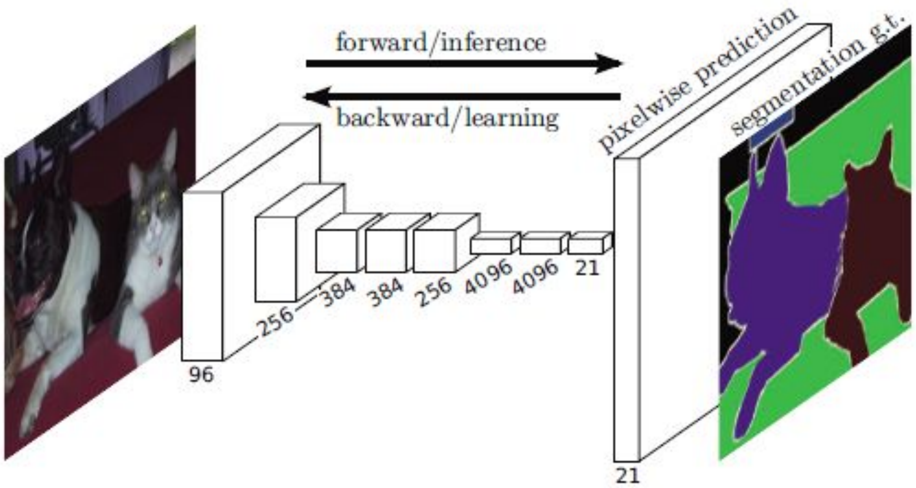
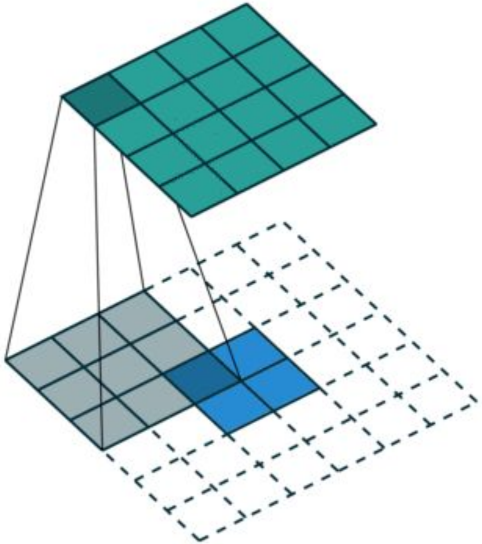
Основные архитектуры для задачи сегментации

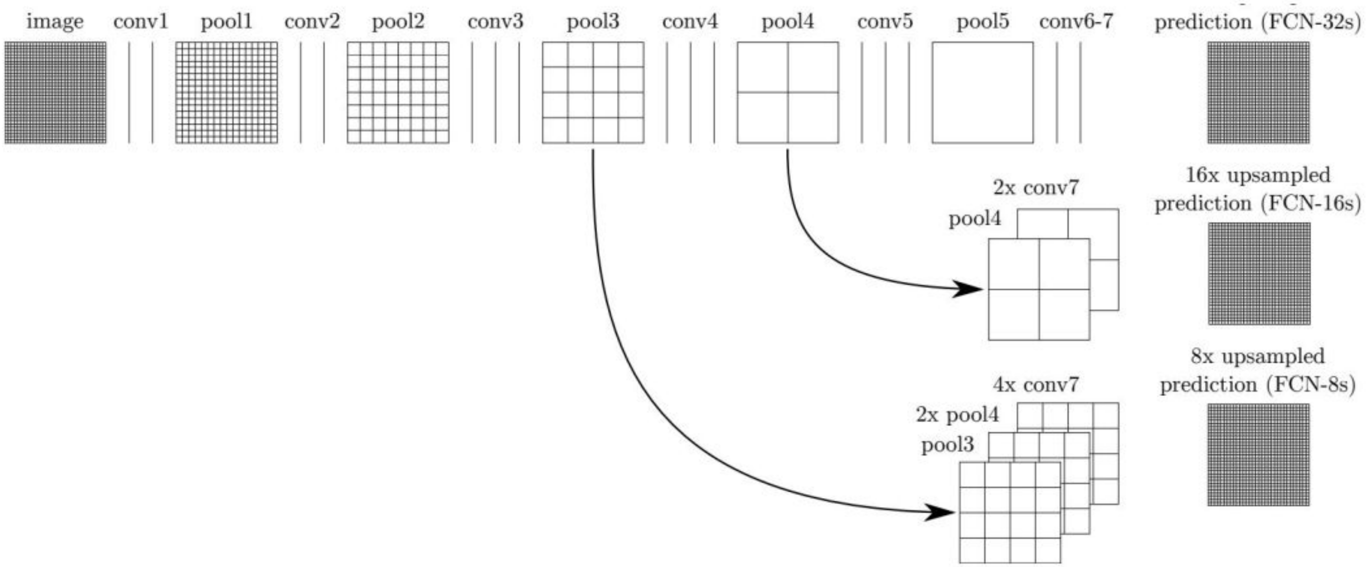
Fully convolutional network



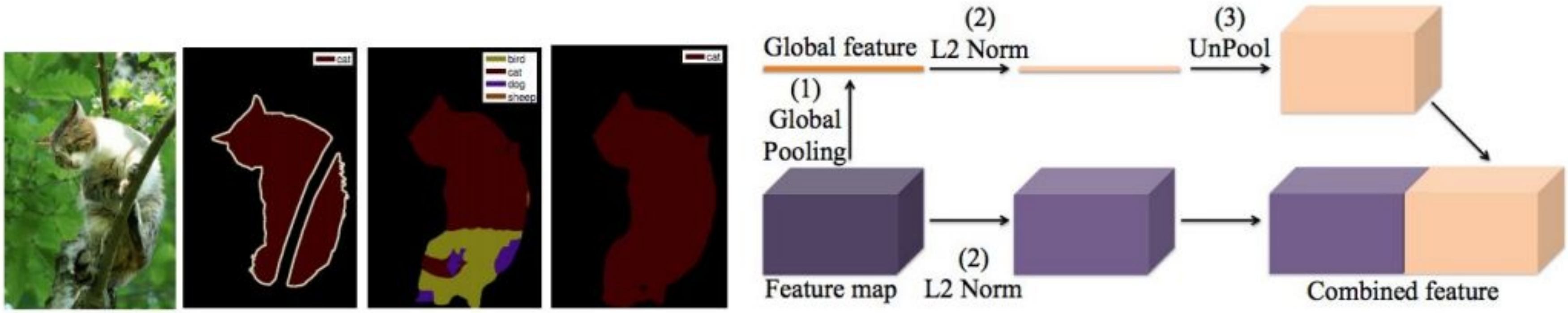
Deconvolution



Fully convolutional network

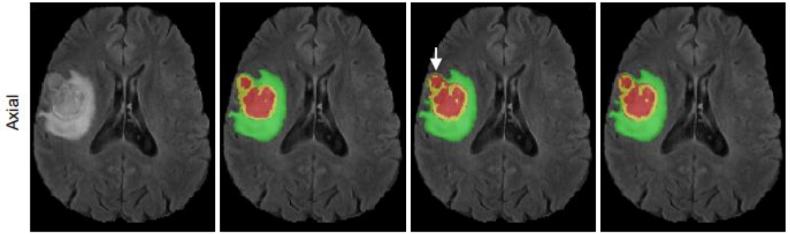


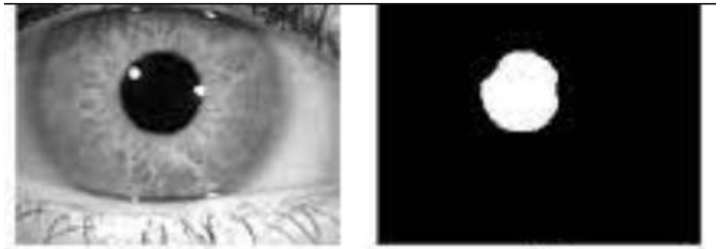
ParseNet



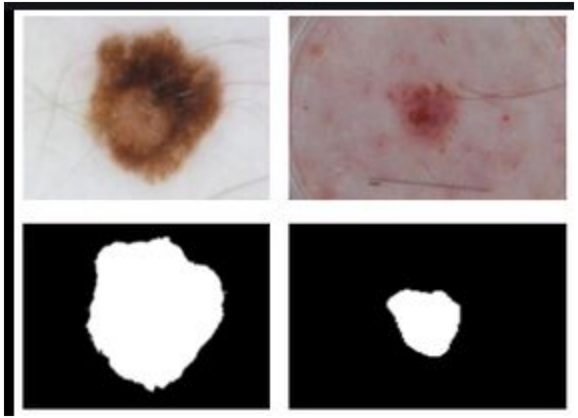
(a) Image (b) Truth (c) FCN (d) ParseNet (e) ParseNet contexture module overview.

Fully convolutional network

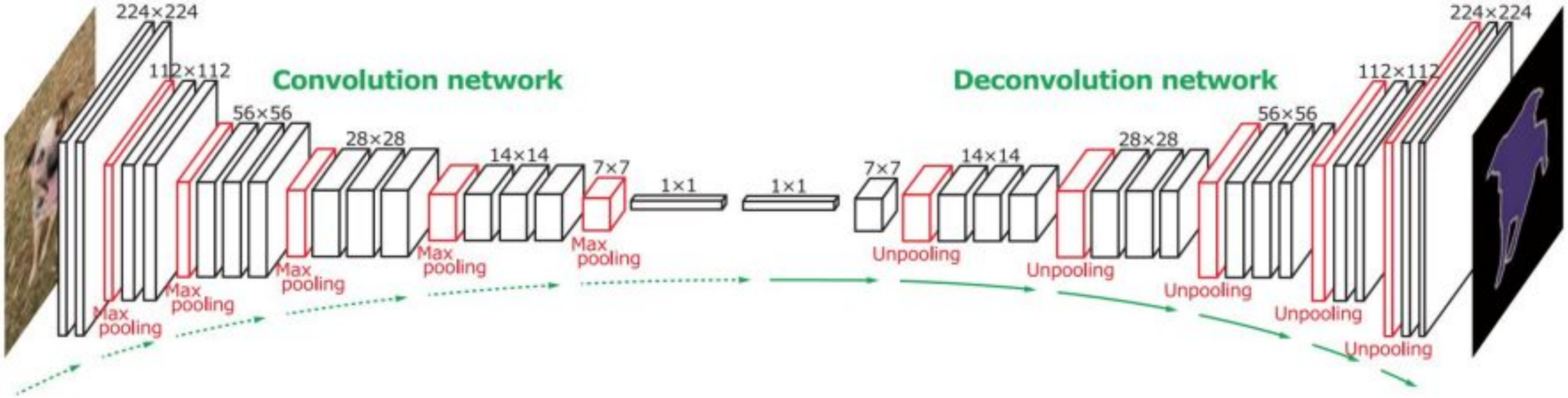




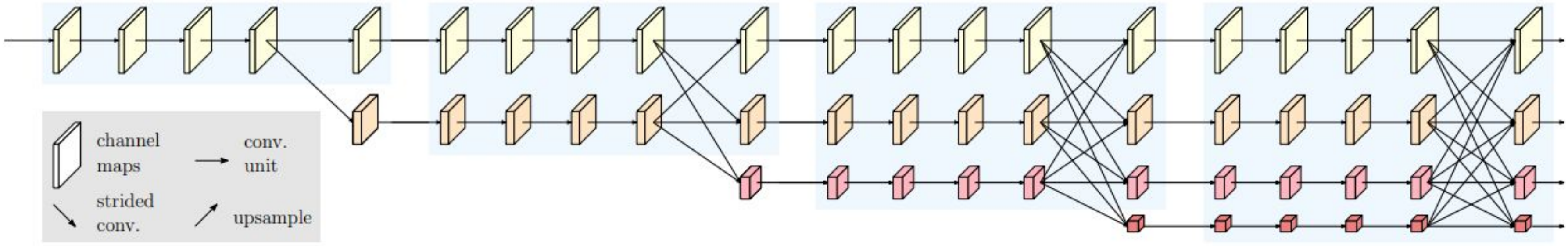
(a) (b)



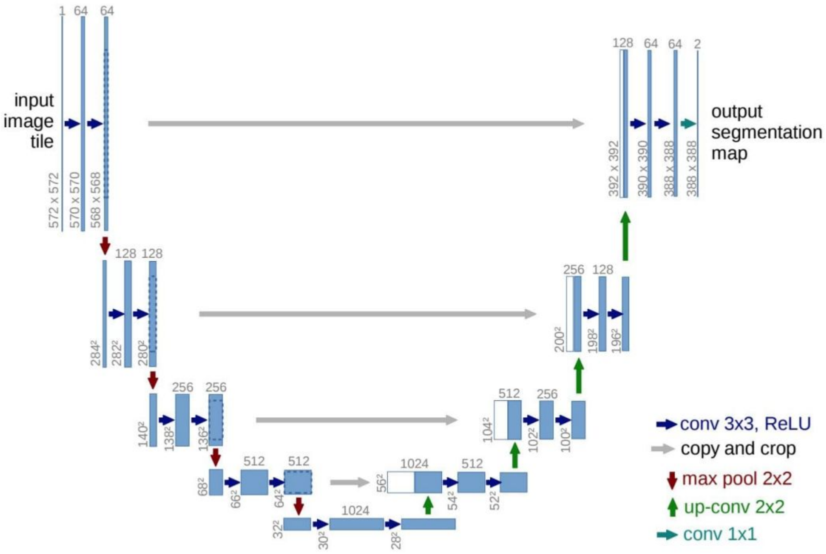
Encoder-Decoder



HRNet

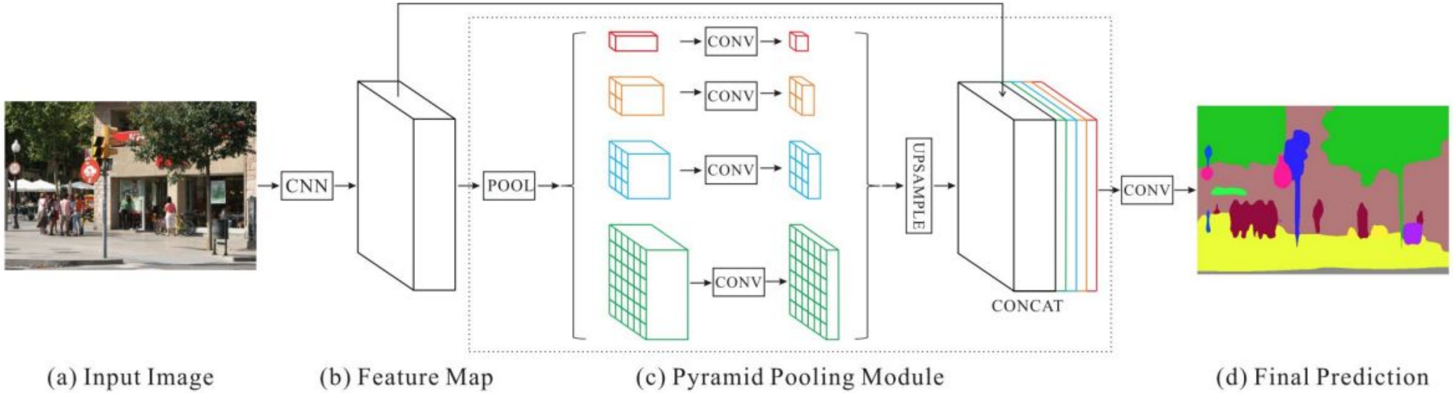


Unet

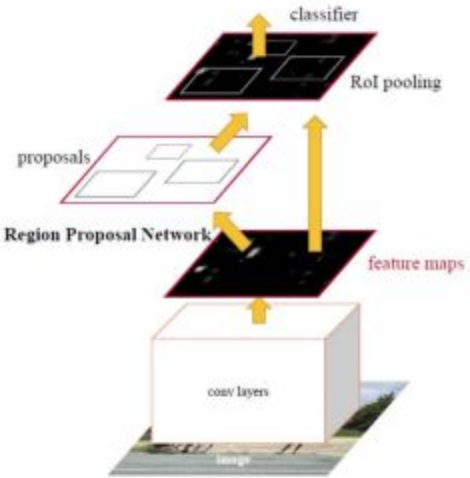


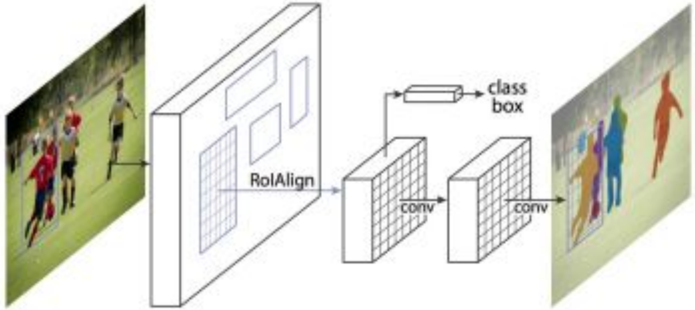
input image tile
output segmentation map
conv 3x3, ReLU
copy and crop
max pool 2x2
up-conv 2x2
conv 1x1

PSPN architecture



R-CNN Based Models





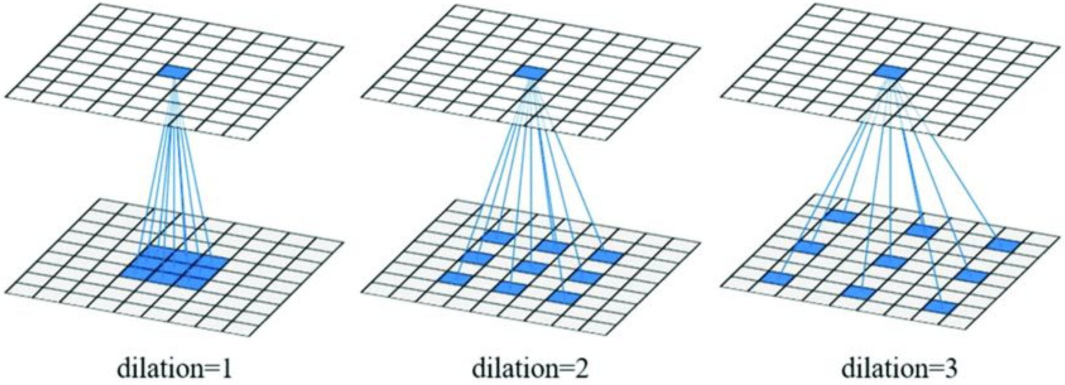
Заключение

Современные архитектуры сегментации

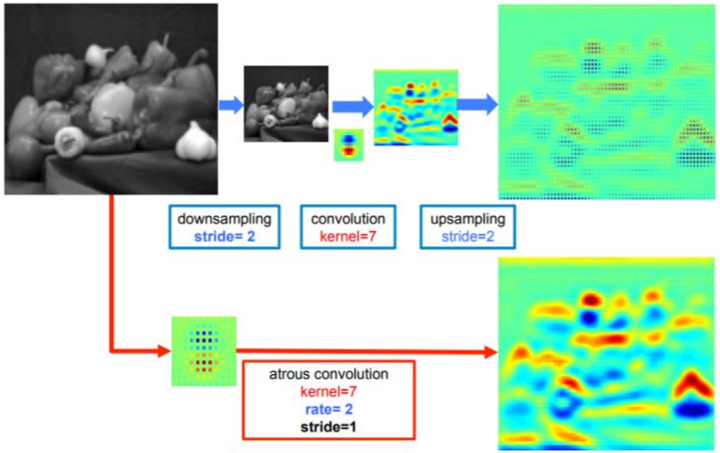
Dilated Convolutional

$$y_i = \sum_{k=1}^{K} x[i + rk]w[k]$$

1. Multiscale context aggregation
2. Dense upsampling convolution, Hybrid dilated convolution
3. Densely connected Atrous Spatial Pyramid Pooling (DenseASPP)
4. Efficient neural network (Enet)



Dilated Convolutional



Deeplab V1

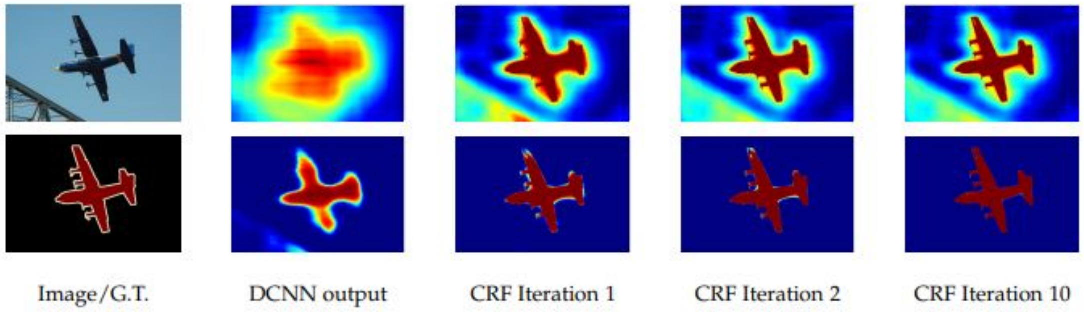
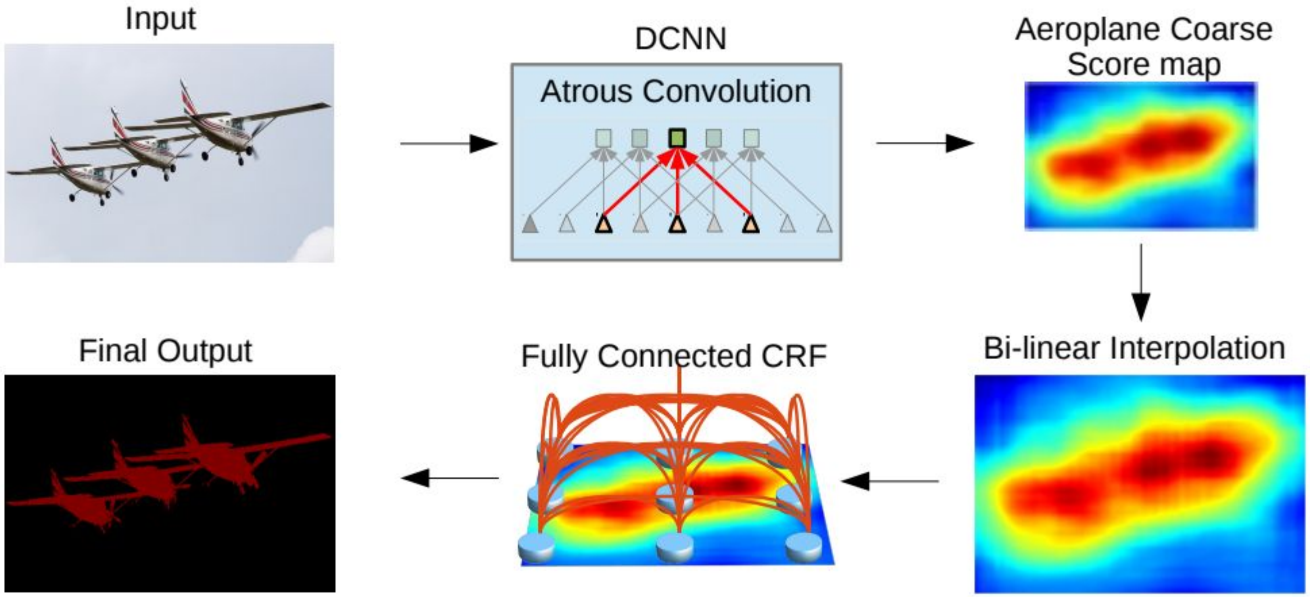
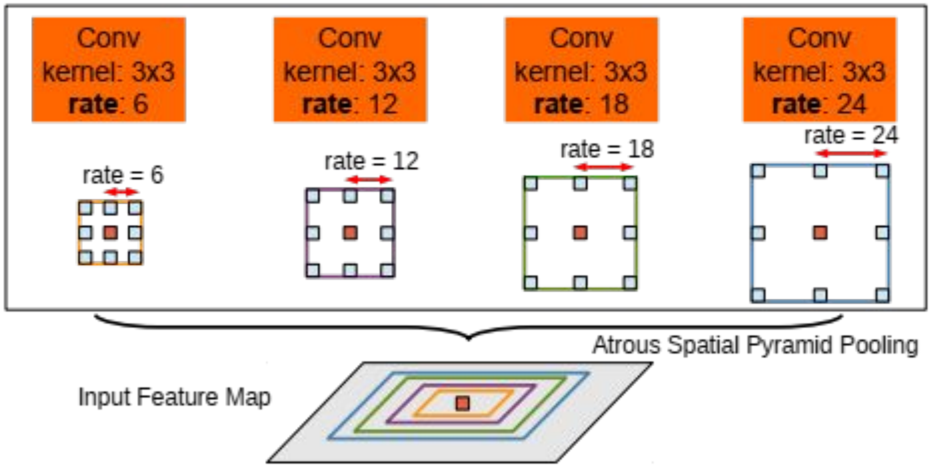


Fig. 5: Score map (input before softmax function) and belief map (output of softmax function) for Aeroplane. We show the score (1st row) and belief (2nd row) maps after each mean field iteration. The output of last DCNN layer is used as input to the mean field inference.

Deeplab V1

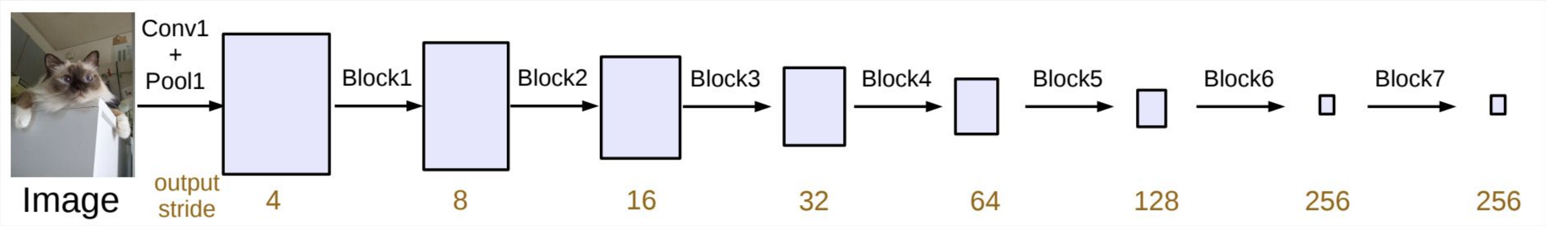

Input
DCNN
Atrous Convolution
Aeroplane Coarse Score map
Bi-linear Interpolation
Fully Connected CRF
Final Output

Deeplab V2

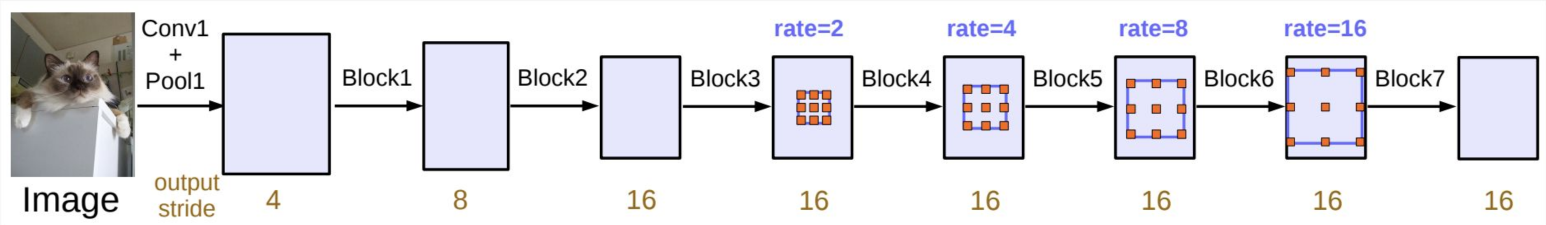


Conv
kernel: 3x3
rate: 6
Conv
kernel: 3x3
rate: 12
Conv
kernel: 3x3
rate: 18
Conv
kernel: 3x3
rate: 24
rate = 6
rate = 12
rate = 18
rate = 24
Atrous Spatial Pyramid Pooling
Input Feature Map

Deeplab V3



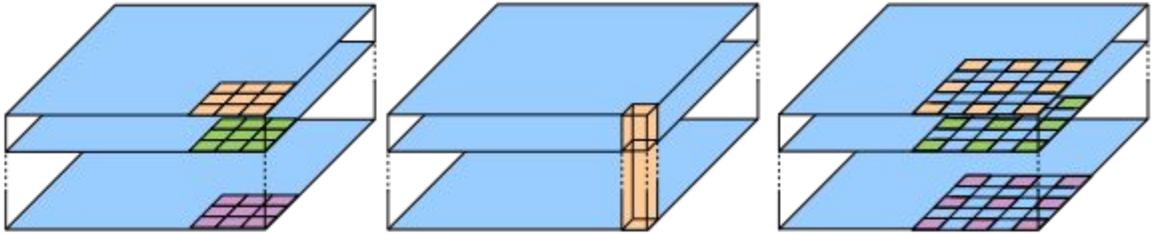
(a) Going deeper without atrous convolution.



(b) Going deeper with atrous convolution. Atrous convolution with $rate > 1$ is applied after block3 when $output\_stride = 16$.

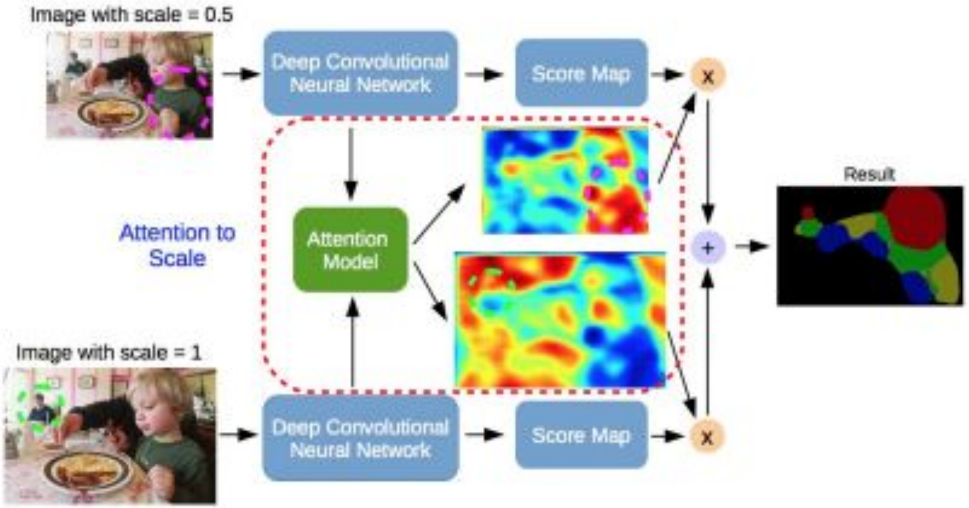
Figure 3. Cascaded modules without and with atrous convolution.

Deeplab V3

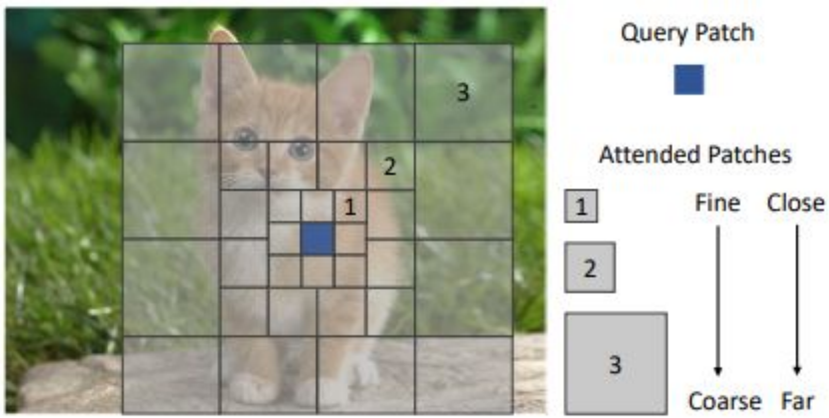


(a) Depthwise conv. (b) Pointwise conv. (c) Atrous depthwise conv.

Attention модели



Self-attention и Патчи



https://arxiv.org/abs/2107.00641

Focal Transformer

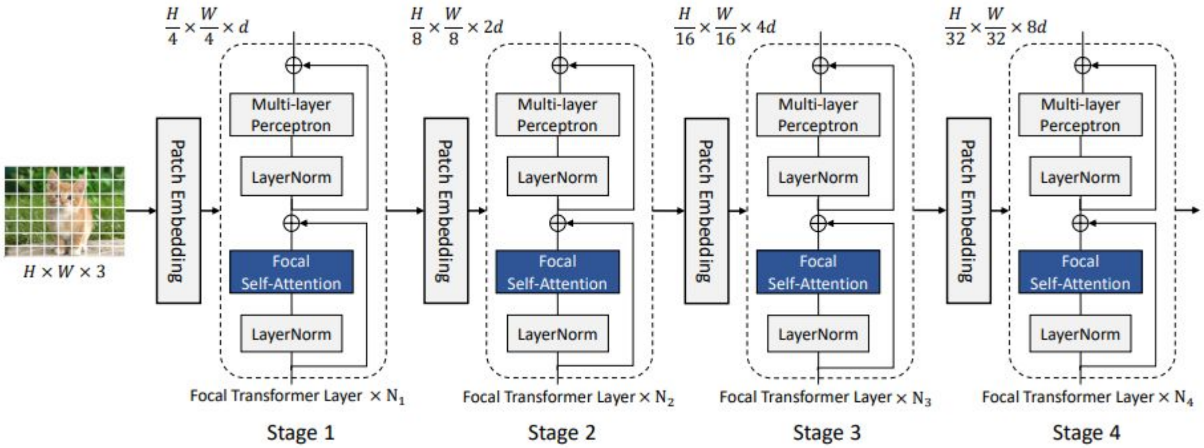
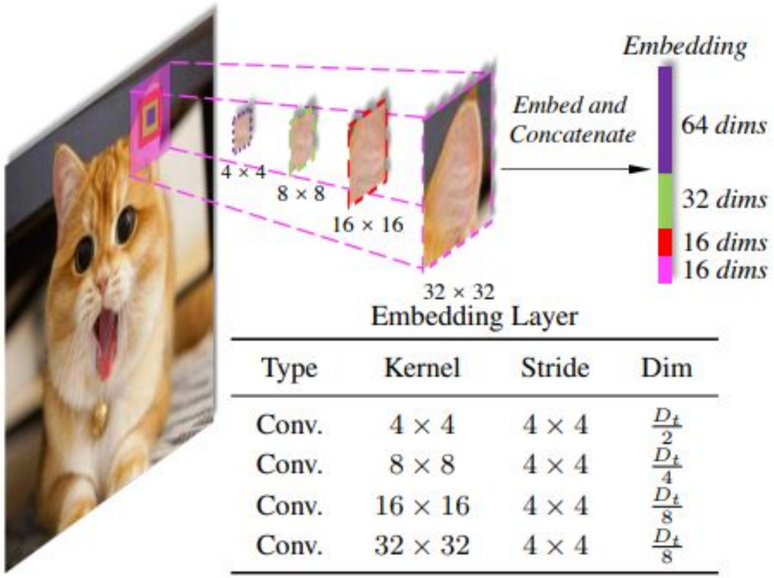


Figure 2: Model architecture for our Focal Transformers. As highlighted in light blue boxes, our main innovation is the proposed focal self-attention mechanism in each Transformer layer.

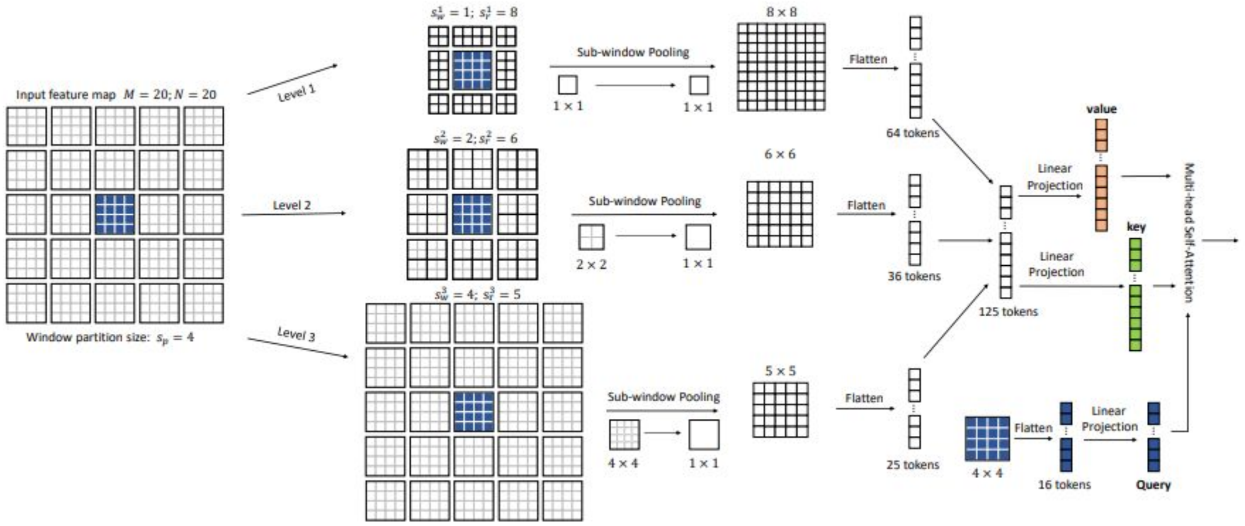
Пример



Embedding Layer

Type	Kernel	Stride	Dim
Conv.	4×4	4×4	$\frac{D_t}{2}$
Conv.	8×8	4×4	$\frac{D_t}{4}$
Conv.	16×16	4×4	$\frac{D_t}{8}$
Conv.	32×32	4×4	$\frac{D_t}{8}$

Self-attention



Заключение