DeepLab & Cityscapes Dataset

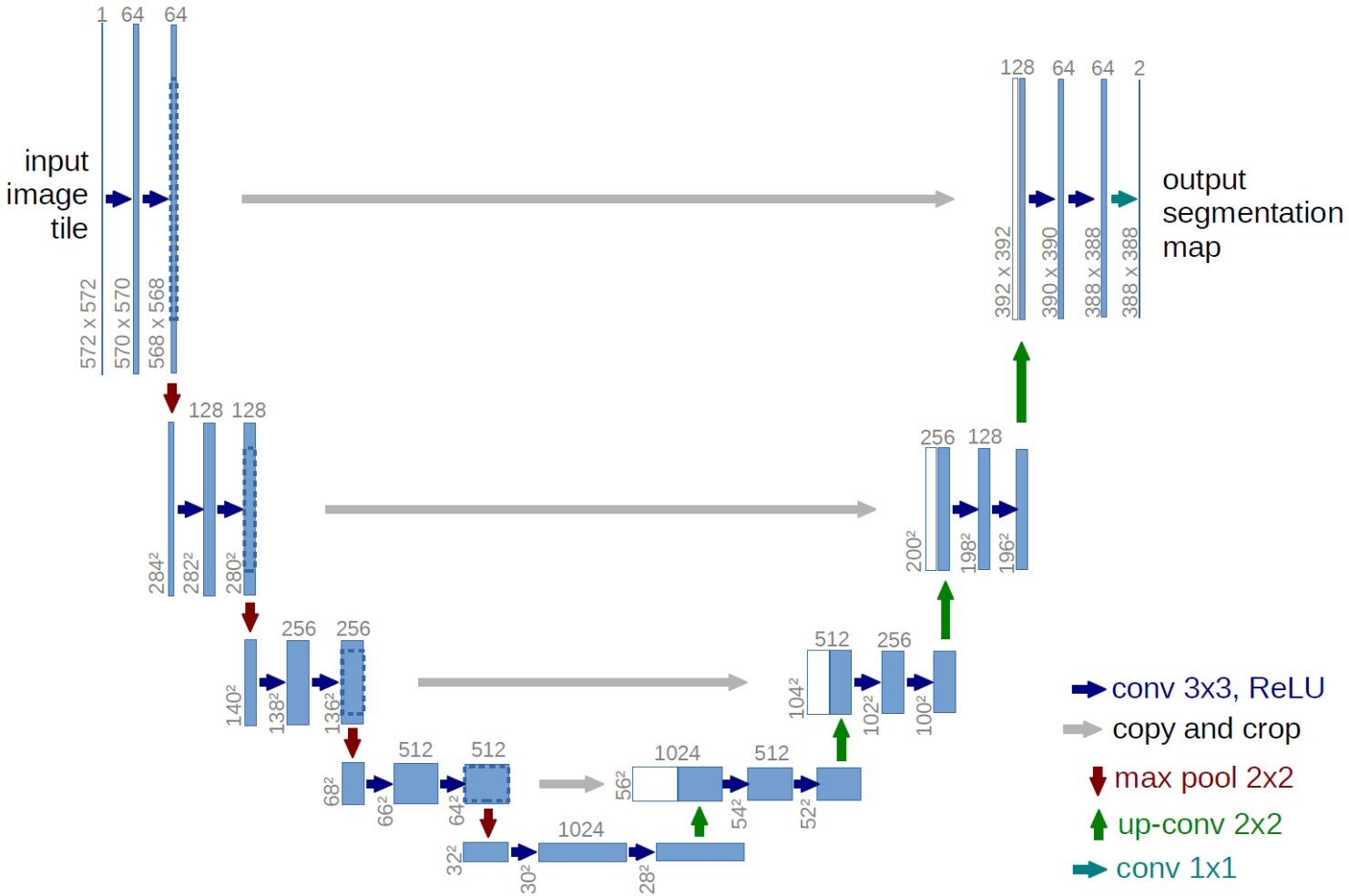
Hochschule Offenburg
University of Applied Sciences

Agenda

- U-Net vs DeepLab
- DeepLab
  - V1, V2, V3, V3+
- Cityscapes Dataset
- ResNet (Experiments)
  - ResNet18
  - ResNet50
  - ResNet101
- Sources

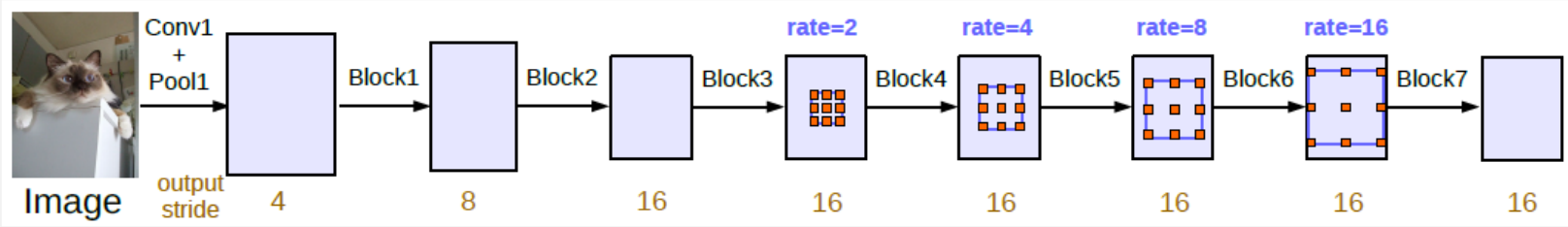
U-Net vs DeepLab

- U-Net (2015):



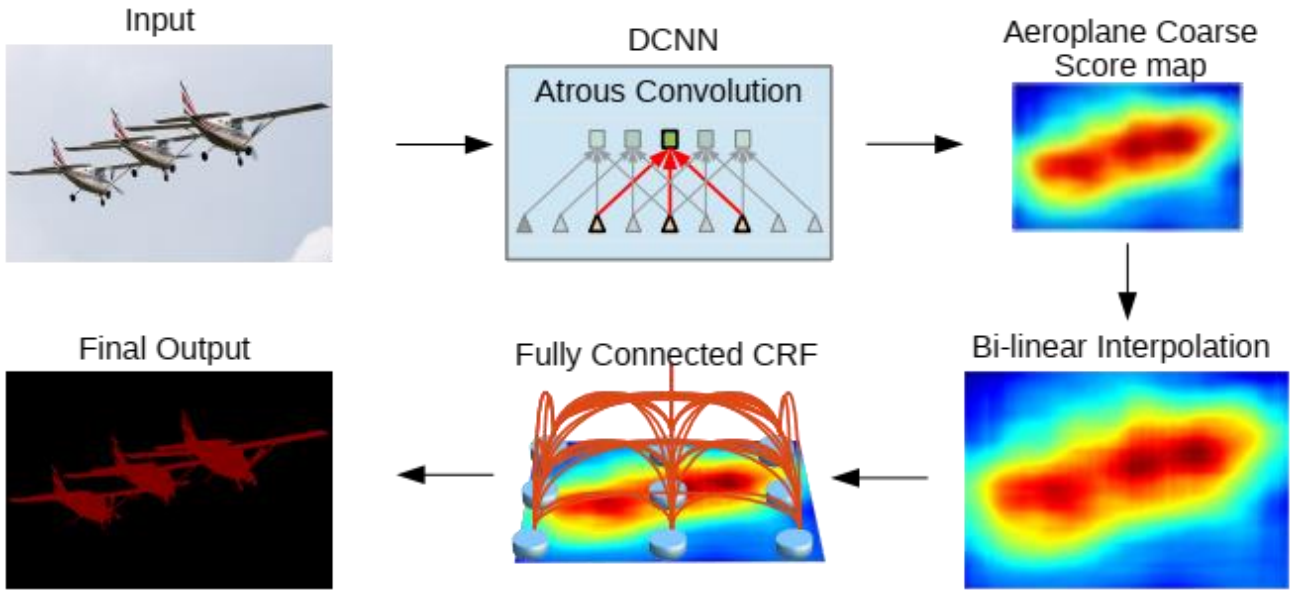
U-Net vs DeepLab

- V1
- DeepLab is a state-of-the-art semantic segmentation model designed and open-sourced by Google.
- Input through usual DCCN layers
- Followed by one or two atrous layer



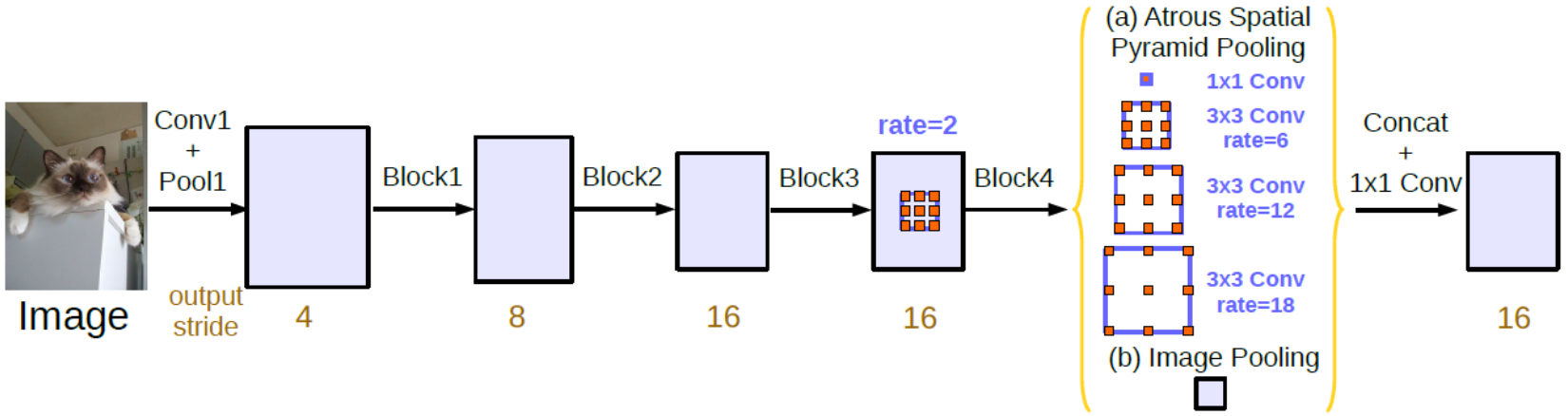
DeepLab

- V1
- Up-sampled to the original size of the image, using bi-linear interpolation
- Fully connected CRF is applied (VGG-16)



DeepLab

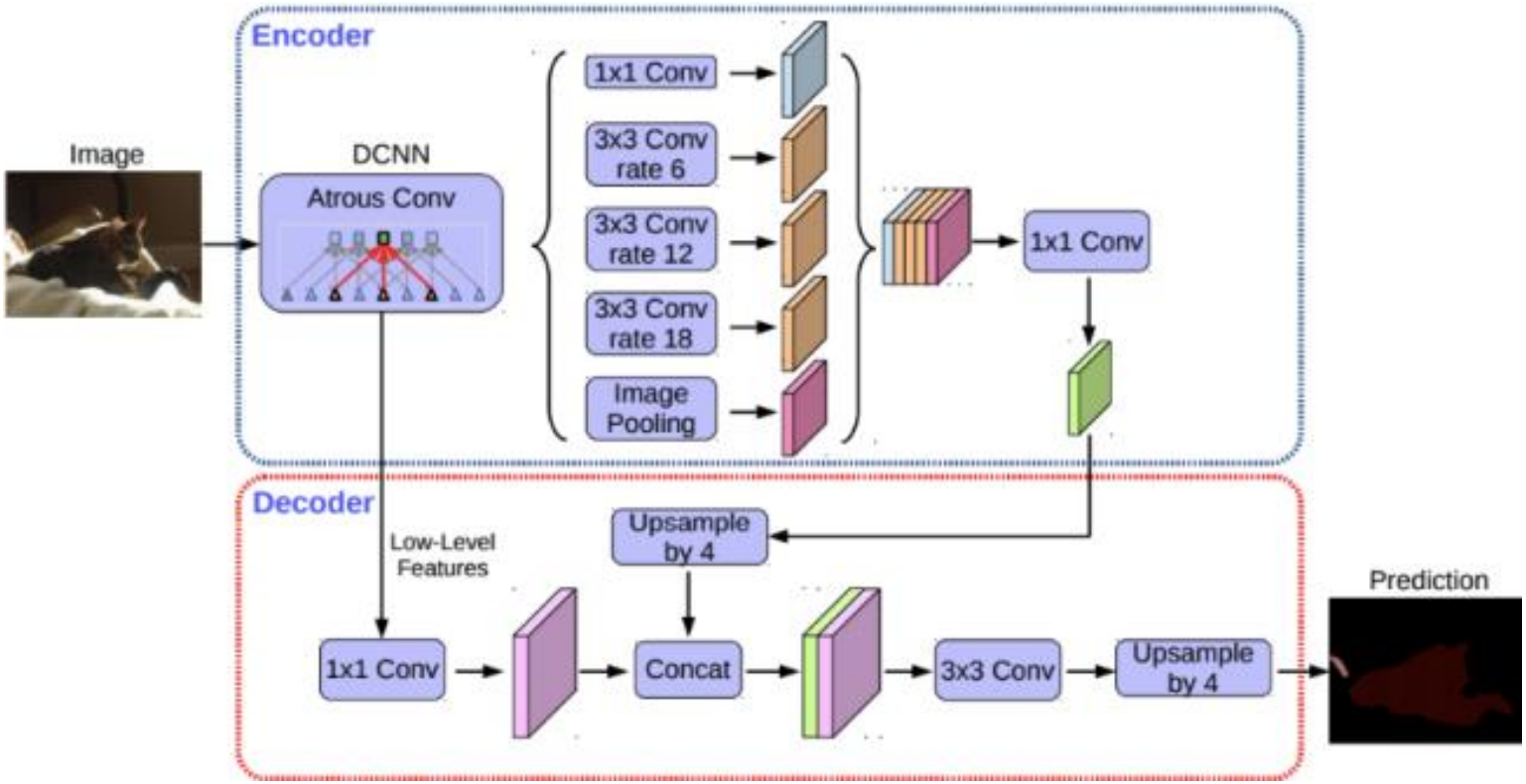
- V2
  - Capture Objekts in diffrent size
  - **Atrous Spatial Pyramid Pooling (ASPP)**



DeepLab

- V3
- Encoder-decoder model
- Challenge: capture sharper object boundaries
- Depth-wise seperable convolution (efficiency)
- V3+
- It uses Aligned Xception instead of ResNet-101 as its main feature extractor
- All max pooling operations are replaced by depth-wise separable convolution (faster)

DeepLab


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

Cityscapes Dataset

- High quality pixel of 5000 frames
  - 1024x2048
  - 11GB
  - 50 different cities
- 30 Classes
- Diffrent annotations (label, depth, etc.)

ResNet

- Types of ResNet

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

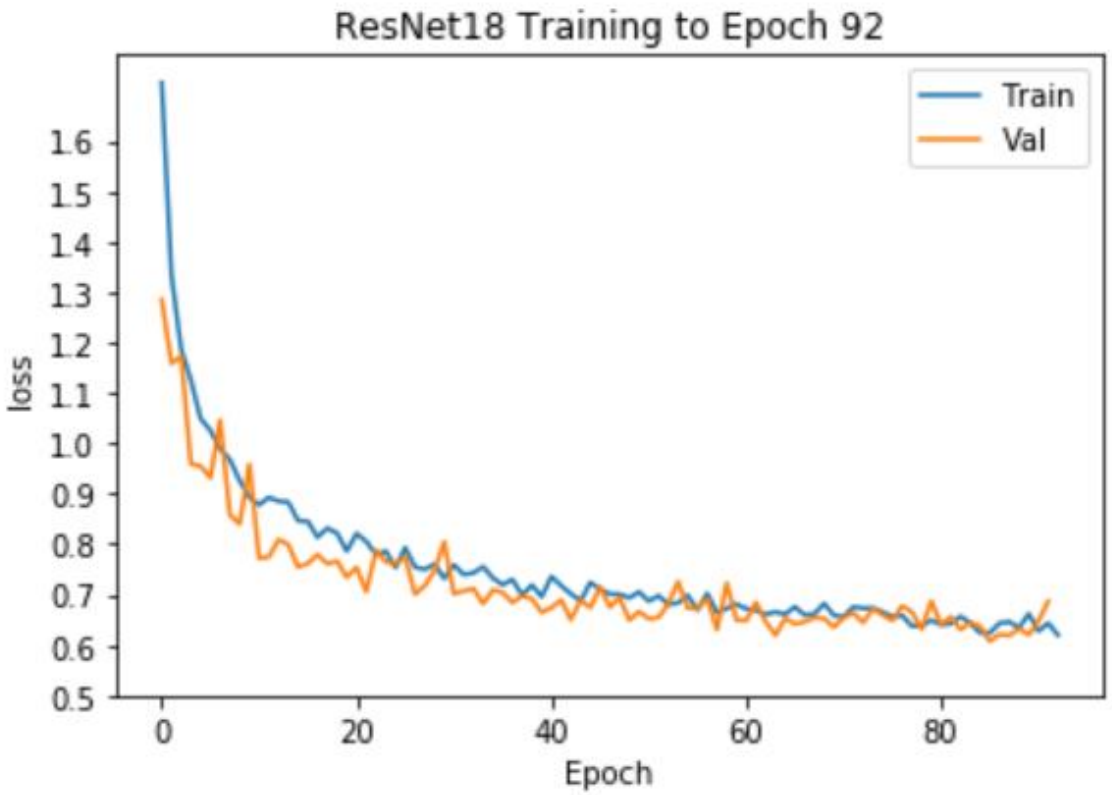
ResNet18

- 18 Layers
- Training Time for one epoch = ~15 min

```
================================================================
Total params: 15,315,028
Trainable params: 15,315,028
Non-trainable params: 0
----------------------------------------------------------------
```

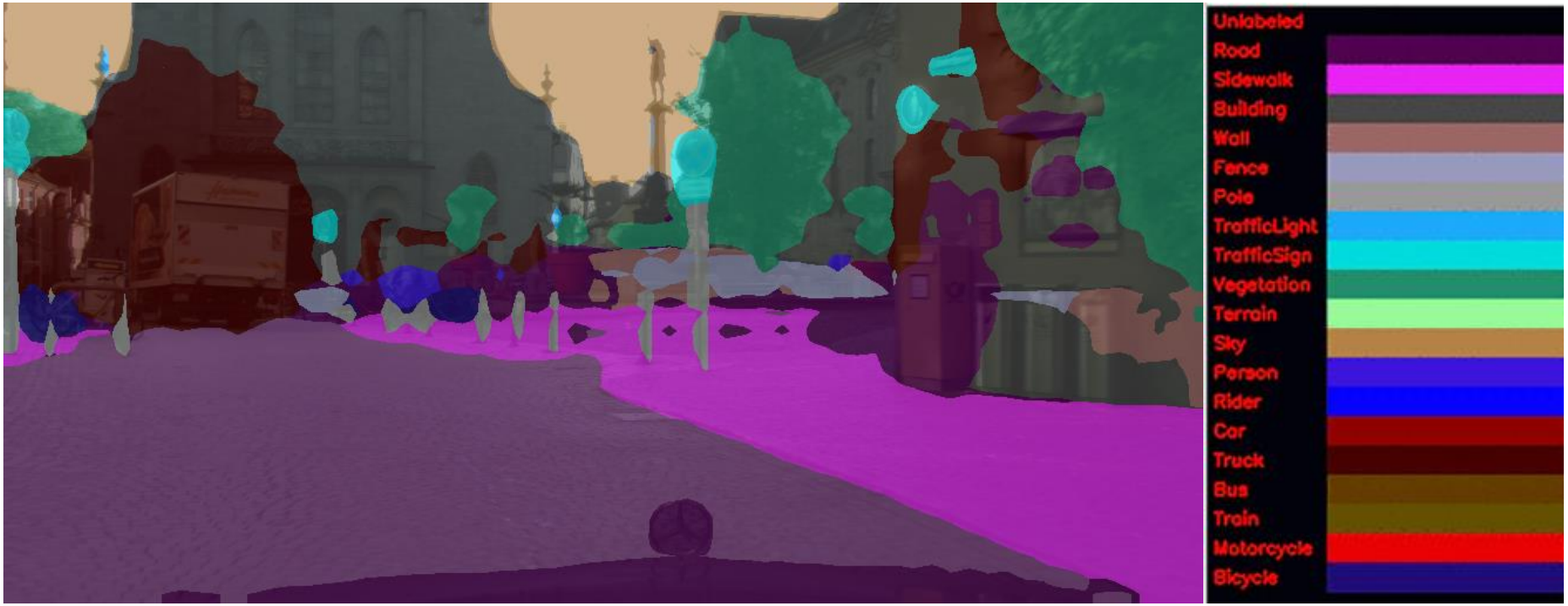
ResNet18

- 92 Epoch
- Val loss: 0.682222
- mIOU is : 0.723001



- Link: https://drive.google.com/open?id=1fmEINKCeWl-2f0G_umkUx9ecccS3Ct2a

Bad example


Unlabeled
Road
Sidewalk
Building
Wall
Fence
Pole
TrafficLight
TrafficSign
Vegetation
Terrain
Sky
Person
Rider
Car
Truck
Bus
Train
Motorcycle
Bicycle

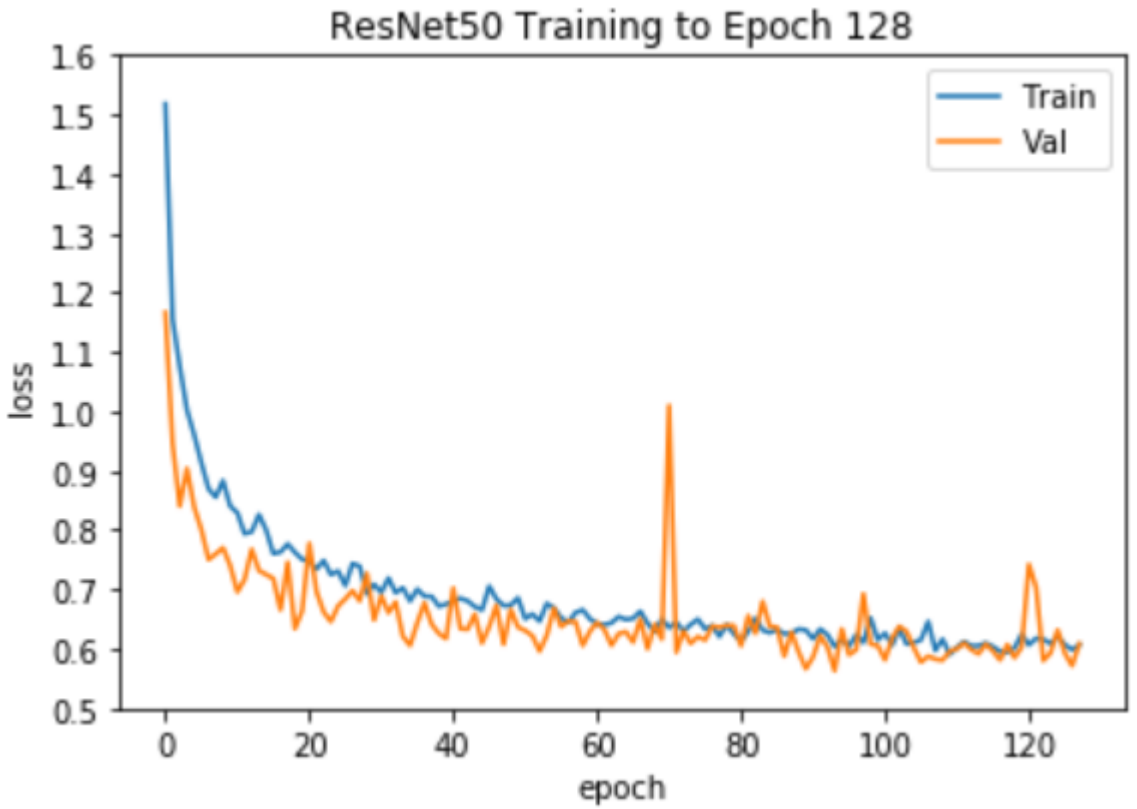
ResNet50

- 50 Layers
- Training Time for one epoch = ~30 min

```
================================================================
Total params: 39,049,812
Trainable params: 39,049,812
Non-trainable params: 0
----------------------------------------------------------------
```

ResNet50

- Epoch 128
- Val loss: 0.606357
- mIOU is : 0.720219



- Link: https://drive.google.com/open?id=1U3m5msu7esP6fTAgY2o9m0BVVHiuniAH

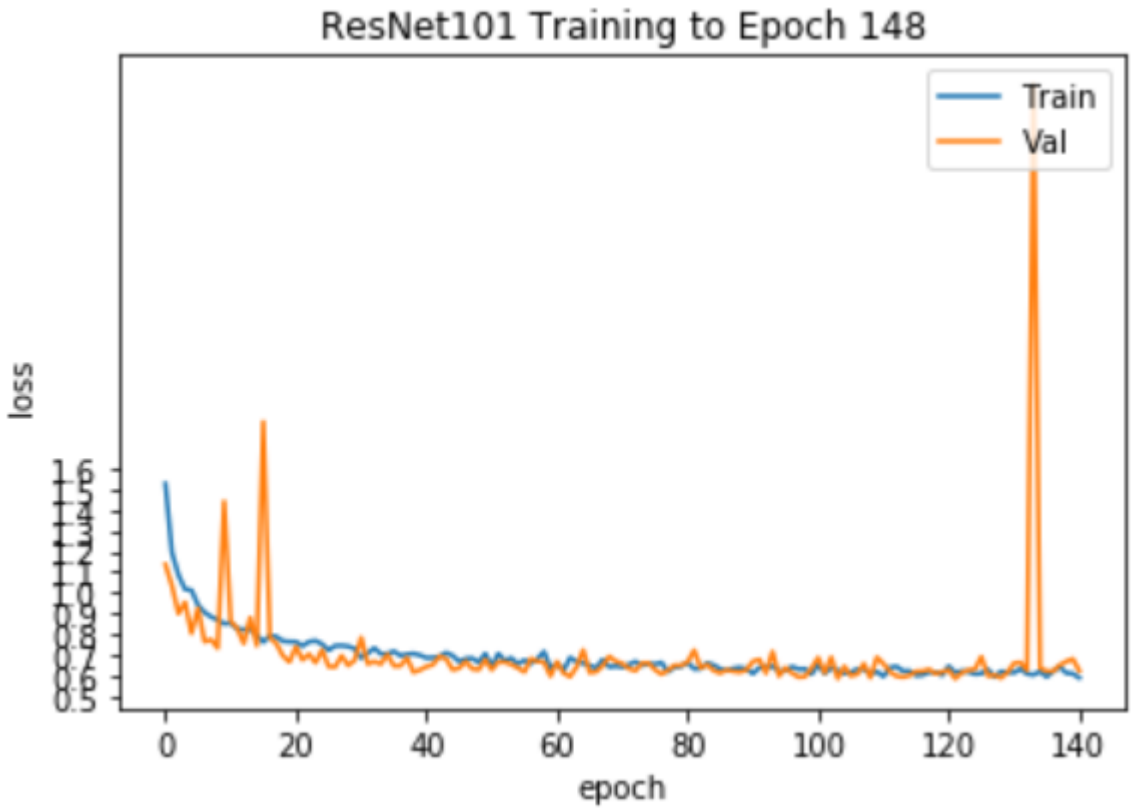
ResNet101

- 101 Layers
- Training Time for one epoch = ~50 min

```
================================================================
Total params: 58,041,940
Trainable params: 58,041,940
Non-trainable params: 0
----------------------------------------------------------------
```

ResNet101

- Retrain trained on 148 epoch model



ResNet101

- 22 more epoch
- LR: 0.0001->0.00001
- Val loss: 0.515101
- mIOU is : 0.69874

- Link: https://drive.google.com/open?id=1nJcNiG_fSUGSGMcORFXB1RbTRdr4xH3c

THANK YOU FOR YOUR ATTENTION

Quellen:

- https://github.com/fregu856/deeplabv3
- https://towardsdatascience.com/the-evolution-of-deeplab-for-semantic-segmentation-95082b025571
- https://arxiv.org/abs/1606.00915
- http://blog.qure.ai/notes/semantic-segmentation-deep-learning-review#deeplabv3
- https://arxiv.org/pdf/1802.02611.pdf
- https://towardsdatascience.com/review-deeplabv3-atrous-convolution-semantic-segmentation-6d818bfd1d74
- https://neurohive.io/en/popular-networks/resnet/
- https://neurohive.io/en/popular-networks/u-net/
- https://towardsdatascience.com/review-deeplabv3-atrous-convolution-semantic-segmentation-6d818bfd1d74
- https://towardsdatascience.com/types-of-convolutions-in-deep-learning-717013397f4d
- https://github.com/vdumoulin/conv_arithmetic
- https://github.com/KlamJ/DeepLab_V3_KL
- https://arxiv.org/abs/1706.05587
- https://colab.research.google.com/drive/1Al4mu6zJjFhNLHbYPod3eCBSJKY0VnP5#scrollTo=hDSBaVoj6dpp