Karlsruhe Institute of Technology

IPF Institute of Photogrammetry and Remote Sensing

Satellite data is for everyone

Insights into modern remote sensing research with open data and Python

Dr.-Ing. Jens Leitloff, Felix M. Riese

GitHub

jensleitloff/CNN-Sentinel

Objective of this talk:

Python tutorial with Keras

Example case:

Classification of land use & cover

Why

multispectral satellite images?

How

to apply out-of-the-box CNNs?

What

do you need to start?

Why

multispectral satellite images?

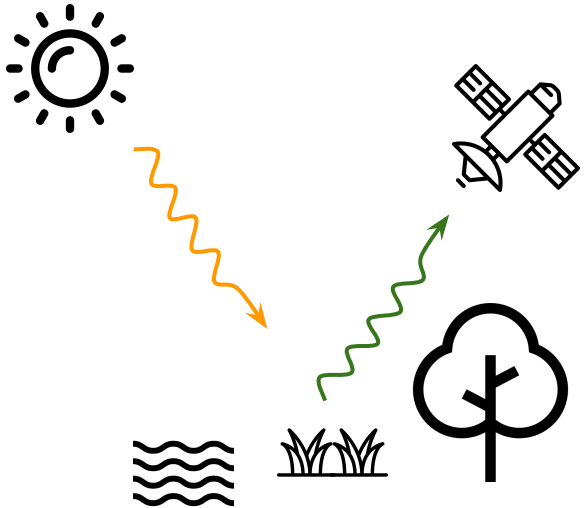
How

to apply out-of-the-box CNNs?

What

do you need to start?

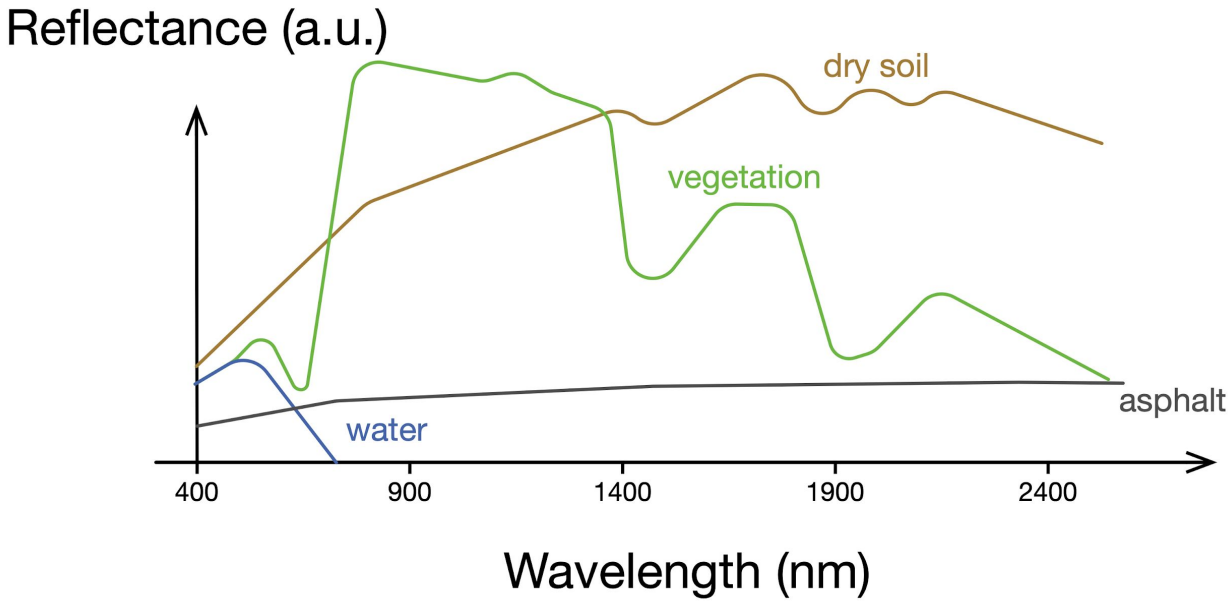
Remote sensing setup



Channels (= features)

- RGB → 3 channels
- Multispectral (e.g. 13 channels)
- Hyperspectral (e.g. 200 channels)

Spectral properties



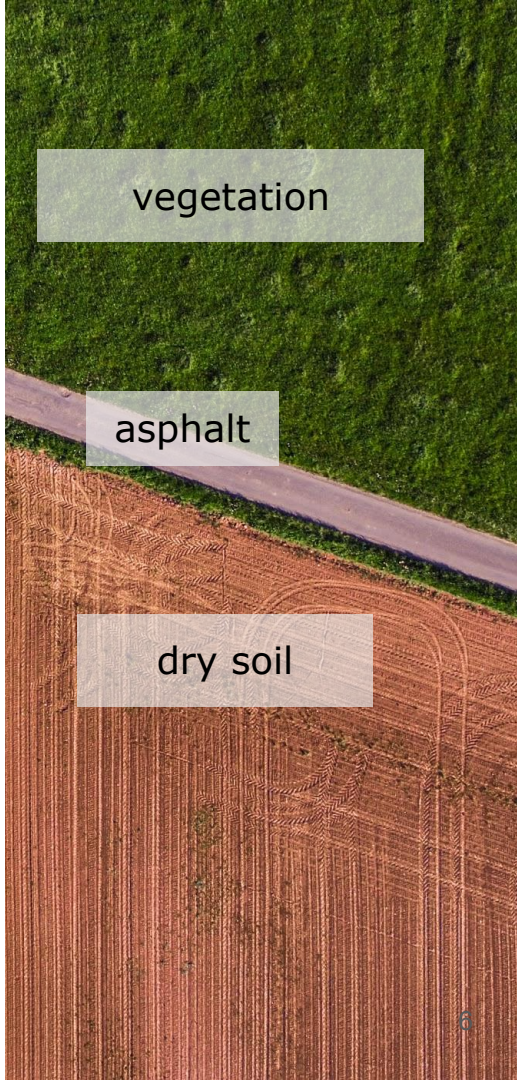
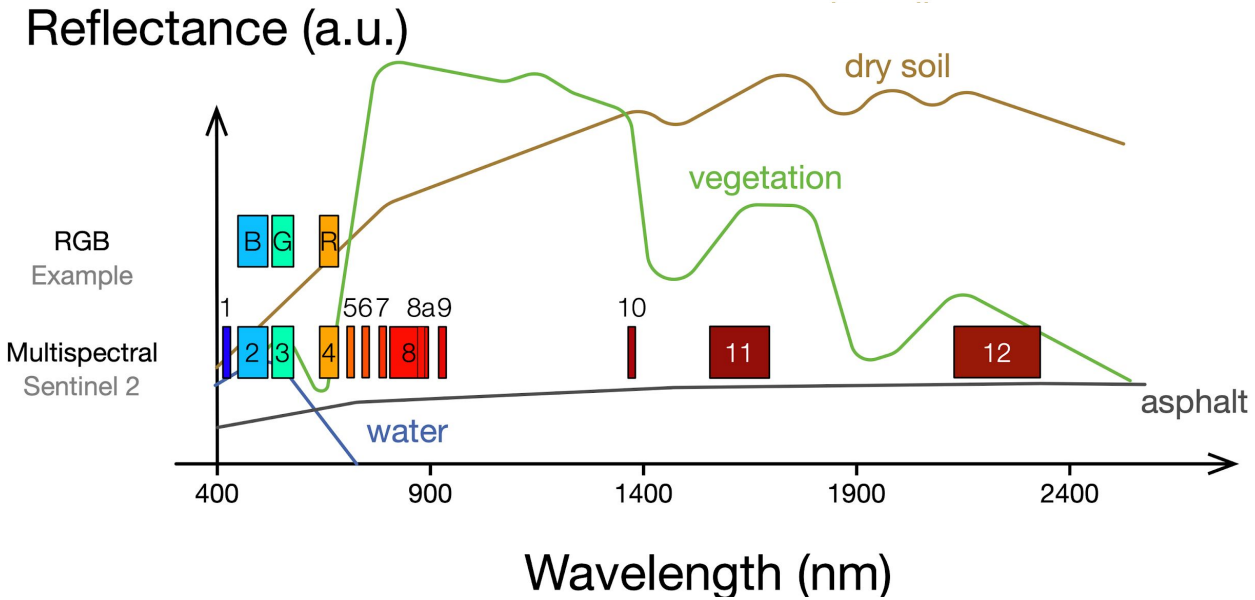


Photo by jean wimmerlin on Unsplash

Sensing spectral properties



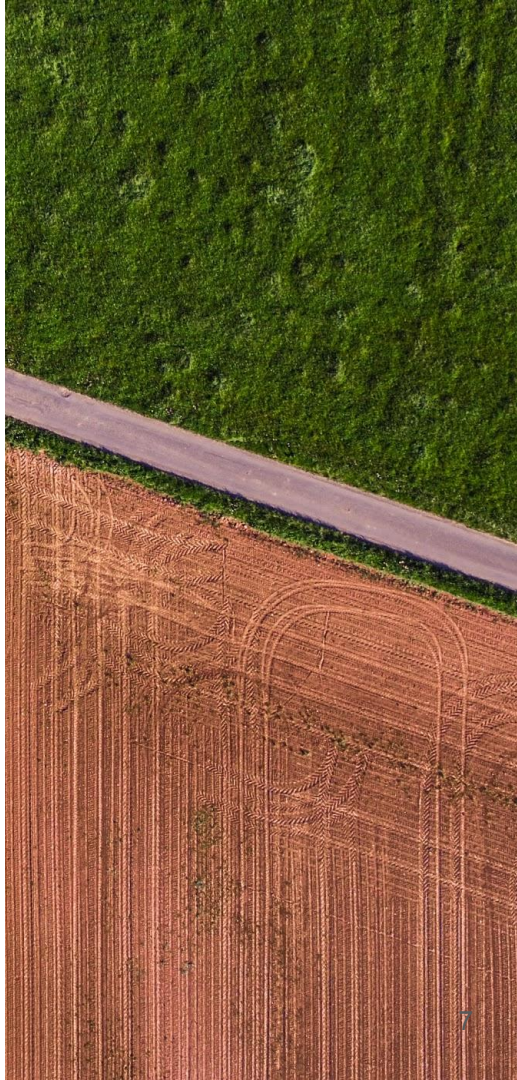


Photo by jean wimmerlin on Unsplash

5 days

does it take for Sentinel-2 satellites to cover the earth’s surface

free

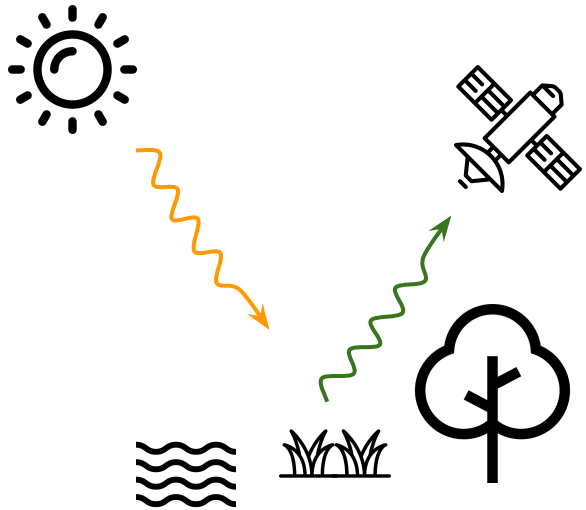
for non-commercial use

Today’s objective:

Implement a land use classifier!

1. Training

→ learn the correlations



Ground truth (labels)

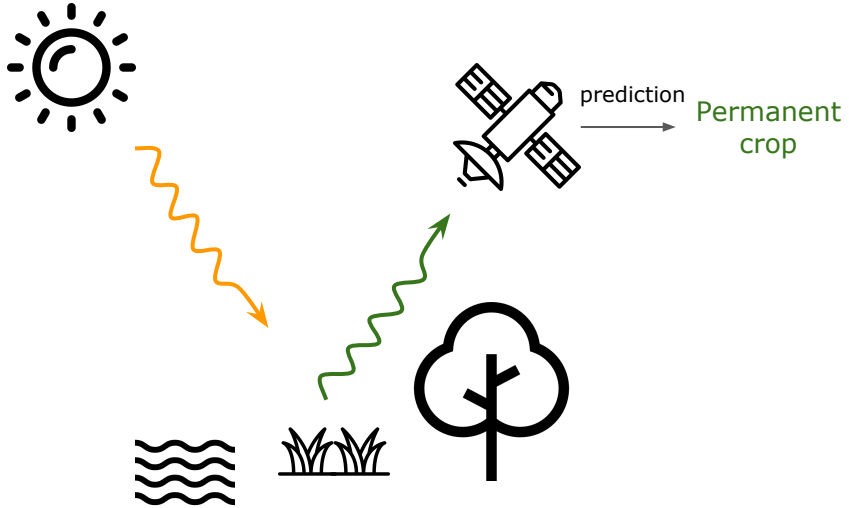
River | Permanent crop | Forest

1. Training
→ learn the correlations

Ground truth (labels)

River | Permanent crop | Forest

2. Prediction
→ apply the learnings



Why

multispectral satellite images?

How

to apply out-of-the-box CNNs?

What

do you need to start?

The EuroSAT dataset

Links:

- Introduced in arXiv:1709.00029
- Download here: http://madm.dfki.de/downloads

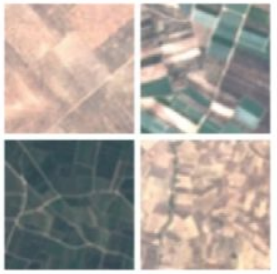
Properties:

- Sentinel-2 satellite images
  - **Raw dataset:** 13 different spectral bands (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 8a)
  - **RGB dataset:** RGB colors
- 27 000 labeled images
- 10 classes (→ next slide)

(a) Industrial

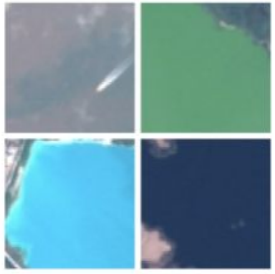
(b) Residential

(c) Annual crop



(d) Permanent crop

(e) River



(f) Sea & lake

(g) Herbaceous vegetation

(h) Highway

(i) Pasture

(j) Forest

The classes

Image source: arXiv:1709.00029

Learning from scratch

1. Network structure with **randomly initialized** weights
2. Training of **complete** network on training dataset

- **slow** convergence
- **lots of** training data necessary
- easily implemented for **every kind** of input data

Transfer learning

Learning from scratch

1. Network structure with **randomly initialized** weights
2. Training of **complete** network on training dataset

- **slow** convergence
- **lots of** training data necessary
- easily implemented for **every kind** of input data

Transfer learning

1. Network structure with **pretrained** weights
2. Training of **selected** layers on the training dataset (= fine tuning)

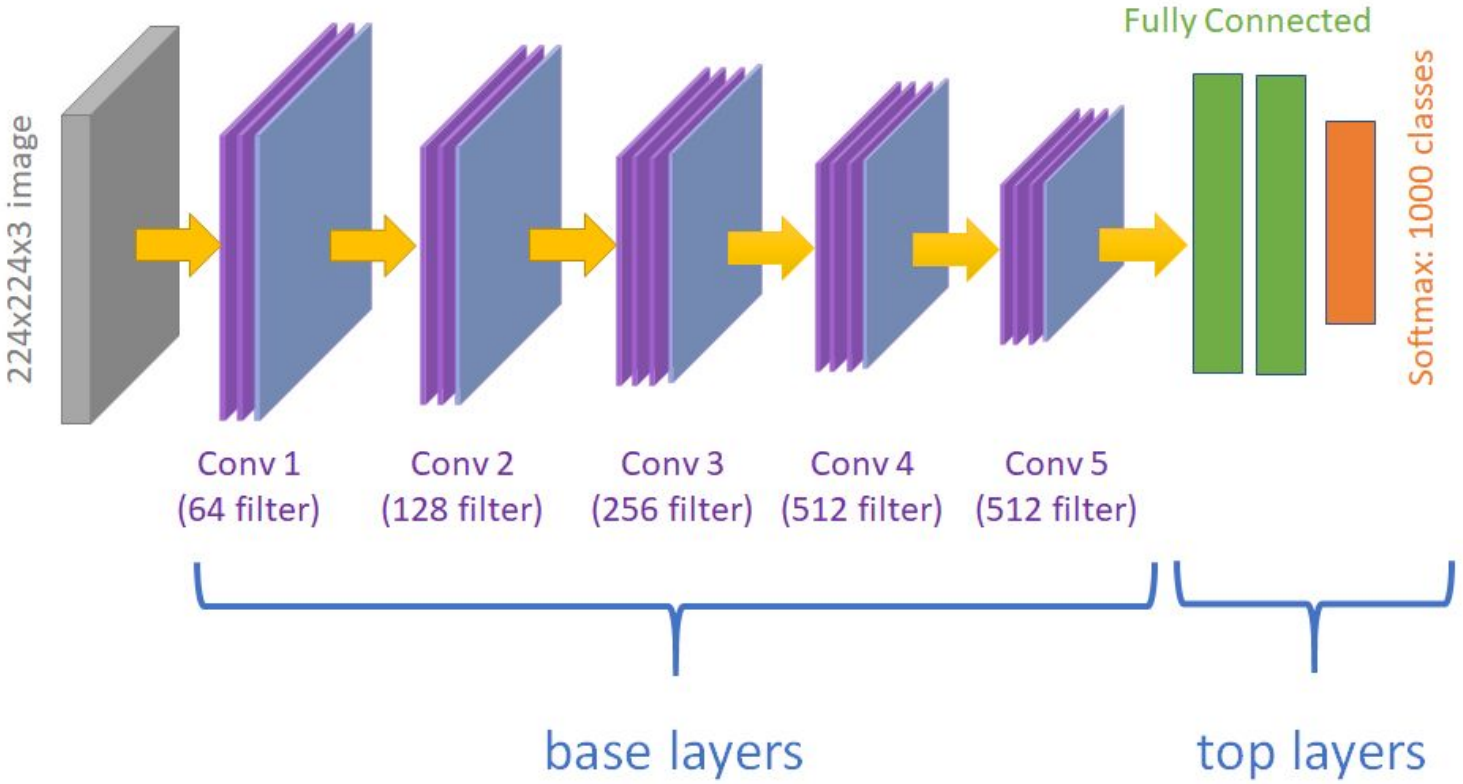
- much **faster** convergence
- **less** training data necessary
- mostly implemented **for RGB**
- modifications for MS necessary

→ Implement for RGB & multispectral

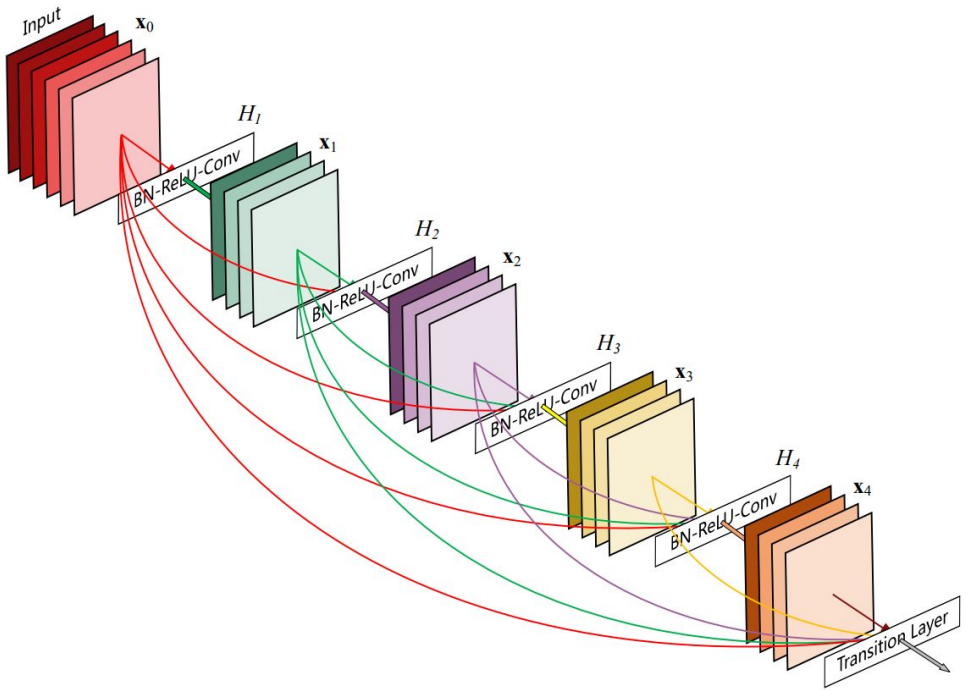
Implement two models:

VGG16 and DenseNet201

VGG16 architecture



DenseNet architecture



Densely Connected Convolutional Networks, arXiv:1608.06993

Detailed look

switch to Jupyter notebooks

GitHub

jensleitloff/CNN-Sentinel

Steps to do

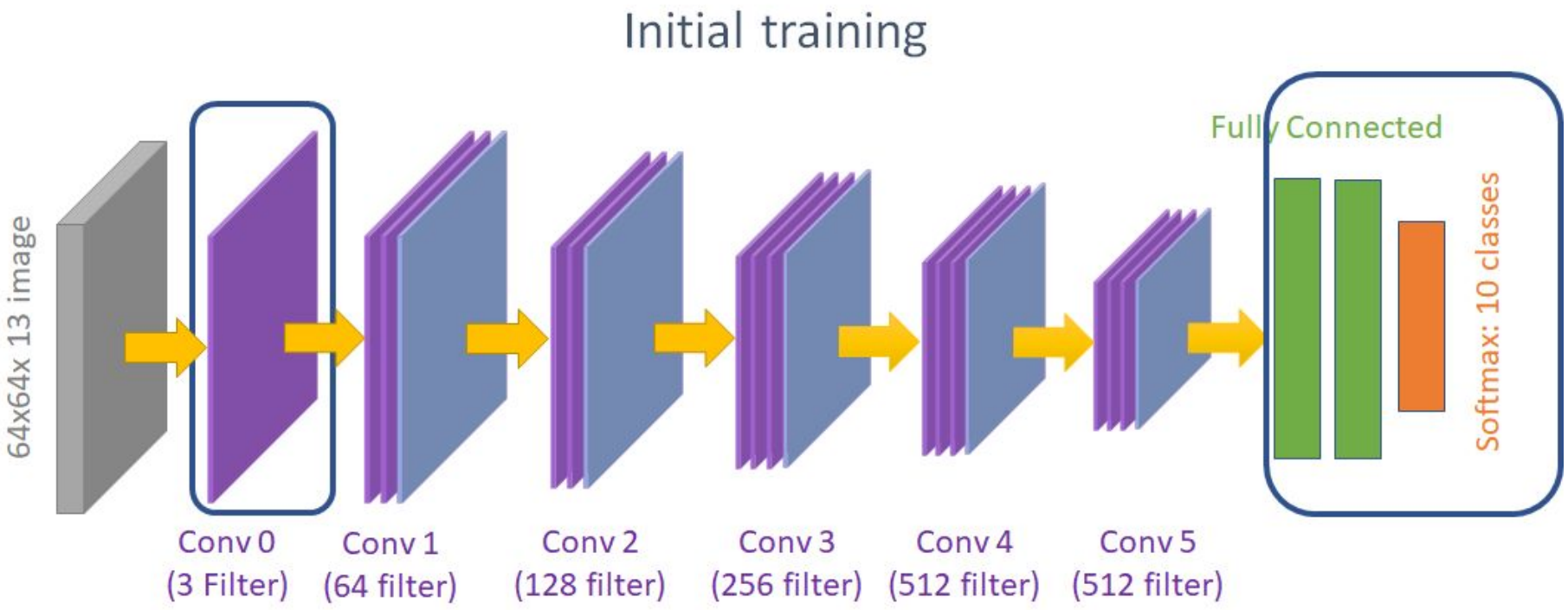
Training from scratch

1. Initialize network model without top layers
2. define new top layers
3. define data augmentation
4. define callbacks
5. -
6. -
7. -
8. fit model

Transfer-learning

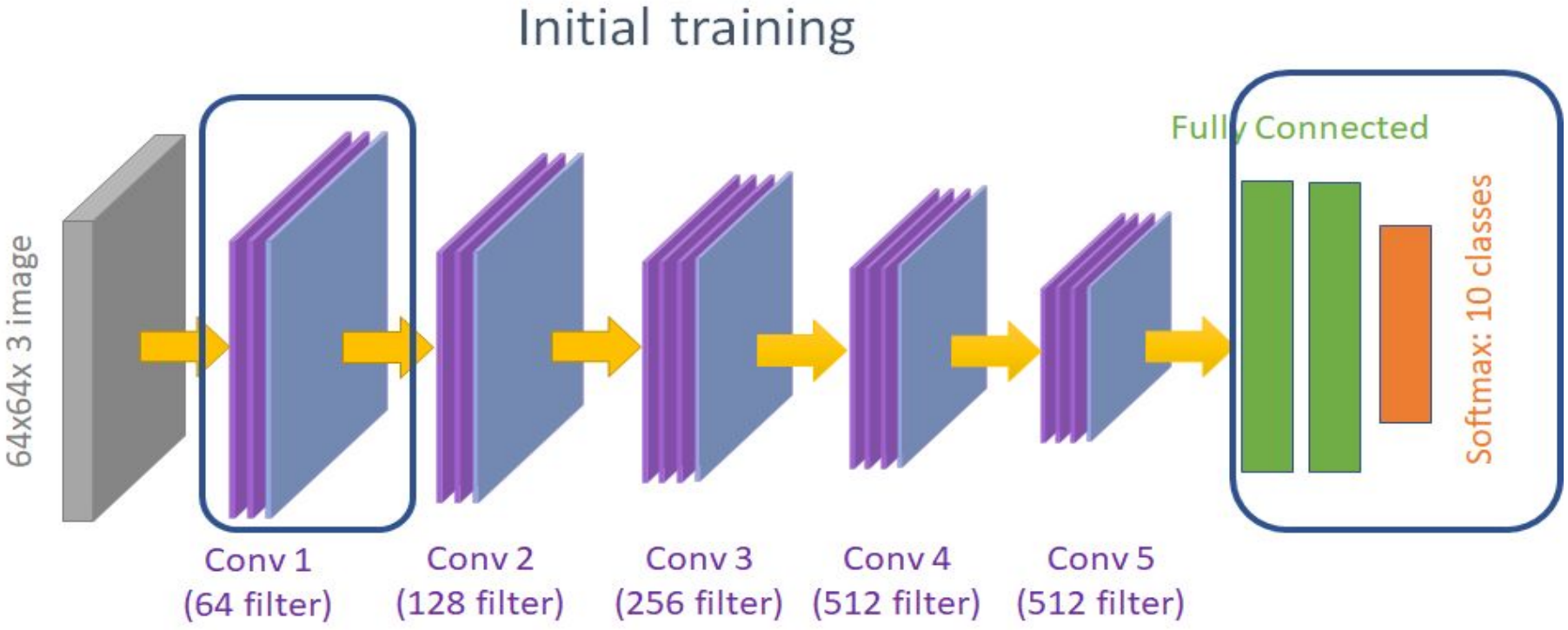
1. Pretrained network model without top layers
2. define new top layers
3. define data augmentation
4. define callbacks
5. set base layers non trainable
6. fit model (train new top layers)
7. set (some) base layers trainable
8. fit model (fine-tune base and top layers)

Adapt transfer learning for MS data



see script **04_train_ms_finetuning.py** in the GitHub repository

Alternative adaption



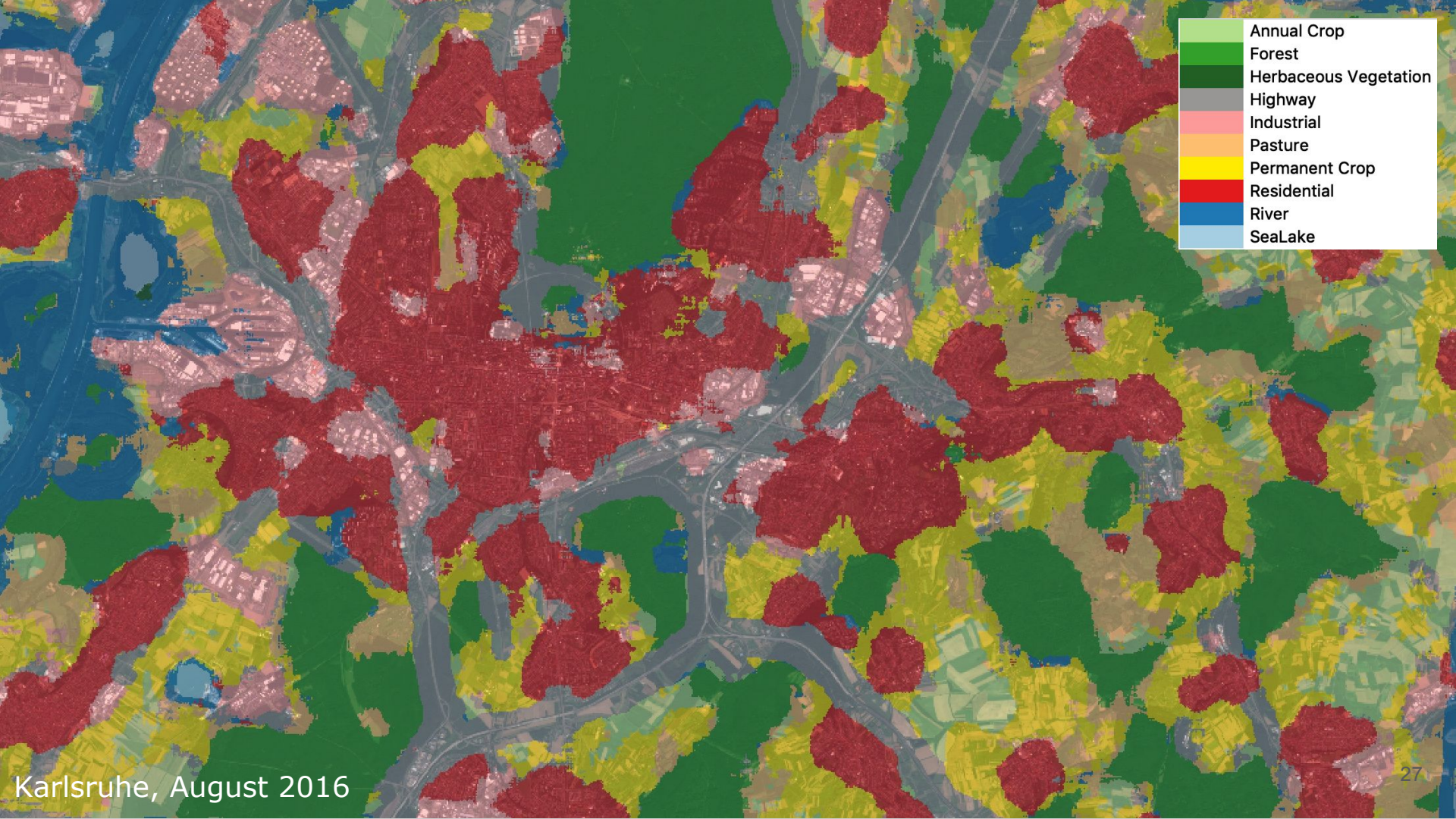
see script **05_train_ms_finetuning_alternative.py** in the GitHub repository

Learning with RGB data

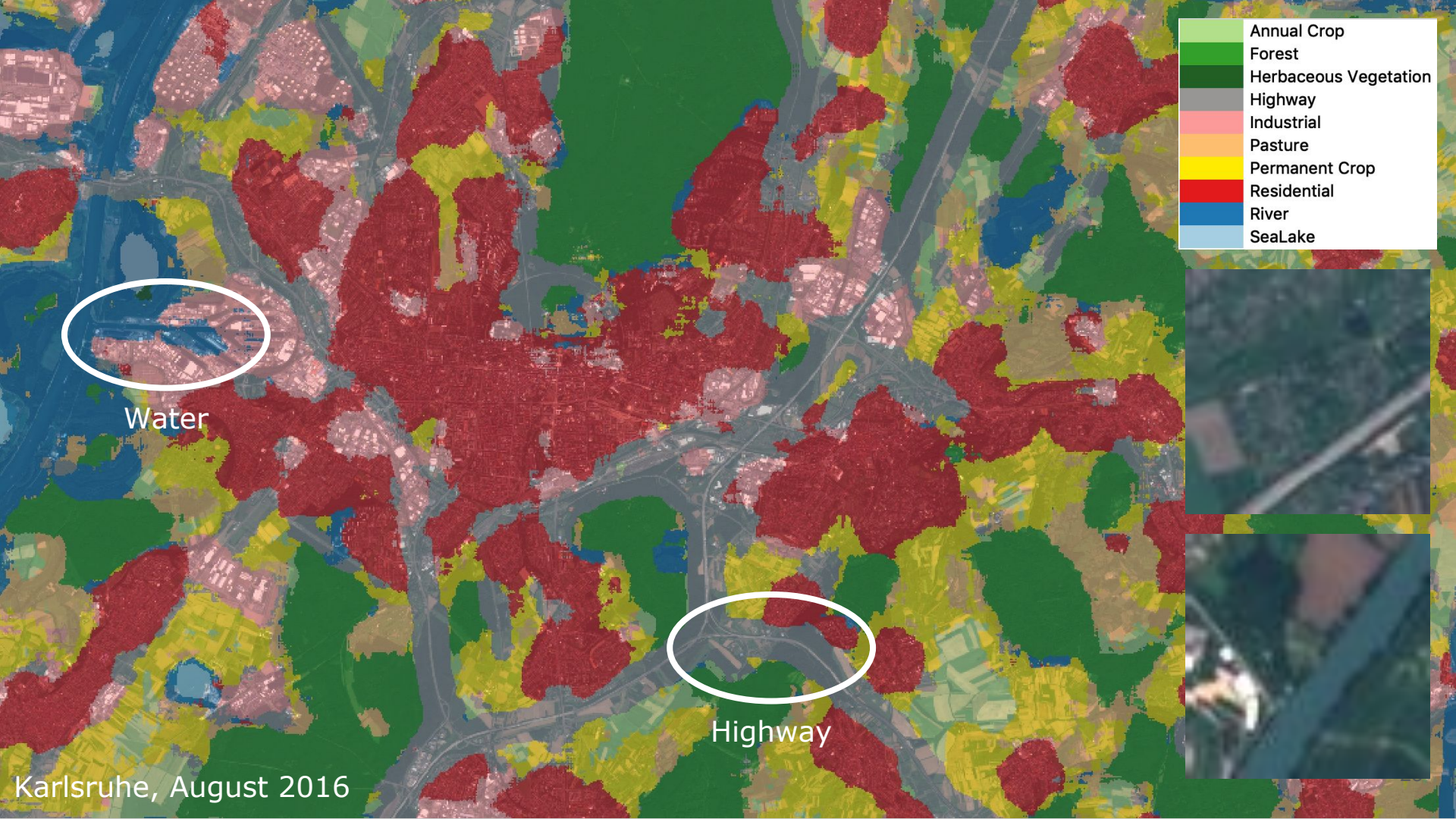
	Accuracy in % (number of epochs)		
	Training Top Layer	**Fine-tune**	**From-scratch**
VGG 16	94.2 (19)	97.7 (206)	97.2 (120)
DenseNet 201	82.9 (29)	97.2 (159)	97.5 (77)

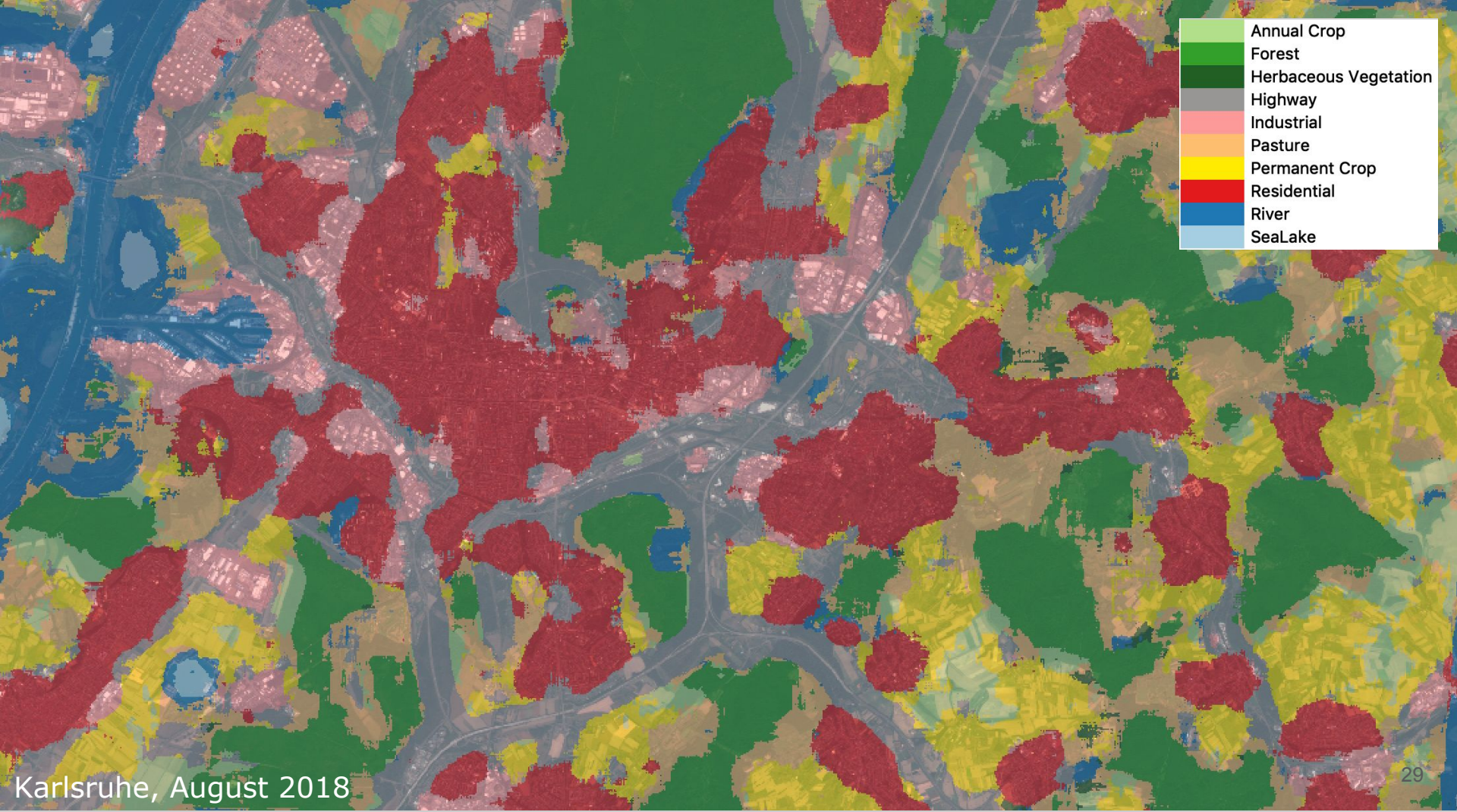
Learning with MS data

	Accuracy in % (number of epochs)			
	Training Top Layer	**Fine-tune**	**Fine-tune alternative**	**From-scratch**
VGG 16	95.1 (14)	98.0 (28)	98.5 (27)	97.6 (76)
DenseNet 201	75.8 (17)	89.5 (15)	36.0 (88)	98.6 (112)



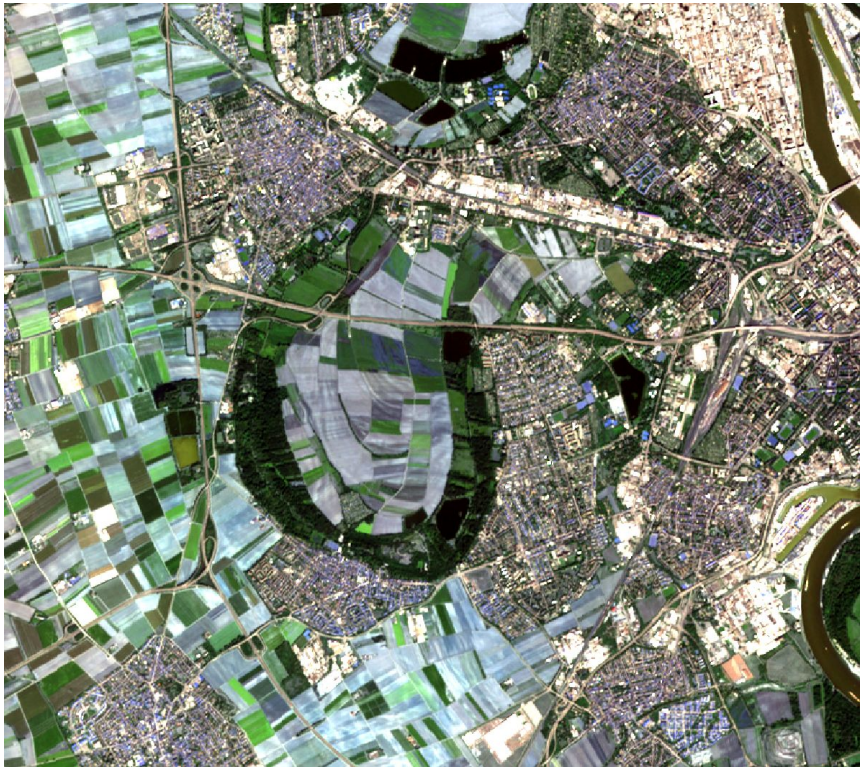
Karlsruhe, August 2016


Annual Crop
Forest
Herbaceous Vegetation
Highway
Industrial
Pasture
Permanent Crop
Residential
River
SeaLake
Water
Highway
Karlsruhe, August 2016

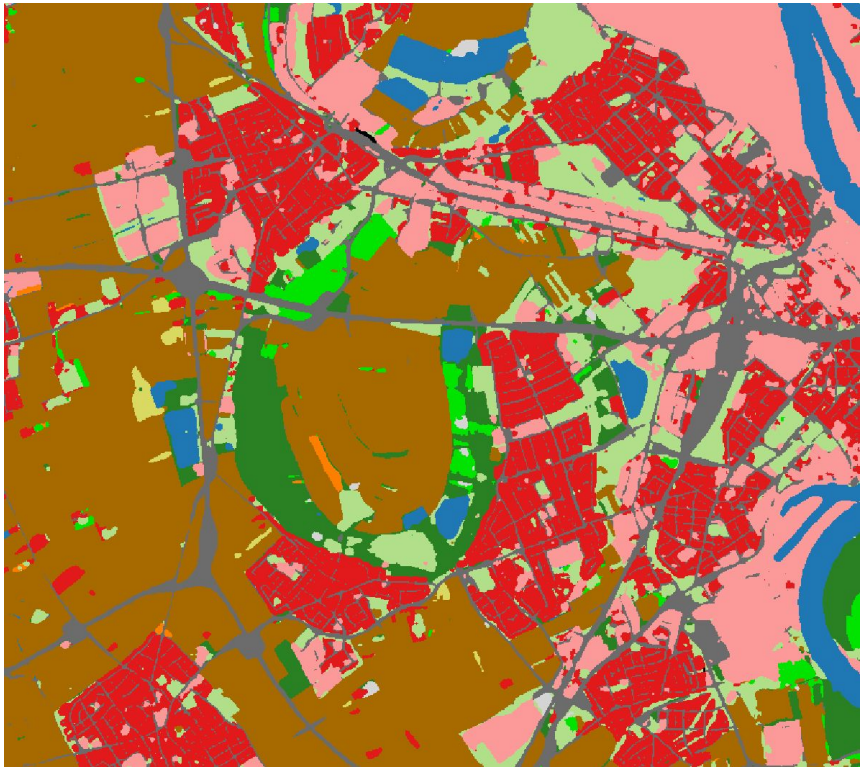


Karlsruhe, August 2018

Results with image segmentation



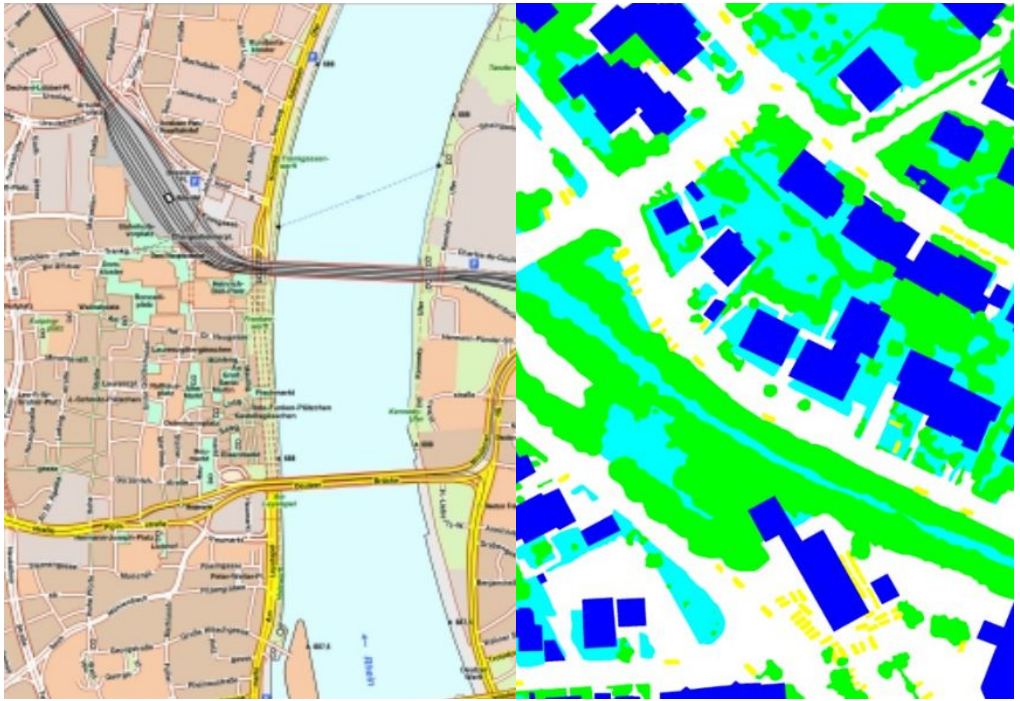
Ludwigshafen, August 2015



Results from project with LVermGeo Rheinland-Pfalz

Image Segmentation

- fully convolutional neural networks (e.g. DenseFCN from keras-contrib)
- corresponding data necessary:
  - https://github.com/mrgloom/awesome-semantic-segmentation
  - http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html
  - https://www.bezreg-koeln.nrw.de/brk_internet/geobasis/landschaftsmodelle/basis_dlm/index.html



Things not covered

- imbalanced data
  - data augmentation
  - “class_weights” in fit functions of Keras
- over-/underfitting during training
  - data augmentation
  - more data
  - network architecture (for better generalization or with reduced complexity)
  - regularization (e.g. dropout)

Take-aways

Using …

- MS data is better than RGB data
- Transfer-Learning is better than from scratch
- Transfer-Learning converged faster than from scratch
- Learning from scratch may more often not converge (without significant modifications)
- generators for data augmentation speeds up training

Use pre-trainend (parts of) networks

Other deep-learning approaches

Convolution approaches:

- No convolution
- 1D convolution over spectral dimension
- 2D convolution over spatial dimension → this talk
- 3D convolution over spectral & spatial dimension

Making use of **multi-temporal data** → e.g. annual crop

- Recurrent neural networks → Rußwurm, Körner (arXiv:1802.02080)

Review paper: arXiv:1710.03959

Why

multispectral satellite images?

How

to apply out-of-the-box CNNs?

What

do you need to start?

What do you need to start? → GitHub repo

GitHub

jensleitloff/CNN-Sentinel

- **Requirements** (Versions …)
- **Environment** setup (Conda …)
- **Code** for
  - setting up the data
  - CNN transfer learning RGB & MS
  - CNN learning-from-scratch RGB & MS
- How to download Sentinel-2 **data**
- **Links** to more datasets

Jens Leitloff
✉ jens.leitloff@kit.edu
jensleitloff/CNN-Sentinel

Felix M. Riese
✉ felix.riese@kit.edu

Thank you for your attention!

Resources

Literature:

- Deep learning in remote sensing: a review (arXiv:1710.03959)
- EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification (arXiv: 1709.00029)

Backup

Setup environment

Append conda-forge to your Anaconda channels:

- conda config --append channels conda-forge

Create new environment:

- conda create -n *pycon* scikit-image gdal tqdm
- conda activate *pycon*
- pip install tensorflow-gpu
- pip install keras (or use tensorflow version, i.e. **from** tensorflow **import** keras)

Other open datasets → Ground truth

Platforms for datasets:

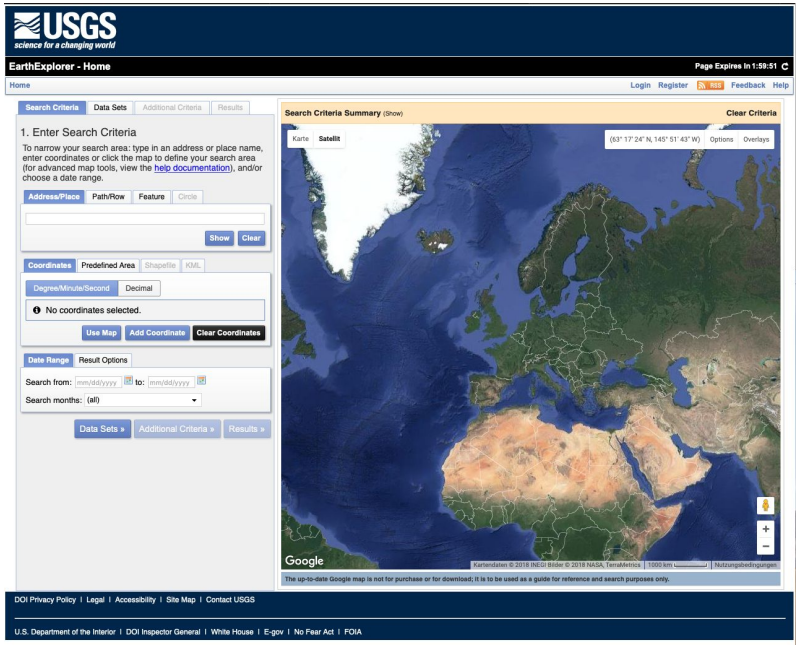
- HyperLabelMe: a Web Platform for Benchmarking Remote Sensing Image Classifiers (Link)
- GRSS Data and Algorithm Standard Evaluation (DASE) website (Link)

Datasets:

- UC Merced Land Use Dataset (Link)
- AID: A Benchmark Dataset for Performance Evaluation of Aerial Scene Classification (Link)
- NWPU-RESISC45 (RGB, Link)
- Zurich Summer Dataset (RGB, Link)

How to get Sentinel-2 data

1. Register at Copernicus Open Access Hub or EarthExplorer
2. Find your region
3. Choose tile(s) (→ area) and date
   - Less tiles makes things easier
   - Less clouds in the image are better
   - Consider multiple dates for classes like “annual crop”
4. Download L1C data
5. Decide of you want to apply L2A atmospheric corrections
   - Your CNN might be able to do this by itself
   - If you want to correct, use Sen2Cor
6. Have fun with the data
   - For quick visualization, QGIS is an option



Screenshot of https://earthexplorer.usgs.gov , date: October 23, 2018