Introduction to Semantic Segmentation

Sergei Belousov

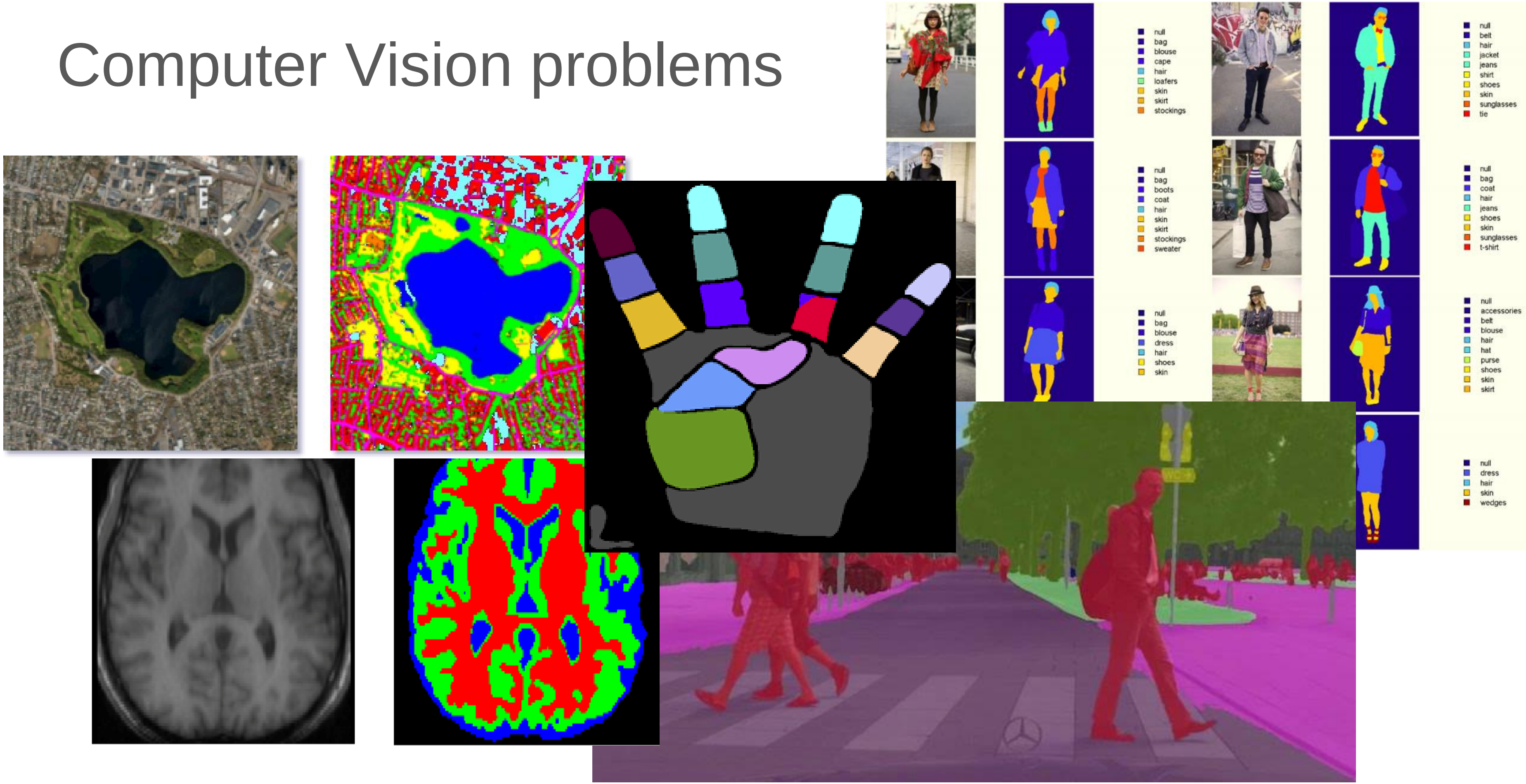
Machine learning R&D Engineer

Internet of Things Group

Agenda

- Problem formulation
- Datasets
- Evaluation metrics
- Architectures
- Loss functions
- Results

Computer Vision problems

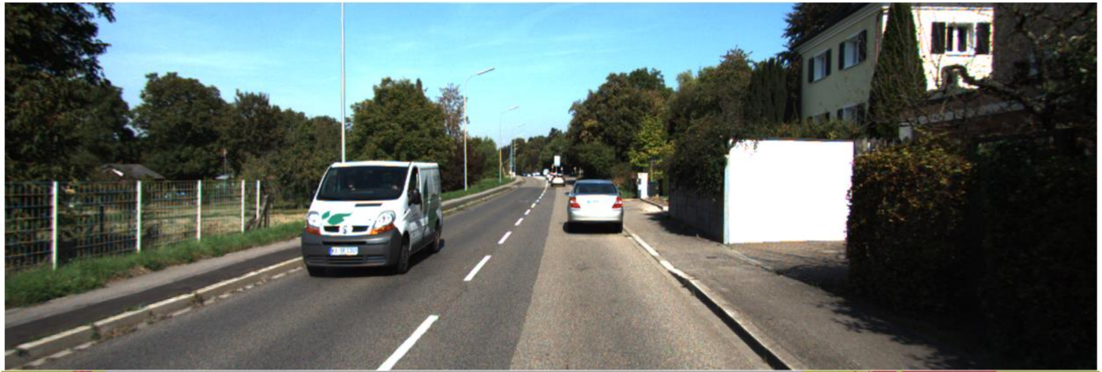


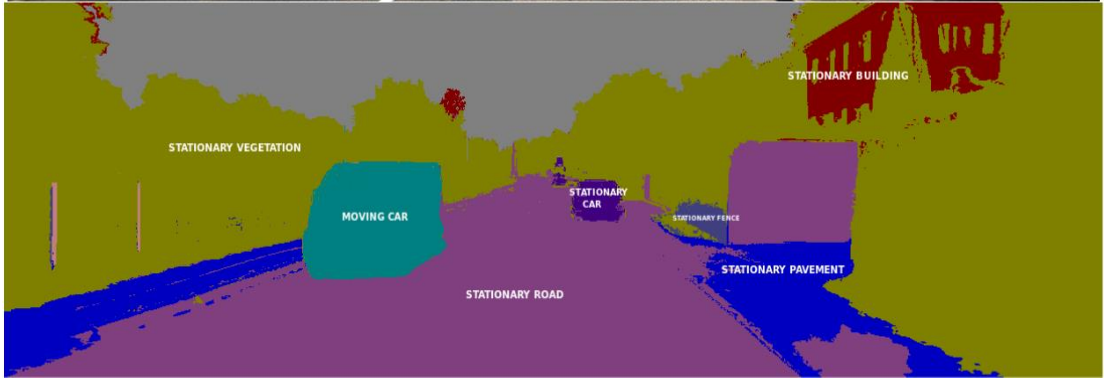
Problem formulation

Input:

$$I \in R^{C*H*W} - input\ image$$

$$L \in [l_0, \ldots l_n] - set\ of\ valid\ labels$$

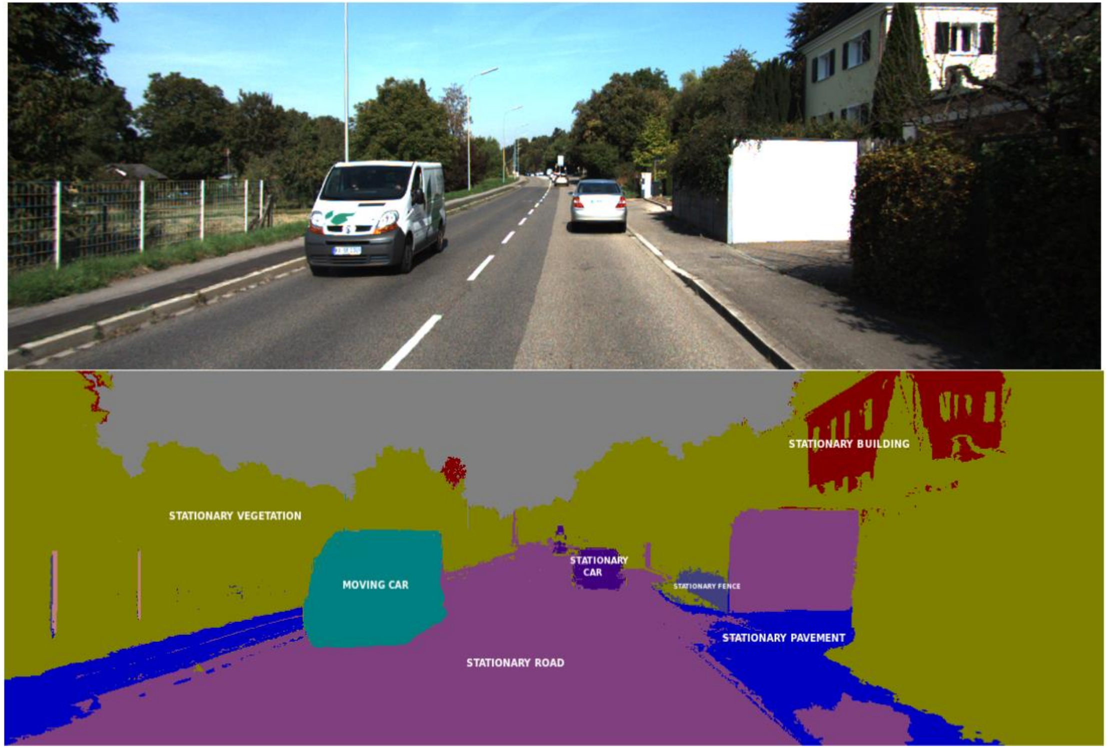




Output:

$$M \in L^{H*W} - labels\ mask$$

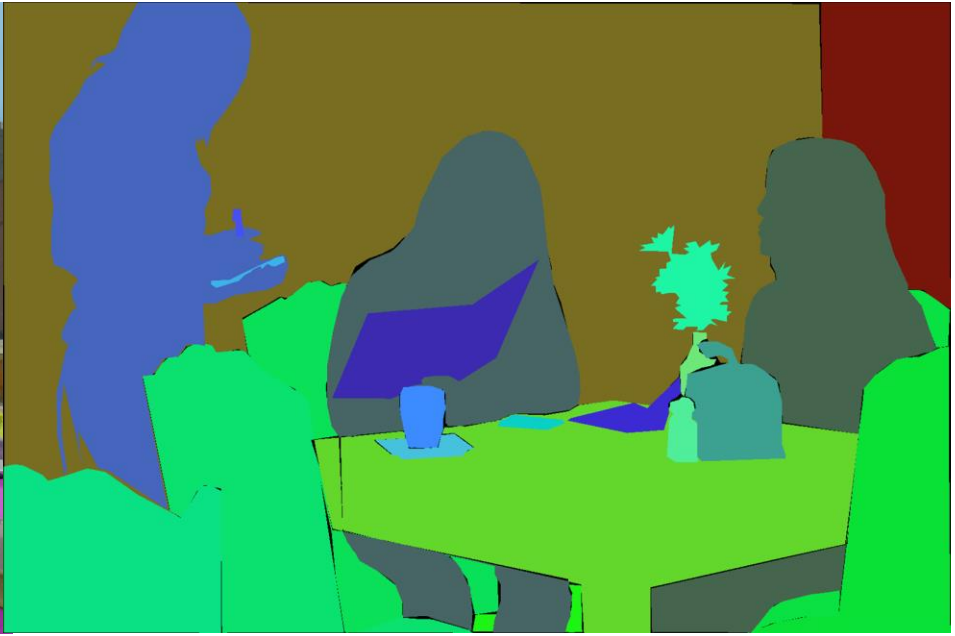
Datasets


STATIONARY BUILDING
STATIONARY VEGETATION
MOVING CAR
STATIONARY CAR
STATIONARY FENCE
STATIONARY PAVEMENT
STATIONARY ROAD

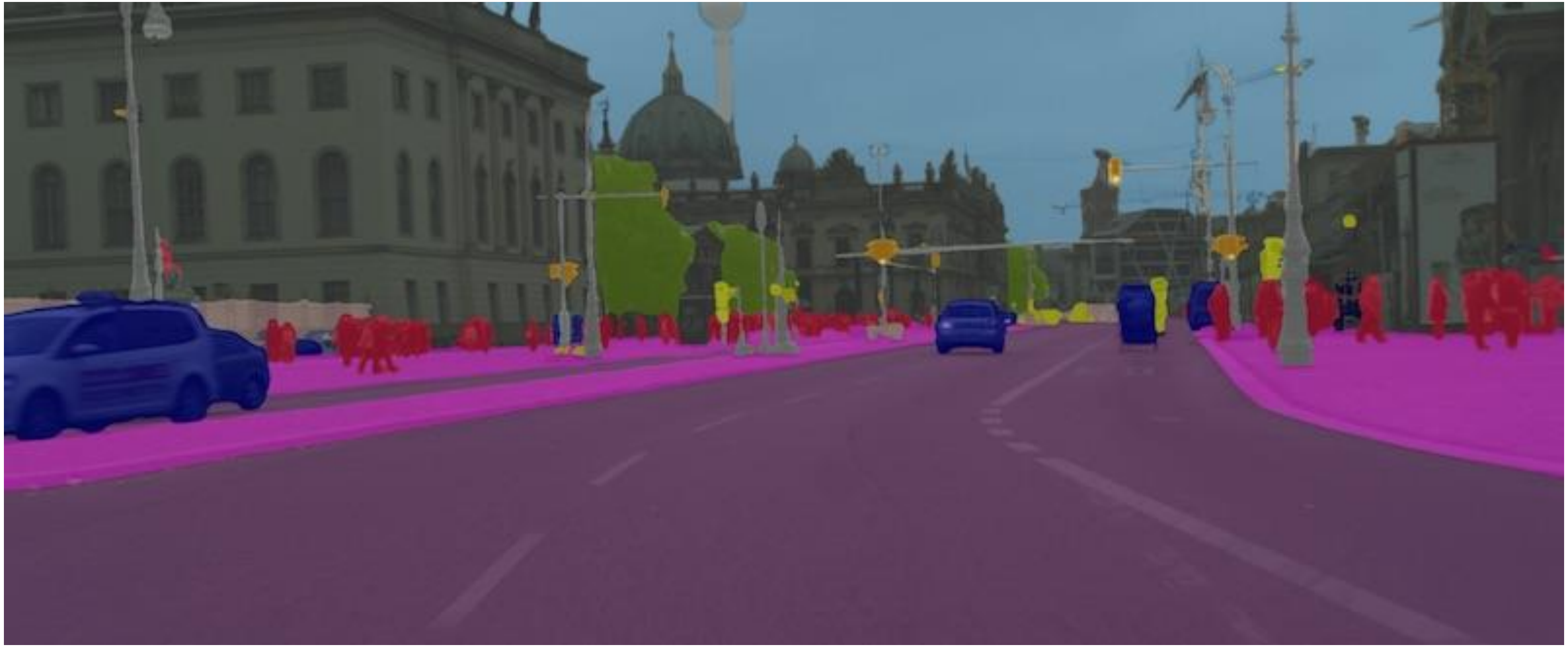
Datasets

Dataset	Labeled Images for Training	Classes
KITTI	200	34
VOC PASCAL 2012	2913	21
Cityscapes	3478	34
BDD100K	8000	19
ADE20K	20210	3169
Mapillary Vistas	20000	66
ApolloScape	147000	36
WAYMO	600000	?

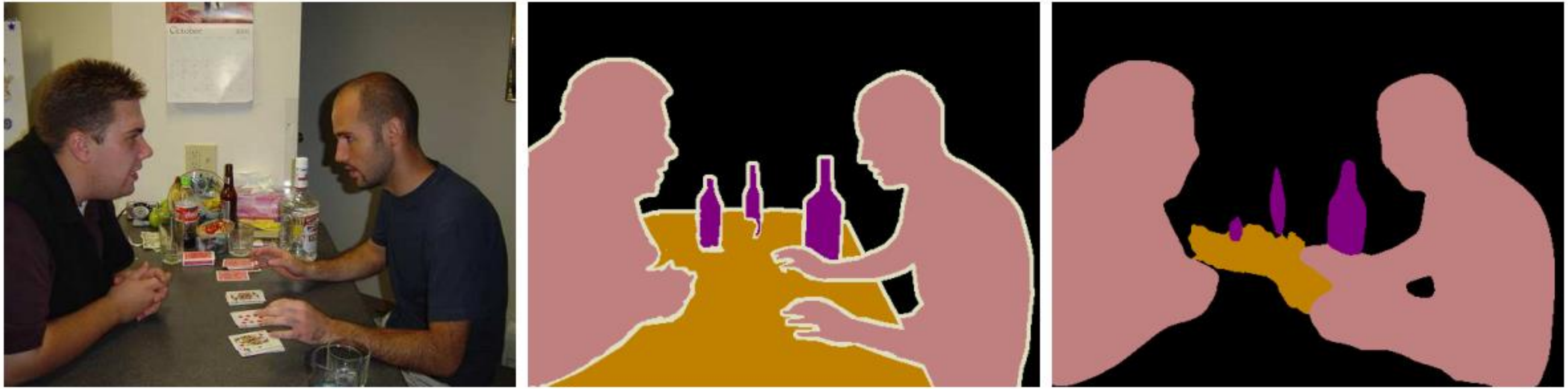
Datasets



Datasets

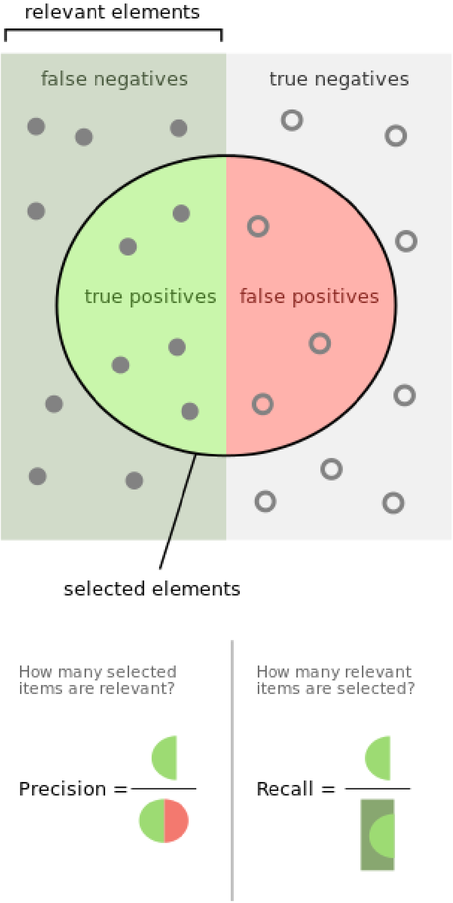


Evaluation metrics



Evaluation metrics

$$accuracy = \frac{TP+TN}{TP+TN+Fp+FN}$$


relevant elements

false negatives

true negatives

true positives

false positives

selected elements

How many selected items are relevant?

How many relevant items are selected?

Precision =

Recall =

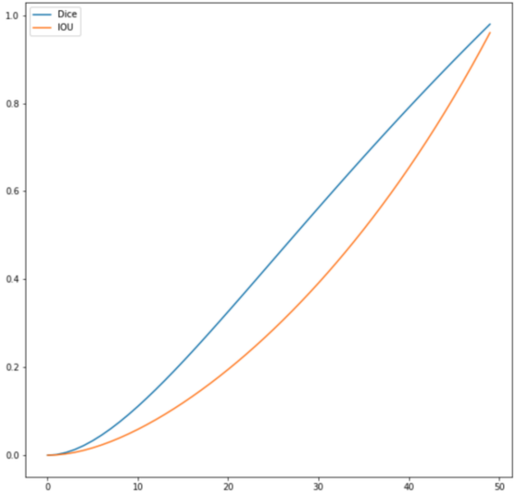
Evaluation metrics

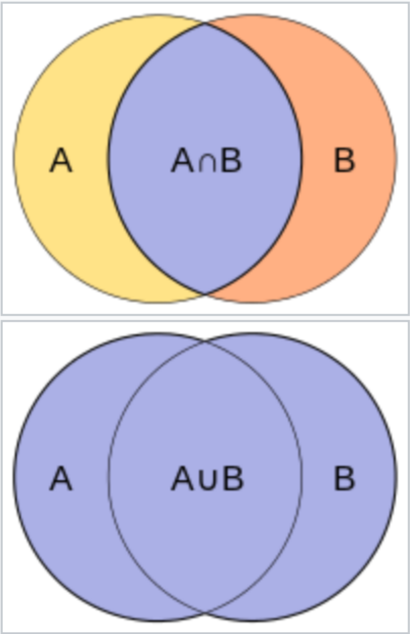
$$Dice(A,B) = 2\frac{|A \cap B|}{|A|+|B|} = \frac{2TP}{2TP+FN+FP}$$

$$IOU(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{TP}{TP+FN+FP}$$

$$IOU = \frac{Dice}{2-Dice}$$

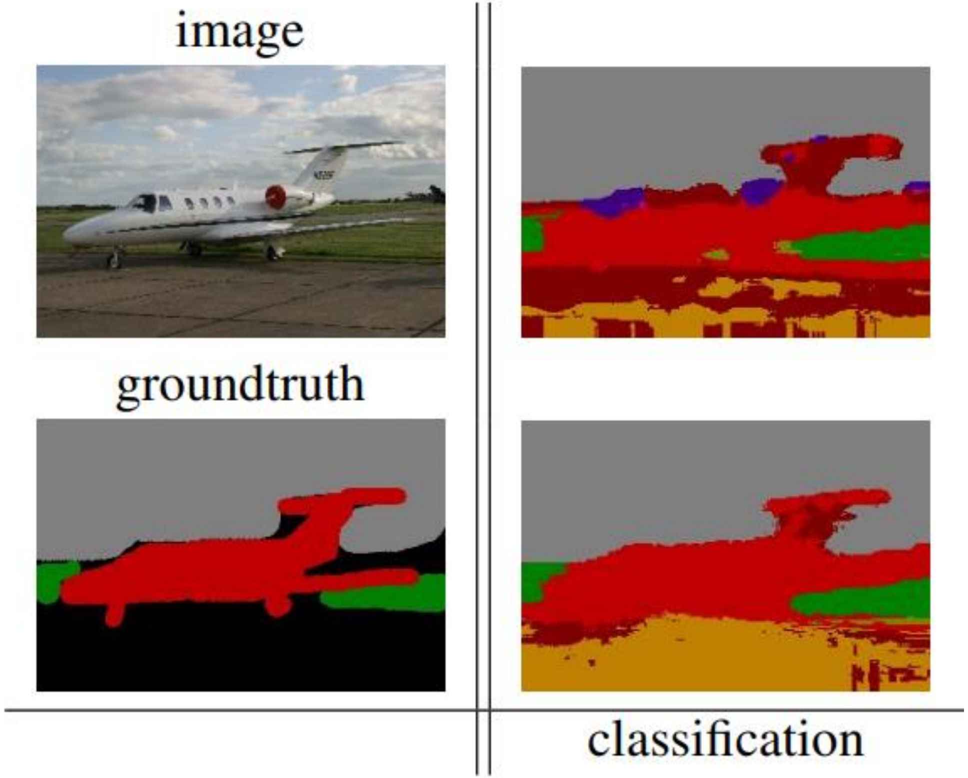
$$Error_{total} = c_0 FP + c_1 FN$$



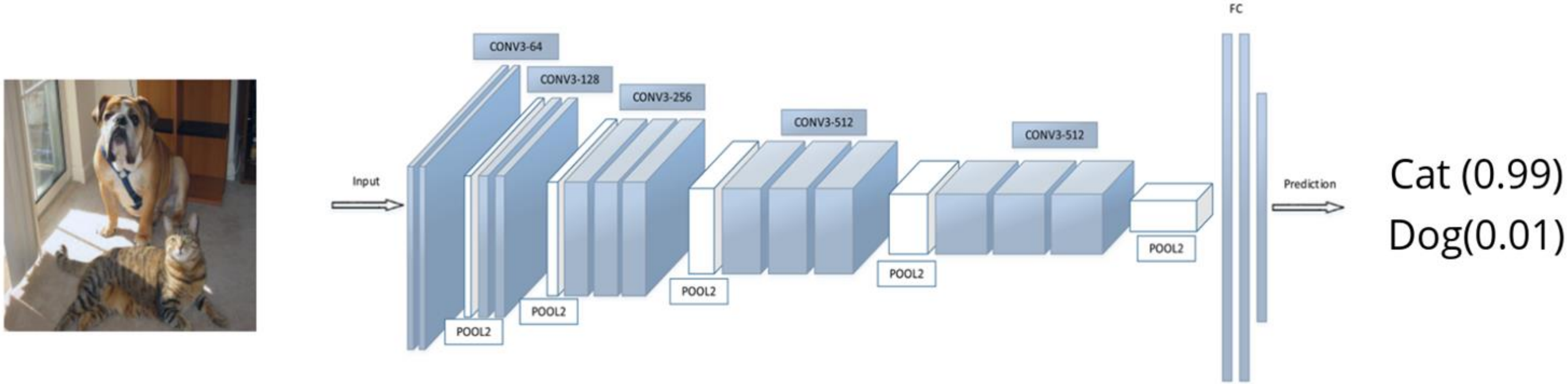


Intersection and union of two sets A and B

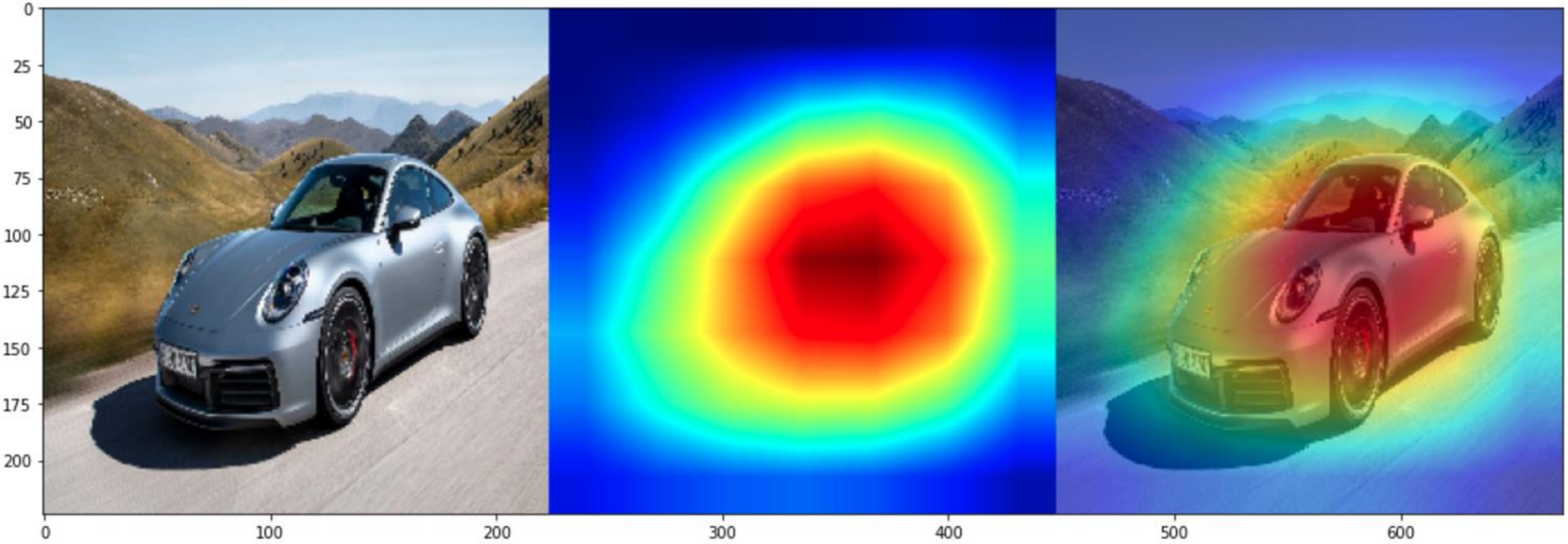
Architectures: In ancient time



Architectures: CNN

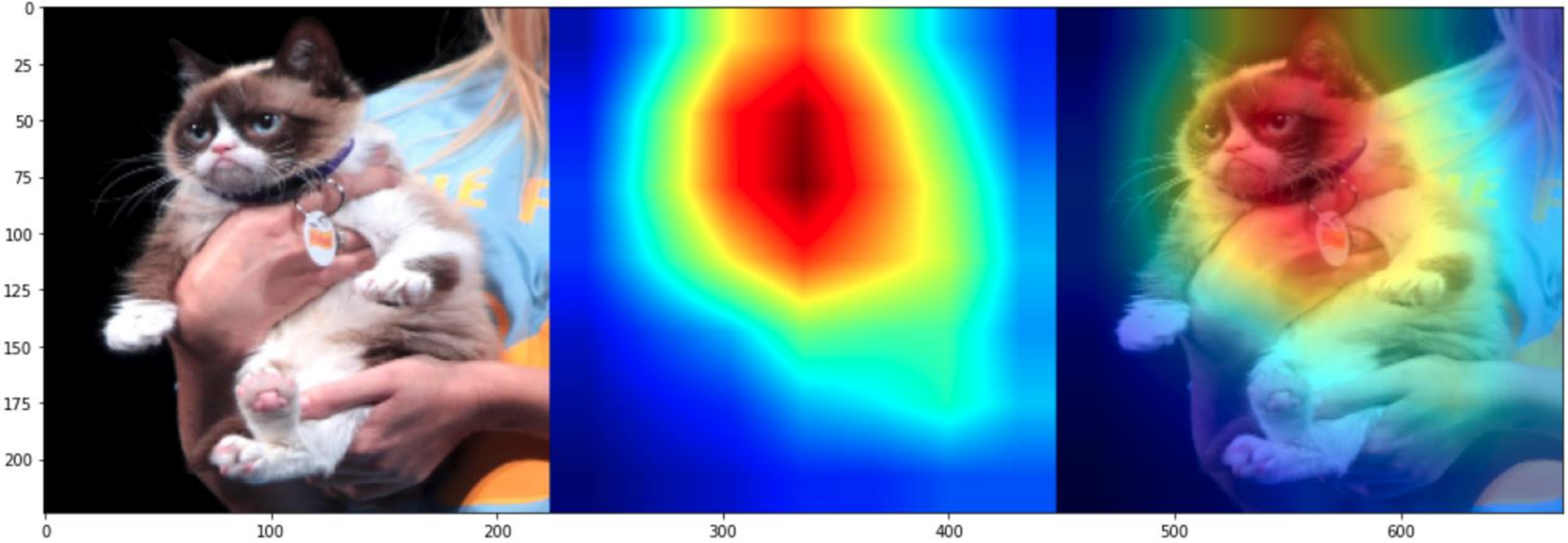


Architectures: Going deeper into ResNet18



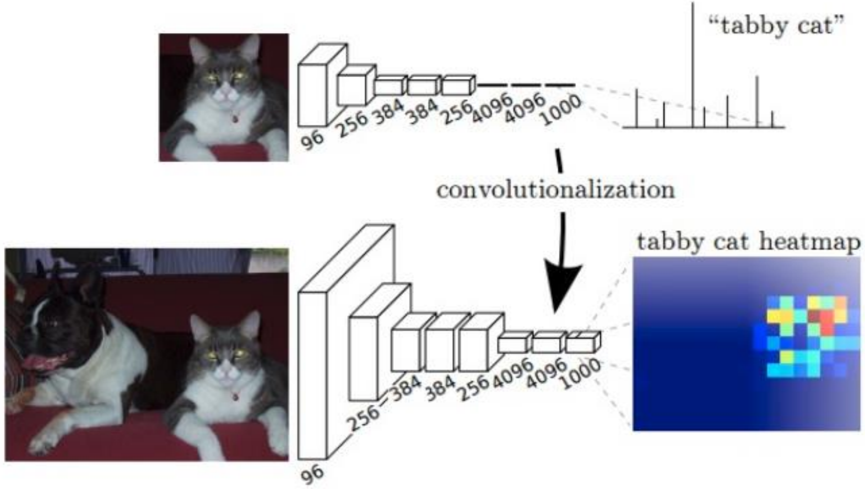
Label: Sport’s car

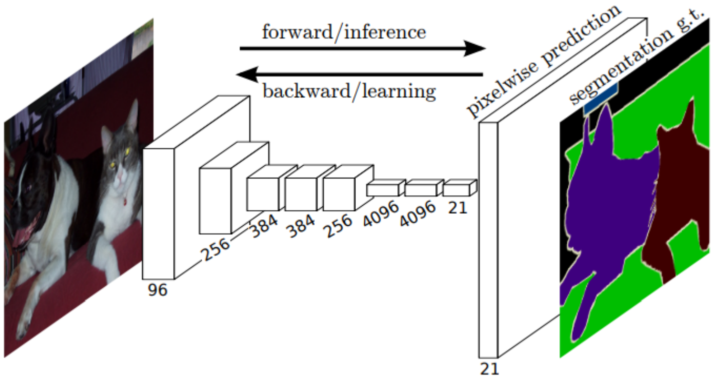
Architectures: Going deeper into ResNet18



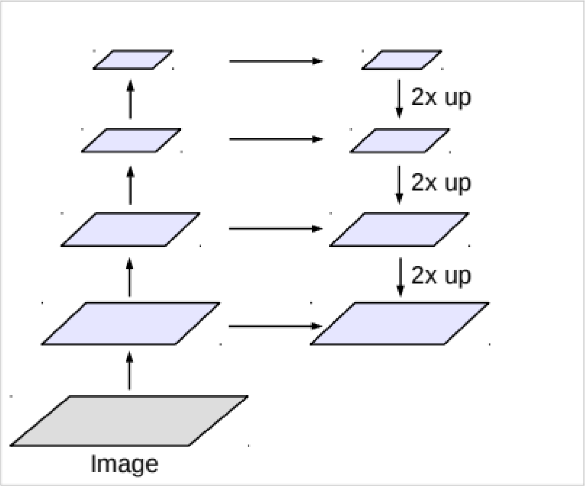
Label: Siamese cat

Architectures: FCN

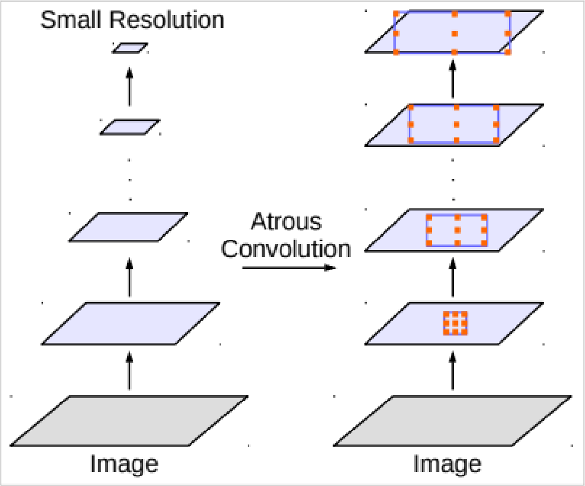




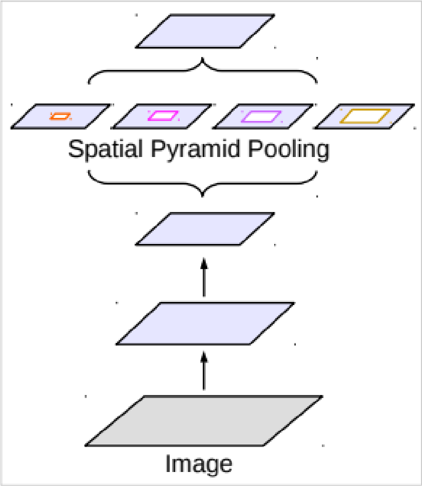
Architectures: Architectures to capture multi-scale context



(b) Encoder-Decoder

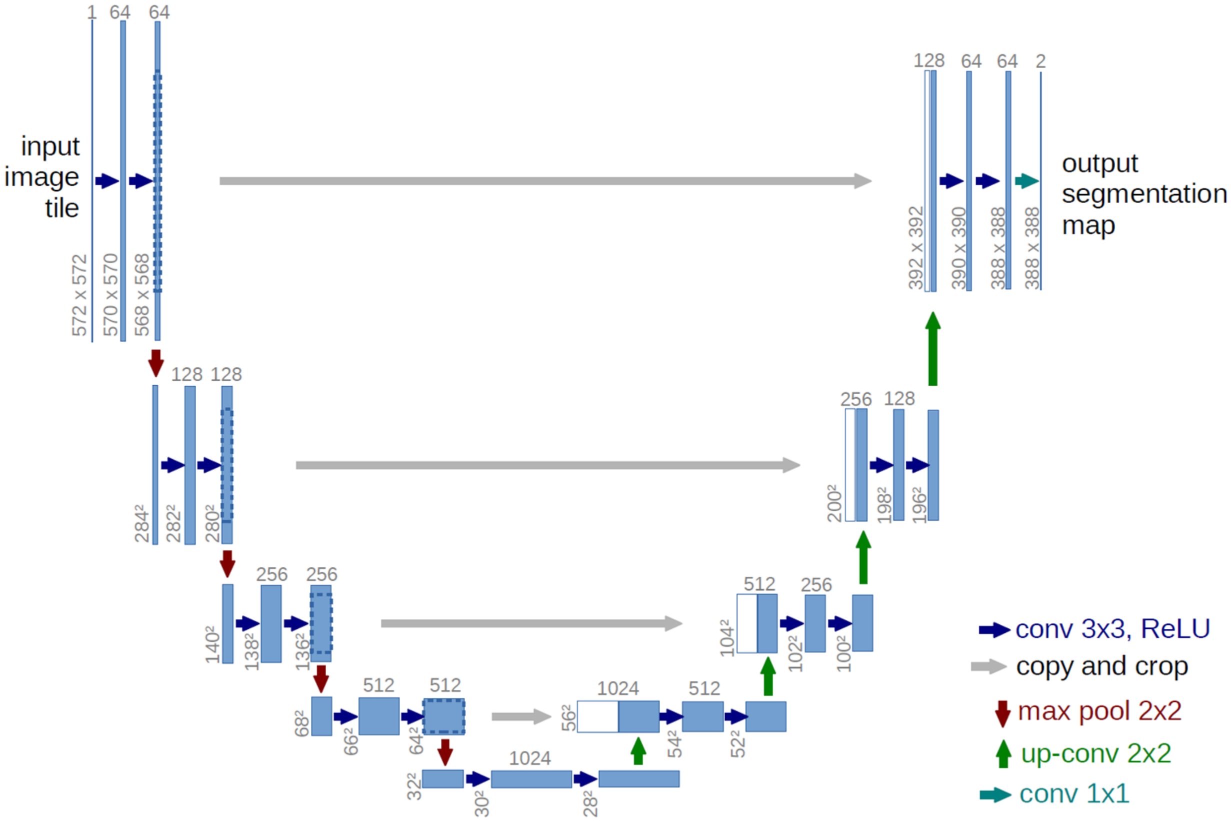


(c) Deeper w. Atrous Convolution

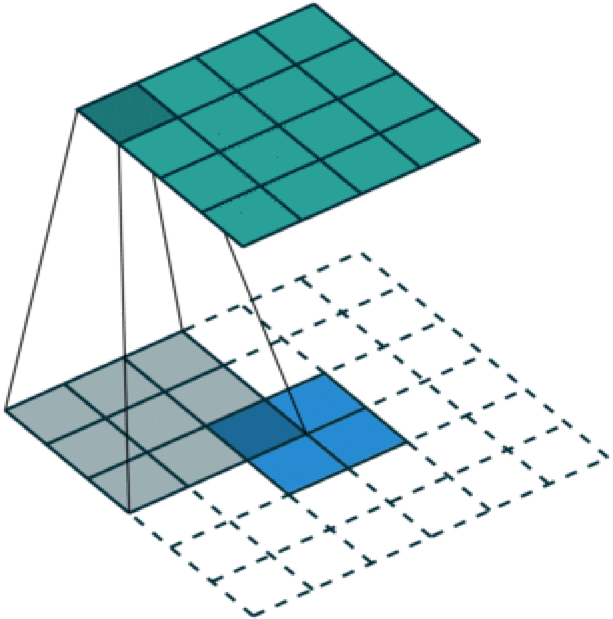


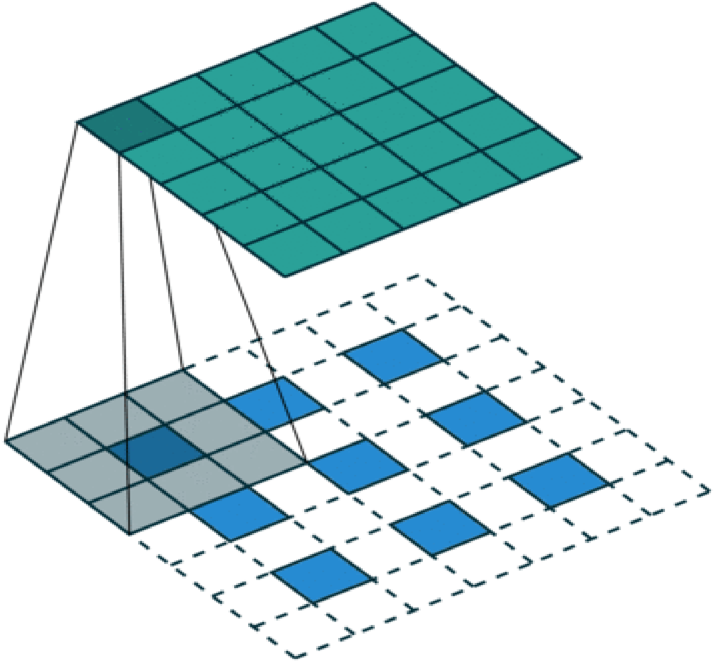
(d) Spatial Pyramid Pooling

CNN: Encoder-Decoder

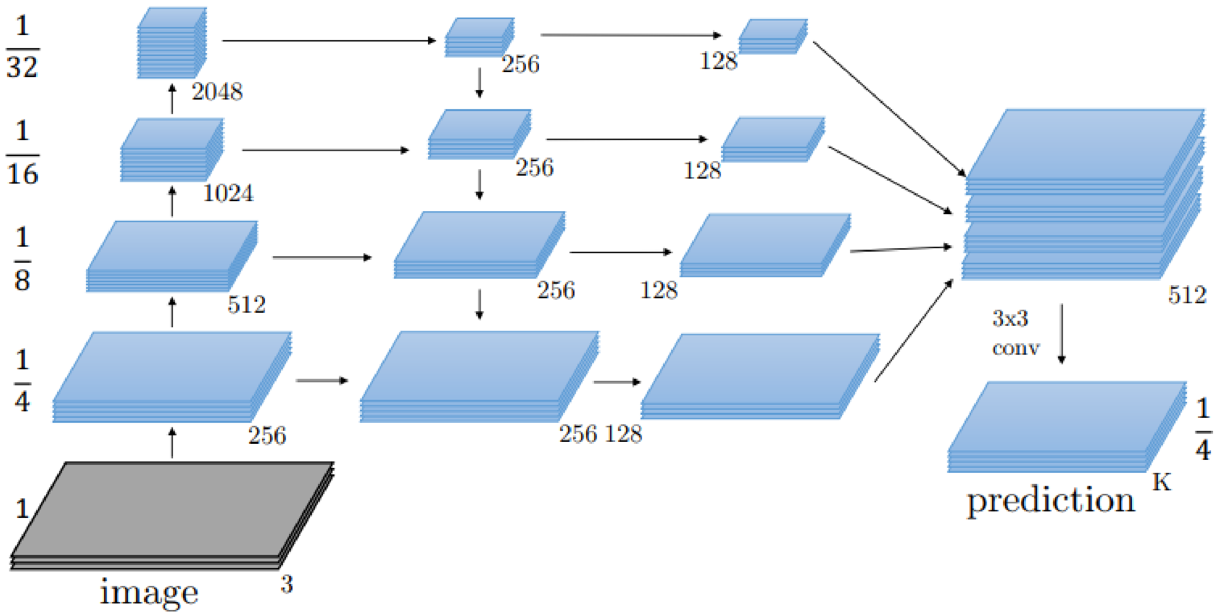


Architectures: Deconvolution

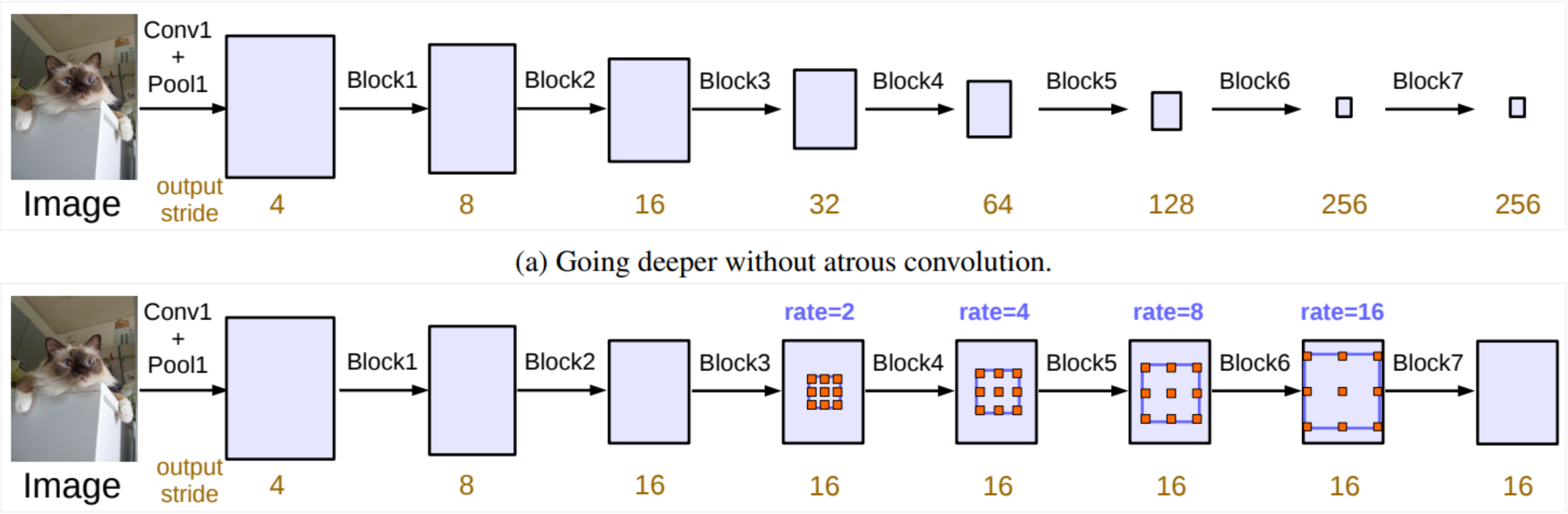




Architectures: Encoder-Decoder

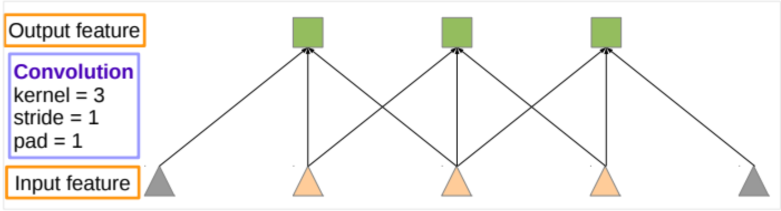


Architectures: w. Atrous Convolution

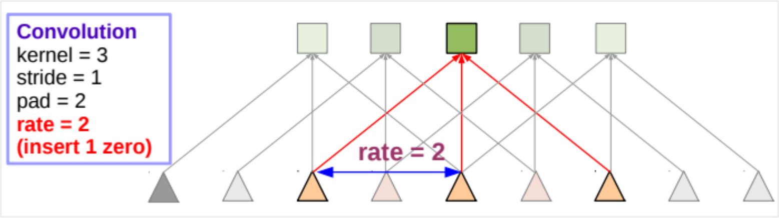


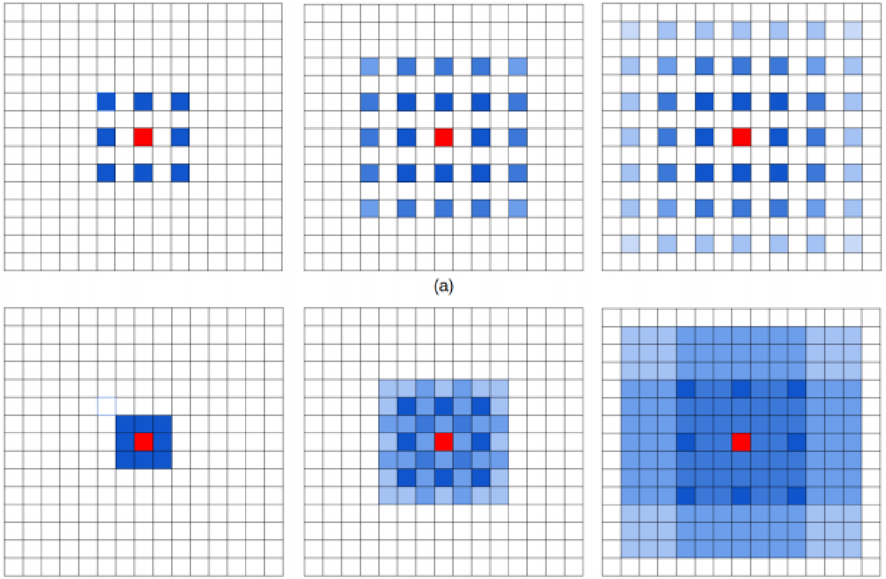
(a) Going deeper without atrous convolution.

CNN: w. Atrous Convolutions



(a) Sparse feature extraction

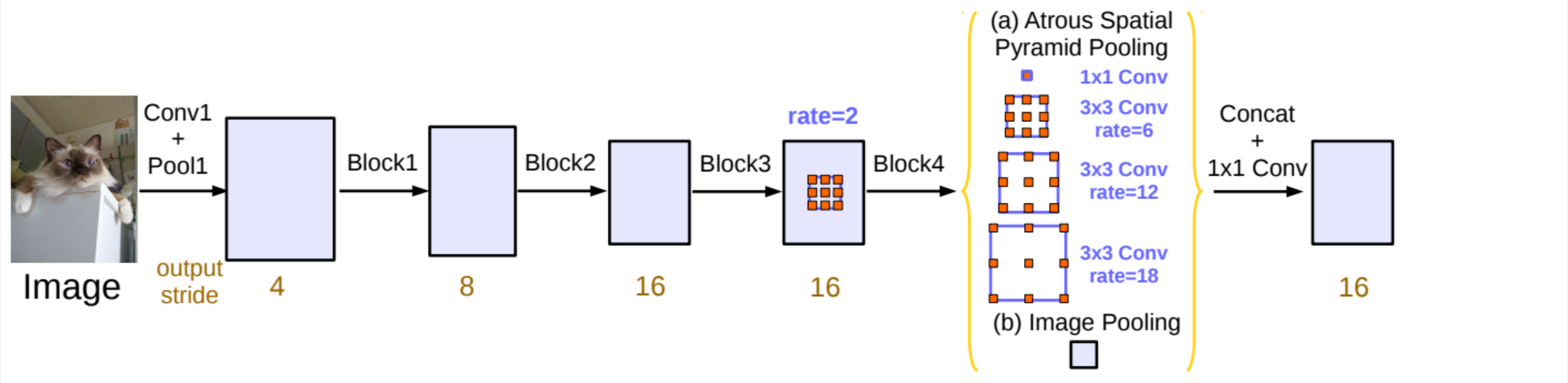




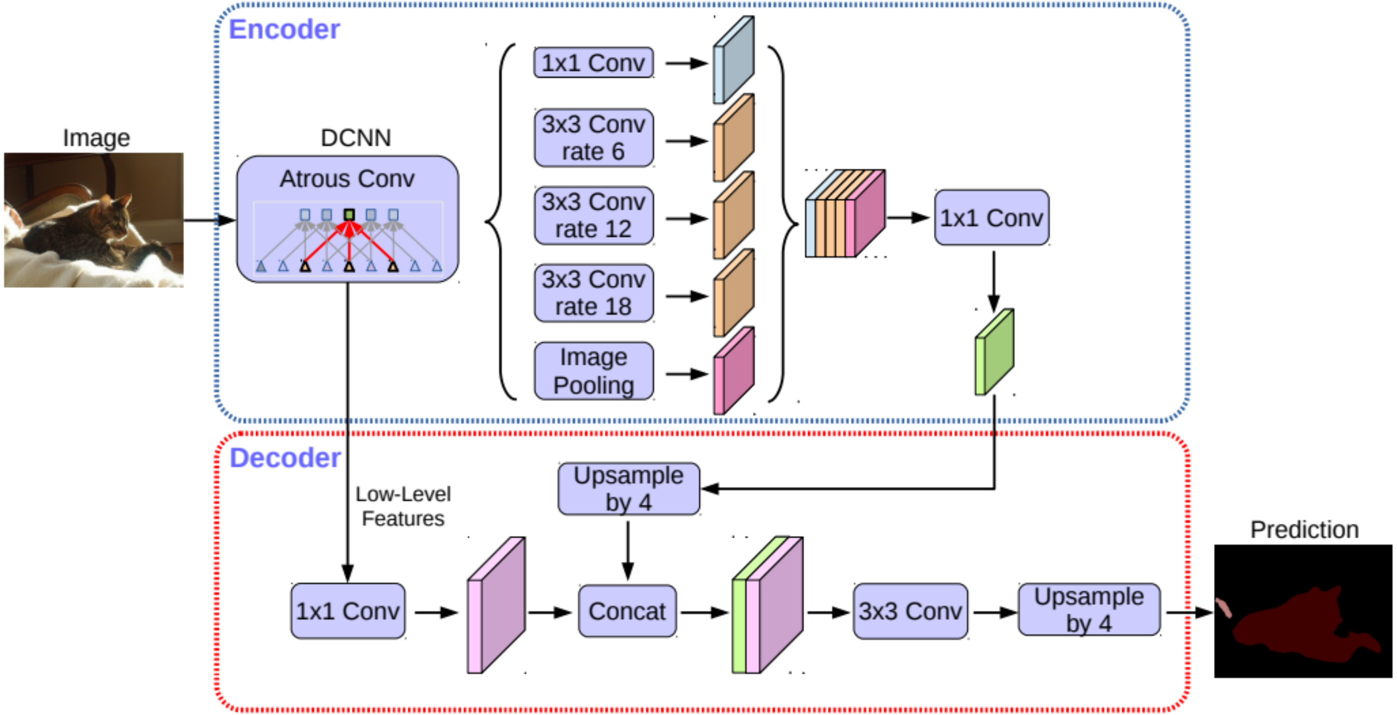
(a)

(b)

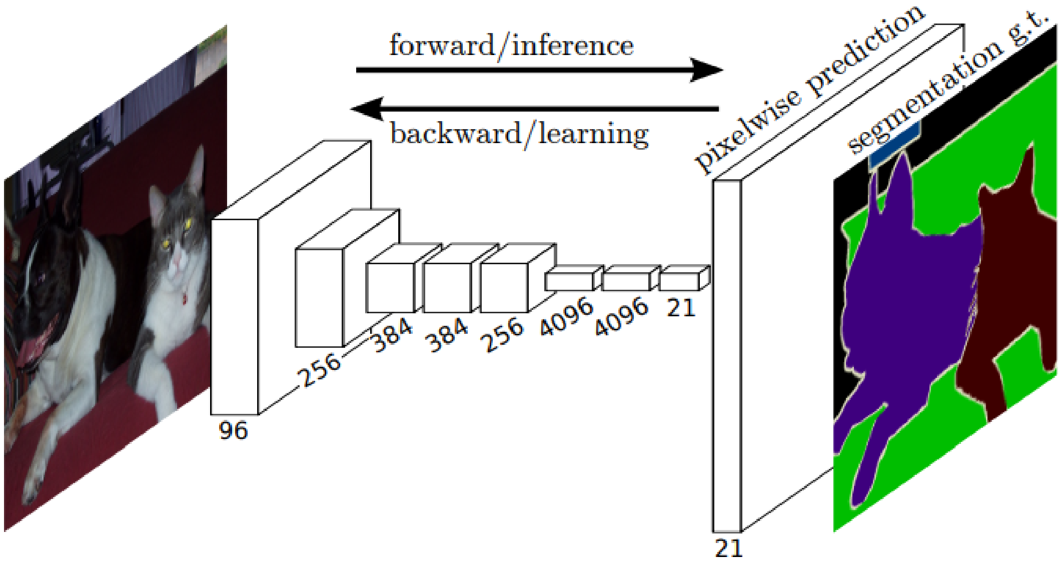
Architectures: Spatial Pyramid Pooling



Architectures: All inclusive

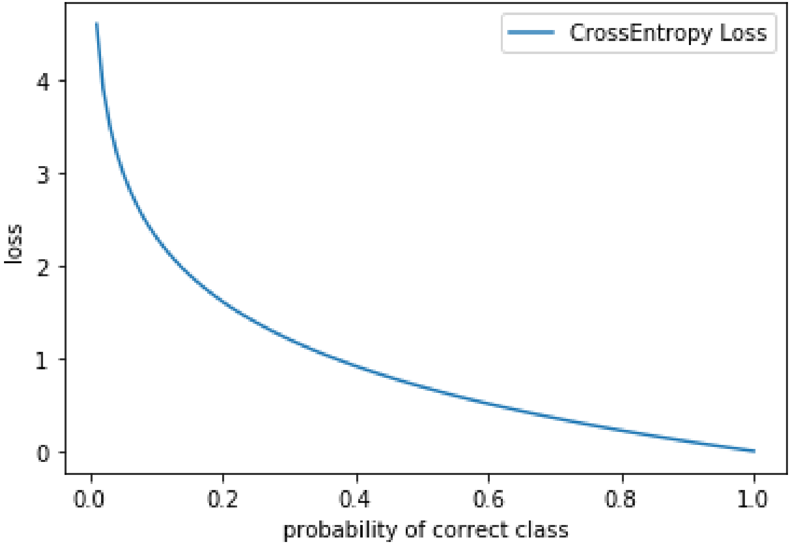


Loss functions



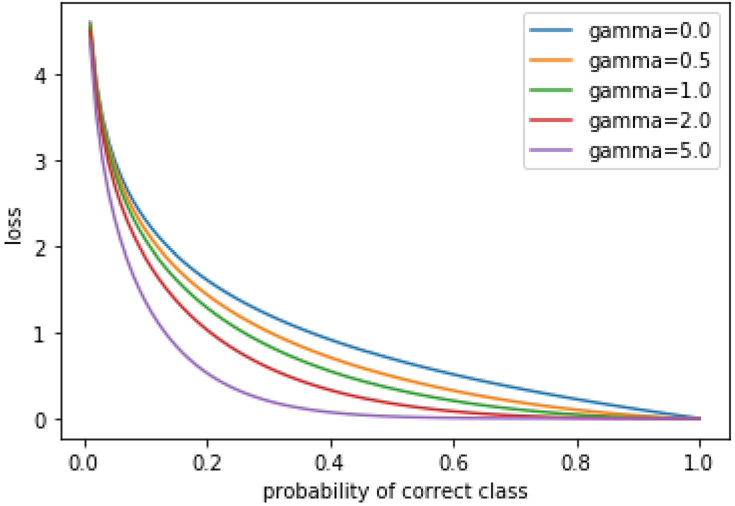
Loss functions

$$L_{CE}(p,y) = -\sum_{c=1}^{M} y_{o,c} \log(p_{o,c})$$



Loss functions

$$L_{Focal}(p, y) = -\sum_{c=1}^{M} y_{o,c} * (1 - p_{o,c})^{\gamma} * \log(p_{o,c})$$

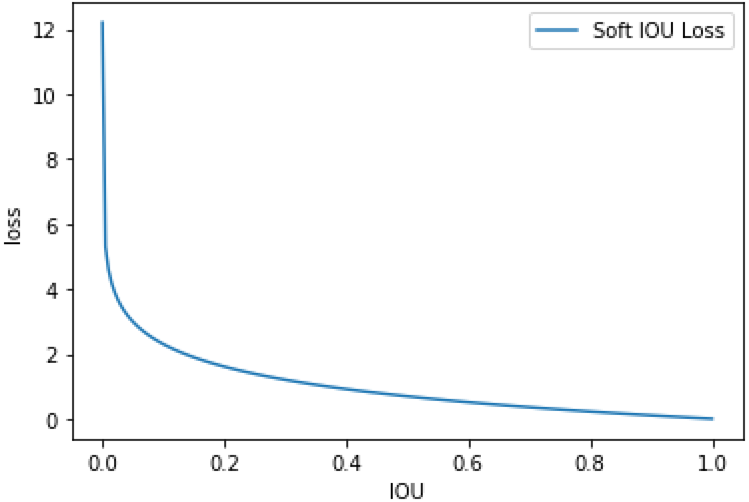


Loss functions

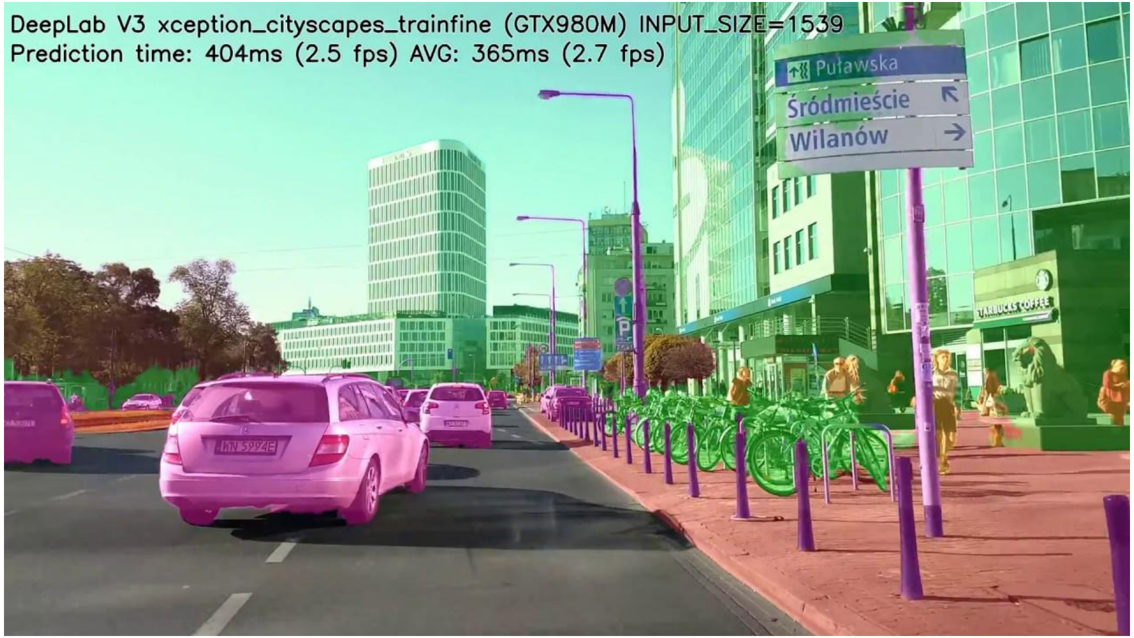
$$IOU(A,B) = \frac{|A \bigcap B|}{|A|+|B|-|A \bigcap B|}$$

$$IOU(p,y) = \frac{\sum_{i=0}^{N} p_i y_i + \epsilon}{\sum_{i=0}^{N} p_i + \sum_{i=0}^{N} y_i - \sum_{i=0}^{N} p_i y_i + \epsilon}$$

$$Loss_{IOU}(p,y) = -log(\frac{\sum_{i=0}^{N} p_i y_i + \epsilon}{\sum_{i=0}^{N} p_i + \sum_{i=0}^{N} y_i - \sum_{i=0}^{N} p_i y_i + \epsilon})$$



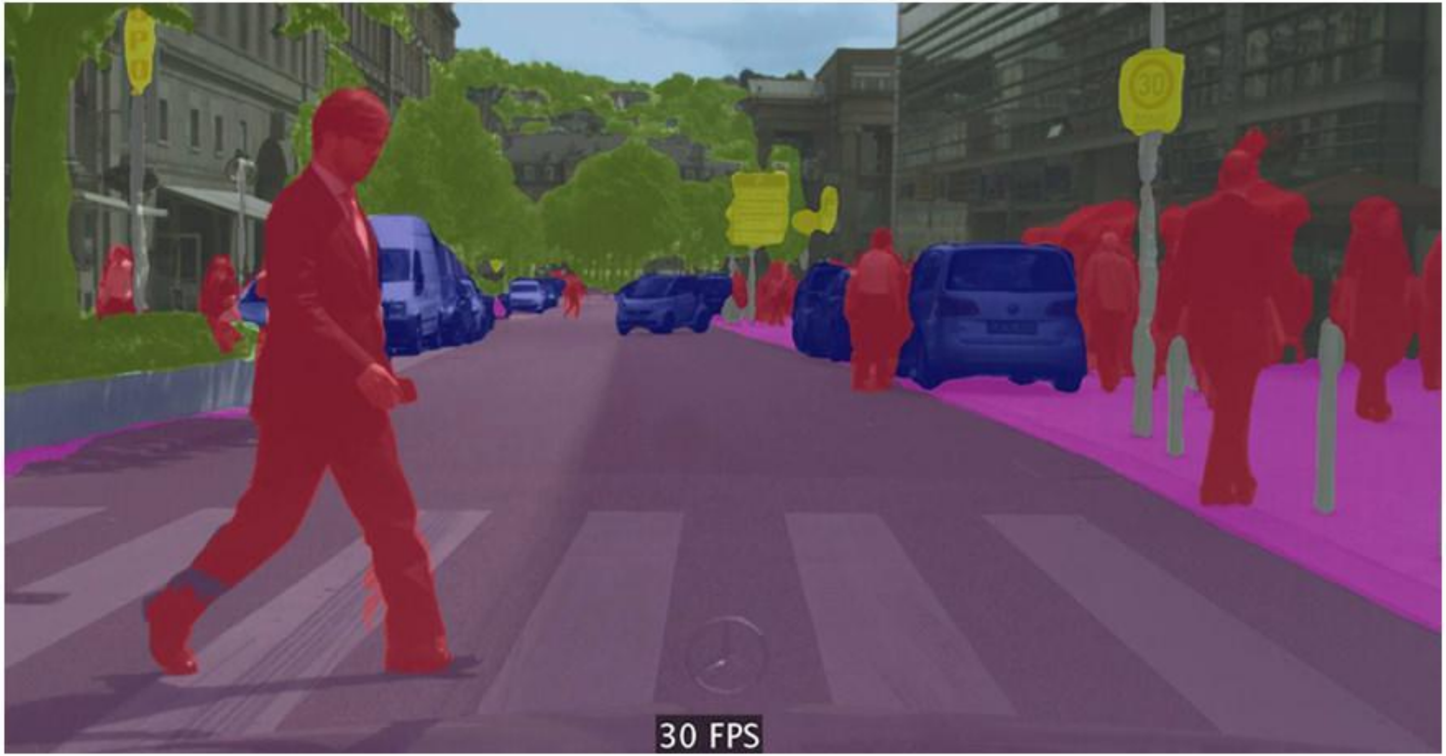
Results



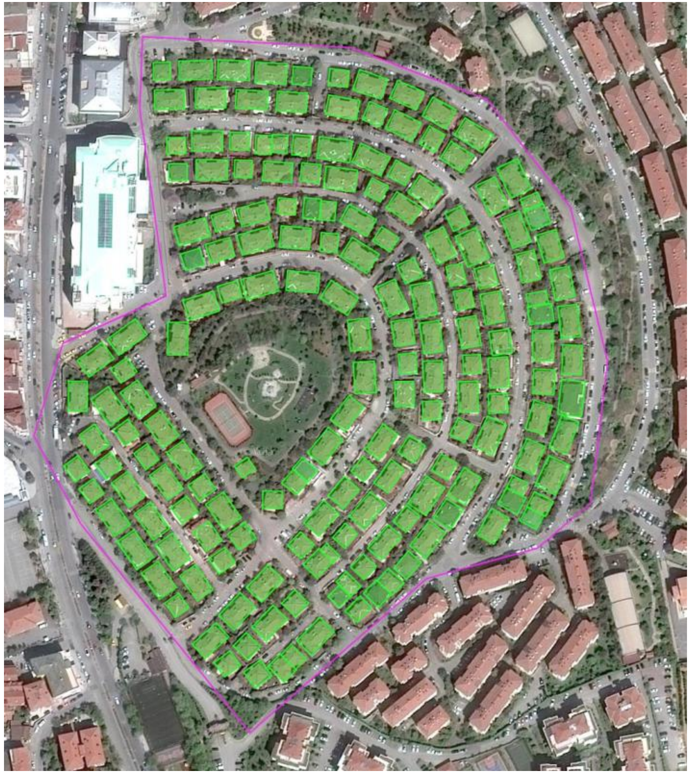
DeepLab V3 xception_cityscapes_trainfine (GTX980M) INPUT_SIZE=1539

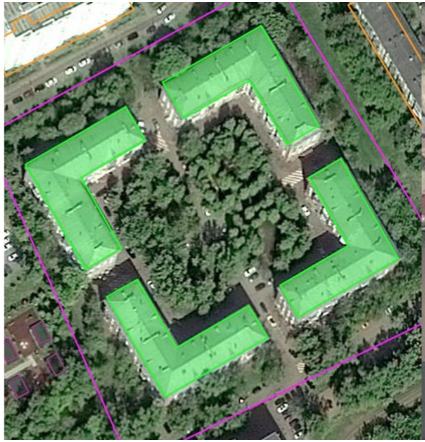
Prediction time: 404ms (2.5 fps) AVG: 365ms (2.7 fps)

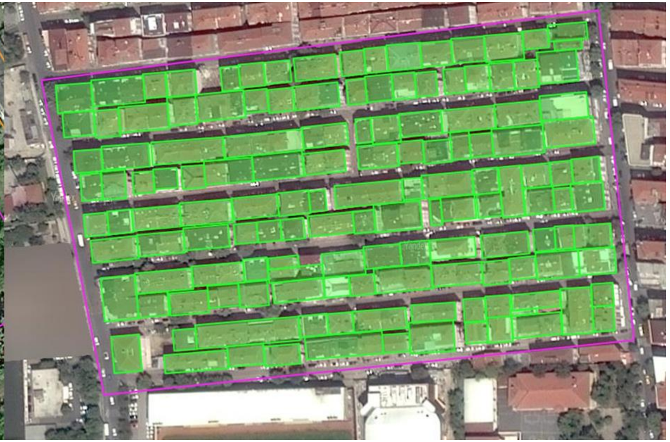
Results



Results







- UNet: https://arxiv.org/abs/1505.04597
- DeepLab: https://arxiv.org/abs/1606.00915
- DeepLabV3: https://arxiv.org/abs/1706.05587
- DeepLabV3+: https://arxiv.org/abs/1802.02611
- SegNet: https://arxiv.org/abs/1511.00561
- FCN: https://arxiv.org/abs/1411.4038
- Grad-CAM: https://arxiv.org/abs/1610.02391

- https://github.com/mrgloom/awesome-semantic-segmentation
- Kaggle: https://www.kaggle.com/
- ODS (@bes): https://ods.ai/ https://opendatascience.slack.com
- Deep Learning Book: https://www.deeplearningbook.org/