WWU MÜNSTER

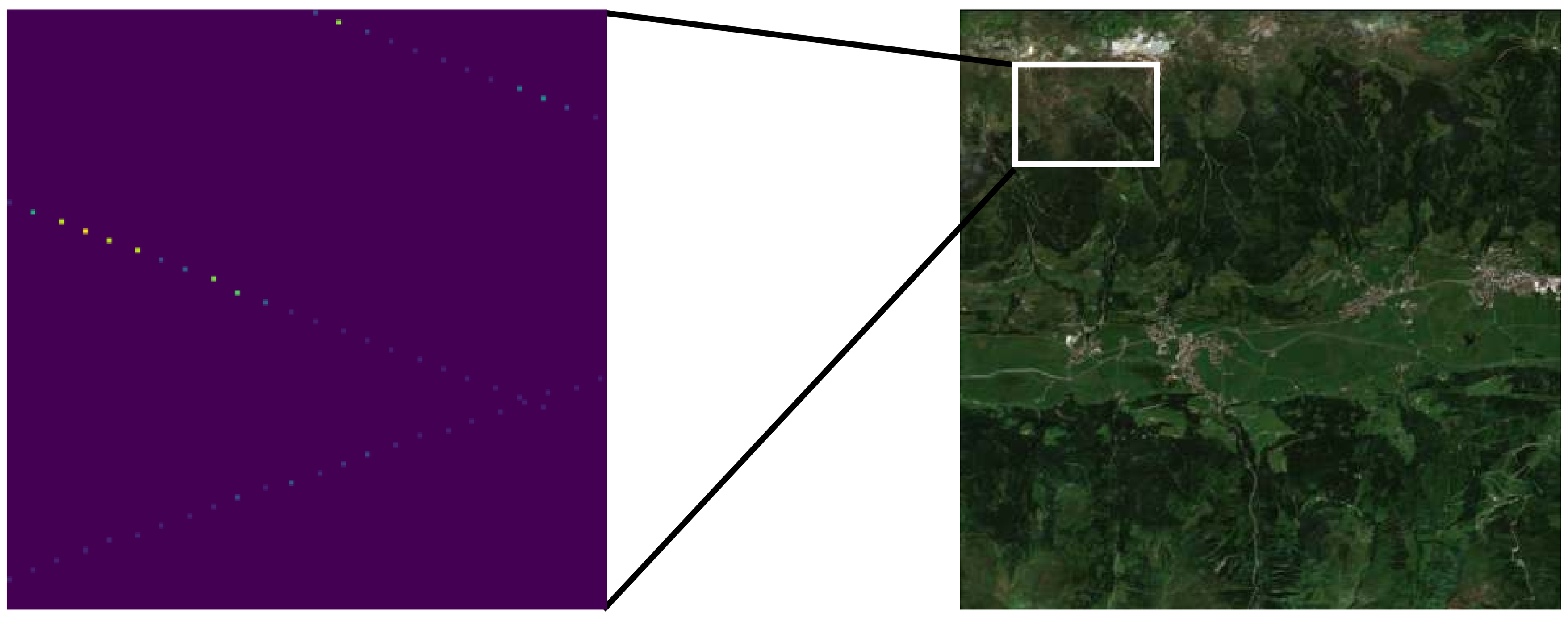
DA2 Case Study: Forest Height

Data Exploration

Reflection Values
- Images have a reflection value and not image values from 0-256.
- Clouds have a much higher reflection value.
- If used raw, the models might not be able to detect the subtle differences between the lower values.

Imbalanced Dataset
- Tree height: many smaller trees and few larger ones.
- Models tend to predict around the median tree height.

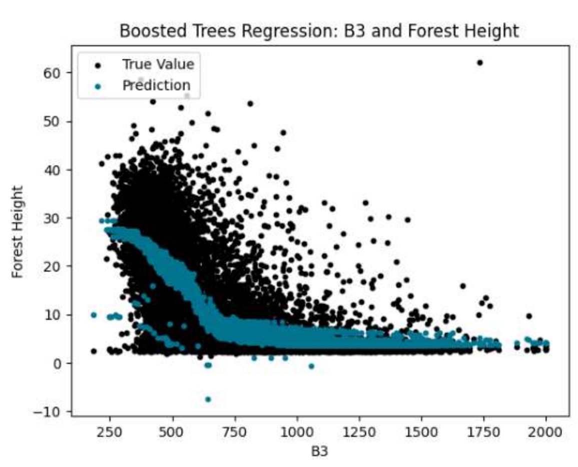
Labels
- Sparse labels (38.863 in all 40 1024x1024px images).
- Insufficient for complete training.



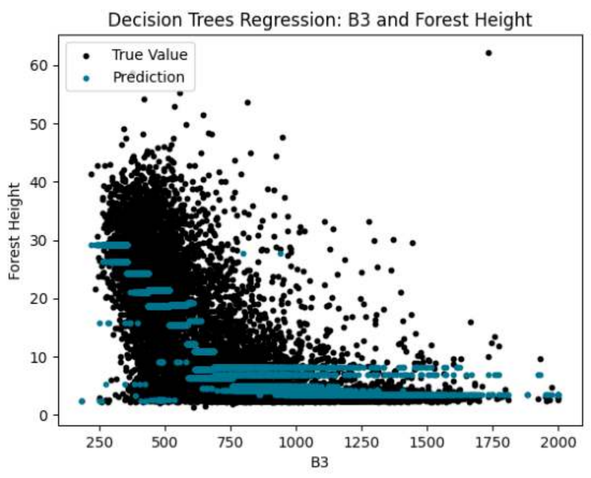
Model Selection and Training

Boosted Trees, Decision Trees, Random Forest
- Grid and Random Search is used to identify best hyperparameters and yield the best possible models.
- Explored performance of models with additional features as Vegetation Indices NDVI, EVI, SAVI, IRECI, S2REP.
- B3 (green color channel) is the most significant feature.
- All three traditional regression models understand the correlation between this feature and the forest height:

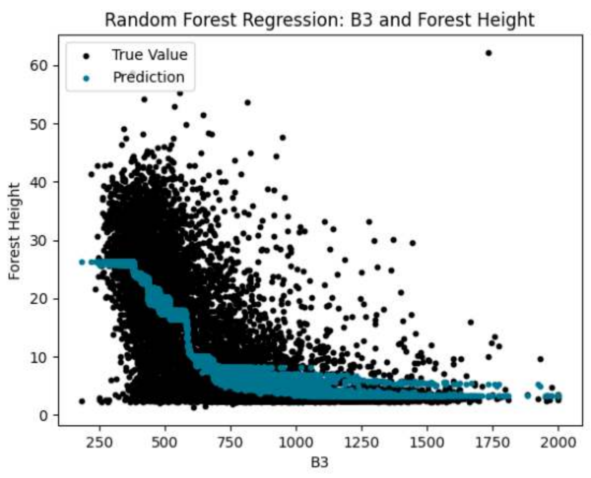
Boosted Trees
- Not brave enough to predict too far off mean.
- MAE = 4.89



Decision Trees
- Predicts whole spectrum of tree heights.
- MAE = 4.79



Random Forest
- Predicts whole spectrum of tree heights.
- MAE = 4.96



Neural Network, Convolutional Neural Network
- Preprocessing results in 38.531 patches.
- Sliding window: Computationally challenging.

Neural Network
- MAE = 4.43, MAPE = 0.49

```
modelNn = Sequential() #build neural network
modelNn.add(Dense(128, input_shape=(10,), kernel_initializer='normal', activation='relu'))
modelNn.add(Dense(256, kernel_initializer='normal', activation='relu'))
modelNn.add(Dense(256, kernel_initializer='normal', activation='relu'))
modelNn.add(Dense(128, kernel_initializer='normal', activation='relu'))
modelNn.add(Dropout(0.4))
modelNn.add(Dense(1, kernel_initializer='normal', activation='linear'))
```

Convolutional Neural Network
- MAE = 4.15, MAPE = 0.47

```
modelCnn = Sequential() #buid cnn
modelCnn.add(InputLayer(input_shape = (10, 5, 5)))
modelCnn.add(Conv2D(filters=128, kernel_size= (3,3), strides= 1 , padding = "same", activation='relu' ))
modelCnn.add(Conv2D(filters=256, kernel_size= (3,3), strides= 1 , padding = "valid", activation='relu' ))
modelCnn.add(MaxPool2D(pool_size = (2,2)))
modelCnn.add(Flatten())
modelCnn.add(Dense(512, activation='relu'))
modelCnn.add(Dense(128, activation='relu'))
modelCnn.add(Dropout(0.4))
modelCnn.add(Dense(1, activation='linear'))
```

Data Preparation

Group

Reflection Values
- Not relevant for us → ceiling irrelevant values off at 2000.

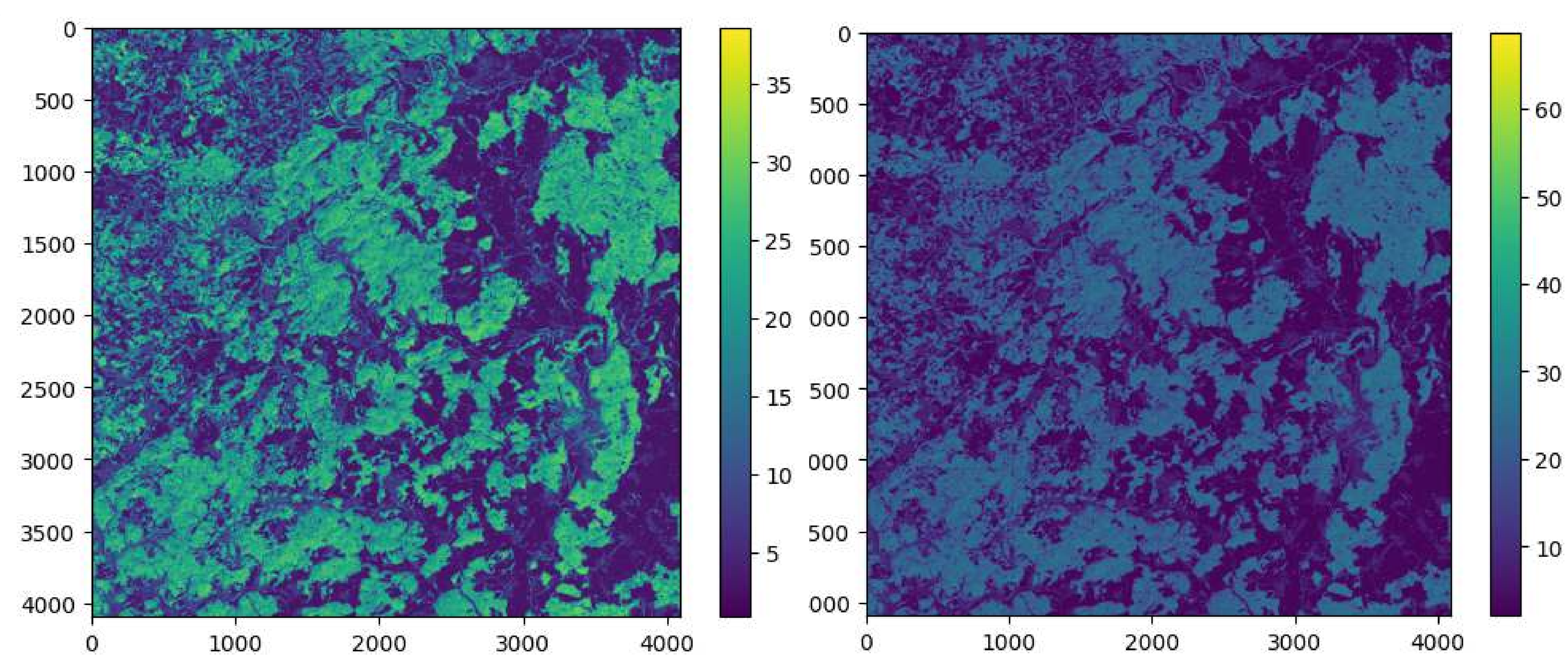
Labels
- Extracting all the labels and corresponding features.
- Training multiple models on given labels.
- Comparing approaches (observing what values do the models predict).

Cropping
- Cropping (10, 5, 5) feature patches where the label is in the middle → used for training of CNN.
- Computing the mean of every feature in one patch → used for training NN.

Evaluation

Comparing Convolutional Neural Network to Neural Network:
- Both models can identify canopy properly and can distinguish between canopy and other landscape.
- Both models have a clear lower limit (both ~ 2.8m).
- CNN can predict higher values (CNN: 68m, NN: 38m).
- CNN has higher median.

Neural Network | **Convolutional Neural Network**



Percentiles	Neural Network	Convolutional Neural Network
0.01	2.7815 m	2.8542 m
0.25	3.0449 m	2.9289 m
0.50	10.8947 m	12.2923 m
0.75	23.8498 m	23.5796 m
0.99	31.0686 m	30.5754 m

Summary & Outlook

CNN works best ! → Used for supplementing sparse labels by predicting all pixels.

Stacking
Considering all models by combining their predictions. Ensemble a model and weight each prediction based on the model's overall accuracy relative to the accuracy of the other models. Leveraging the strength of each model would potentially result in higher level of accuracy at the cost of high computational time.

Data Augmentation
Giving CNN model more data to learn from with random flipping and rotation of patches.

Equally Distributed Dataset
Grouping tree heights by a 3 m interval → down- and upsampling to represent each height group equally.

Standardization of features
Scale features from 0 to 1.

living.knowledge

Leo Giesen, Henrik Fliegel, Amna Mubashar, Konstantinos Koutris, Raquel Martinez Blasco, Michael Krämer, Neja Bevc