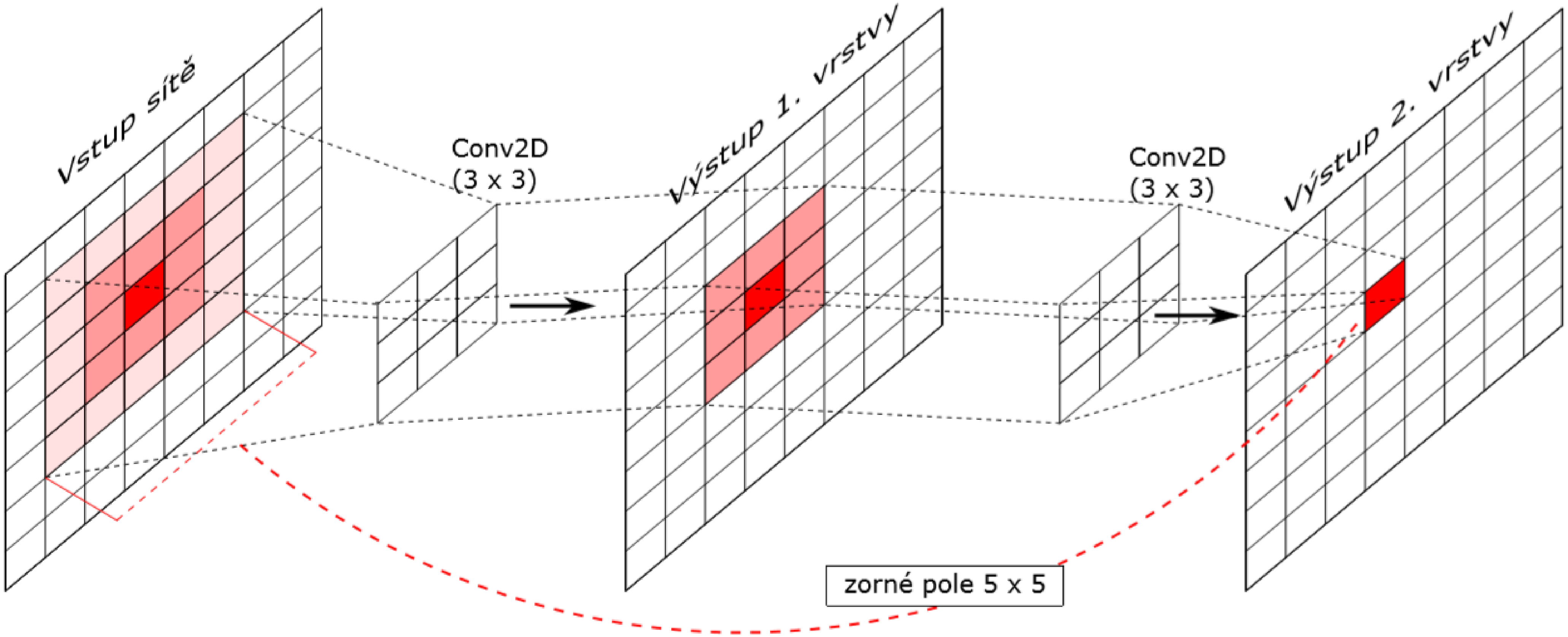
Semantic segmentation



Semantic segmentation

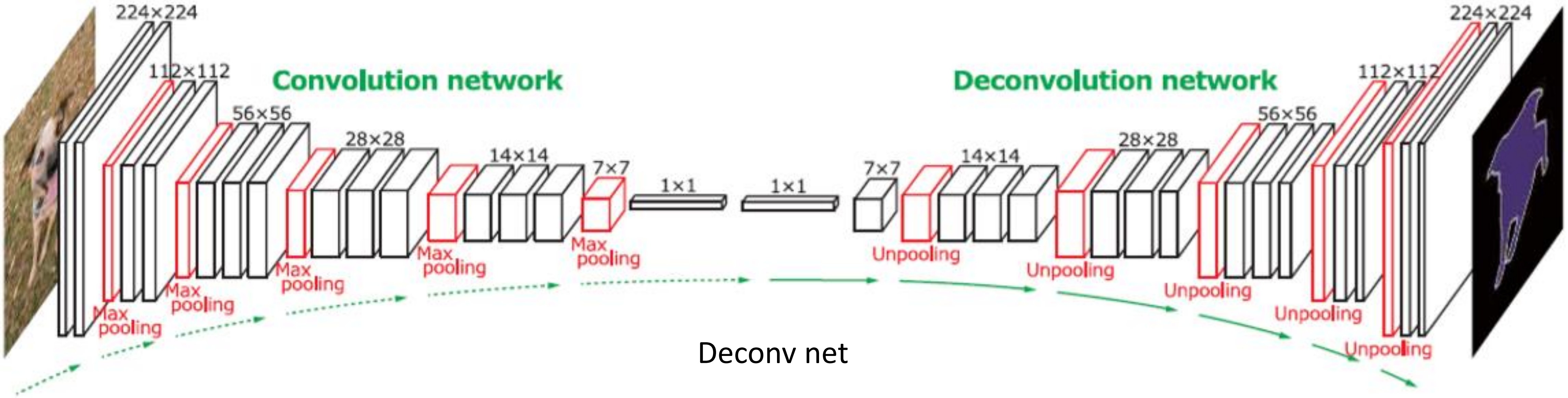


CNN – receptive field



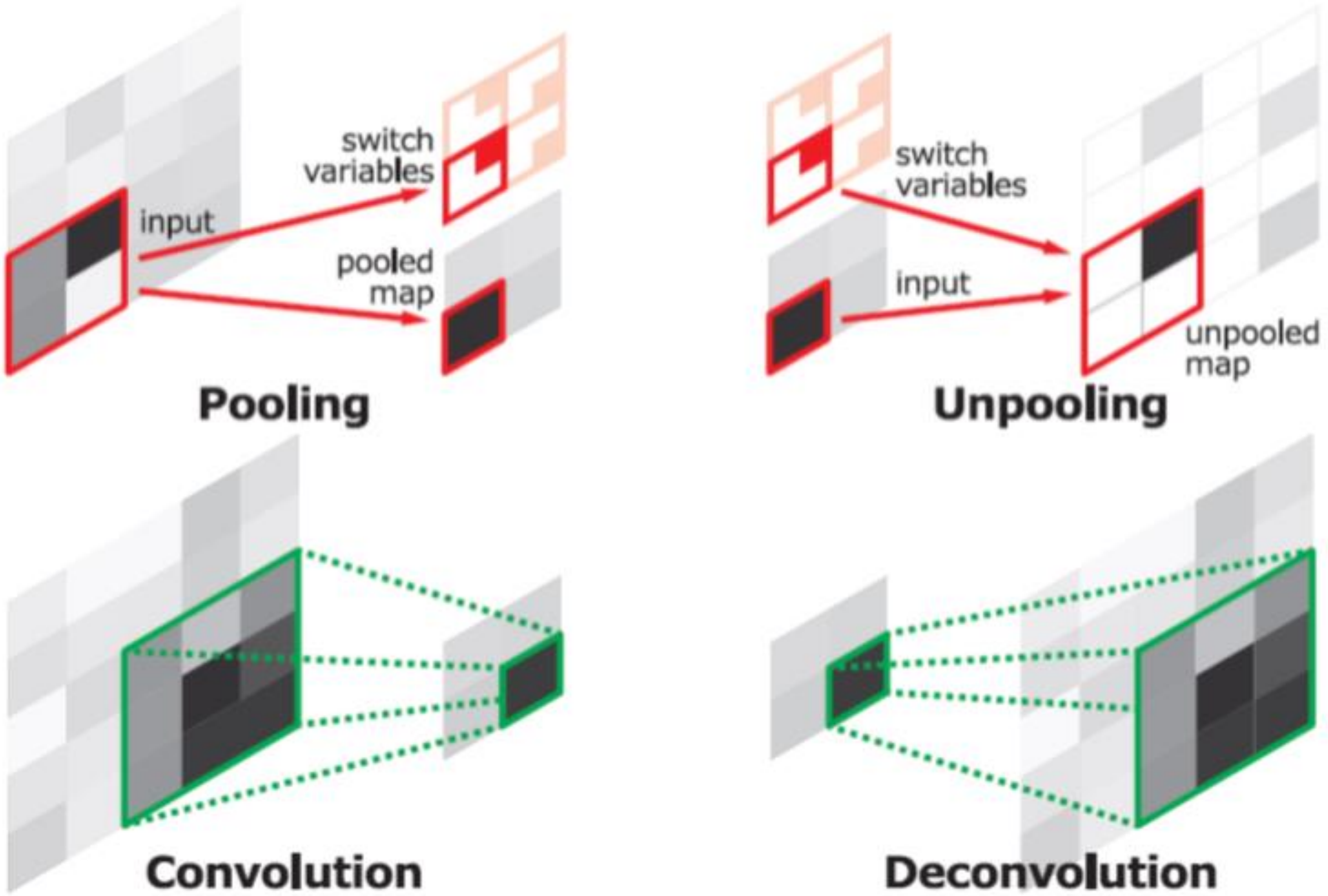
Obrázek 7.1: Ukázka zorného pole v hlubších vrstvách.

Learning Deconvolution Network for Semantic Segmentation



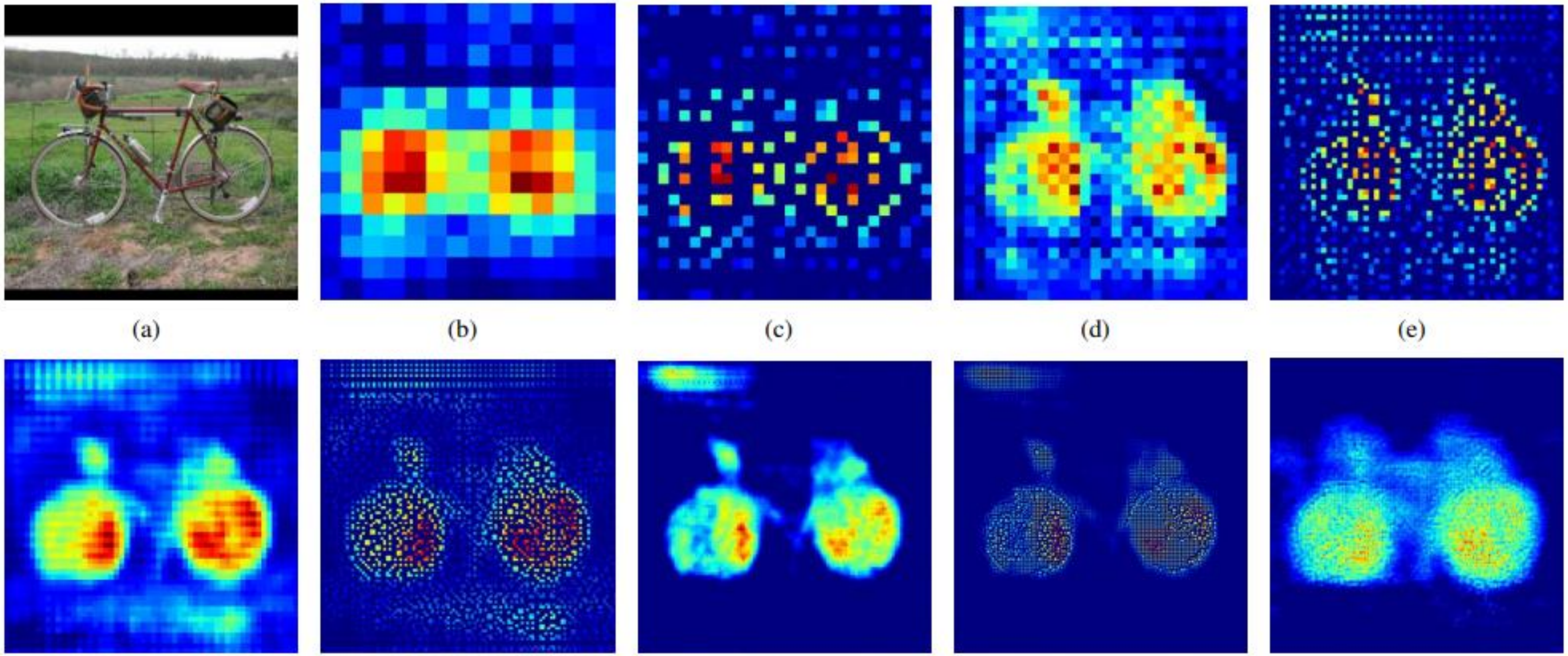
Deconv net

Unpooling & Deconvolution



Deconvolution = Transposed Convolution

- https://github.com/vdumoulin/conv_arithmetic



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

U-Net

- 3x3 convolutions in encoder
- 2x2 max pool, no overlap
- Doubling the number of channels (features)

- Encoder – upsamping
- Transposed convolution 2x2@2
- Halving number of channels (features)
- Cropped and concatenated feature maps

- Used in medical imaging segmentation
- Fully convolutional

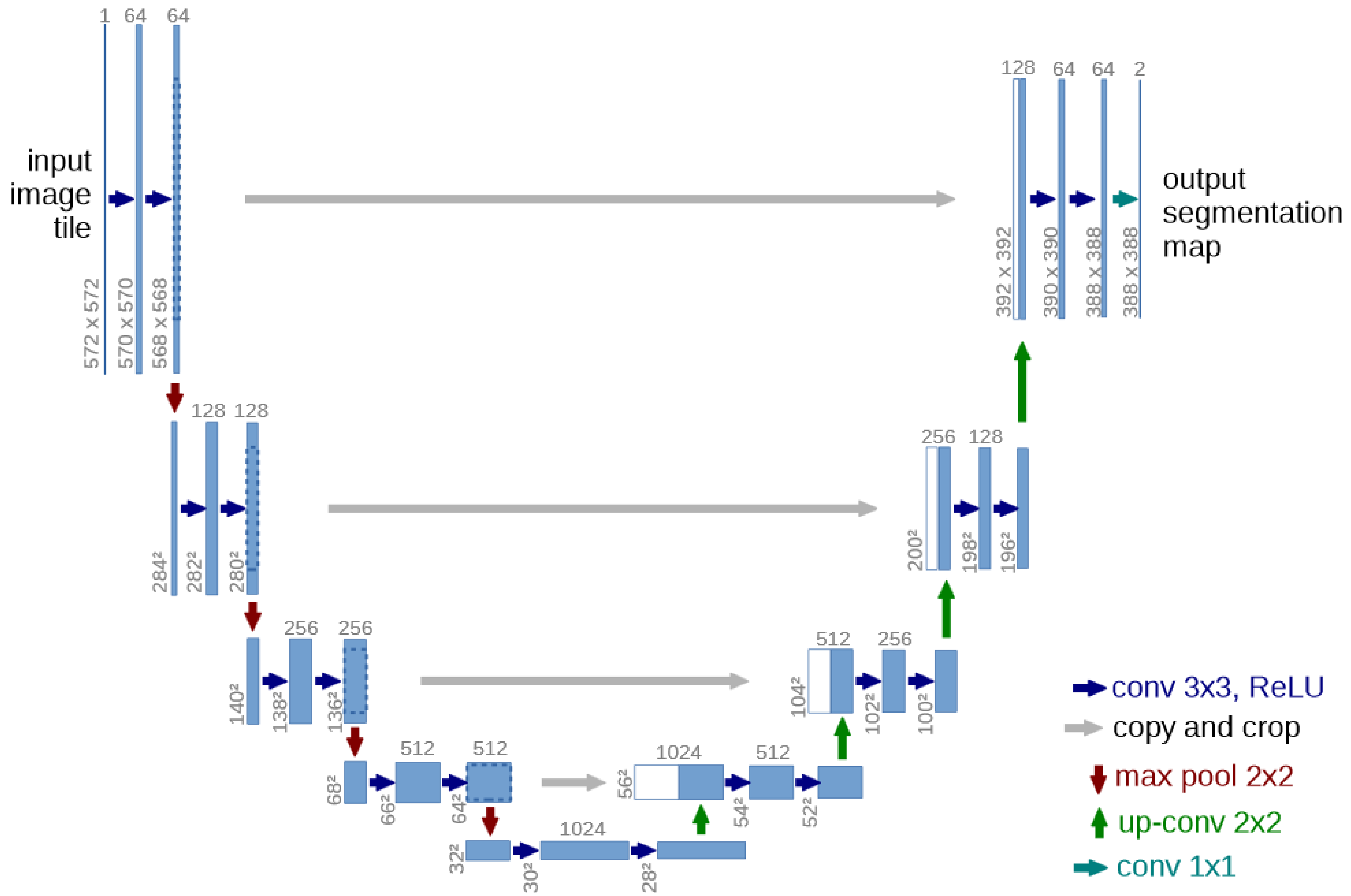
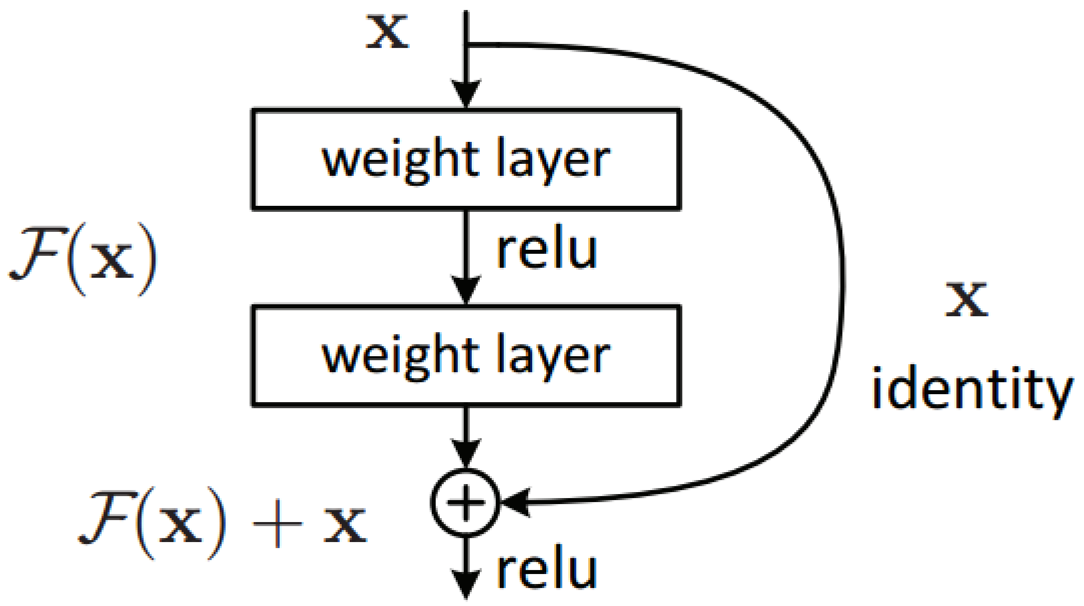


Fig. 1. U-net architecture (example for 32x32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

Residual connections

- Best practice when training deep neural networks
- Works, no one really knows why
- The residual (skip) connection can be – identity or learned projection (possibly into space of different dimension)



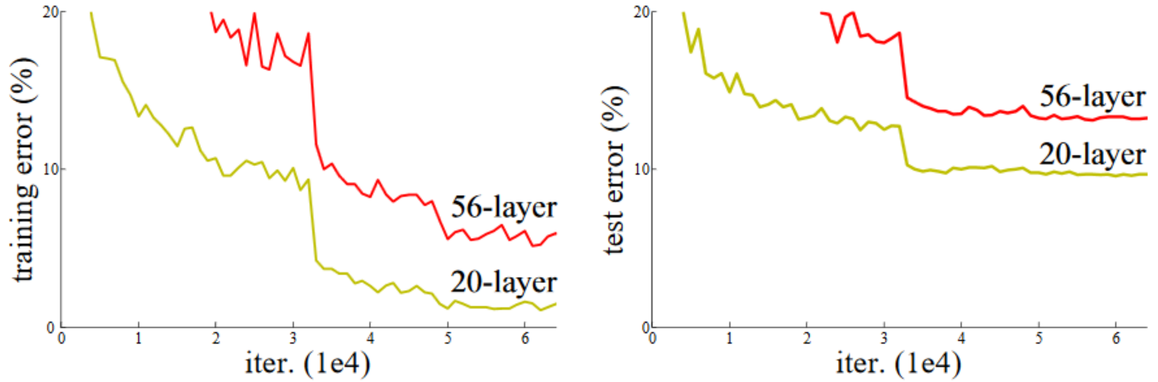
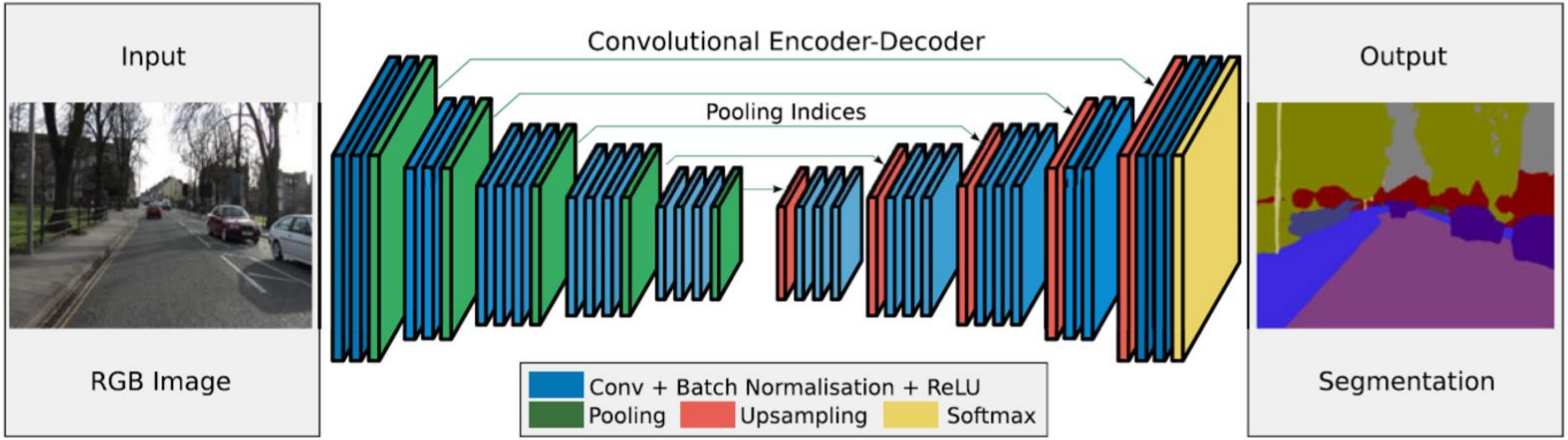


Figure 1. Training error (left) and test error (right) on CIFAR-10 with 20-layer and 56-layer "plain" networks. The deeper network has higher training error, and thus test error. Similar phenomena on ImageNet is presented in Fig. 4.

SegNet



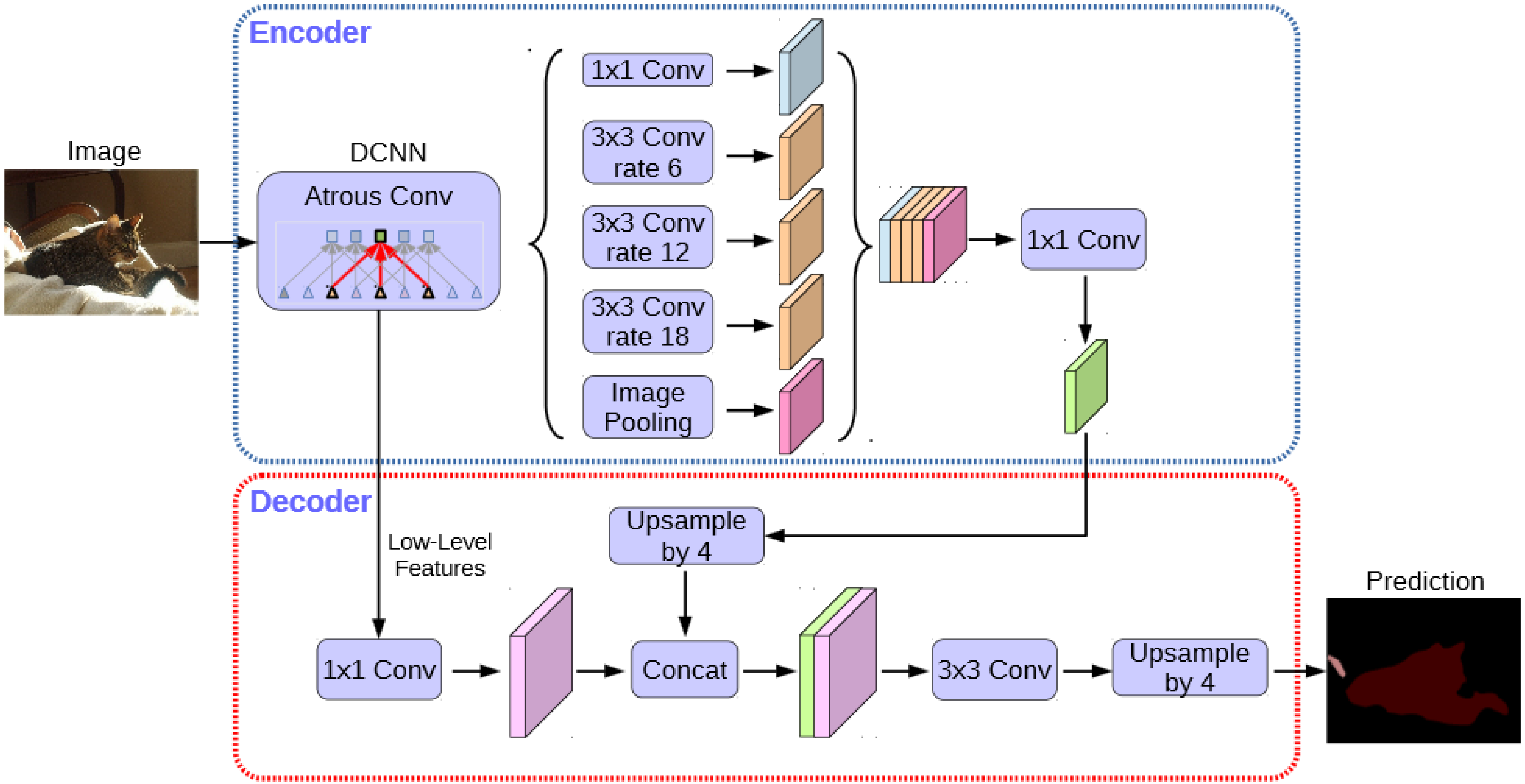
Obrázek 10.2: Ilustrace architektury SegNet, která byla převzata z práce [12].

http://mi.eng.cam.ac.uk/projects/segnet/

SegNet – details

- Semantic segmentation
- Fully Convolutional (unlike DeconvNet)
- Unpooling + Deconvolution
- VGG backbone
- Tested on task of road scene segmentation

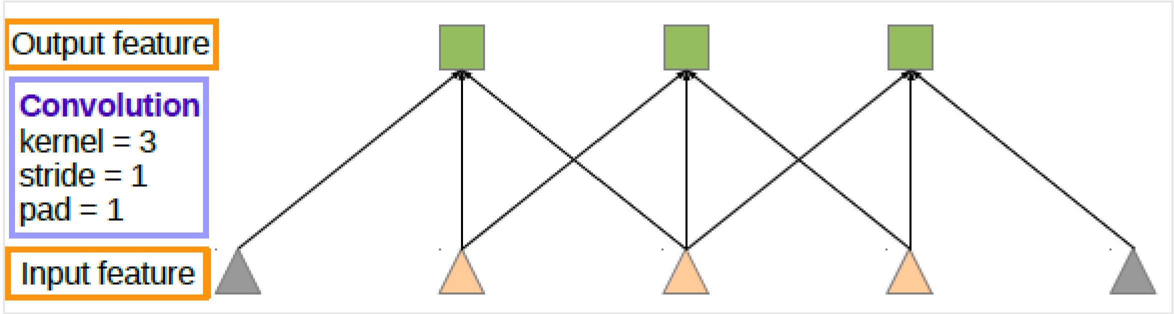
DeepLab v3+



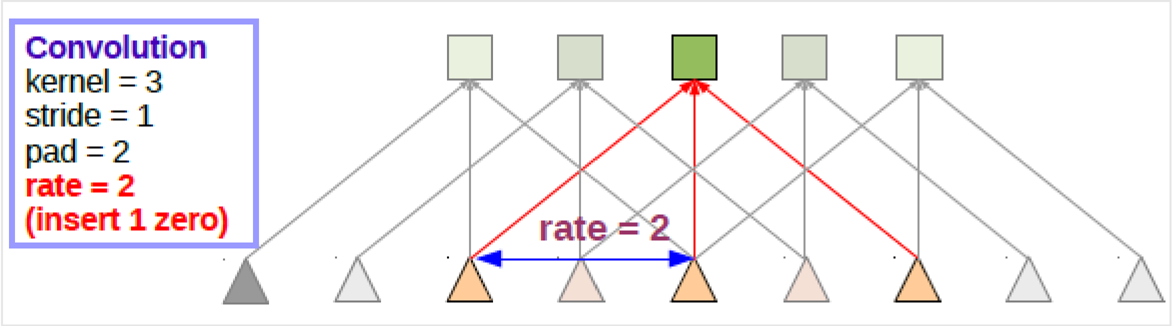
Encoder
Image
DCNN
Atrous Conv
1x1 Conv
3x3 Conv rate 6
3x3 Conv rate 12
3x3 Conv rate 18
Image Pooling
1x1 Conv
Decoder
Low-Level Features
Upsample by 4
1x1 Conv
Concat
3x3 Conv
Upsample by 4
Prediction

Atrous (Dilated) Convolution

- $y[i] = \sum_{k=1}^{K} x[i + r \cdot k]\omega[k]$

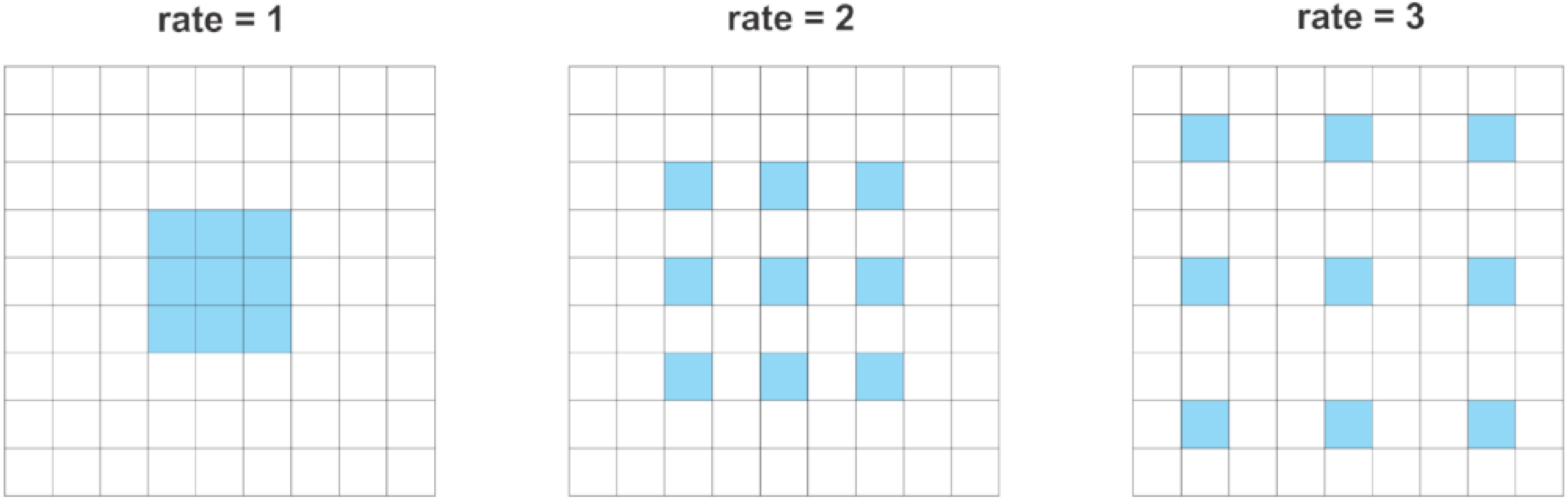


(a) Sparse feature extraction

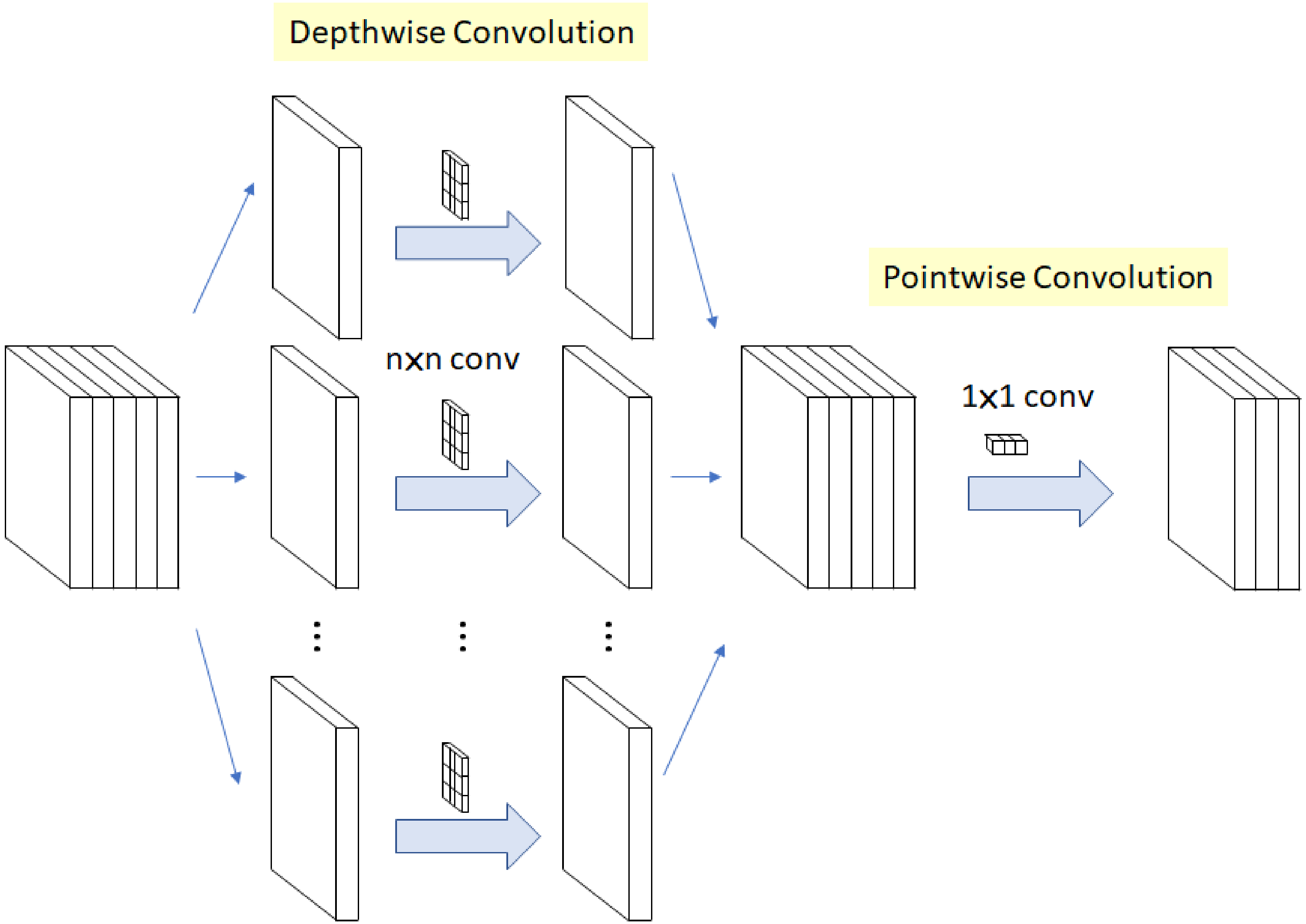


(b) Dense feature extraction

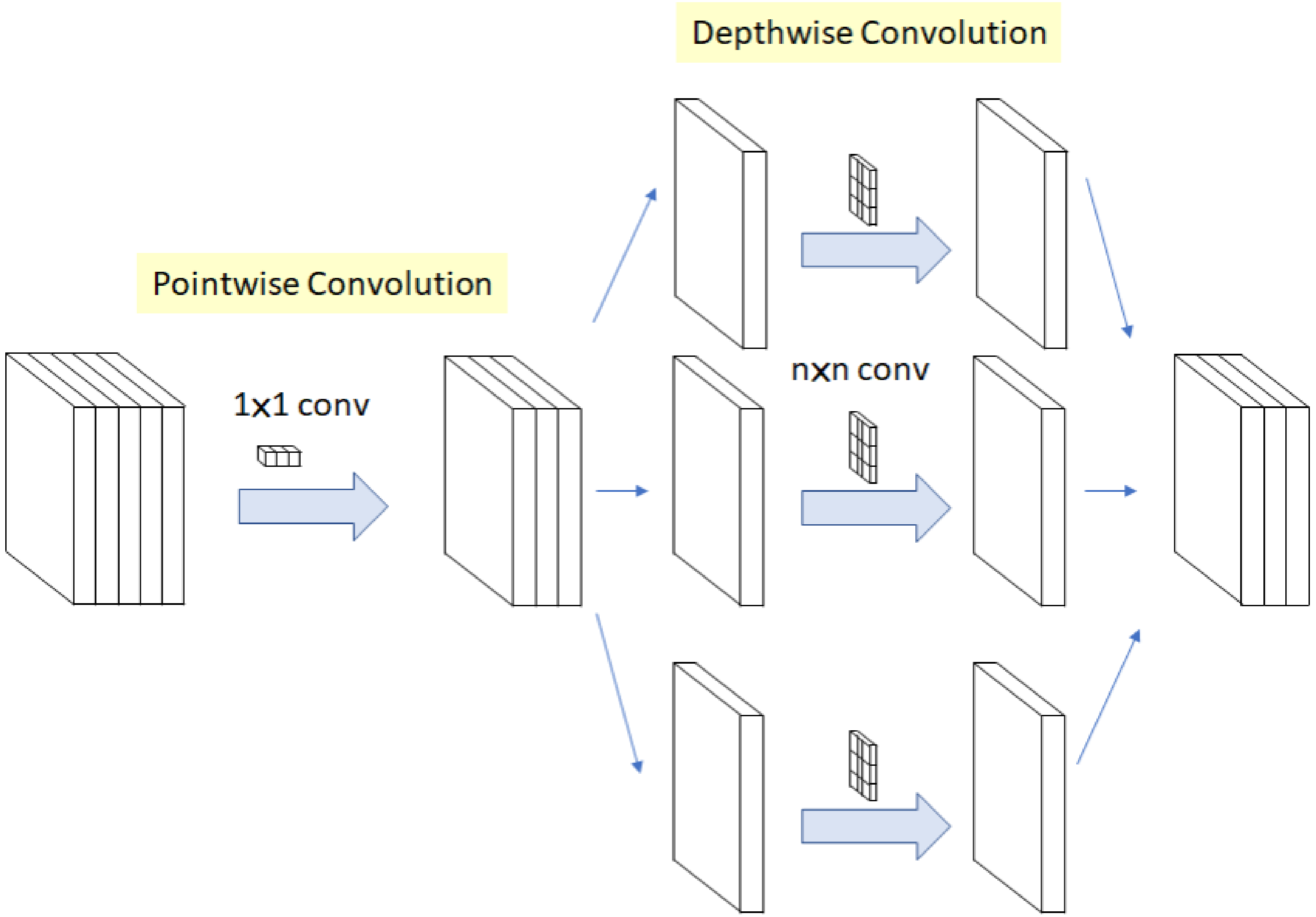
Atrous (Dilated) Convolution



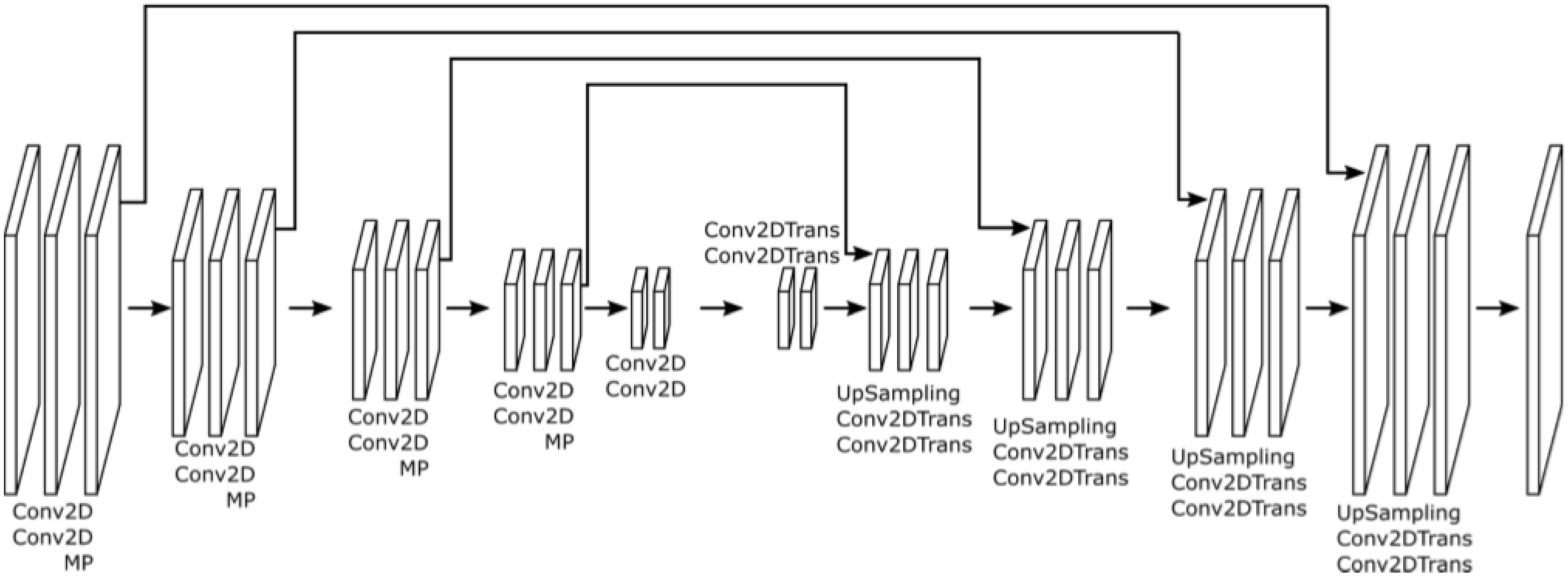
Depth wise conv - Inception version



Depth wise conv - Xeption version

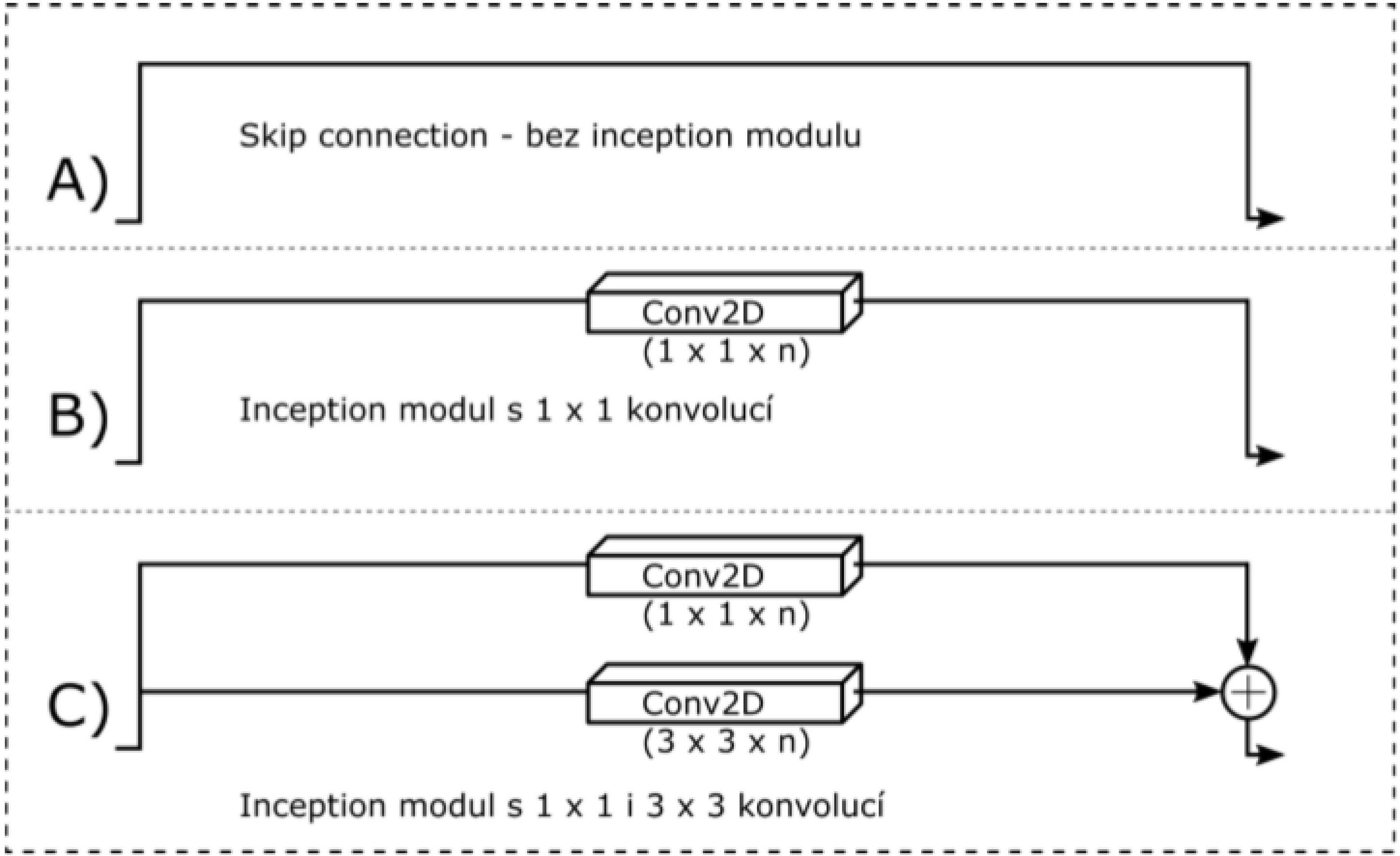


Using skip connections

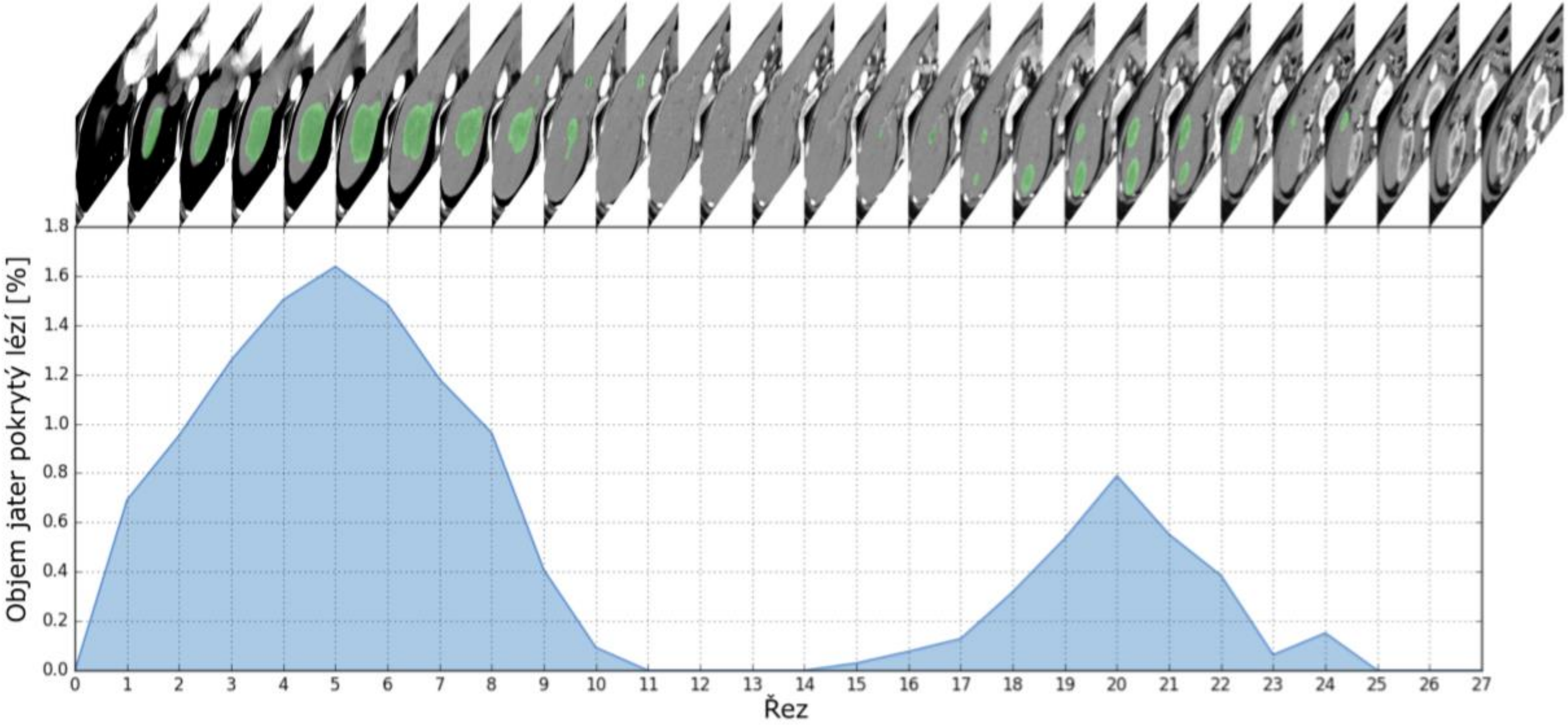


Obrázek 10.5: Architektura sítě se skip connections.

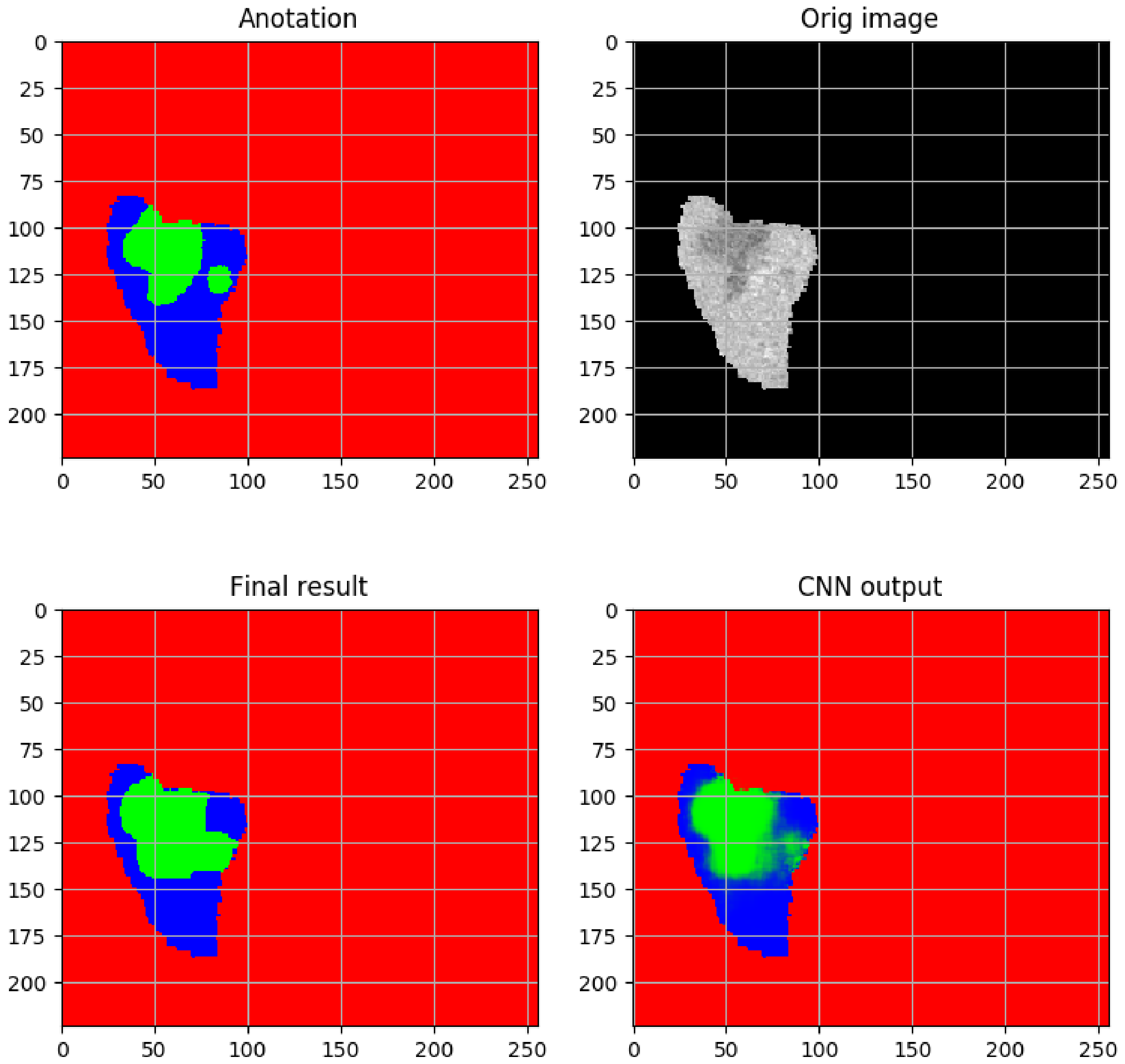
Inceptions in skip connections

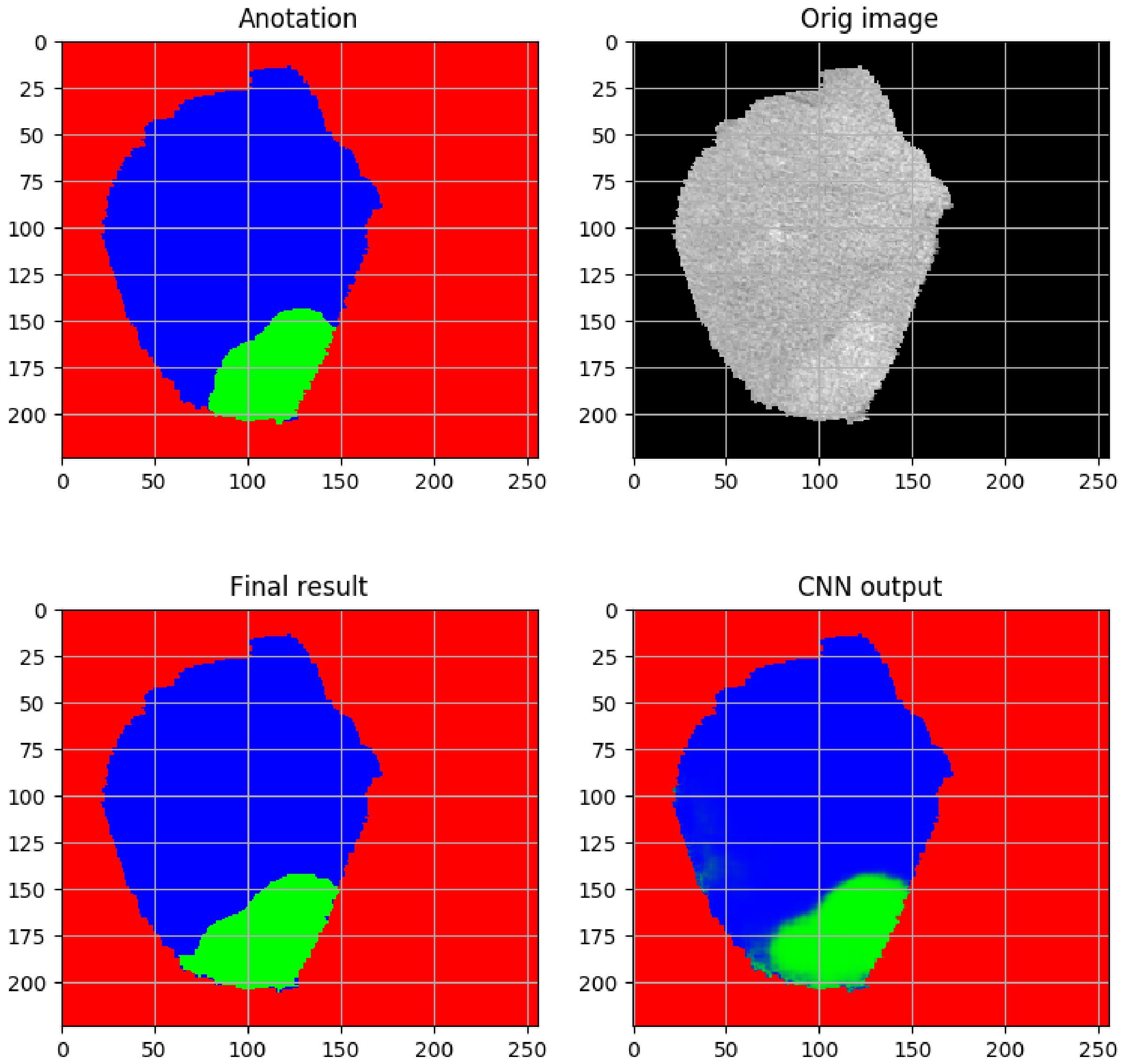


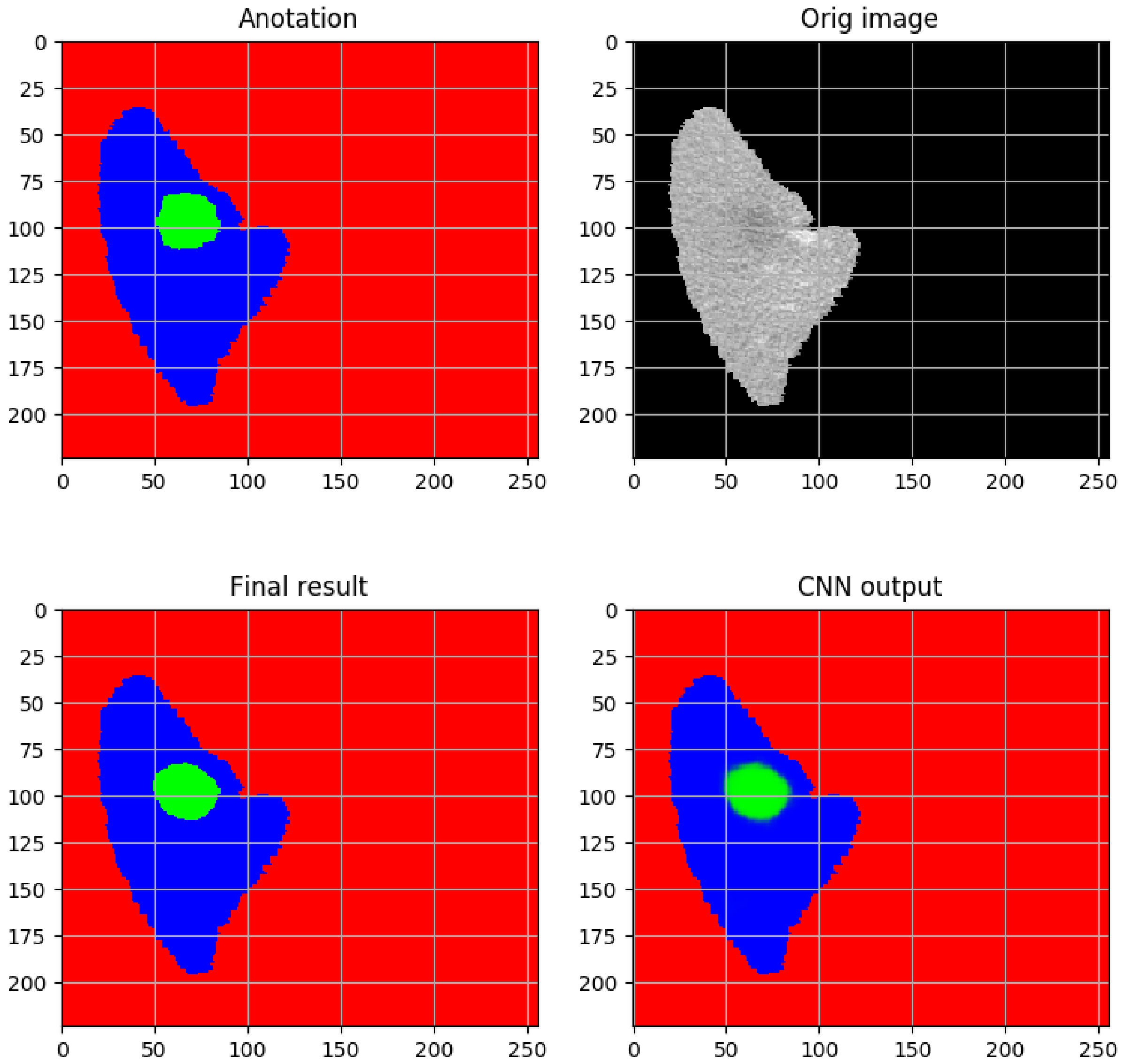
Application

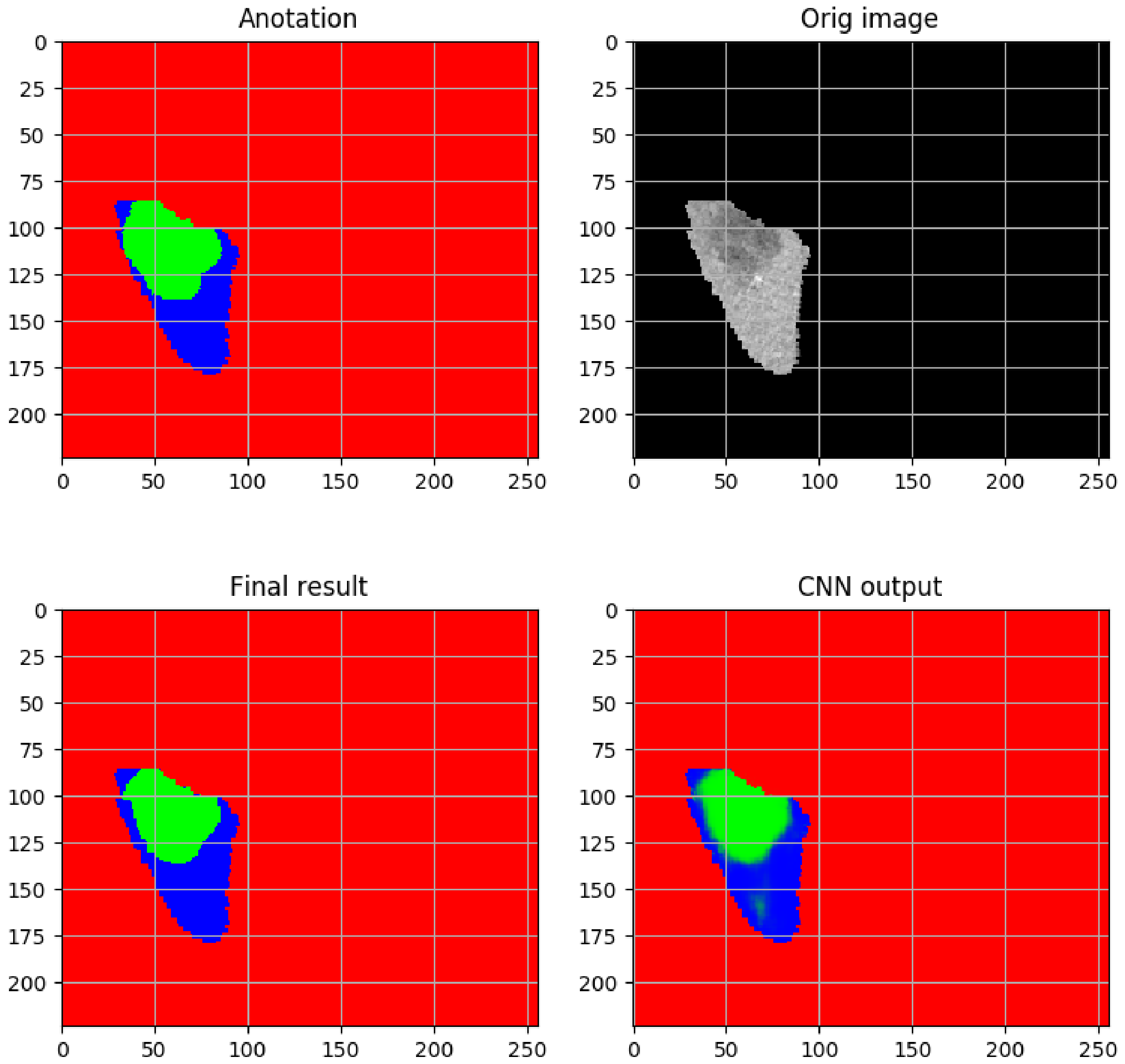


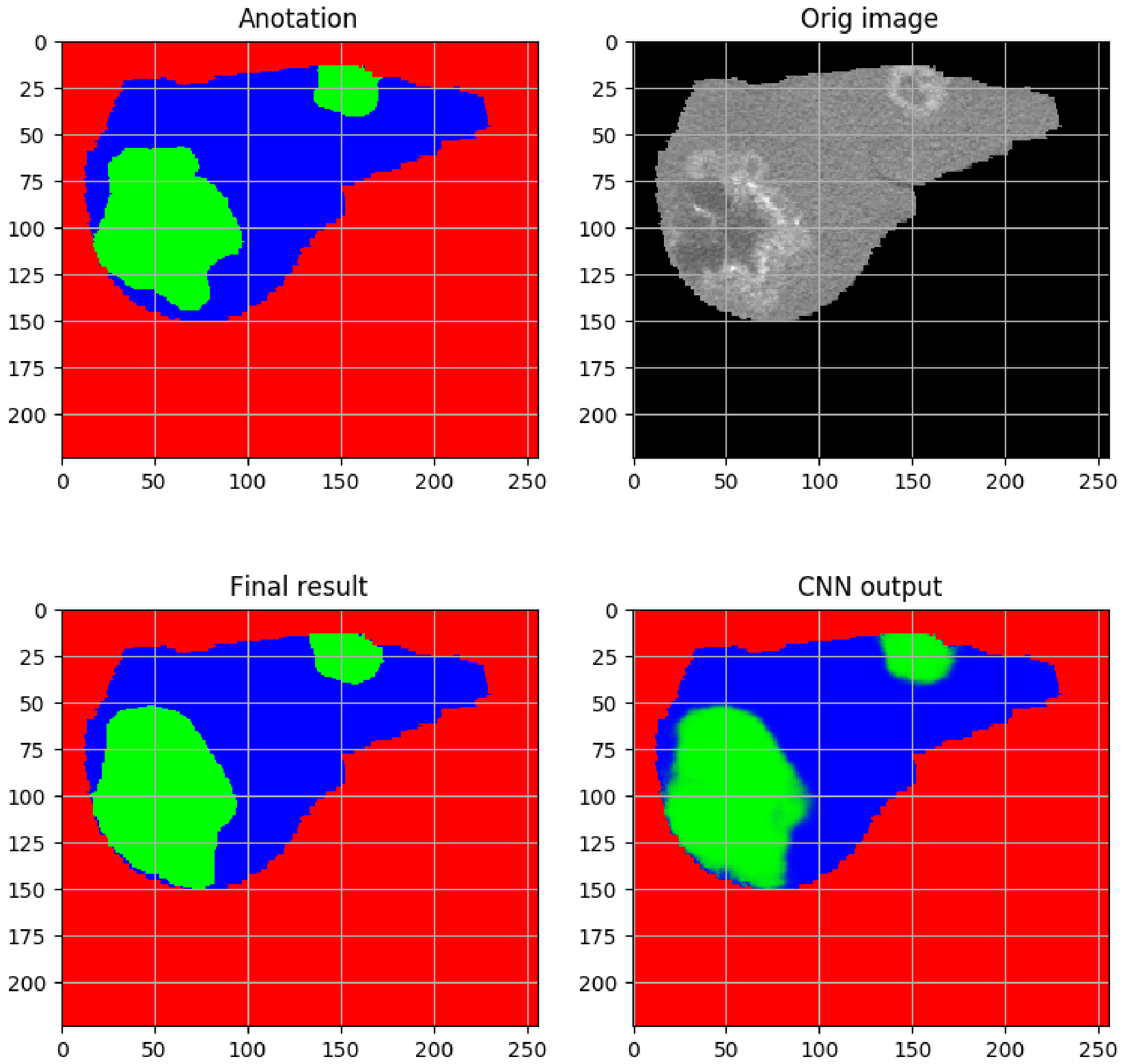
Obrázek 11.1: Křivka zastoupení lézí v jednotlivých řezech.

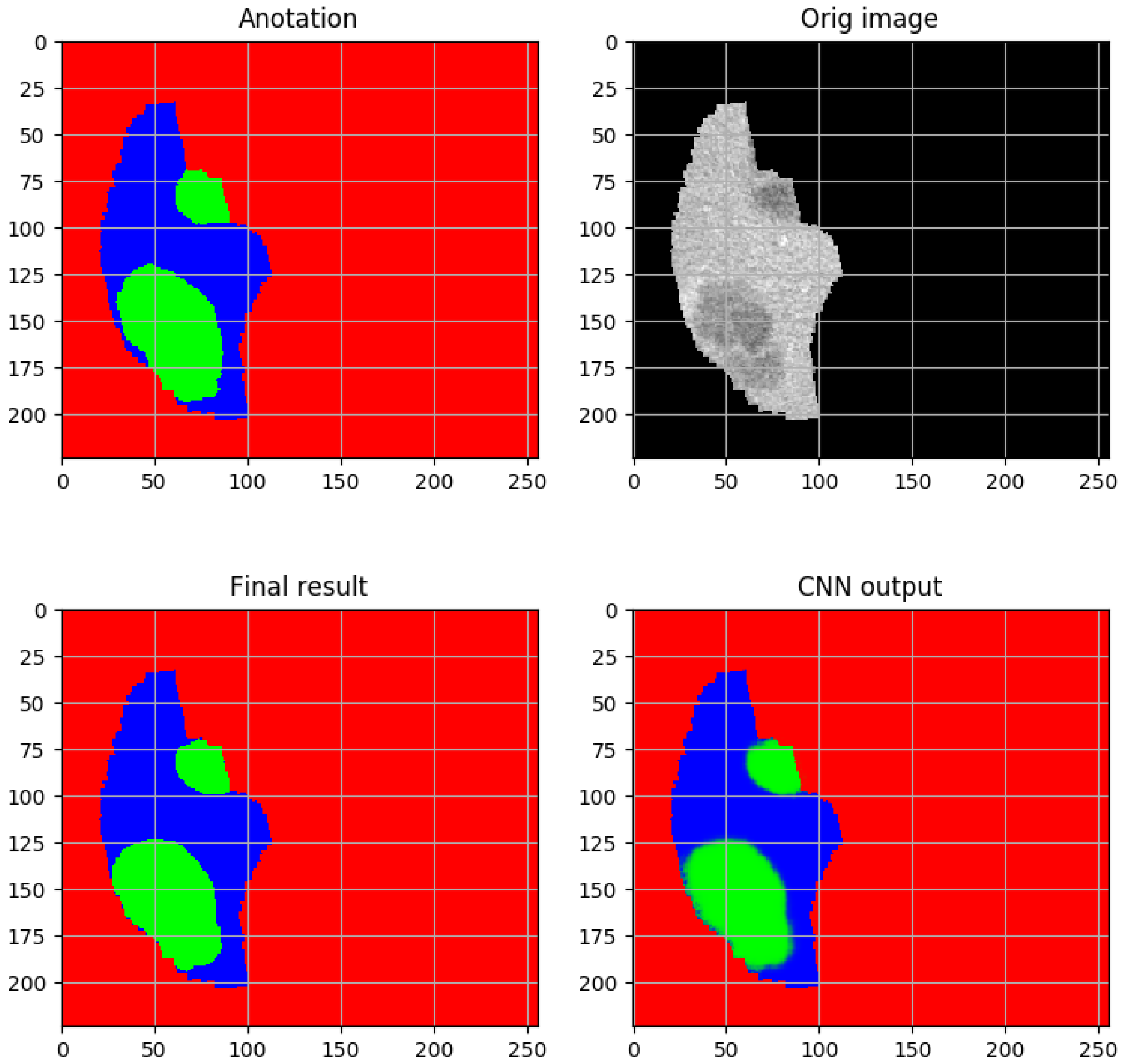

Anotation
Orig image
Final result
CNN output
0
25
50
75
100
125
150
175
200
0
50
100
150
200
250


Anotation
Orig image
Final result
CNN output
0
25
50
75
100
125
150
175
200
0
50
100
150
200
250

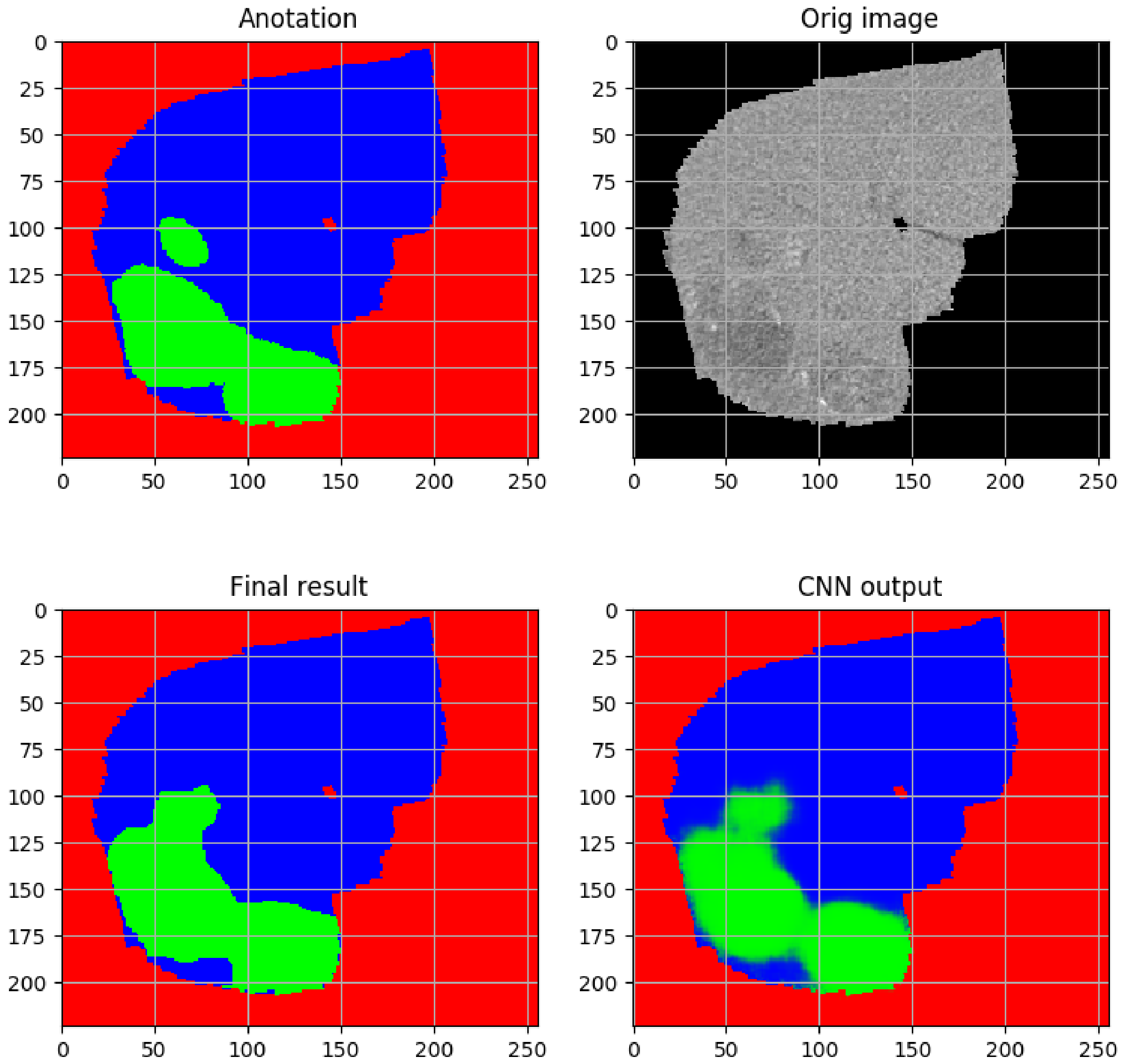

Anotation
Orig image
Final result
CNN output
0
25
50
75
100
125
150
175
200
0
50
100
150
200
250

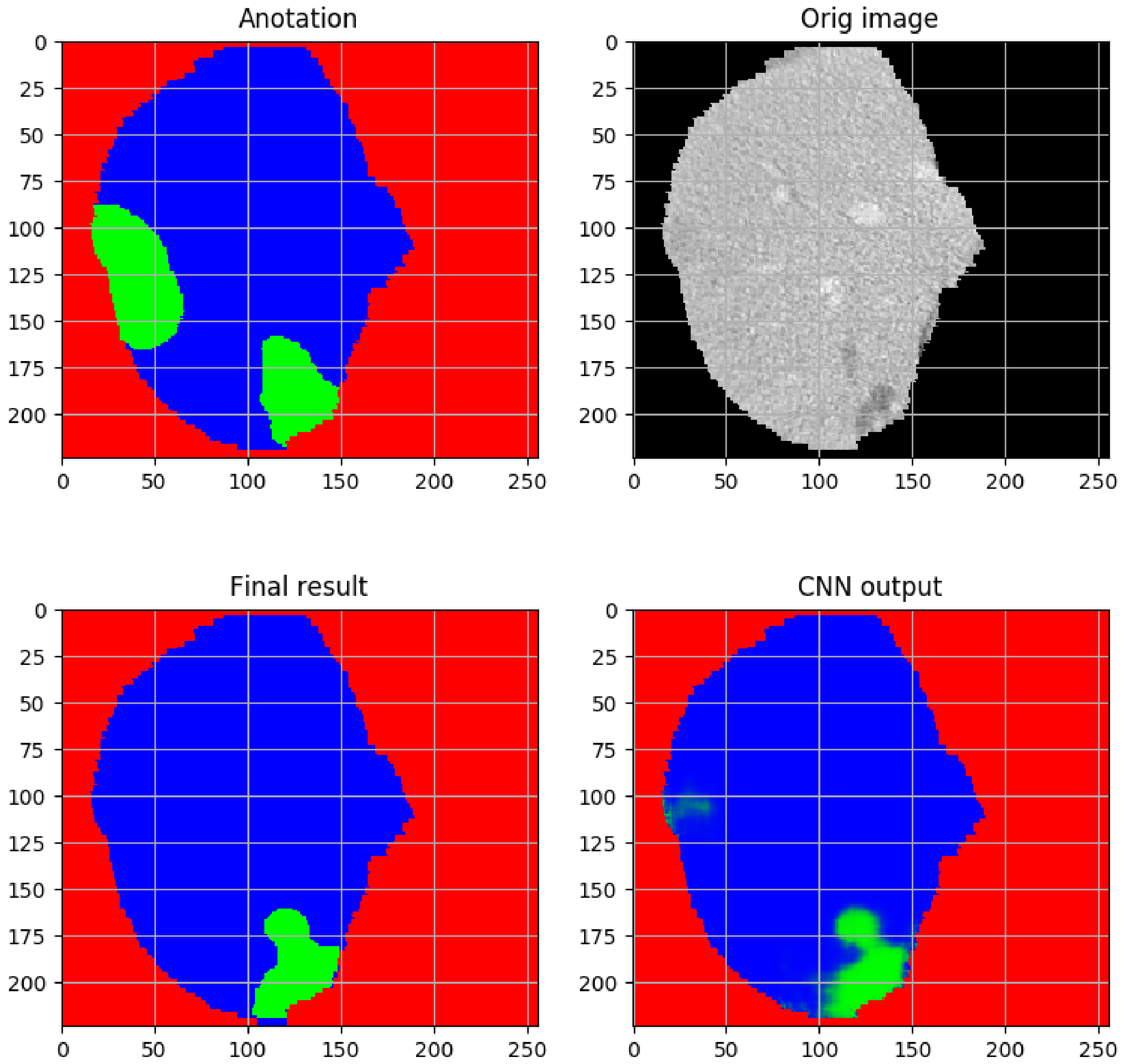

Anotation
Orig image
Final result
CNN output

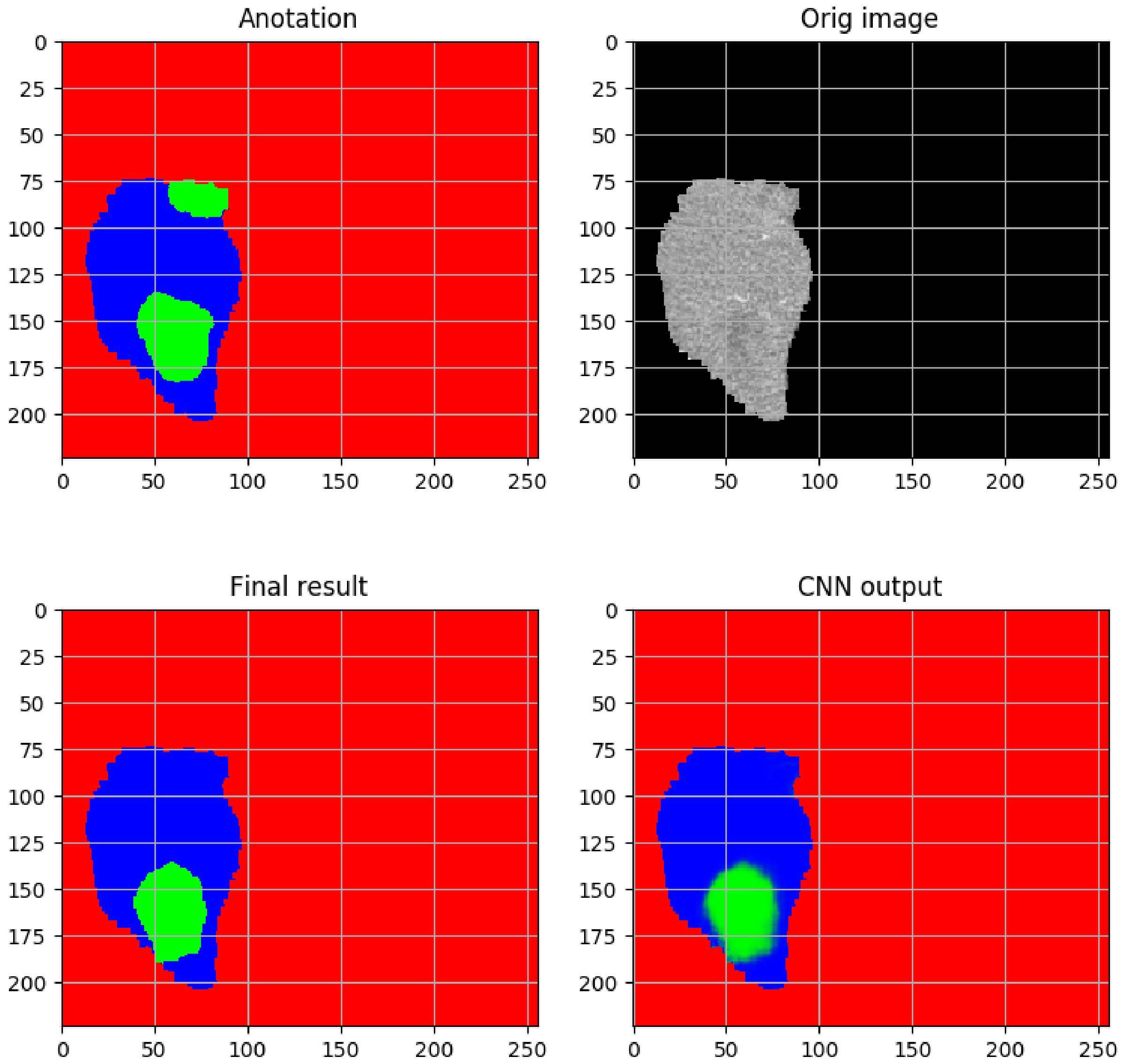

Anotation
Orig image
Final result
CNN output
0
25
50
75
100
125
150
175
200
250



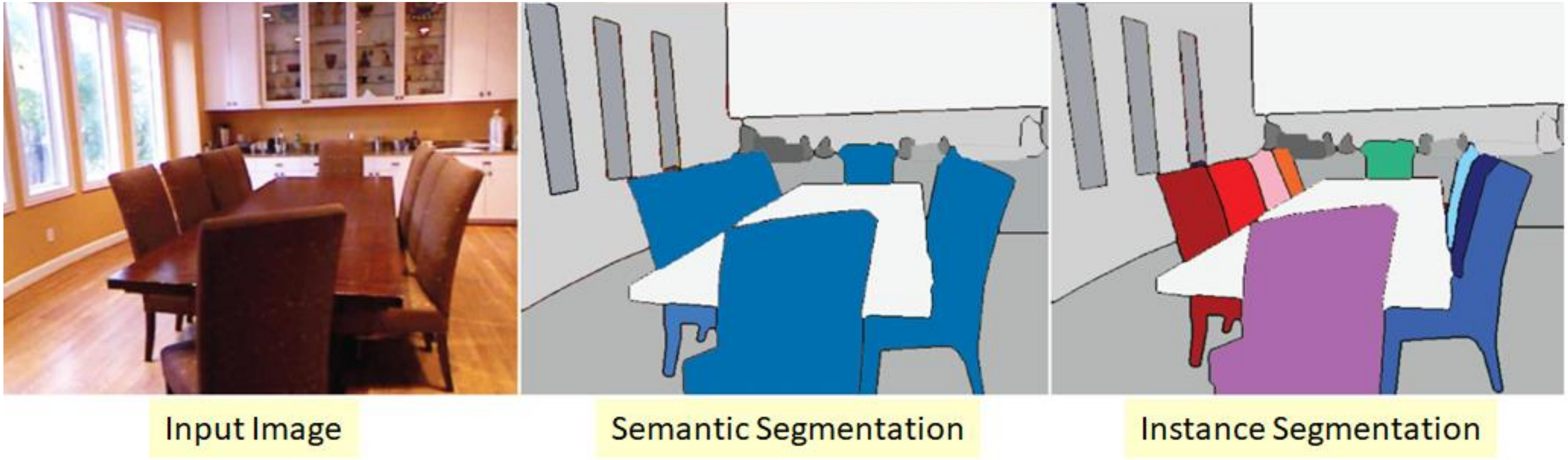
Anotation
Orig image
Final result
CNN output
0
25
50
75
100
125
150
175
200
250


Anotation
Orig image
Final result
CNN output
0
25
50
75
100
125
150
175
200
0
50
100
150
200
250


Anotation
Orig image
Final result
CNN output
0
25
50
75
100
125
150
175
200
0
50
100
150
200
250

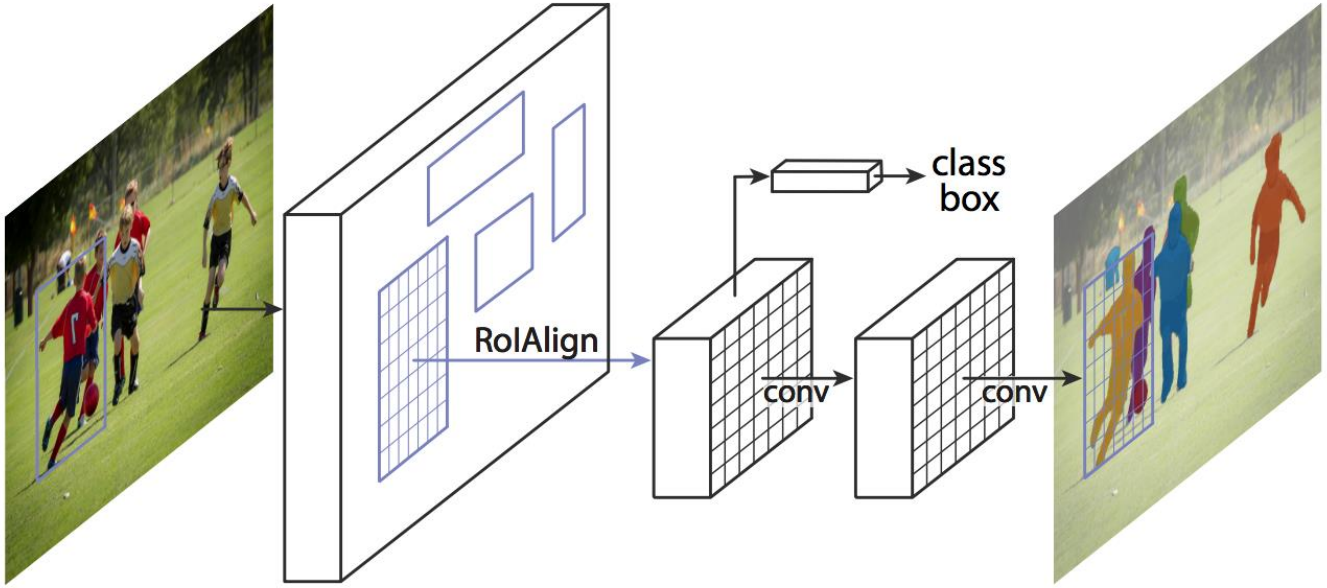

Anotation
Orig image
Final result
CNN output
0
25
50
75
100
125
150
175
200
0
50
100
150
200
250

Instance segmentation



Mask R-CNN

- Faster R-CNN for object region proposal and object classification
- Parallel branch that predicts mask of the object
- Pixel wise convolution (1x1) – binary cross entropy for mask learning



- ROI Align – sub pixel precision of the mask features – uses bilinear interpolation of features
- 28x28 target maps
- At prediction time the 28x28 maps are resized to the predicted object region