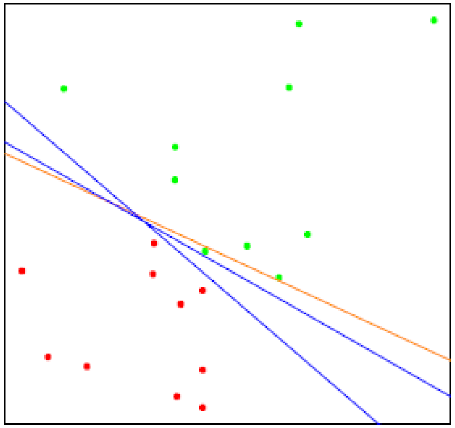


Deep Learning

Introduction to Deep Learning Models

Dr. – Ing. Gabriele Cavallaro

Postdoctoral Researcher

High Productivity Data Processing Group

Juelich Supercomputing Centre, Germany

LECTURE 3

Deep Learning in Remote Sensing: Challenges

June 6th, 2018

Juelich Supercomputing Centre, Germany, Germany

UNIVERSITY OF ICELAND
SCHOOL OF ENGINEERING AND NATURAL SCIENCES
FACULTY OF INDUSTRIAL ENGINEERING, MECHANICAL ENGINEERING AND COMPUTER SCIENCE

JÜLICH FORSCHUNGSZENTRUM

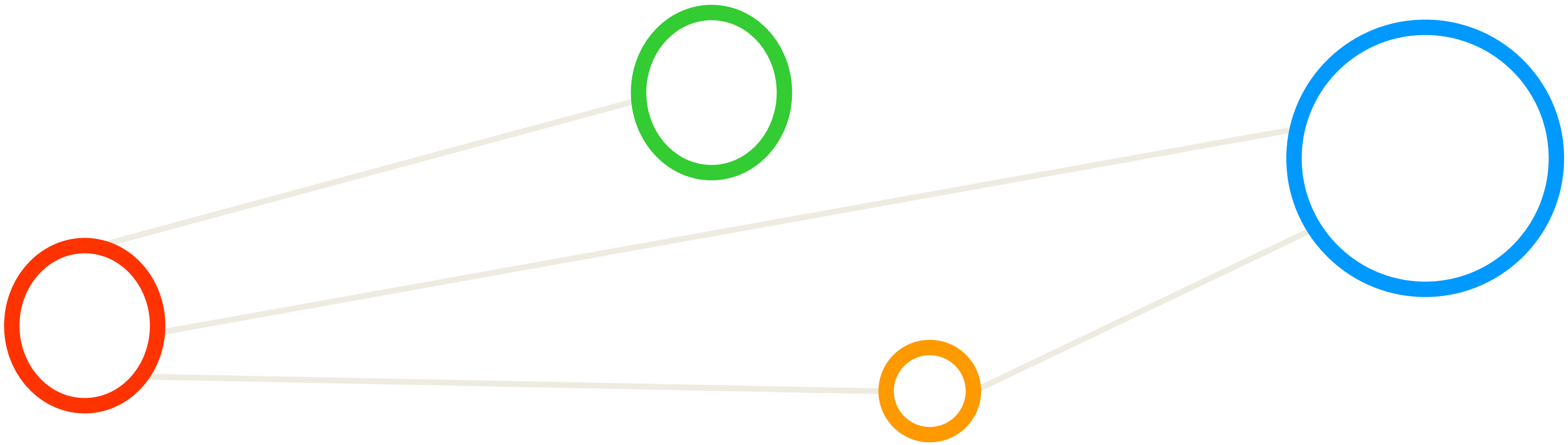
HELMHOLTZ RESEARCH FOR GRAND CHALLENGES

DEEP Projects

Outline of the Course

1. Introduction to Deep Learning
2. Fundamentals of Convolutional Neural Networks (CNNs)
3. Deep Learning in Remote Sensing: Challenges
4. Deep Learning in Remote Sensing: Applications
5. Model Selection and Regularization
6. Fundamentals of Long Short-Term Memory (LSTM)
7. LSTM Applications and Challenges
8. Deep Reinforcement Learning

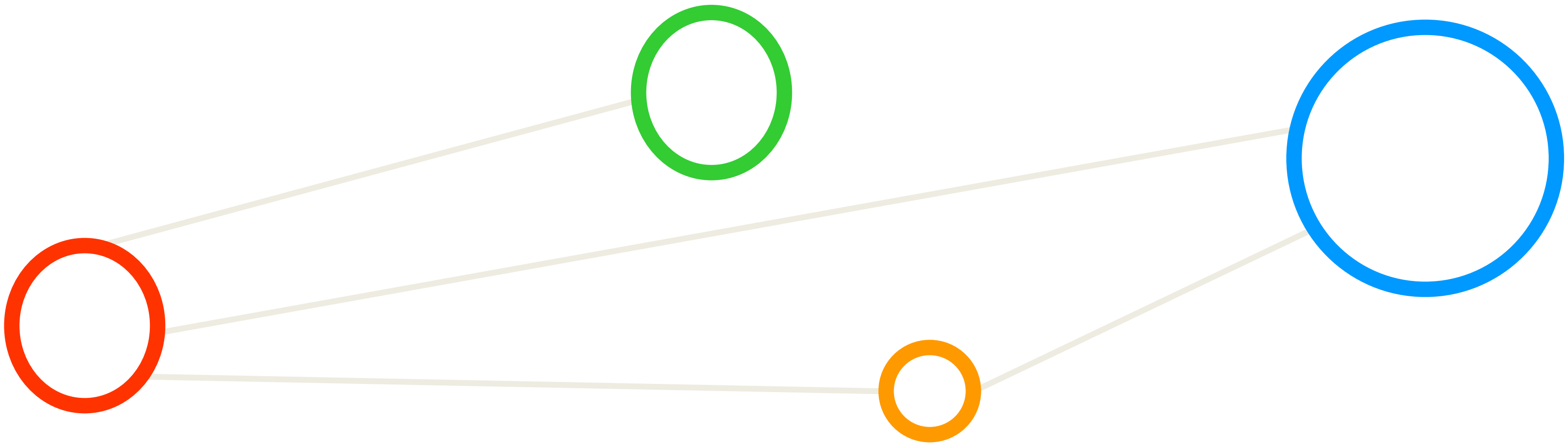
Outline



Outline

- Remote Sensing (RS) Overview
  - RS Background
  - Challenges for Deep Learning (DL) in RS
  - ISPRS 2D Semantic Labeling Contest
  - Data Augmentation
  - Transfer Learning
  - DL and Shallow Learning
  - Hiercarchial Features of CNNs

Remote Sensing Background

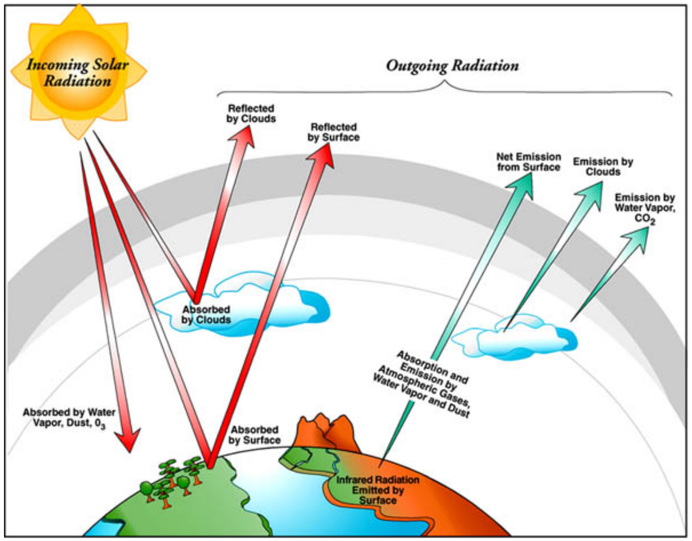


Remote Sensing

Remote (without physical contact) **Sensing** (measurement of information)

[1] Satellite (1960)

- The term **remote sensing** was first used in the United States in the 1950s by Ms. Evelyn Pruitt of the U.S. Office of Naval Research



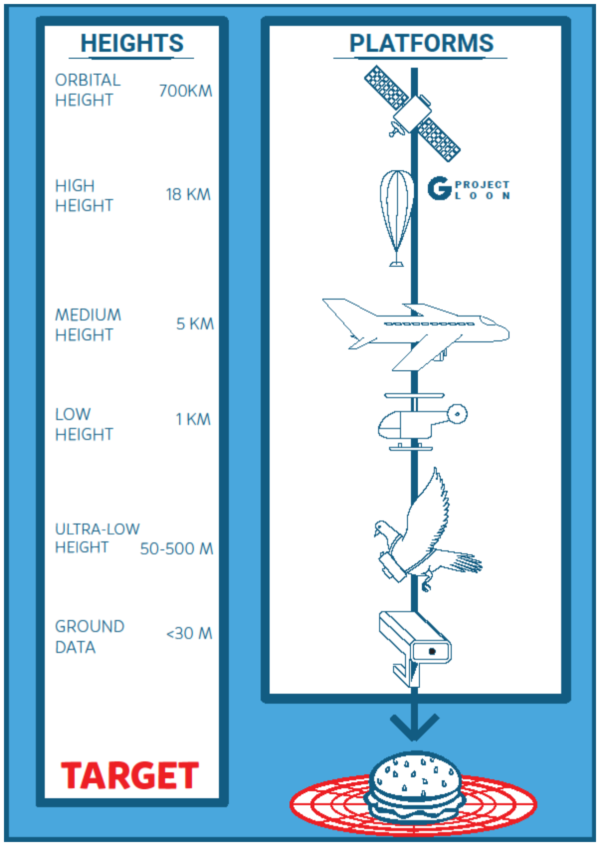
[2] The Earth-Atmosphere Energy Balance

- Measurement of radiation of different wavelengths reflected or emitted from distant objects or materials

- They may be categorized by class/type, substance, and spatial distribution

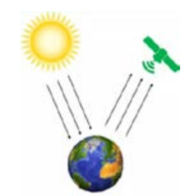
Platforms and Sensors

- ***Platform:*** selected according to the application



- ***Active Sensor**:* own source of illumination
  - Capture image in day and night
  - Any weather or cloud conditions

- ***Passive Sensor:*** natural light available
  - Great quality satellite imagery
  - Multispectral and Hyperspectral technology

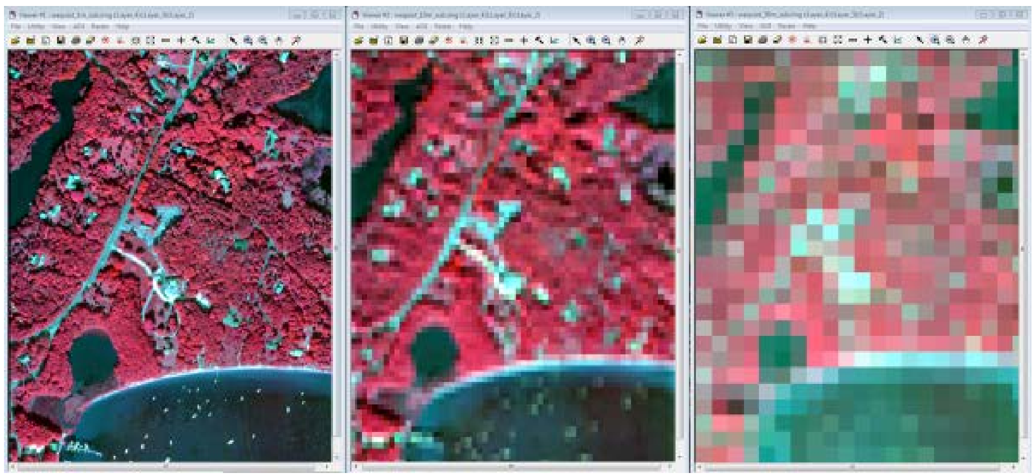


[3] Active-and-passive-remote-sensing

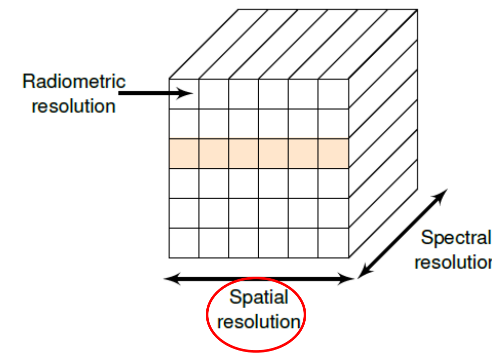
DATA ACQUISITION → PREPROCESSING → PROCESSING → INFORMATION ABSTRACTION

Spatial Resolution

- The spatial minimum area discernable by a pixel (i.e., picture element)
- Influenced primarily by the sensor and the platform altitude



1m *10m* *30m*

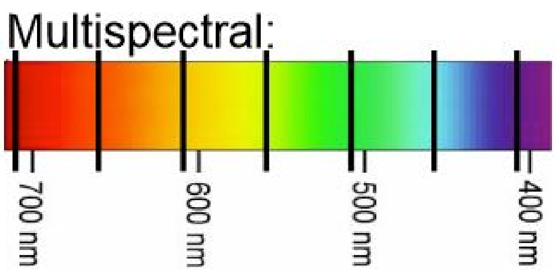


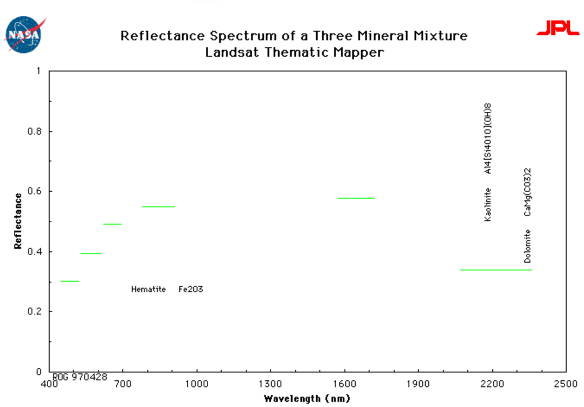
- Landscapes vary greatly in their spatial complexity
  - Some may be represented clearly at coarse levels of detail
  - Others are so complex that the finest level of detail is required

Spectral Resolution

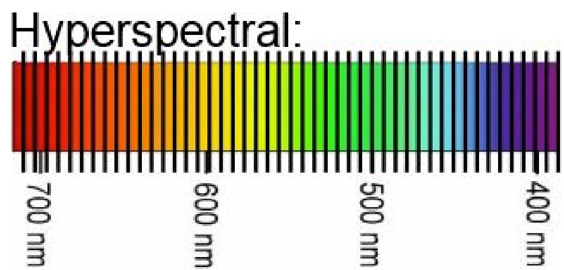
- Remote sensing observes at varied wavelengths
- Ability to define fine wavelength intervals
- The finer the spectral resolution, the narrower the wavelength range of a particular band

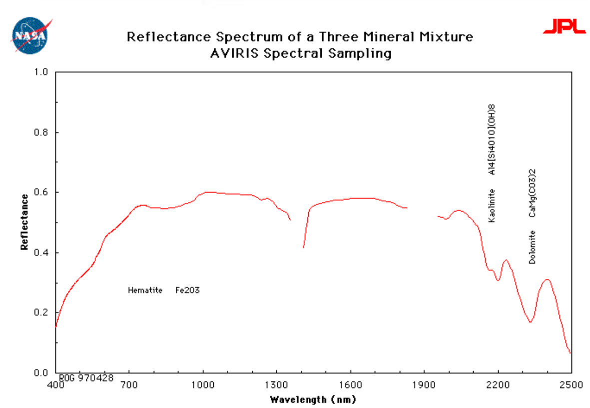
Landsat (7-11 bands)

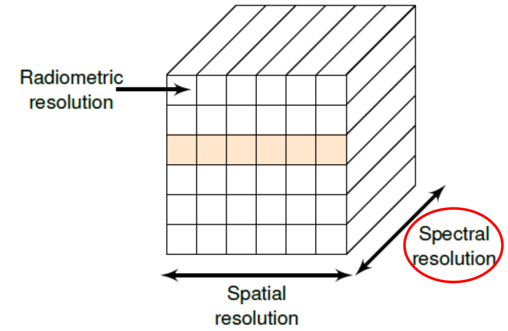




AVIRIS (256 bands)







Spectrum of a mixture of three common minerals: kaolinite, Dolomite, Hematite

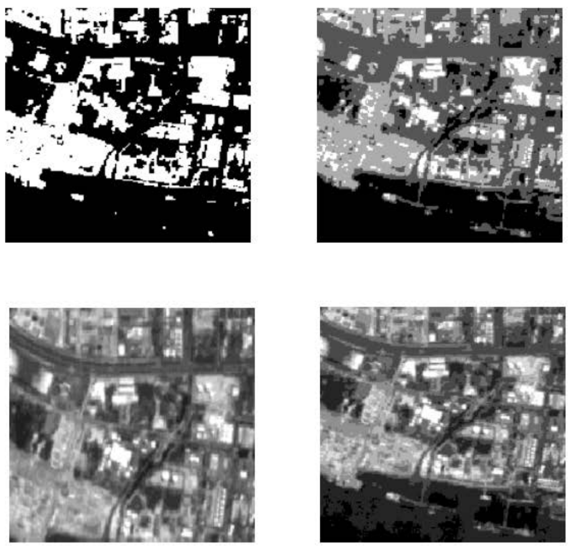
[4] Airborne Hyperspectral

Radiometric Resolution

- Maximum number of brightness levels available
- Depends on the number of bits used in representing the energy recorded
- Higher radiometric resolution, sharper images

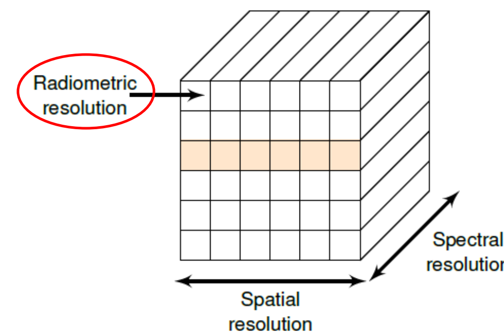
2 - levels

4 - levels



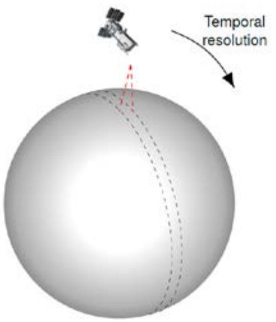
8 - levels

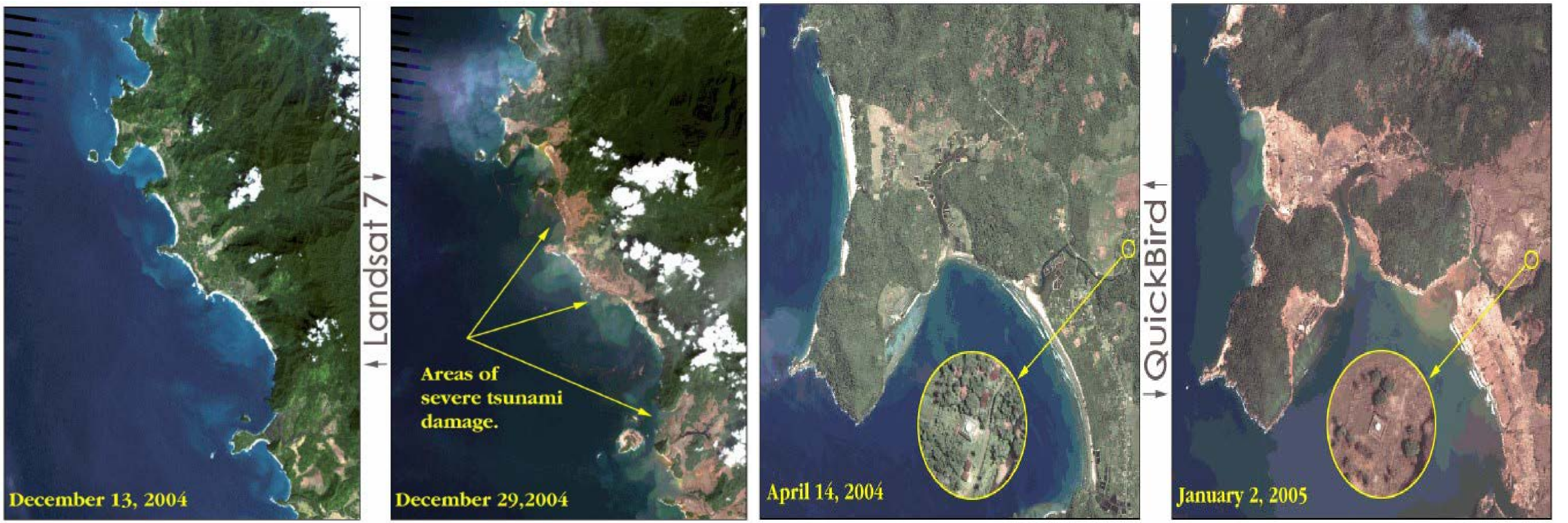
16 - levels



Temporal Resolution

- The time it takes for a satellite to complete one orbit cycle (revisit time)
- Depends on the satellite/sensor capabilities, swath overlap and latitude



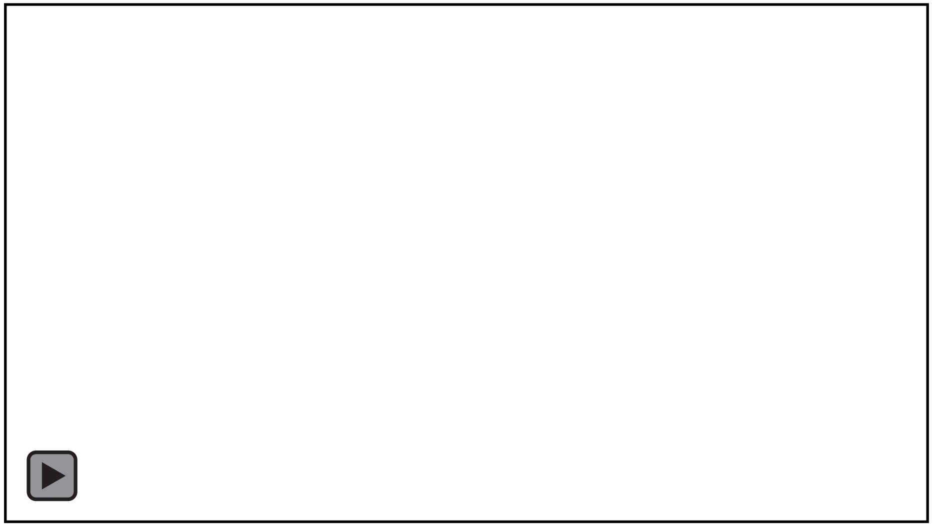


[5] Tsunami Damage, Gleebruk, Indonesia January 7, 2005

Repeated imaging enables assessment of changes in the type or condition of surface features

Sentinel 2 Mission

- **Platform:** Twin polar-orbiting satellites, phased at 180° to each other
- **Temporal resolution** of 5 days at the equator in cloud-free conditions



~23 TB data stored per day

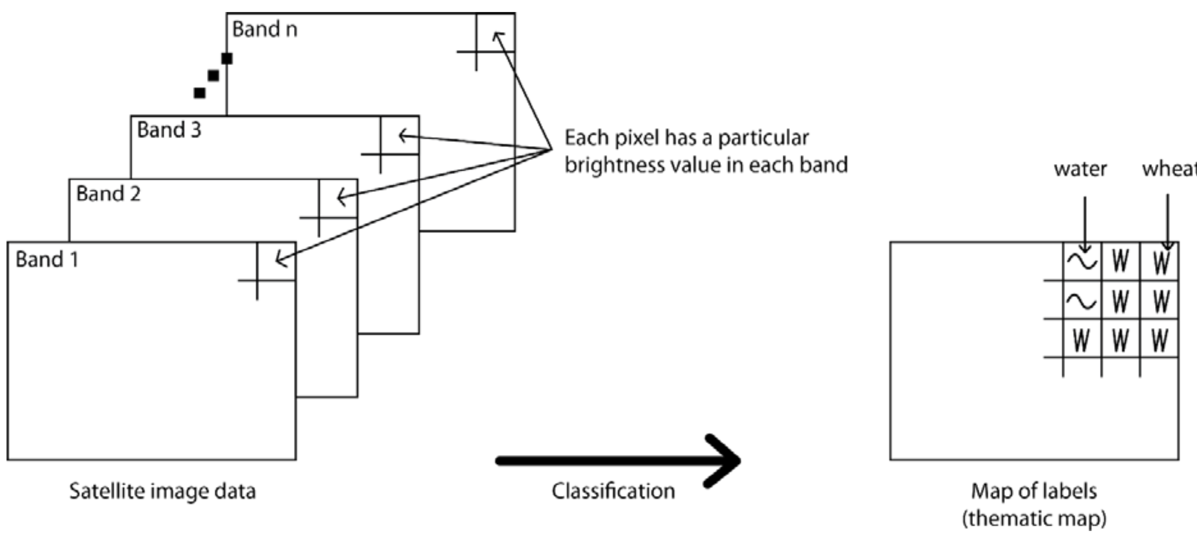
Spatial Resolution (m)	Band Number	S2A		S2B	
		Central Wavelength (nm)	Bandwidth (nm)	Central Wavelength (nm)	Bandwidth (nm)
10	2	496.6	98	492.1	98
	3	560.0	45	559	46
	4	664.5	38	665	39
	8	835.1	145	833	133
20	5	703.9	19	703.8	20
	6	740.2	18	739.1	18
	7	782.5	28	779.7	28
	8a	864.8	33	864	32
	11	1613.7	143	1610.4	141
	12	2202.4	242	2185.7	238
60	1	443.9	27	442.3	45
	9	945.0	26	943.2	27
	10	1373.5	75	1376.9	76

[6] Earth Observation Mission Sentinel 2

- Images for mapping land-use change, land-cover change biophysical variables
- Monitor the coastal and inland waters and help with risk and disaster mapping

Classification of Remote Sensing Images

- Perhaps the most common form of image interpretation



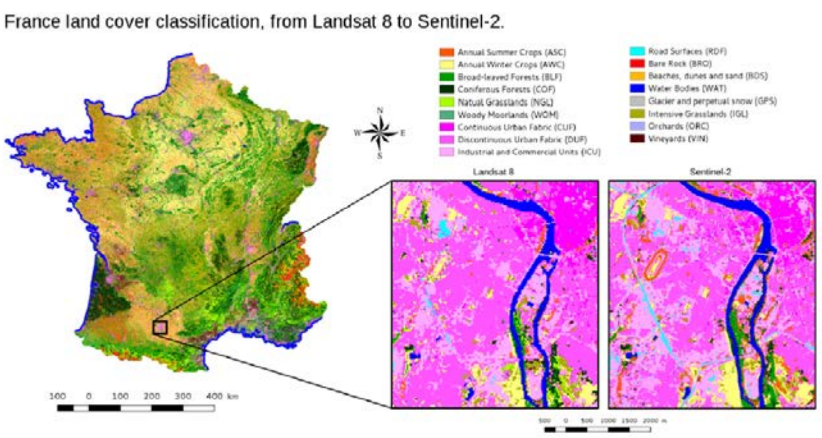
Generation of thematic maps

- Applications: environmental management, agricultural planning, health studies, climate and biodiversity monitoring, and land change detection

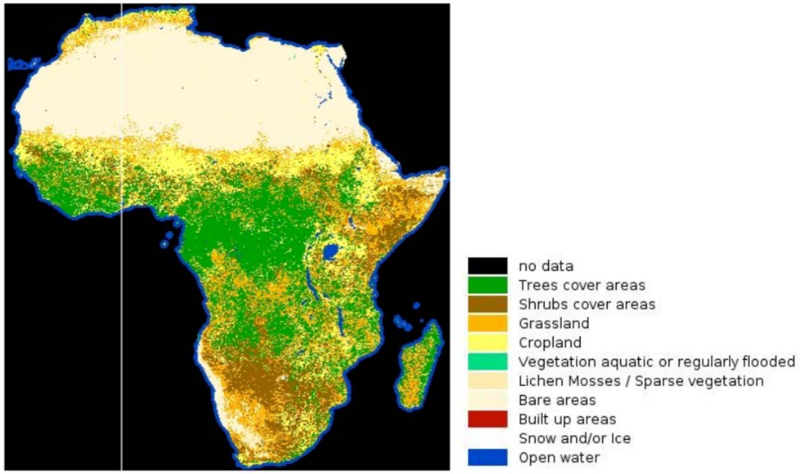
Update of Land-Cover Maps at Country Scale

- Important task for regularly monitoring the Earth's surface
- Generation of reliable maps with spatial consistency is challenging
  - Data from several satellite orbit tracks, different dates and presence of clouds
- Cannot rely on field campaigns
  - Use existing databases to build the reference sets for supervised classification

[7] CORINE land cover



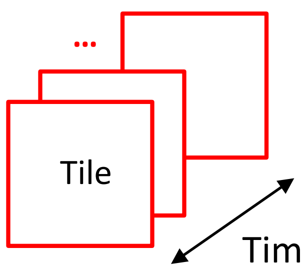
[8] France land cover classification 2016



[9] S2 prototype LC map at 20m of Africa 2016

Germany – 1 year of Sentinel 2 Data

- Sentinel-2 tile gridding is based on the NATO Military Grid Reference System
- Each tile covers an area of 100 km x 100 km (excluding overlapping edges of 9.8 km)
- Typical file size of all bands with a tile: ~700 MB
- Germany can be covered with 56 tiles



Time serie of tile images for 1 year
(365 days / 5 days = 73 acquisitions)

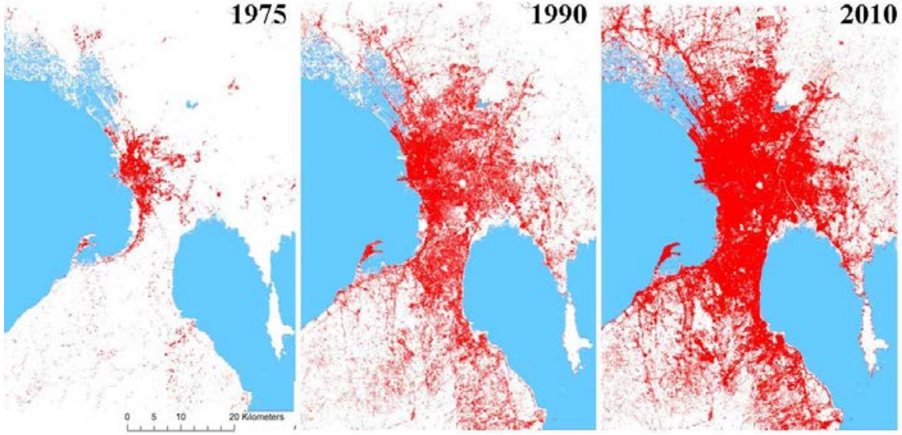
- Data size to be processed: 73 acq. * 56 tiles * 700Mb = 2.72 TB

[10] Military Grid Reference System

Other Applications

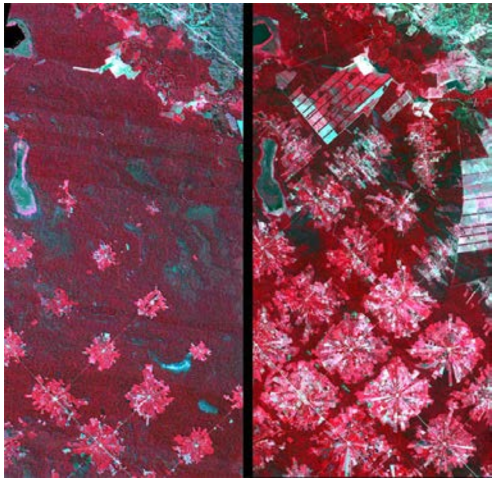
- Environmental assessment and monitoring: ***Urban growth***

[11] Population growth from 1975 - 2010 of Manila



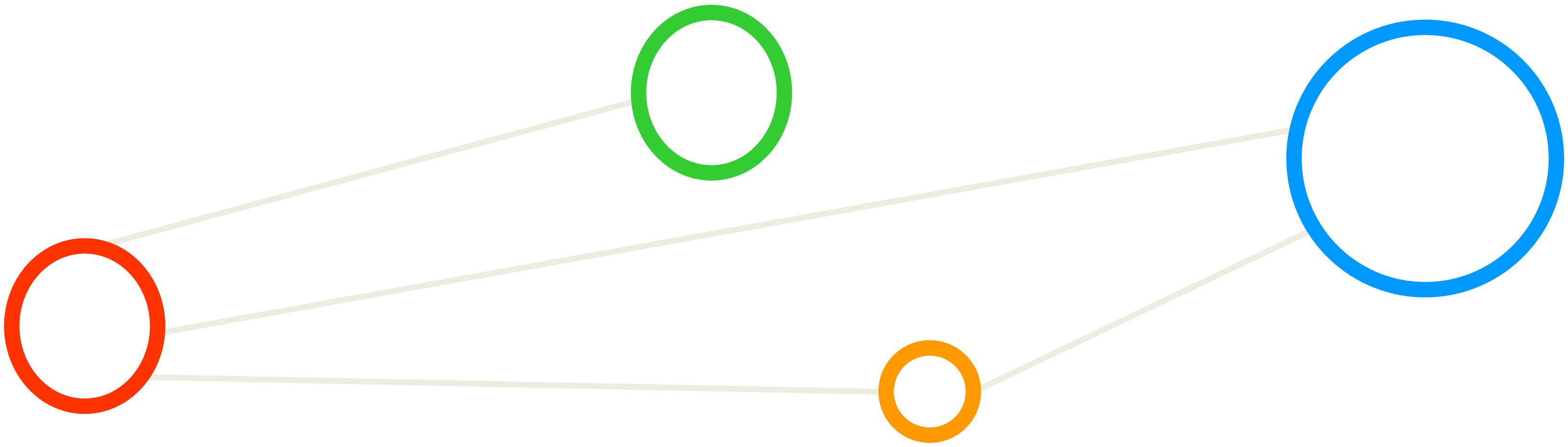
- Global change detection and monitoring: ***Deforestation***

[12] Deforestation in Bolivia from 1986 to 2001



1986 ***2001***

Challenges for Deep Learning in Remote Sensing



Deep Learning in Remote Sensing

- DL has risen to the top in numerous areas in the last years
  - Computer vision, speech recognition, etc.

- Deep learning is also taking off in RS

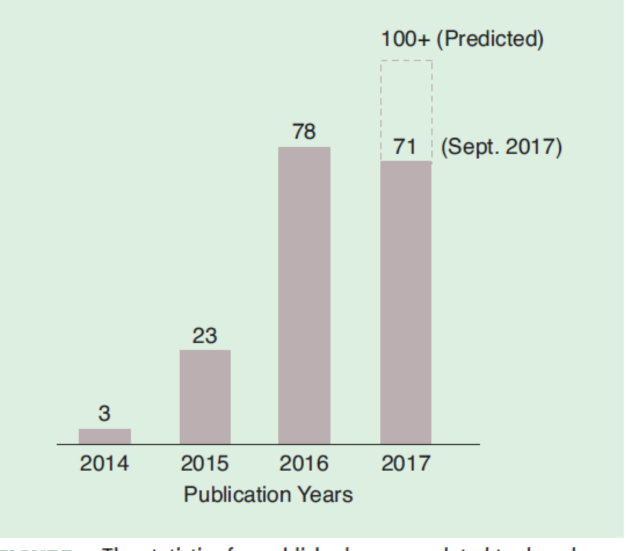


FIGURE 4. The statistics for published papers related to deep learning in remote sensing [187].

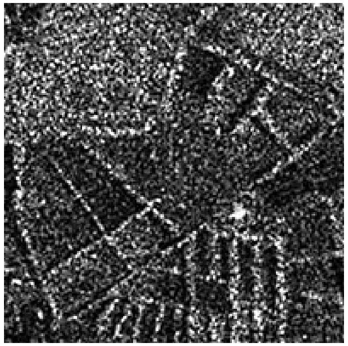
Exponential increase of the paper published

[13] X. X. Zhu et al

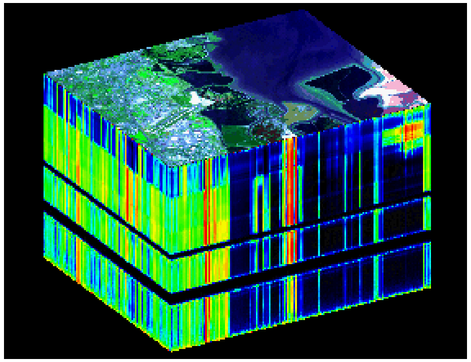
- RS possesses a number of unique challenges for deep learning

Main Challenges for DL in RS

- **Multimodal data:** geometries and content are completely different
  - From optical (multi- and hyperspectral), Lidar, and synthetic aperture radar (SAR) sensors
- **High temporal resolutions data**
  - Shift from individual image analysis to time-series processing
- **Big remote sensing data:** high spatial and more-so spectral dimensionality
  - Traditional DL systems operate on relatively small grayscale or RGB imagery



SAR images: noisy data



Optical data: From four to hundreds to of channels

Limited Number of Remote Sensing Datasets

- The existing datasets have a number of limitations
  - Small scale of scene classes and the image numbers
  - Lack of image variations and diversity
  - Saturation of accuracy

Table 5 HSI Dataset Usage.

Dataset and Reference	Number of uses
IEEE GRSS 2013 Data Fusion Contest[338]	4
IEEE GRSS 2015 Data Fusion Contest[339]	1
IEEE GRSS 2016 Data Fusion Contest[340]	2
Indian Pines[341]	27
Kennedy Space Center[342]	8
Pavia City Center[343]	13
Pavia University[343]	19
Salinas[344]	11
Washington DC Mall[345]	2

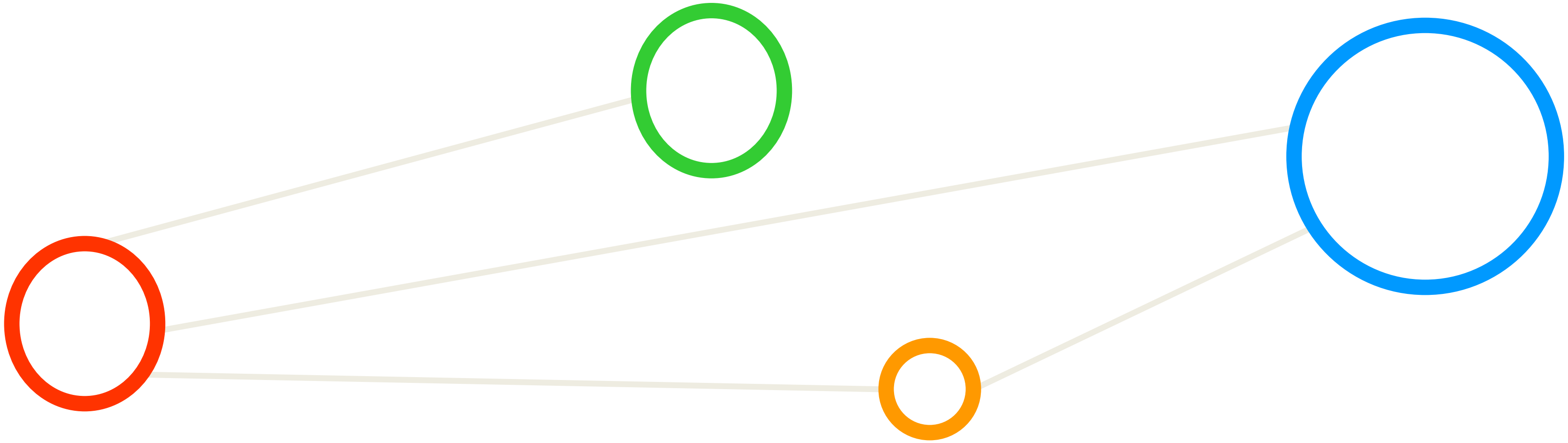
[14] J. E. Ball

- They severely limit the development of new deep learning-based methods

Datasets	Image per class	Scene classes	Total images	Spatial resolution (m)	Image sizes	Year	Ref.
UC Merced Land-Use	100	21	2100	0.3	256x256	2010	***[15]***
WHU-RS19	~50	19	1005	up to 0.5	600x600	2012	***[16]***
RSSCN7	400	7	2800	--	400x400	2015	***[17]***
SAT-6	--	6	405000	1	28x28	2015	***[18]***
SIRI-WHU	200	12	2400	2	200x200	2016	***[19]***
RSC11	~100	11	1323	0.2	512x512	2016	***[20]***
Brazilian Coffee Scene	1438	2	2876	--	64x64	2016	***[21]***
NWPU-RESISC45	700	45	31500	~30 to 0.2	256x256	2016	***[22]***
AID	~300	30	10000	0.6	600x600	2016	***[23]***
EuroSAT	~2500	10	27000	10	64x64	2017	***[24]***

- ImageNet dataset: 14,197,122 images ***[25] ImageNet project***

ISPRS 2D Semantic Labeling Contest



ISPRS 2D Semantic Labeling Contest

- 2D semantic segmentation that assigns labels to multiple object categories

- Acquired by airborne sensors
  - Very high resolution true ortho photo tiles
  - Digital surface models (DSMs) derived from dense image matching techniques

- Very heterogeneous appearance of objects
  - High intra-class variance and low inter-class variance

Vaihingen

Town with many detached buildings and small multi story buildings

[26] 2D Semantic Labeling Contest

Potsdam

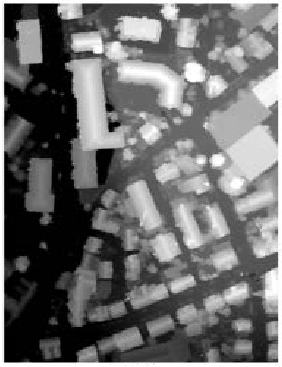
Historic city with large building blocks, narrow streets and dense settlement structure

Vaihingen Dataset (1)

- 33 patches of different sizes with 9 cm spatial resolution
- Manually classified into six land cover classes
  - *Impervious surfaces, Building, Low vegetation, Tree, Clutter/background*

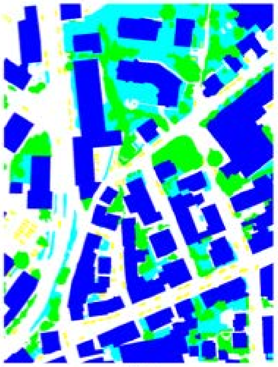
True orthophoto

- Near infrared
- Red
- Green

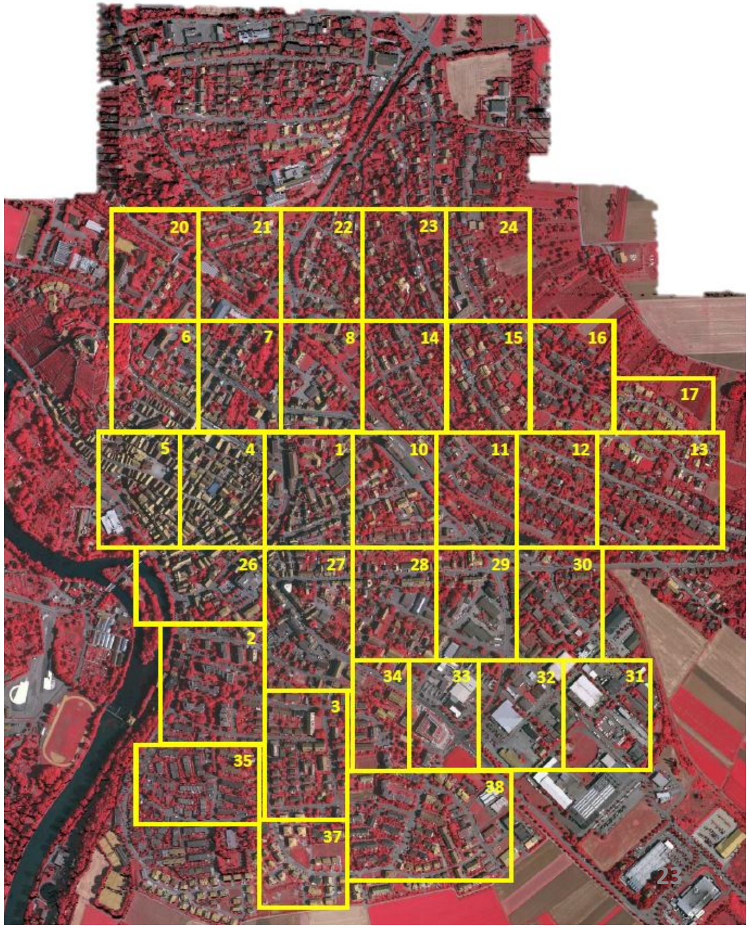


DSM

One band
Grey levels mapped to DSM heights



Groundtruth



Vaihingen Dataset (2)

- The groundtruth is provided for only 16 patches
- For the remaining scenes is unreleased and used for evaluation of submitted results

TOP	DSM	N_{col}	N_{row}	GT
top_mosaic_09cm_area1	dsm_09cm_matching_area1	1919	2569	top_mosaic_09cm_area1
top_mosaic_09cm_area2	dsm_09cm_matching_area2	2428	2767	
top_mosaic_09cm_area3	dsm_09cm_matching_area3	2006	3007	top_mosaic_09cm_area3
top_mosaic_09cm_area4	dsm_09cm_matching_area4	1887	2557	
top_mosaic_09cm_area5	dsm_09cm_matching_area5	1887	2557	top_mosaic_09cm_area5
top_mosaic_09cm_area6	dsm_09cm_matching_area6	1887	2557	
top_mosaic_09cm_area7	dsm_09cm_matching_area7	1887	2557	top_mosaic_09cm_area7
top_mosaic_09cm_area8	dsm_09cm_matching_area8	1887	2557	
top_mosaic_09cm_area10	dsm_09cm_matching_area10	1887	2557	
top_mosaic_09cm_area11	dsm_09cm_matching_area11	1893	2566	top_mosaic_09cm_area11
top_mosaic_09cm_area12	dsm_09cm_matching_area12	1922	2575	
top_mosaic_09cm_area13	dsm_09cm_matching_area13	2818	2558	top_mosaic_09cm_area13
top_mosaic_09cm_area14	dsm_09cm_matching_area14	1919	2565	
top_mosaic_09cm_area15	dsm_09cm_matching_area15	1919	2565	top_mosaic_09cm_area15
top_mosaic_09cm_area16	dsm_09cm_matching_area16	1919	2565	
top_mosaic_09cm_area17	dsm_09cm_matching_area17	2336	1281	top_mosaic_09cm_area17
top_mosaic_09cm_area20	dsm_09cm_matching_area20	1866	2315	
top_mosaic_09cm_area21	dsm_09cm_matching_area21	1903	2546	top_mosaic_09cm_area21
top_mosaic_09cm_area22	dsm_09cm_matching_area22	1903	2546	
top_mosaic_09cm_area23	dsm_09cm_matching_area23	1903	2546	top_mosaic_09cm_area23
top_mosaic_09cm_area24	dsm_09cm_matching_area24	1903	2546	
top_mosaic_09cm_area26	dsm_09cm_matching_area26	2995	1783	top_mosaic_09cm_area26
top_mosaic_09cm_area27	dsm_09cm_matching_area27	1917	3313	
top_mosaic_09cm_area28	dsm_09cm_matching_area28	1917	2567	top_mosaic_09cm_area28
top_mosaic_09cm_area29	dsm_09cm_matching_area29	1917	2563	
top_mosaic_09cm_area30	dsm_09cm_matching_area30	1934	2563	top_mosaic_09cm_area30
top_mosaic_09cm_area31	dsm_09cm_matching_area31	1980	2555	
top_mosaic_09cm_area32	dsm_09cm_matching_area32	1980	2555	top_mosaic_09cm_area32
top_mosaic_09cm_area33	dsm_09cm_matching_area33	1581	2555	
top_mosaic_09cm_area34	dsm_09cm_matching_area34	1388	2555	top_mosaic_09cm_area34
top_mosaic_09cm_area35	dsm_09cm_matching_area35	2805	1884	
top_mosaic_09cm_area37	dsm_09cm_matching_area37	1996	1995	top_mosaic_09cm_area37
top_mosaic_09cm_area38	dsm_09cm_matching_area38	3816	2550	

ISPRS Semantic Labeling Contest (2D): Results

Click here for a description of evaluation measures

Vaihingen: 2D Labelling challenge

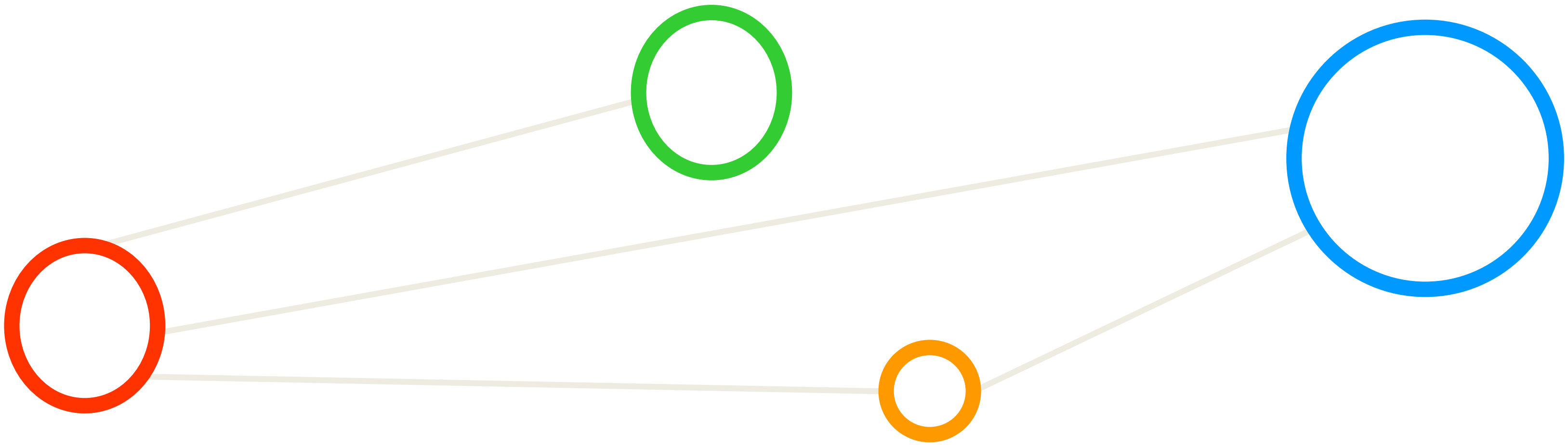
All quality measures except for *overall* are F1 scores in [%] using the reference with eroded boundaries. Mouse over the column headings will display more information.

Abbrev.	imp surf	building	low_veg	tree	car	Overall	Strategy	Deta
SVL_1	86.3	90.8	78.2	84.2	56.8	84.7	s	D F
SVL_2	82.1	82.8	71.6	81.6	51.9	79.4	s	D F
SVL_3	86.6	91.0	77.0	85.0	55.6	84.8	s	D F
SVL_4	86.1	90.9	77.6	84.9	59.9	84.7	s	D F
SVL_5	86.1	90.3	75.6	84.7	45.9	84.0	s	D F
SVL_6	86.0	90.2	75.6	82.1	45.4	83.2	s	D F
ADL_1	88.1	92.0	79.0	86.5	59.0	86.1	s	D F
ADL_2	89.0	93.0	81.0	87.8	59.5	87.3	s	D F
ADL_3	89.5	93.2	82.3	88.2	63.3	88.0	s	D F
UT_Mev	84.3	88.7	74.5	82.0	9.9	81.8	u	D F
HUST	86.9	92.0	78.3	86.9	29.0	85.9	s	D F
ONE_1	83.0	84.4	75.0	84.9	44.7	81.0	s	D F

[26] 2D Semantic Labeling Contest

“Participants shall use all data with ground truth for training or internal evaluation of their method”

Practical 1: Access JURECA and Find the Dataset



Vaihingen Dataset in JURECA

- Access JURECA: **$ ssh –X train???@jureca.fz-juelich.de**
- Data location: ***/homea/hpclab/train001/data/vaihingen/***

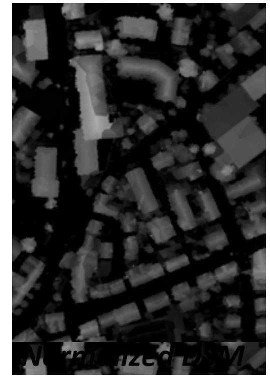
```
[train002@jrl05 ~]$ ls /homea/hpclab/train001/data/vaihingen/
vaihingen_11.hdf5  vaihingen_1.hdf5   vaihingen_28.hdf5  vaihingen_37.hdf5
vaihingen_13.hdf5  vaihingen_21.hdf5  vaihingen_30.hdf5  vaihingen_3.hdf5
vaihingen_15.hdf5  vaihingen_23.hdf5  vaihingen_32.hdf5  vaihingen_5.hdf5
vaihingen_17.hdf5  vaihingen_26.hdf5  vaihingen_34.hdf5  vaihingen_7.hdf5
[train002@jrl05 ~]$
```

- HDF5 files creation:

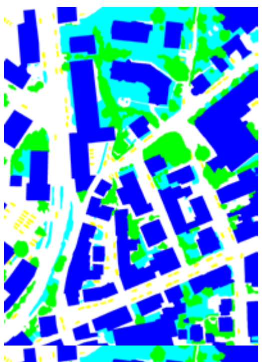
  str = 'Vaihingen_1.hdf5';
  h5write(str , '/x_1', cat(3, Near Infrared, Red, Green, Normalized DSM,NDVI*))*
  h5write(str , '/y_1', Groundtruth)
  h5write(str , '/m_1', Boundaries)

E.G., Vaihingen_1.hdf5

/x_1



/y_1



Groundtruth

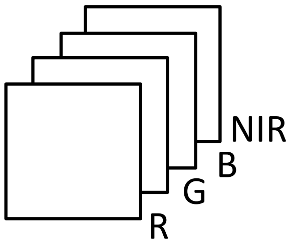
Impervious Surface	
Building	
Low Vegetation	
Trees	
Car	
Clutter/Background	

/m_1

- Uncertainty associated with the boundary of objects in the groundtruth
- These boundaries are ignored for the evaluation
- Groundtruth images with boundary pixels (in a 3 pixel radius) provided

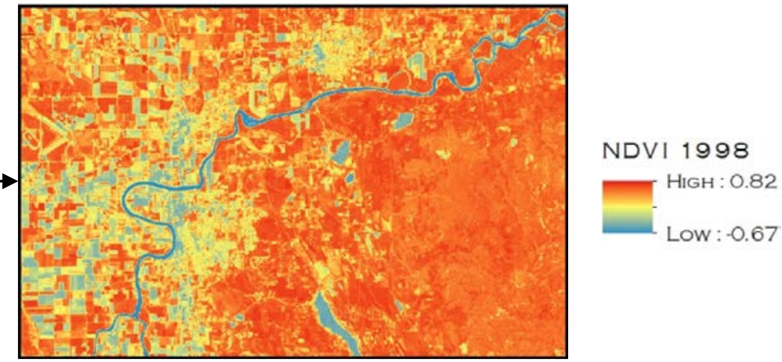
Normalized Difference Vegetation Index (NDVI)

- Create additional relevant features from the existing raw features in the data
- Increase the predictive power of the classifier

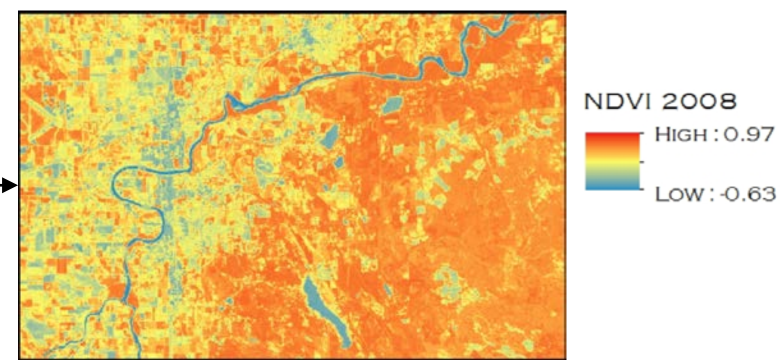


NDVI = (NIR – R) / (NIR + R)

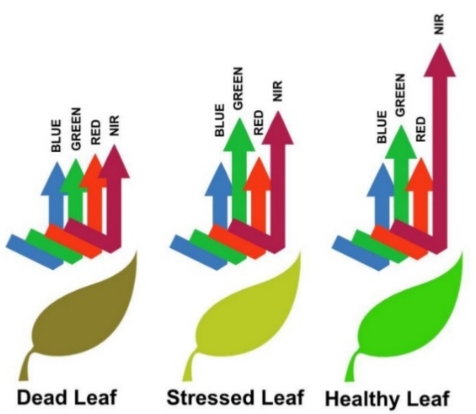
PREPROCESSING → PREPROCESSING → INFORMATION ABSTRACTION



Bellingham, WA , US



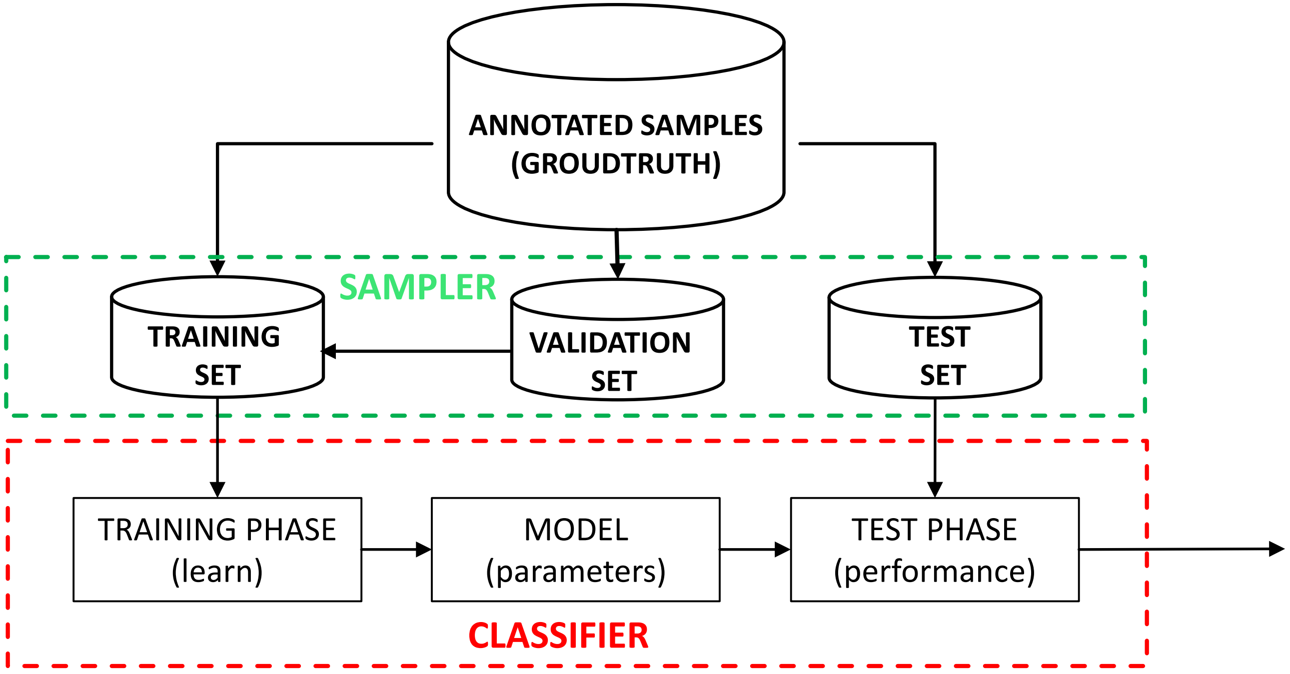
[27] NDVI



[28] NDVI & Classification

Assess Classifier Performance

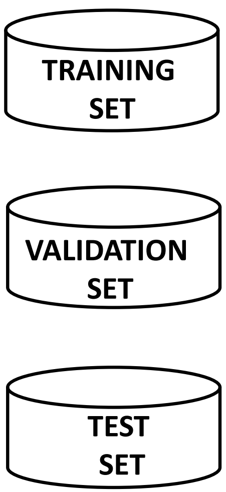
- If annotated samples are available, the classifier parameters are learned in a supervised way
- How to estimate the generalization error: split the groundtruth into three disjoint sets

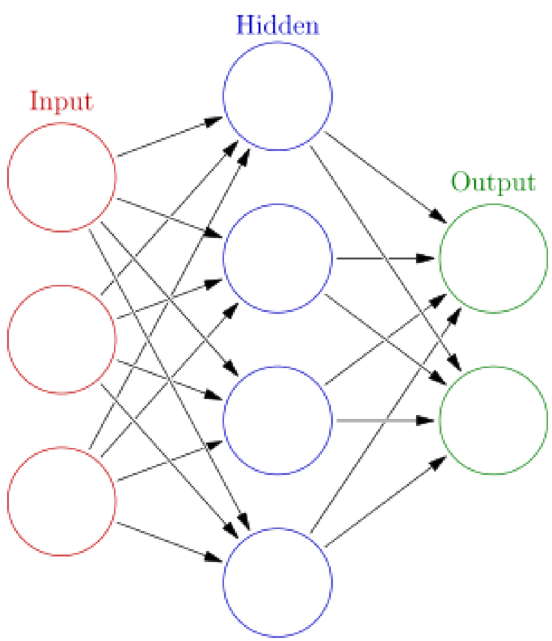


Performances: usually more influenced by the amount and quality of the training samples (i.e., sampling design) rather than the classifier/model complexity

Assess Classifier Performance

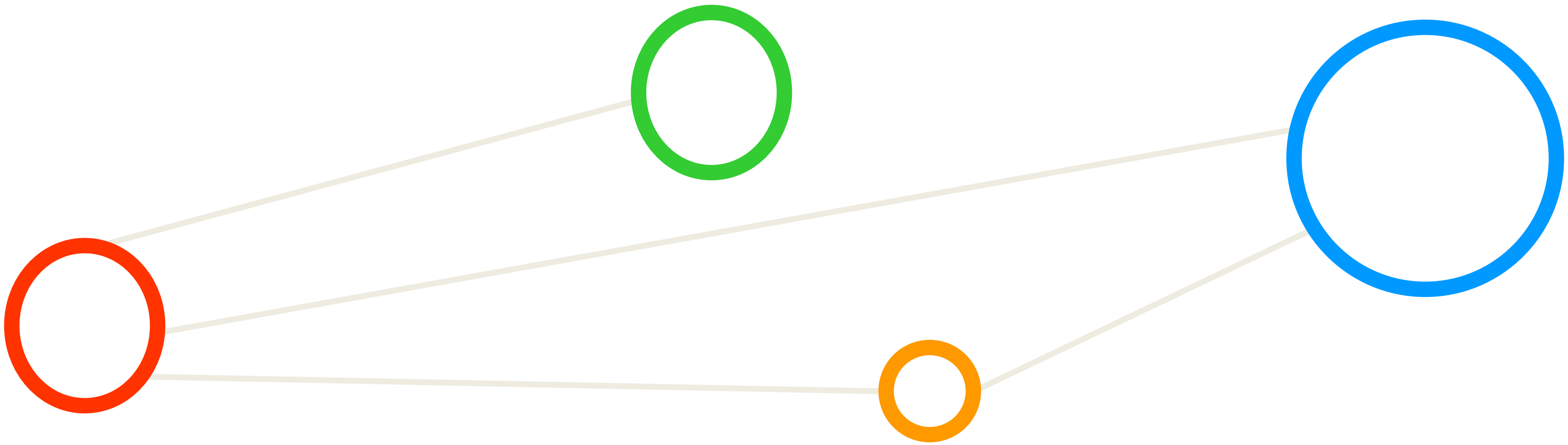
- **Training set**: used to train the model
  - How do we ensure that the model is not overfitting to the data in the training set?
- **Validation set**: used to validate the model during training
  - Its classification is based only on the model that is learnt from the training set
  - The model weights are updated based on this set
  - Help to adjust the hyperparameters (e.g., number of hidden layers, learning rate, etc..)
- **Test set:** used to test the model after it has been trained





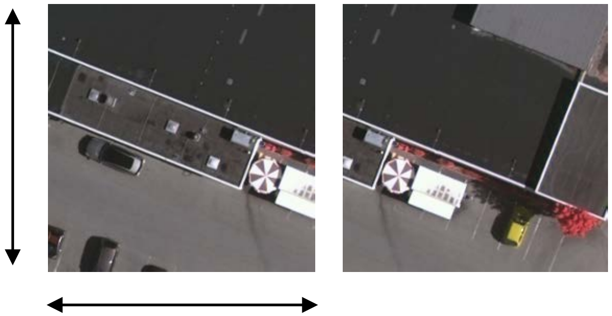

```
Epoch 7/10
2152/2152 [==============================] - 34s 16ms/step - loss: 0.2539 - acc: 0.9021 - v
al_loss: 0.4802 - val_acc: 0.8374
Epoch 8/10
2152/2152 [==============================] - 34s 16ms/step - loss: 0.2087 - acc: 0.9198 - v
al_loss: 0.4097 - val_acc: 0.8589
Epoch 9/10
2152/2152 [==============================] - 34s 16ms/step - loss: 0.2465 - acc: 0.9080 - v
al_loss: 0.8998 - val_acc: 0.7405
Epoch 10/10
2152/2152 [==============================] - 34s 16ms/step - loss: 0.3857 - acc: 0.8568 - v
al_loss: 0.4116 - val_acc: 0.8549
```

Practical 2:
Generate your Copy of the Training and Validation Sets



Approach for Training and Validation Set Generation

256



...

256 → **Not a random number, will be later explained …**

- Generate dataset of 256x256 sized image patches

```
def main(arguments):

    data_path = arguments[1]
    output_path = arguments[2]

    # files used for training:
    training_nums = [1, 3, 5, 7, 11, 13, 17, 21, 26, 28, 34, 37]

    # files used for validation:
    validation_nums = [30, 32]

    # generate and save the training and validation set:
    overlap = 0.6
```

Get the Code and Test the Python Environment

1. Get a copy of the folder **/homea/hpclab/train001/tools/resnet50-fcn**
   - Create a new folder in your local path **$ ~/semseg**
   - Copy **$ cp -R /homea/hpclab/train001/tools/resnet50-fcn ~/semseg/**

2. All modules and python packages have been already prepared
   - Just run -> **$ module restore dl_tutorial**
   - How was it setup?
     - $ module use /usr/local/software/jureca/OtherStages
     - $ ml Stages/Devel-2017a
     - $ ml GCC/5.4.0 MVAPICH2
     - $ ml TensorFlow/1.4.0-Python-2.7.13
     - $ pip install --user virtualenv
     - $ pip install --user h5py
     - $ pip install --user keras
     - $ pip install --user sklearn
     - $ module store dl_tutorial

3. Check if Keras is available
   - **$ python**
   - **>>> import keras**

```
[train002@jrl12 code]$ python
Python 2.7.13 (default, Feb 14 2018, 10:29:12)
[GCC 5.4.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import keras
/homea/hpclab/train002/.local/lib/python2.7/site-packages/h5py/__init__.py:36: FutureWarnin
g: Conversion of the second argument of issubdtype from `float` to `np.floating` is depreca
ted. In future, it will be treated as `np.float64 == np.dtype(float).type`.
  from ._conv import register_converters as _register_converters
Using TensorFlow backend.
```

Generate the Training and Validation Set

4. Use the function ***~/semseg/resnet50-fcn/data_io.py***

 - *If you run* ***$ python data_io.py***

```
[train002@jrl12 resnet50-fcn]$ python data_io.py
/homea/hpclab/train002/.local/lib/python2.7/site-packages/h5py/__init__.py:36: FutureWarning: Conve
 future, it will be treated as `np.float64 == np.dtype(float).type`.
  from ._conv import register_converters as _register_converters

************************************
Two paremeters need to be specified:
1. Location of the input image patches (e.g., /homea/hpclab/train001/data/vaihingen/ )
2. Location to write training and validation sets (e.g., /homea/hpclab/train002/semseg/vaihingen/ )
************************************
```

- Create a new folder where to save the sets **$ makdir ~/semseg/vaihingen**
- Run the function:
  - **$ python data_io.py /homea/hpclab/train001/data/vaihingen/ ~/semseg/vaihingen/**

The Outcome

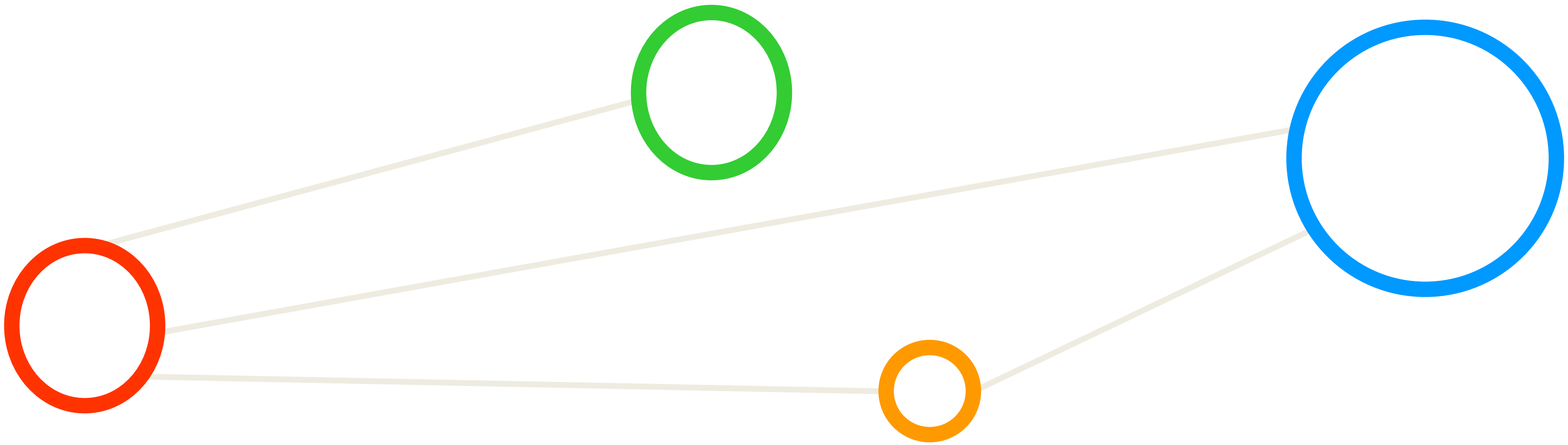
- The patches are assigned to the training and validation sets

```
[train002@jrl03 resnet50-fcn]$ python data_io.py /homea/hpclab/train001/data/vaihingen/ ~/semseg/vaihingen/
/homea/hpclab/train002/.local/lib/python2.7/site-packages/h5py/__init__.py:36: FutureWarning: Conversion of t
econd argument of issubdtype from `float` to `np.floating` is deprecated. In future, it will be treated as `n
oat64 == np.dtype(float).type`.
  from ._conv import register_converters as _register_converters
/homea/hpclab/train001/data/vaihingen/vaihingen_1.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_3.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_5.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_7.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_11.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_13.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_17.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_21.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_26.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_28.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_34.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_37.hdf5
Generated 2083 samples!
/homea/hpclab/train001/data/vaihingen/vaihingen_30.hdf5
/homea/hpclab/train001/data/vaihingen/vaihingen_32.hdf5
Generated 368 samples!
```

  - These sets will be used for training the network

```
[train002@jrl10 vaihingen]$ pwd
/homea/hpclab/train002/semseg/vaihingen
[train002@jrl10 vaihingen]$ ls
vaihingen_train.hdf5  vaihingen_val.hdf5
[train002@jrl10 vaihingen]$
```

Transfer Learning and Data Augmentation

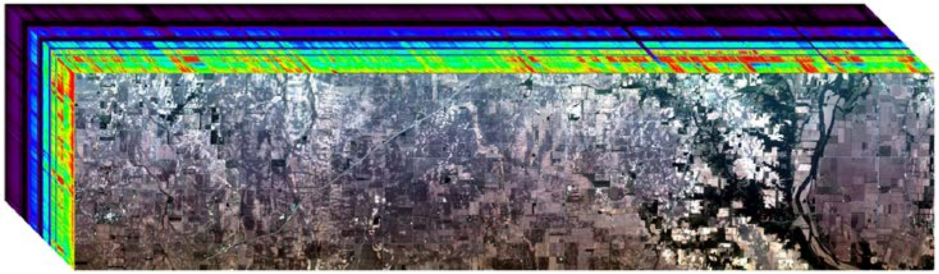


Limited Remote Sensing Training Data

- RS applications have massive amounts of temporal and spatial data (e.g., Sentinel 2)

- But not enough labeled training samples, which usually don't fully represent:
  - Seasonal variations
  - Object variation (e.g., plants, crops, etc.)

- Most online hyperspectral data sets have little-to-no variety



[30] Indian Pines dataset

- DL systems with many parameters require large amounts of training data
  - Else they can easily overtrain and not generalize well

- DL systems in CV use very large training sets
  e.g., millions or billions of faces in different illuminations, poses, inner class variations, etc.

DL systems with limited training data

- Possible approaches to mitigate small training samples:

  1. **Data augmentation**
     - Affine transformations, rotations, small patch removal, etc.

  2. **Transfer learning**
     - Train on other imagery to obtain low-level to mid-level features

  3. **Use ancillary data**
     - Other sensor modalities (e.g., LiDAR, SAR, etc.)

  4. **Unsupervised training**
     - training labels not required

Need for a Large Amount of Training Data

- State of the art DL networks have parameters in the order of millions
  - The learning model needs a proportional amount of examples
  - The number of parameters should be proportional to the complexity of the task

	VGGNet	DeepVideo	GNMT
Used For	Identifying Image Category	Identifying Video Category	Translation
Input	Image	Video	English Text
Output	1000 Categories	47 Categories	French Text
Parameters	140M	~100M	380M
Data Size	1.2M Images with assigned Category	1.1M Videos with assigned Category	6M Sentence Pairs, 340M Words
Dataset	ILSVRC-2012	Sports-1M	WMT'14

[31] Data Augmentation

- The available dataset is taken in a limited set of conditions
  - Different orientation, location, scale, brightness etc.

Data Augmentation

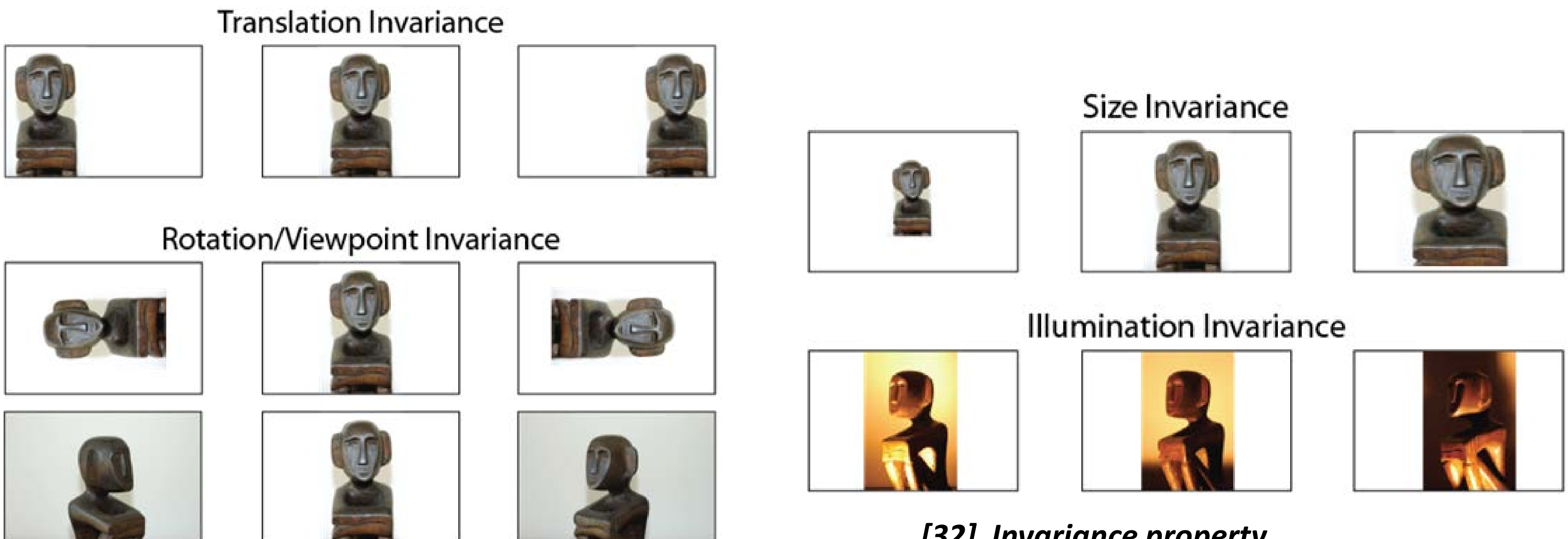
- Train with additional synthetically modified data
- Techniques to artificially increase the size of the training set
- Make minor changes such as flips, translations and rotations to the existing dataset
- Employed to counteract overfitting

[31] Data Augmentation

"A poorly trained neural network would think that these three tennis balls, are distinct, unique images"

Essential Assumption: Invariance

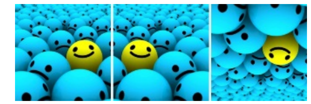
- Ability to recognize an object as an object, even when its appearance varies in some way
- It allows to abstract an object's identity from the specifics of the visual input
  - E.g., relative positions of the viewer/camera and the object.
- Well-trained CNNs can be invariant to translation, viewpoint, size or illumination



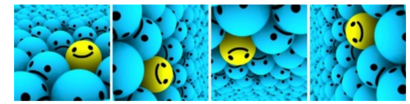
[32] Invariance property

Popular Augmentation Techniques

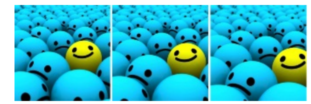
- Flip horizontally and vertically



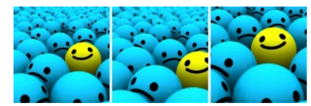
- Rotate



- Scaled outward or inward

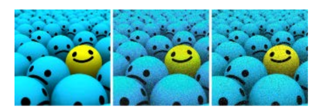


- Crop: random sample a section



- Translate: moving the image along the X or Y direction

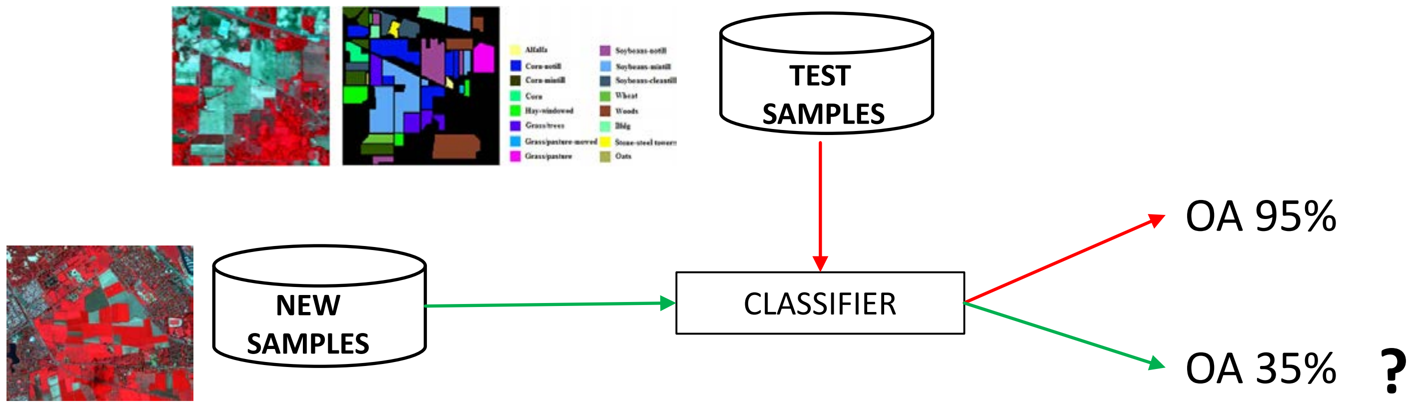
- Add noise



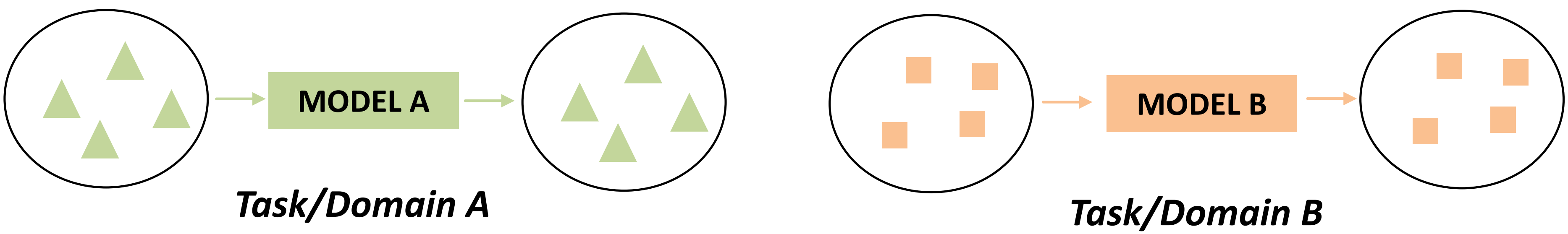
- Data augmentation is more challenging for remote sensing
  - Images exist in a variety of conditions (e.g., different seasons)
  - They cannot be accounted for by the above simple methods

[31] Data Augmentation

Classification Results Assessment



Traditional classifiers are based on the assumption that training and test samples are generated from the same feature space and distribution



Remote sensing data usually present heterogeneous feature spaces and distributions due to differences in acquisition or changes in the nature of the object observed.

Most of the statistical models are likely to fail the prediction of new samples

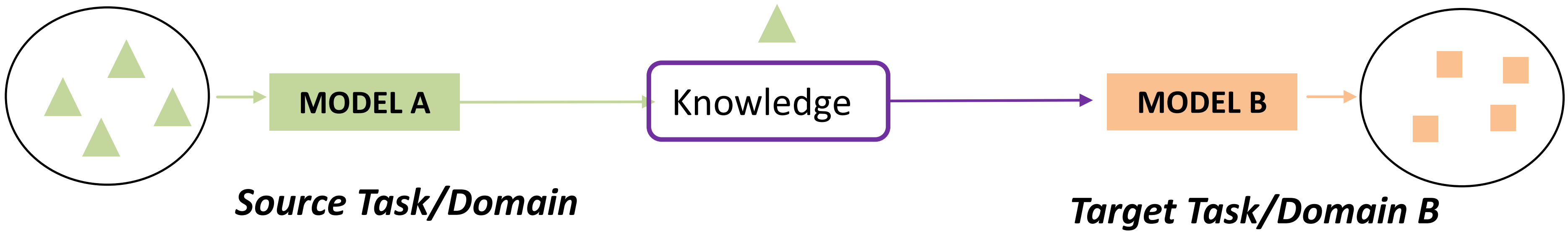
Transfer of Knowledge

- *Direct solution*: rebuild from scratch the predictive model using new training samples

- However it is preferable to reduce the need for and effort in recollecting new samples

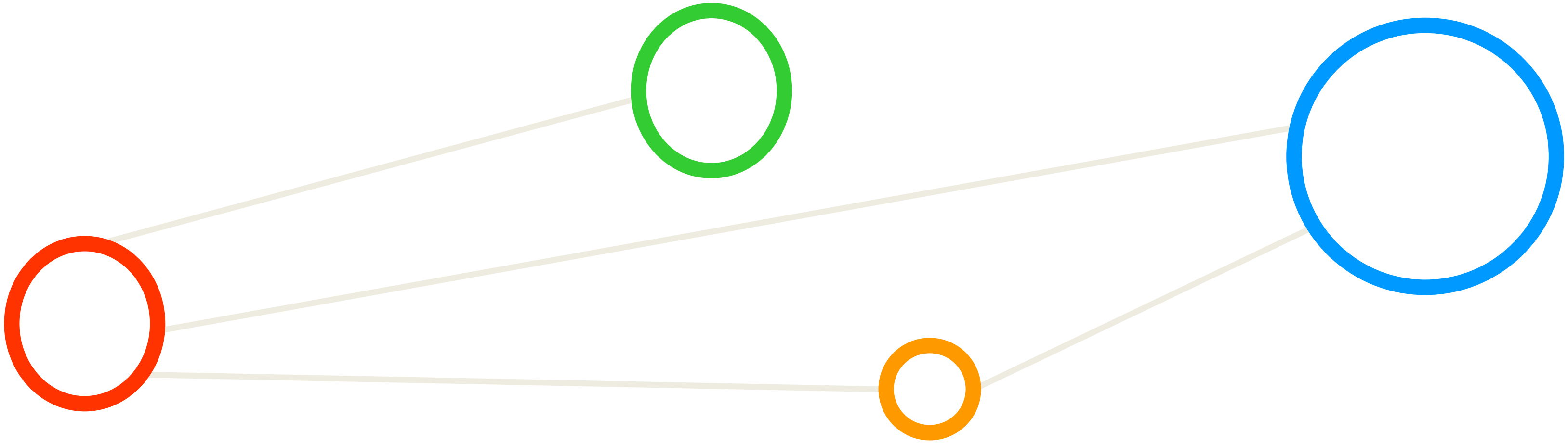
- *Other solutions*: **transfer learning**, **domain adaptation** and **active learning** approaches

- Exploit the knowledge acquired by the available reference samples for classifying new images acquired over different geographical locations at diverse times with different sensors



[33] Domain Adaptation

Practical 3: Submit two Training Jobs



Edit the Script and Submit the First Training Job

1. **Get a copy of the script /homea/hpclab/train001/script/submit_train_resnet50_fcn.sh**
2. **Modify the highlighted parts**

```
#!/bin/bash -x
#SBATCH--nodes=1
#SBATCH--ntasks=1
#SBATCH--output=train_resnet50_fcn_out.%j
#SBATCH--error=train_resnet50_fcn_err.%j
#SBATCH--time=01:00:00
#SBATCH--mail-user=g.cavallaro@fz-juelich.de
#SBATCH--mail-type=ALL
#SBATCH--job-name=train_resnet50_fcn

#SBATCH--partition=gpus
#SBATCH --gres=gpu:1

#SBATCH--reservation=deep_learning

### location executable
RESNET50_FCN=/homea/hpclab/train002/semseg/resnet50-fcn/train_resnet50_fcn.py

module restore dl_tutorial

### submit
python $RESNET50_FCN /homea/hpclab/train002/semseg/data/
/homea/hpclab/train002/semseg/models/resnet50_fcn_weights.hdf5 True False
```

Location of training and validation sets

Create a new folder where the trained model will be saved

3. **Submit: $ sbatch submit_train_resnet50_fcn.sh**

Edit the Script and Submit the Second Training Job

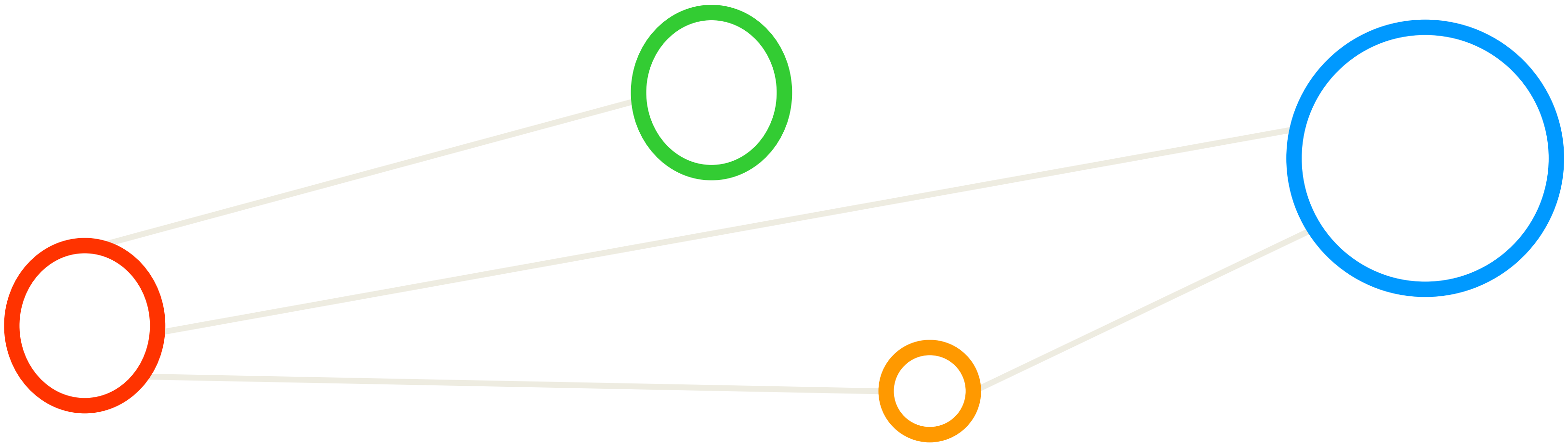
1. **Get a copy of the script**
   /homea/hpclab/train001/script/submit_train_resnet50_fcn_pretrained.sh
2. **Modify the highlighted parts**

```
#!/bin/bash -x
#SBATCH--nodes=1
#SBATCH--ntasks=1
#SBATCH--output=train_resnet50_fcn_out.%j
#SBATCH--error=train_resnet50_fcn_err.%j
#SBATCH--time=01:00:00
#SBATCH--mail-user=g.cavallaro@fz-juelich.de
#SBATCH--mail-type=ALL
#SBATCH--job-name=train_resnet50_fcn

#SBATCH--partition=gpus
#SBATCH --gres=gpu:1

#SBATCH--reservation=deep_learning

### location executable
RESNET50_FCN=/homea/hpclab/train002/semseg/resnet50-fcn/train_resnet50_fcn.py

module restore dl_tutorial

### submit
python $RESNET50_FCN /homea/hpclab/train002/semseg/data/
/homea/hpclab/train002/semseg/models/resnet50_fcn_weights.hdf5 True True
```

Location of training and validation sets

Create a new folder where the trained model will be saved

Deep Learning and Shallow Learning



How to Extract Spatial Features?

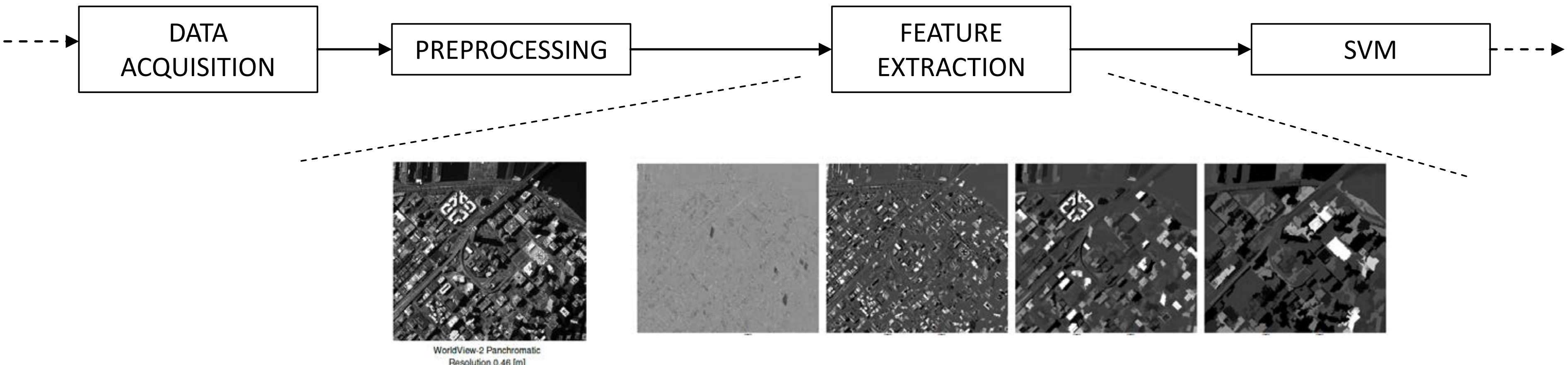
- Very High Spatial Resolution images: huge amount of details

WorldView-2 Panchromatic - Resolution 0.46 [m]

- Sub-metric resolution
- Allows for accurate analysis
- Objects with different scales and shapes

Traditional RS Pipeline

- Traditional feature extraction methods involve approaches to extract information
  - Based on spatial, spectral, textural, morphological content, etc.
- The features are designed for a specific task
  - e.g. Enhancing the spatial information (Self-Dual Attribute Profiles - SDAPs)



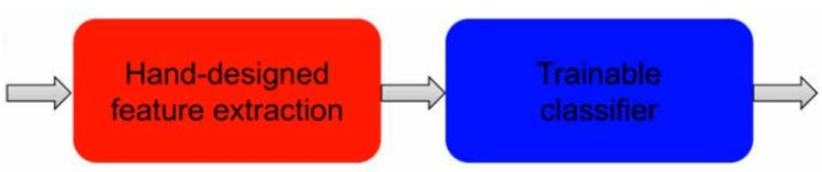
[34] G. Cavallaro

Representative: they contain salient structures of the input image

Non-redundant: same objects are present only in one or few levels of the SDAP

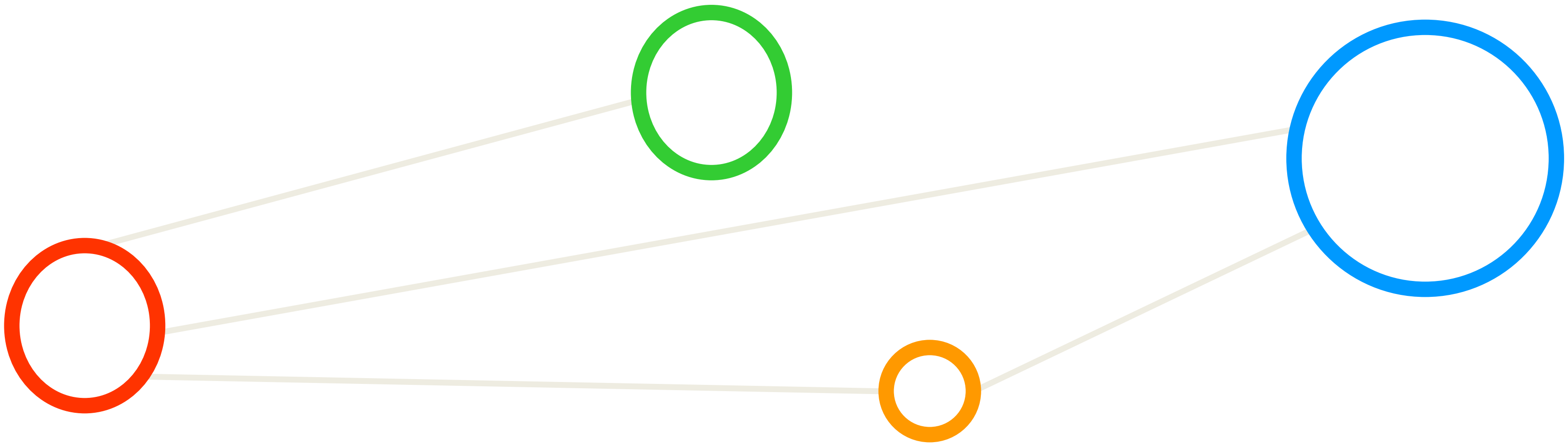
Deep Learning and Shallow Learning

- Shallow learning: learning networks that usually have at most one to two layers
  - E.G. Support Vector Machines (SVMs)
- They compute linear or nonlinear functions of the data (often hand-designed features)

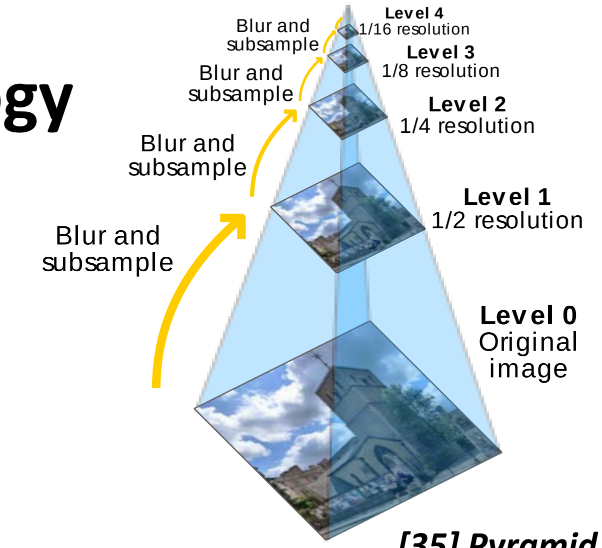


- DL means a deeper network with many layers of non-linear transformations
  - No universally accepted definition of how many layers constitute a “deep” learner
  - Typical networks are typically at least four or five layers deep.

Hiercarchial Features of CNNs

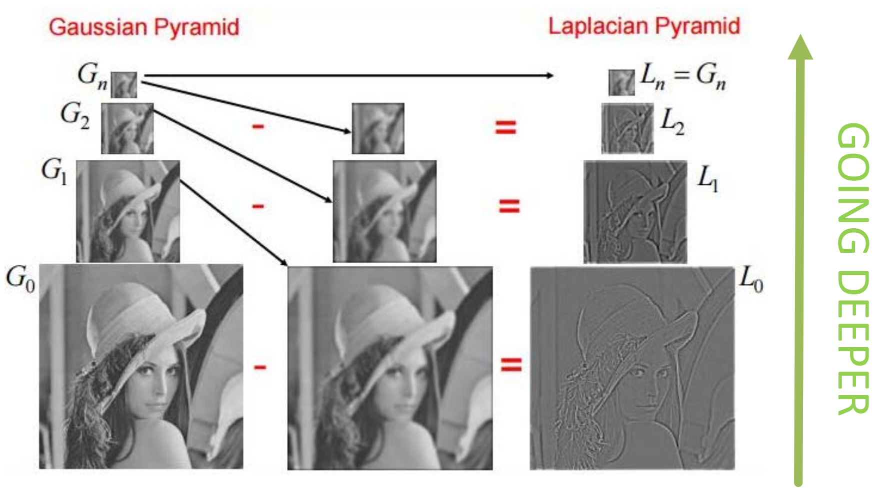


Pyramid and CNNs Analogy

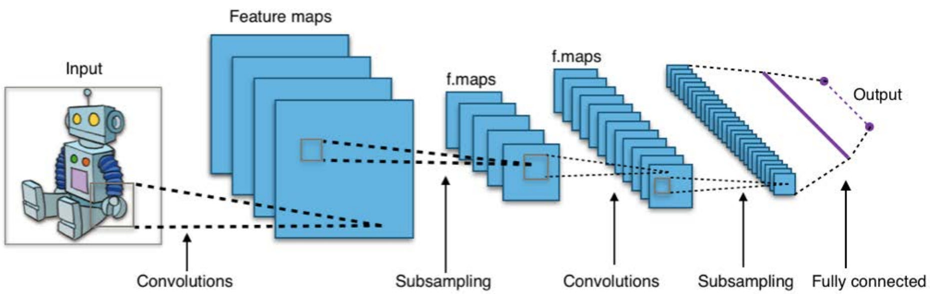


[35] Pyramid

- Multi-scale signal representation
- An image is subject to repeated smoothing and subsampling
- Used for doing tasks at multiple scales
- **Gaussian pyramid**
  - Subsequent images are weighted and scaled down
  - Each pixel is local average of the neighborhood on a lower level
- **Laplacian pyramid**
  - It saves the difference image of the blurred versions between each levels.

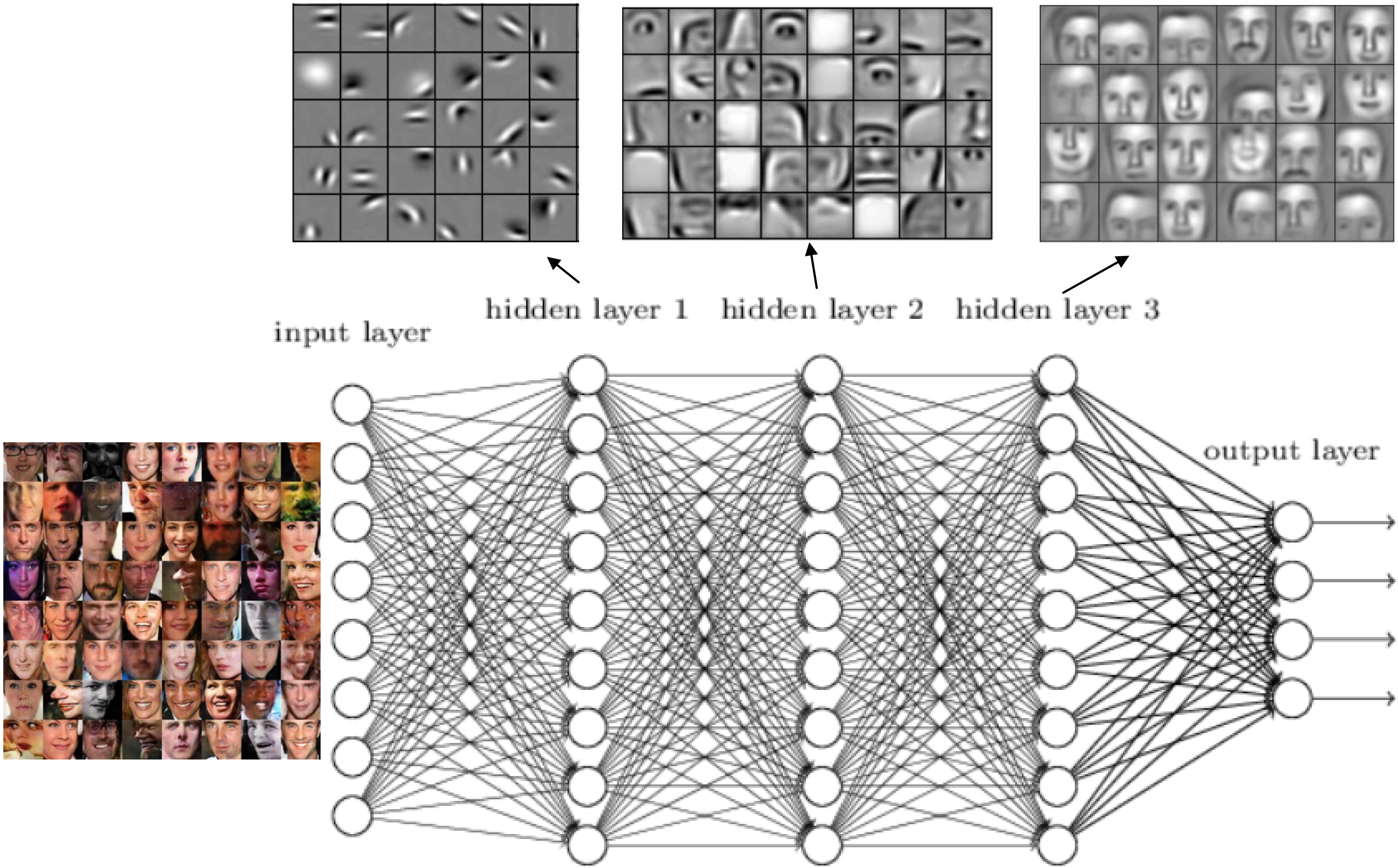


[36] The Laplacian Pyramid



[37] Typical CNN

Deep Networks Learn Hierarchical Feature Representations



[38] H. Lee et al.

Lecture Bibliography (1)

- [1] CORONA: American's First Satellite Program: first photograph.
  Online: https://www.oneonta.edu/faculty/baumanpr/geosat2/RS%20History%20II/RS-History-Part-2.html
- [2] The Earth-Atmosphere Energy Balance
  Online: http://theatmosphere.pbworks.com/w/page/27058542/The%20Earth-Atmosphere%20Energy%20Balance
- [3] Active-and-passive-remote-sensing
  Online: http://grindgis.com/remote-sensing/active-and-passive-remote-sensing
- [4] Airborne Hyperspectral Remote Sensing Systems
  Online: http://uregina.ca/piwowarj/Satellites/Hyperspectral.html
- [5] Tsunami Damage, Gleebruk, Indonesia January 7, 2005
  Online: https://earthobservatory.nasa.gov/IOTD/view.php?id=5145
- [6] Earth Observation Mission Sentinel 2
  Online: https://sentinel.esa.int/web/sentinel/missions/sentinel-2
- [7] CORINE land cover
  Online: https://land.copernicus.eu/pan-european/corine-land-cover
- [8] France land cover classification 2016
  Online: http://www.cesbio.ups-tlse.fr/multitemp/?p=11778
- [9] S2 prototype LC map at 20m of Africa 2016
  Online: http://2016africalandcover20m.esrin.esa.int/
- [10] Military Grid Reference System
  Online: https://en.wikipedia.org/wiki/Military_Grid_Reference_System
- [11] Population growth from 1975 - 2010 of Manila
  Online: https://manilabydaniellaandisabel.weebly.com/location-and-characteristics.html
- [12] Deforestation in Bolivia from 1986 to 2001
  Online: https://www.satimagingcorp.com/gallery/more-imagery/aster/aster-deforestation-bolivia/

Lecture Bibliography (2)

- [13] X. X. Zhu et al., "Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources," in IEEE Geoscience and Remote Sensing Magazine, vol. 5, no. 4, pp. 8-36, Dec. 2017.
- [14] J. E. Ball, D. T. Anderson, and C. S. Chan, “A Comprehensive Survey of Deep Learning in Remote Sensing: Theories, Tools and Challenges for the Community,” in Proceedings of the SPIE Journal of Applied Remote Sensing, 2017.
- [15] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in Proc. ACM SIGSPATIAL Int. Conf. Adv. Geogr. Inform. Syst., 2010, pp. 270-279.
- [16] J. Li, J. M. Bioucas-Dias, and A. Plaza, “Spectral–spatial classification of hyperspectral data using loopy belief propagation and active learning,” IEEE Trans. Geosci. Remote Sens., vol. 51, no. 2, pp. 844-856, 2013.
- [17] Q. Zou, L. Ni, T. Zhang, and Q. Wang, “Deep learning based feature selection for remote sensing scene classification,” IEEE Geosci. Remote Sens. Lett., vol. 12, no. 11, pp. 2321-2325, 2015.
- [18] S. Basu, S. Ganguly, S. Mukhopadhyay, R. DiBiano, M. Karki, and R. Nemani. Deepsat: a learning framework for satellite imagery. In Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, page 37. ACM, 2015
- [19] B. Zhao, Y. Zhong, G.-S. Xia, and L. Zhang, “Dirichlet-derived multiple topic scene classification model for high spatial resolution remote sensing imagery,” IEEE Trans. Geosci. Remote Sens., vol. 54, no. 4, pp. 2108-2123, 2016
- [20] L. Zhao, P. Tang, and L. Huo, “Feature significance-based multibag-of-visual-words model for remote sensing image scene classification,” J. Appl. Remote Sens., vol. 10, no. 3, pp. 035004-035004, 2016.
- [21] O. A. Penatti, K. Nogueira, and J. A. dos Santos, "Do deep features generalize from everyday objects to remote sensing and aerial scenes domains?," in Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit. Workshops, 2015, pp. 44-51.
- [22] G. Cheng, J. Han and X. Lu, "Remote Sensing Image Scene Classification: Benchmark and State of the Art," in Proceedings of the IEEE, vol. 105, no. 10, pp. 1865-1883, Oct. 2017.
- [23] G.-S. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, and L. Zhang. Aid: A benchmark dataset for performance evaluation of aerial scene classication. arXiv preprint arXiv:1608.05167, 2016.
- [24] P. Helber, B. Bischke, A. Dengel, D. Borth. 2017. “EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification”, ArXiv
- [25] ImageNet project

  Online: http://image-net.org/about-overview

Lecture Bibliography (3)

- [26] 2D Semantic Labeling Contest
  Online: http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html
- [27] Normalized Difference Vegetation Index (NDVI)
  Online: http://www.agasyst.com/portals/NDVI.html
- [28] NDVI & Classification
  Online: https://lholmesmaps.wordpress.com/my-work-2/environmental-studies-421-gis-iv-advanced-gis-applications/2-2/
- [29] G. Cavallaro, N. Falco, M. Dalla Mura and J. A. Benediktsson, "Automatic Attribute Profiles," in *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 1859-1872, April 2017.
- [30] Indian Pines dataset: 220 Band AVIRIS Hyperspectral Image
  Online: https://purr.purdue.edu/publications/1947/1
- [31] Data Augmentation - How to use Deep Learning when you have Limited Data
  Online: https://medium.com/nanonets/how-to-use-deep-learning-when-you-have-limited-data-part-2-data-augmentation-c26971dc8ced
- [32] Invariance property
  Online: https://i.stack.imgur.com/iY5n5.png
- [33] D. Tuia, C. Persello and L. Bruzzone, "Domain Adaptation for the Classification of Remote Sensing Data: An Overview of Recent Advances," in IEEE Geoscience and Remote Sensing Magazine, vol. 4, no. 2, pp. 41-57, June 2016.
- [34] G. Cavallaro, N. Falco, M. Dalla Mura and J. A. Benediktsson, "Automatic Attribute Profiles," in IEEE Transactions on Image Processing, vol. 26, no. 4, pp. 1859-1872, April 2017.
- [35] Pyramid: image processing
  Online: https://en.wikipedia.org/wiki/Pyramid_(image_processing)#Gaussian_pyramid
- [36] The Laplacian Pyramid
  Online: https://hypjudy.github.io/2017/05/10/panorama-image-stitching/
- [37] Typical CNN
  Online: https://commons.wikimedia.org/wiki/File:Typical_cnn.png
- [38] H. Lee, R. Grosse, R. Ranganath and A. Y. Ng, "Unsupervised Learning of Hierarchical Representations with Convolutional Deep Belief Networks", in Commun. ACM, vol. 54, no. 10, pp. 95-103, 2011.