Tropical Forest Carbon Stock Estimation

Team 3B - 15.12.2022

Supervised Carbon Prediction from Drone Imagery and Field Data

6 Forestry Site in Ecuador

Predict → Carbon Stock

	lat	lon	diameter	year	site	carbon
0	-2.181226	-79.576630	0.000000	2016.0	Nestor Macias RGB	2.123249
1	-2.181312	-79.576412	0.000000	2016.0	Nestor Macias RGB	2.123249
2	-2.181438	-79.576322	0.000000	2016.0	Nestor Macias RGB	2.123249
3	-2.181593	-79.576154	0.000000	2016.0	Nestor Macias RGB	2.123249
4	-2.181498	-79.576179	0.000000	2016.0	Nestor Macias RGB	2.123249
5	-2.181485	-79.576235	0.000000	2016.0	Nestor Macias RGB	2.123249

Field measurements of trees

Agenda

1. **Problem statement**

2. **Results**
   a. Rerunning **Benchmark** - Tree Crown detection
   b. **Patching** site to ensure that model sees all trees
   c. Applying **gaussian** prior to account for GPS noise
   d. Estimating **tree density** to counter large carbon values

3. **Conclusion**

Running benchmark

Benchmark pipeline to rerun approach of the ReforesTree paper

Drone images of 6 forestry sites

Detect tree crowns

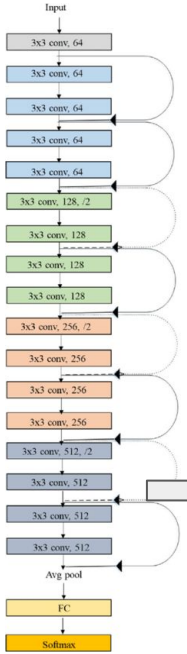
Field measurements of trees

Matching: Tree crowns + carbon labels

Carbon: 5.8016

5 x Train

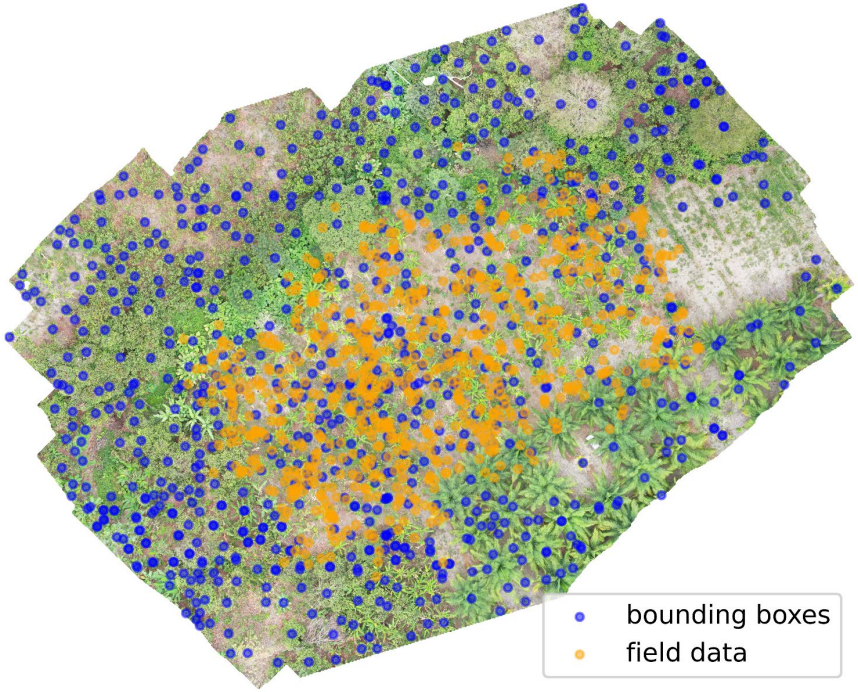
1x Test



Resnet 18

Carbon prediction (for each tile in each site)

Not all bounding boxes are considered

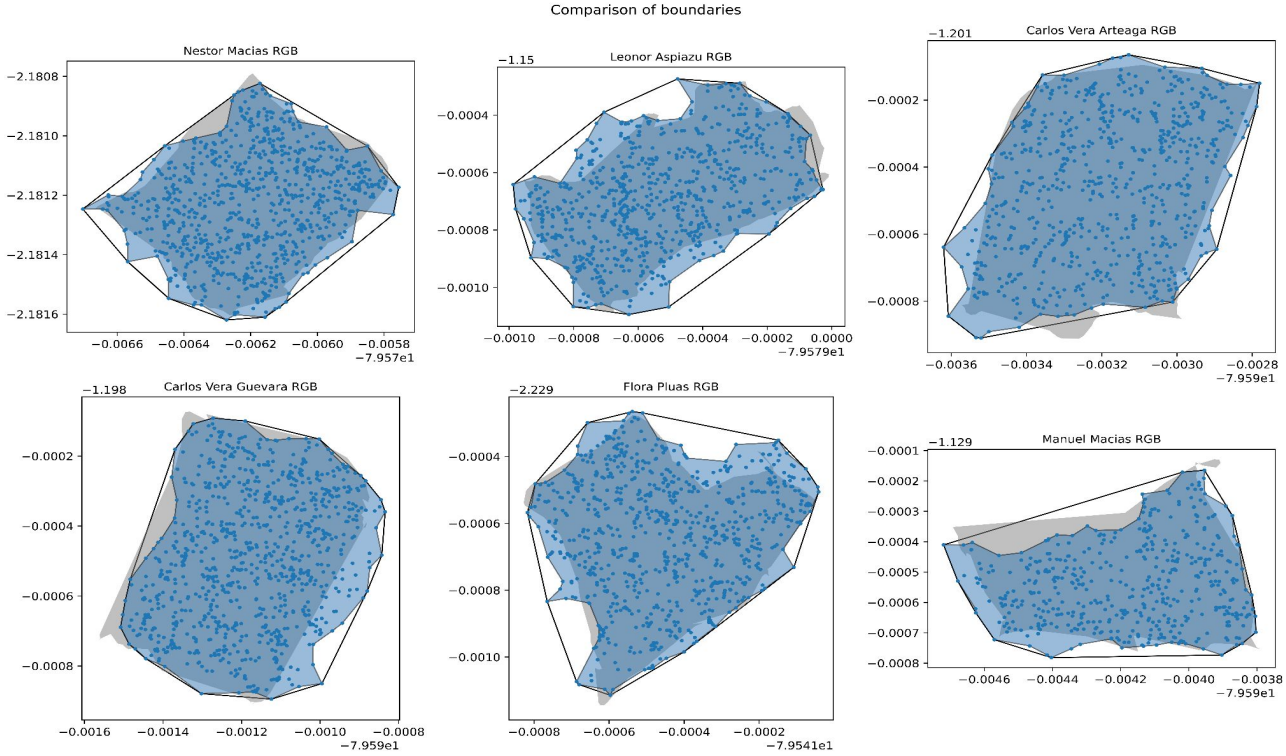


Assumption: Site were field data was measured is only a **subset** of the RGB images

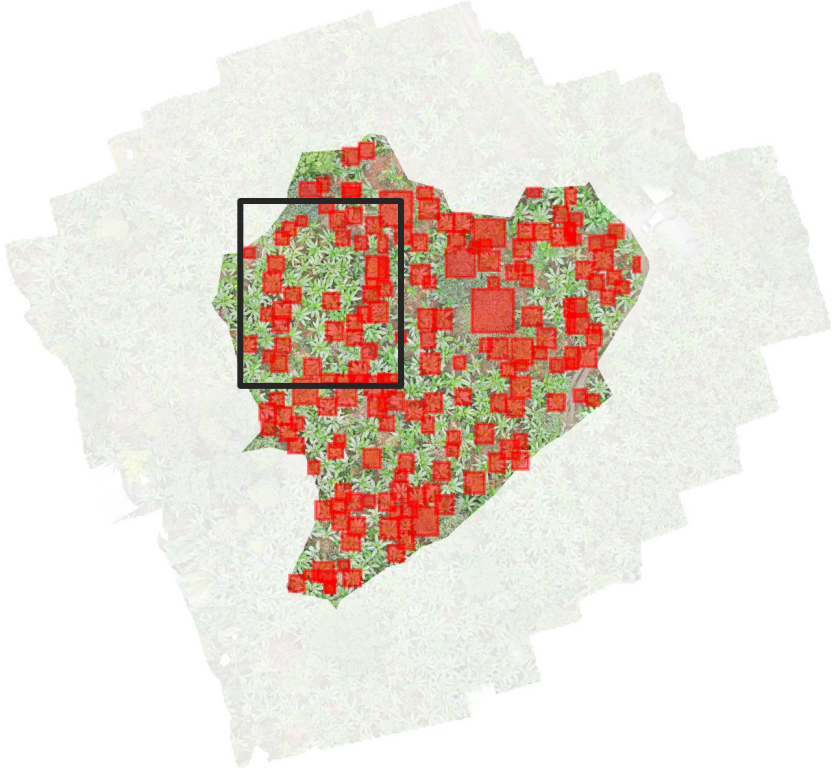
Justified by:

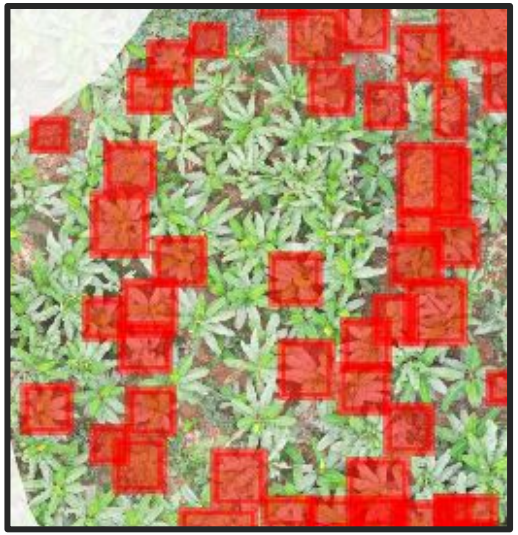
- Field data
- Visual clues
- Shape files provided
- WWF Report

We considered different shapes to mask RGB Images



Tree crown detection is not suited for dense forests





Optimal transport matching taking species into account

Bounding boxes

- **1564** bounding boxes
- classified into banana and non-banana

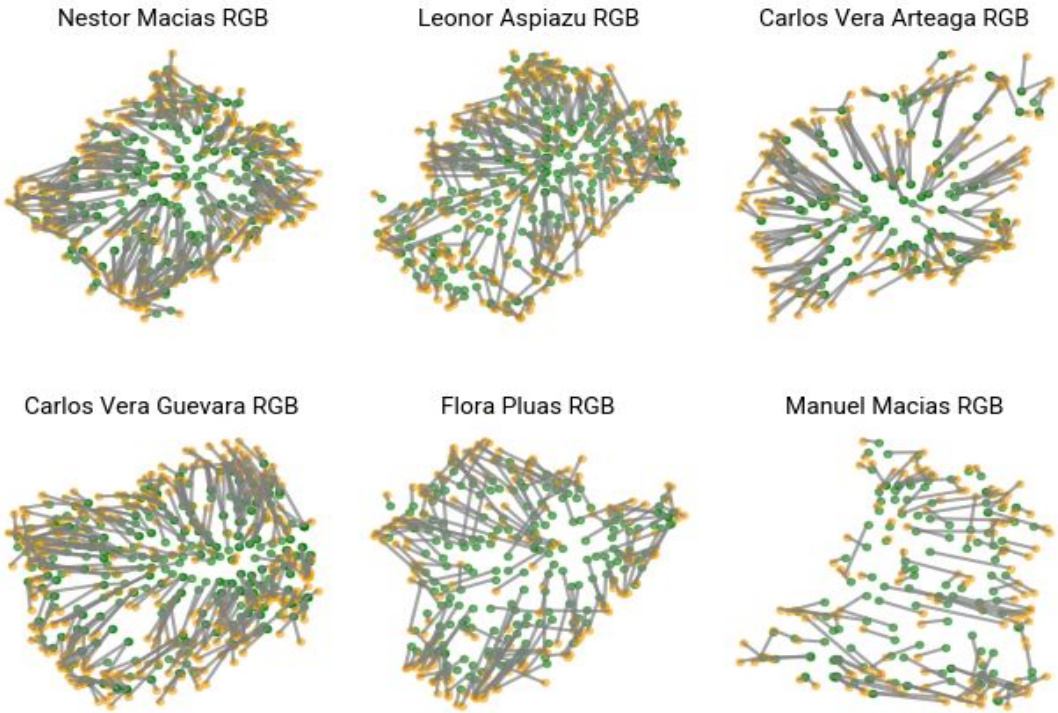
matching

Field data points

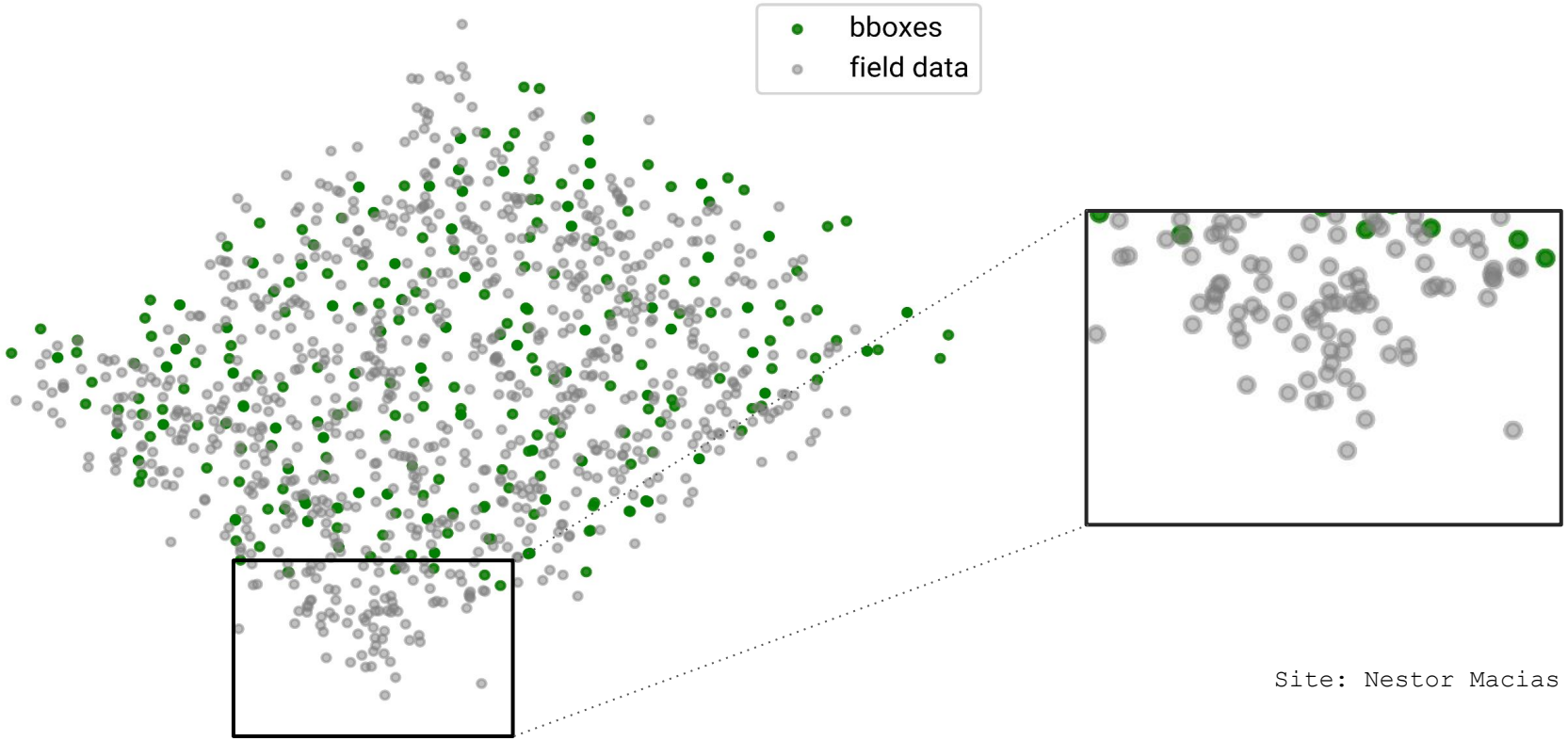
- **4663** field data points
- species label
- noisy location

Algorithm: Species respecting Sinkhorn optimal transport with greedy matching

Matching bounding boxes to field data



Field data points near boundary skew matching



Dataset

Drone images of 6 forestry sites

Detect tree crowns

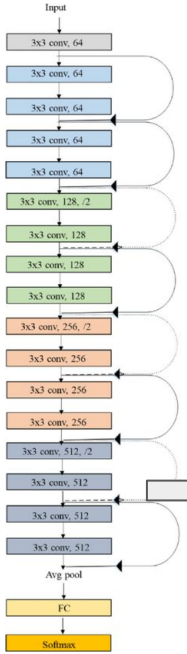
Field measurements of trees

Matching: Tree crowns + carbon labels

Carbon: 5.8016

5 x Train

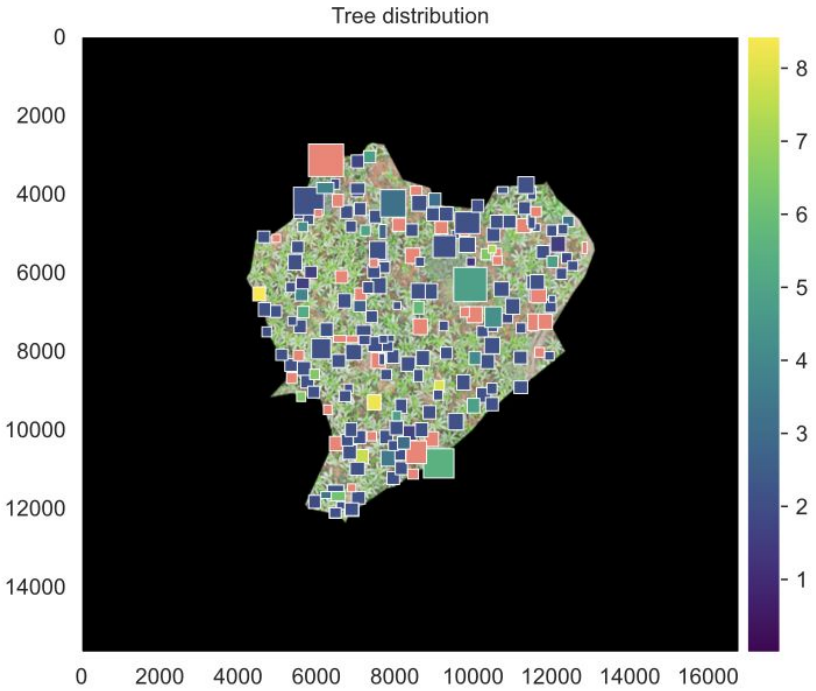
1x Test



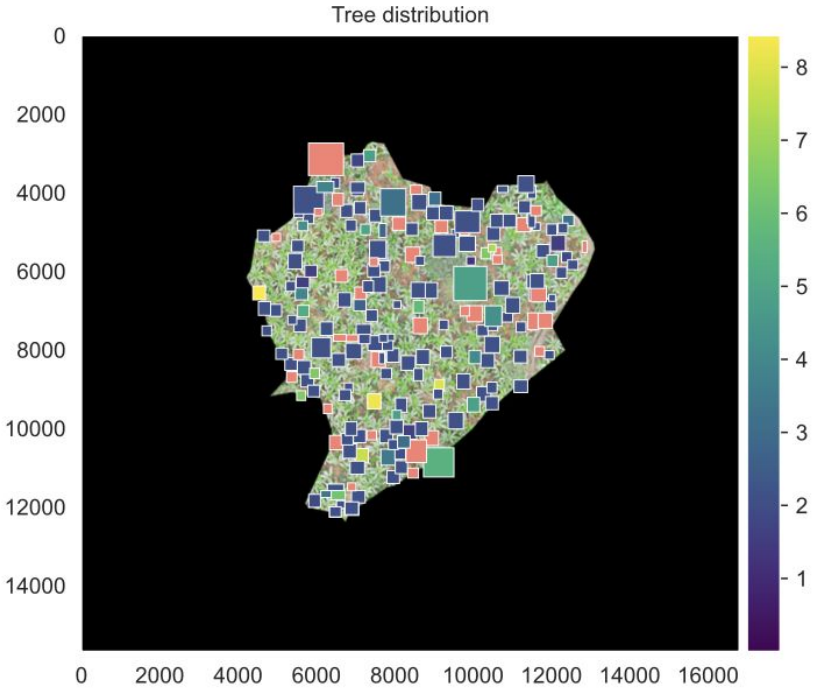
Resnet 18

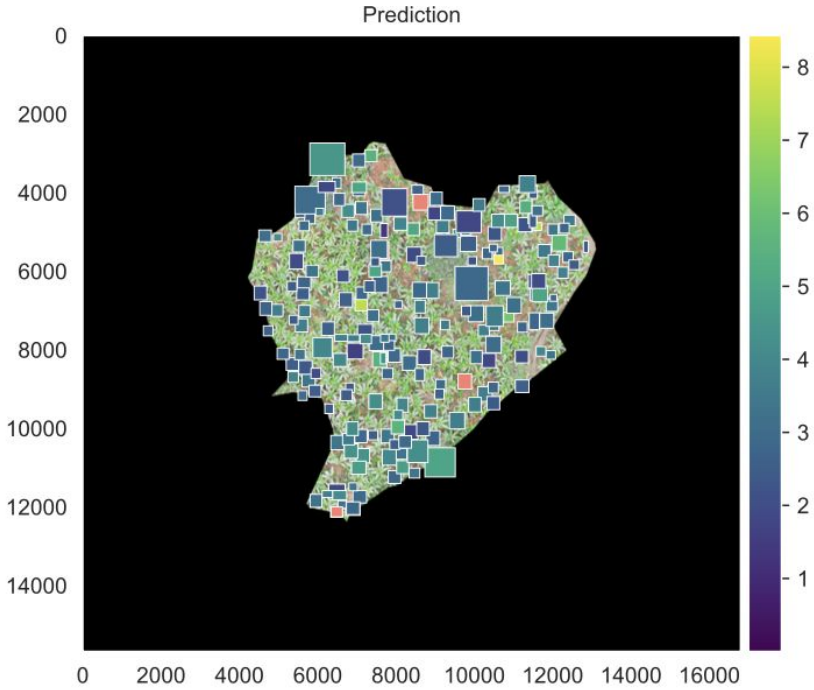
Carbon prediction (for each tile in each site)

Large trees are not properly matched



The model is unable to make meaningful predictions





Patching site to ensure that model sees all trees

Introducing patching to ensure all trees are used

Recall: After filtering out the out-of-bounds tree crowns, there are only **1564** bounding boxes across 6 sites left

Issues with previous approach

- Tree crown **detection** failed to detect majority of trees
- The **matching** between bounding boxes and field data
- Not enough training **data**

Patching

With patching

- **No** more need for tree **detection or matching**: remove two sources of approximation errors
- Control over the amount of training **data**

Summing patch-wise over the tree carbon values to get targets

Drone images of 6 forestry sites

Split the images into patches

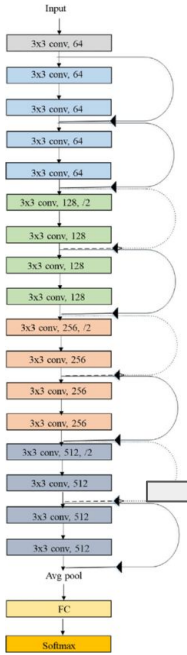
Field measurements of trees

Matching: **Patches + sum of carbon labels**

Carbon: 5.8016

5 x Train

1x Test

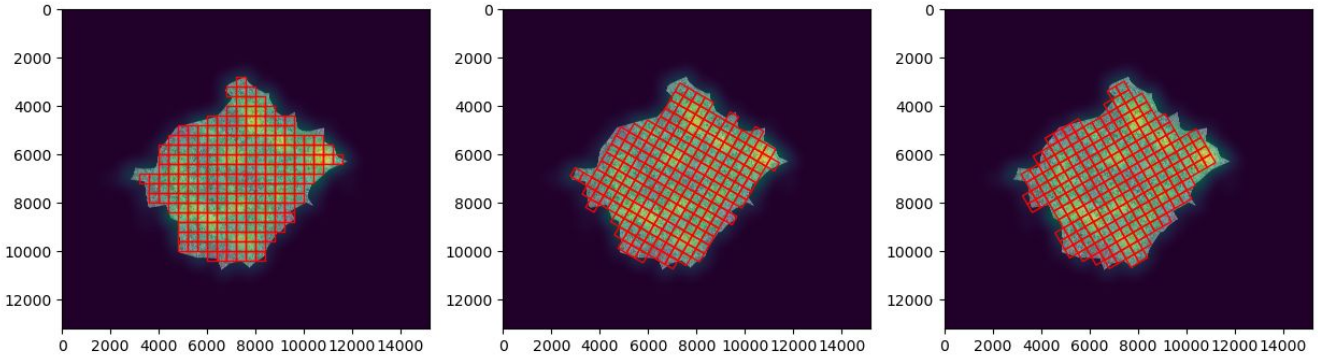


Resnet 18

Carbon prediction (for each tile in each site)

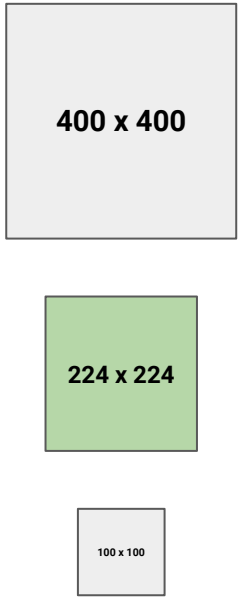
The patching approach allows for more training data

Variation of patch rotation angle

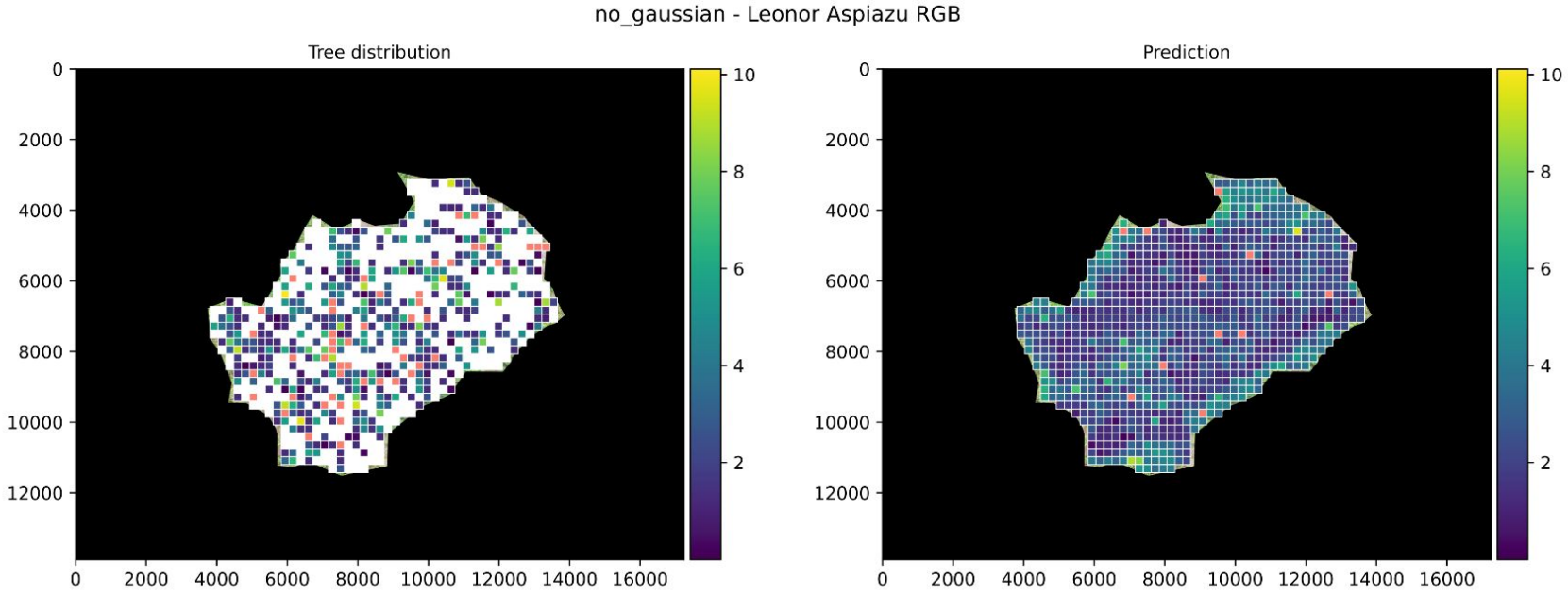


Rotation angle: 0 | Rotation angle: 30 | Rotation angle: 60

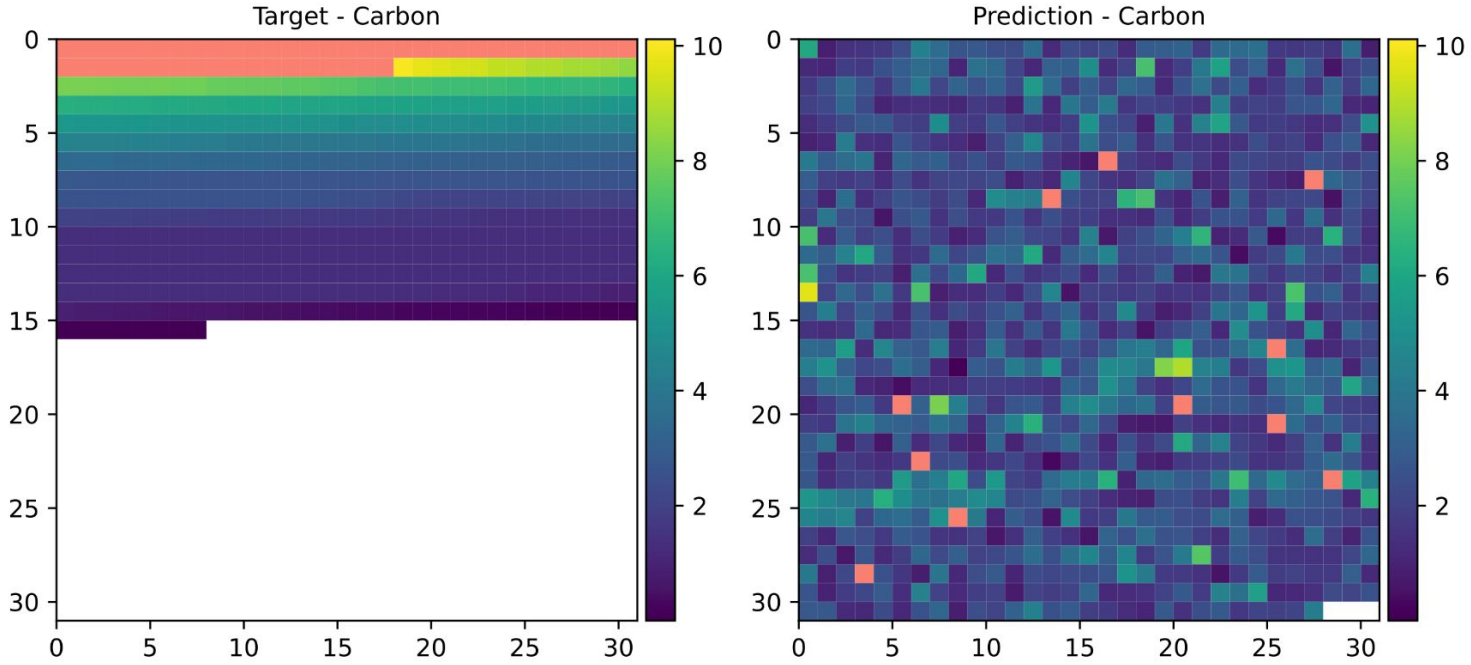
Patch Size



The training data has a very high variance and many zero targets

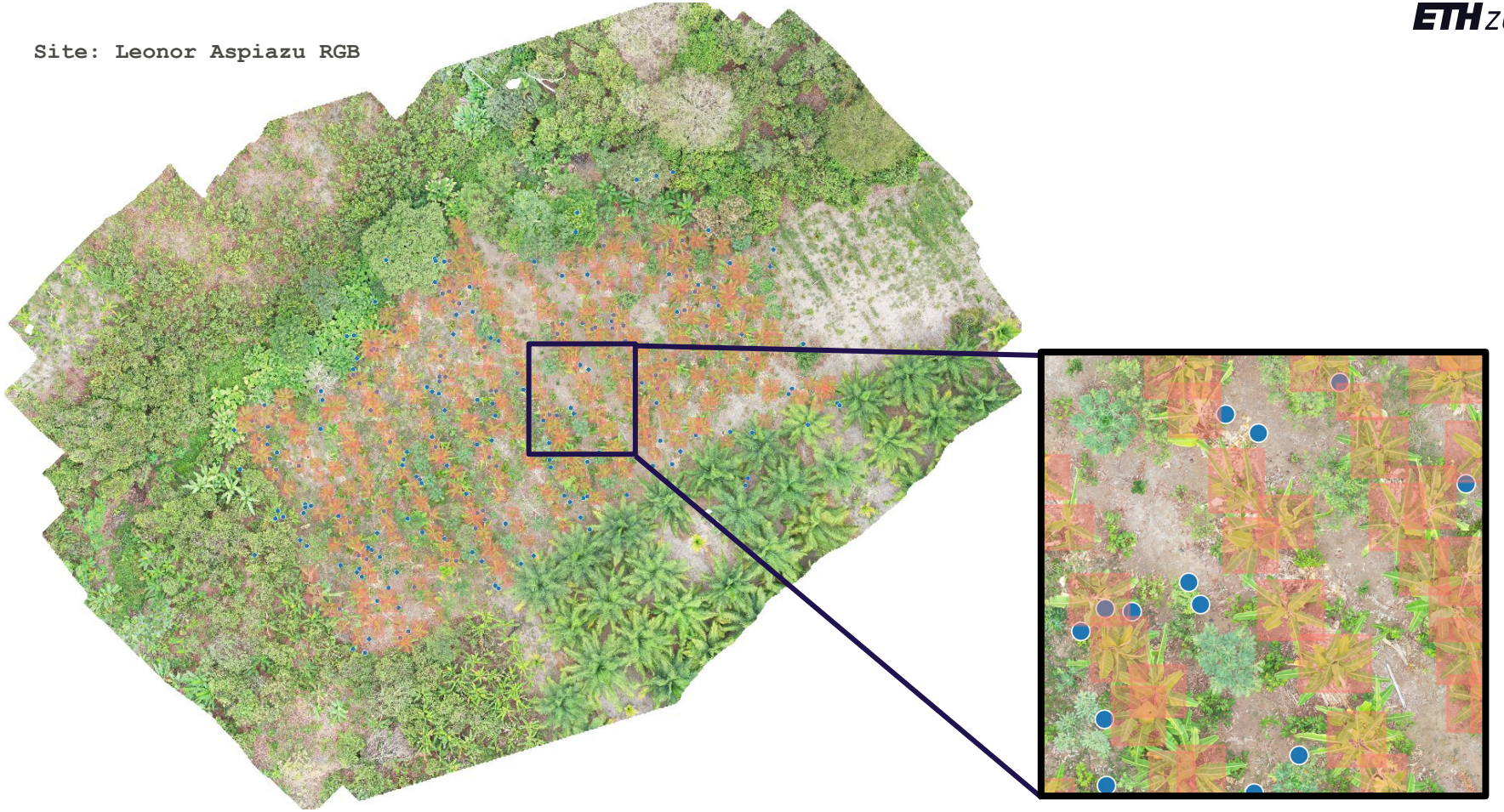


The training data has a very high variance and many zero targets

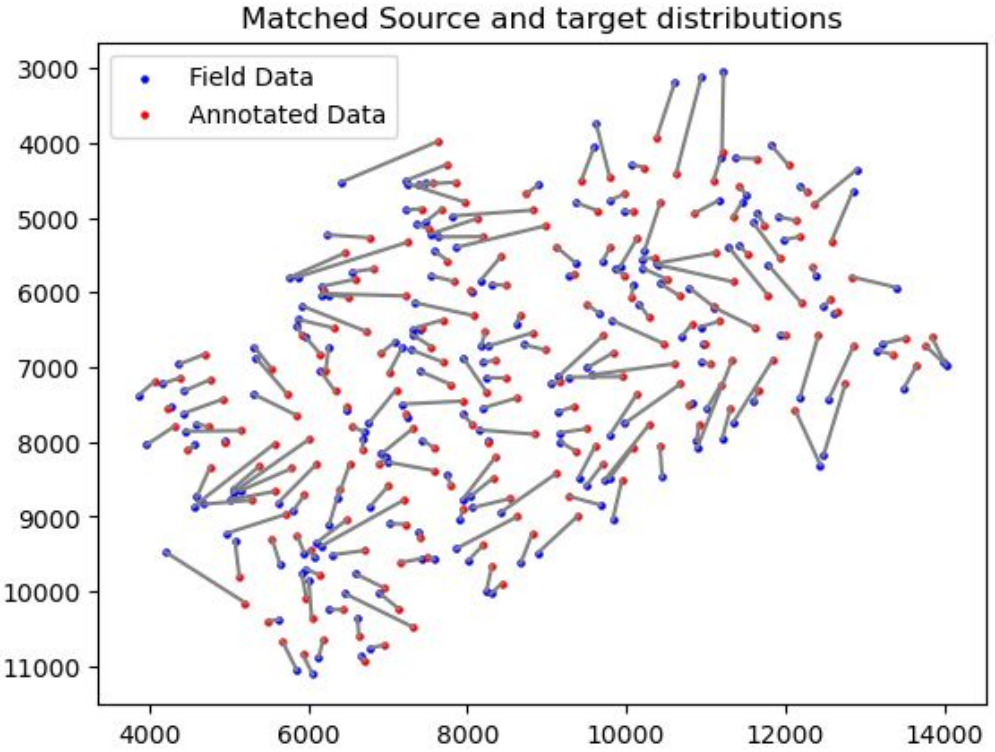


Applying gaussian prior to account for GPS noise

Site: Leonor Aspiazu RGB



Use optimal transport algorithm to match field & annotated data



Using **K-Means Clustering** to get equal number of samples: **210 -> 210**

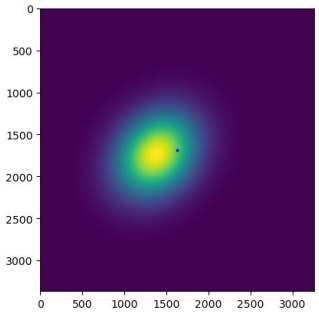
Applying **Optimal Transport algorithm** and matching sample to target with highest probability

Weierstrass: 432.89

Mean error: 433 px | **4.59 m**
Standard Deviation: 308 px | **3.27 m**

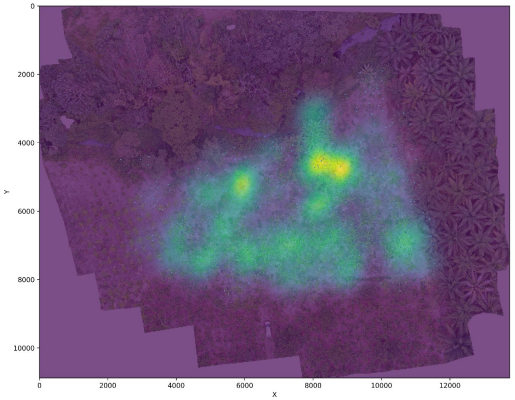
Computing a gaussian carbon distribution to account for GPS Error

Previously computed GPS error, represented as 2D Gaussian



apply to each tree

normalized with carbon value

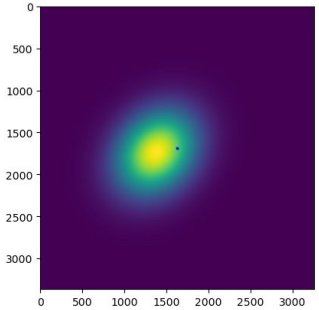


Site: Manuel Macias RGB

well balanced carbon distribution, roughly following actual distribution of trees/carbon

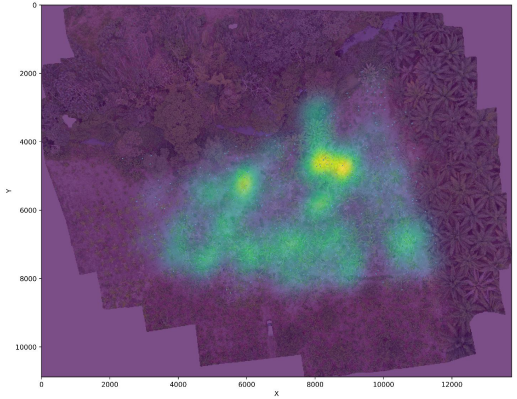
Computing a gaussian carbon distribution to account for GPS Error

Previously computed GPS error, represented as 2D Gaussian



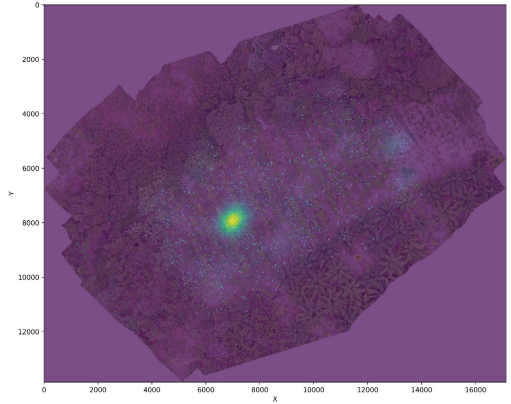
apply to each tree

normalized with carbon value



Site: Manuel Macias RGB

well balanced carbon distribution, roughly following actual distribution of trees/carbon



Site: Leonor Aspiazu RGB

One tree has especially **high carbon value**, but might not be at that **position**…

Summing patch-wise over the carbon distribution to get targets

Drone images of 6 forestry sites

Split the images into patches

Field measurements of trees

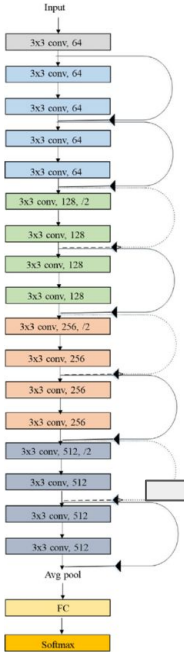
Target carbon distribution

Matching: **Patches + sum over carbon distribution**

Carbon: 5.8016

5 x Train

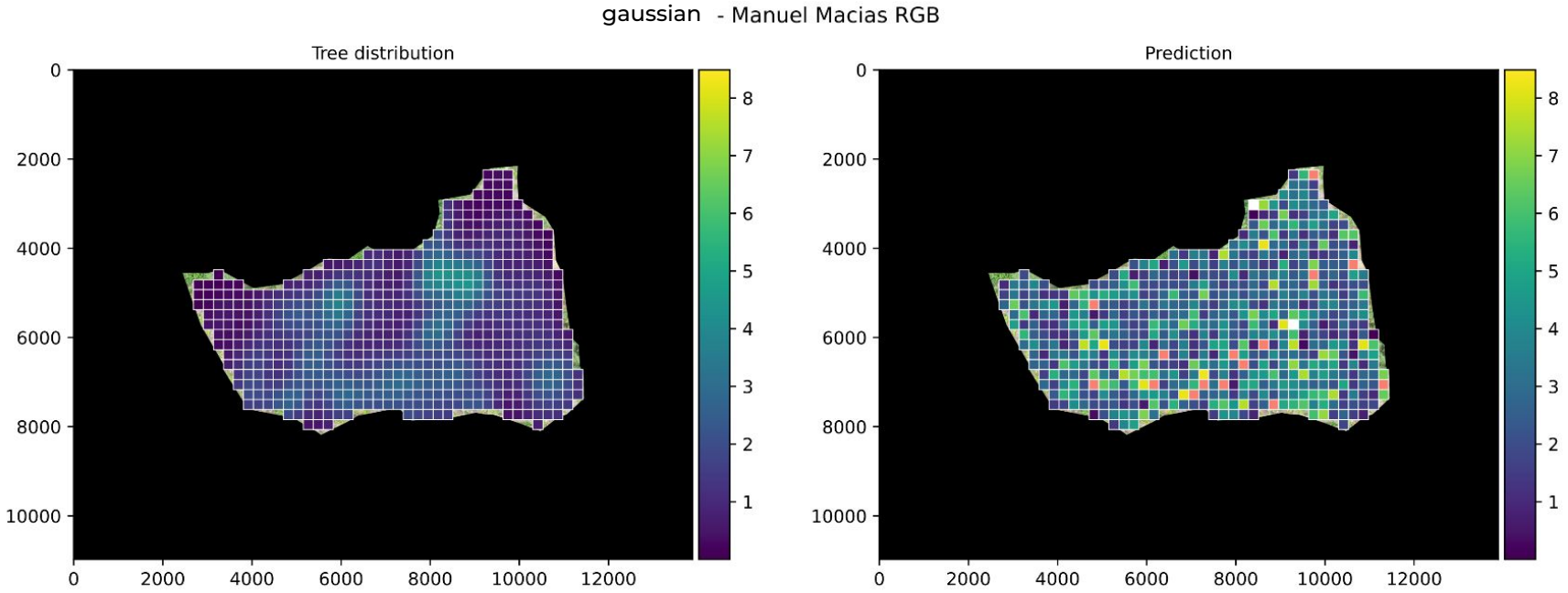
1x Test



Resnet 18

Carbon prediction (for each tile in each site)

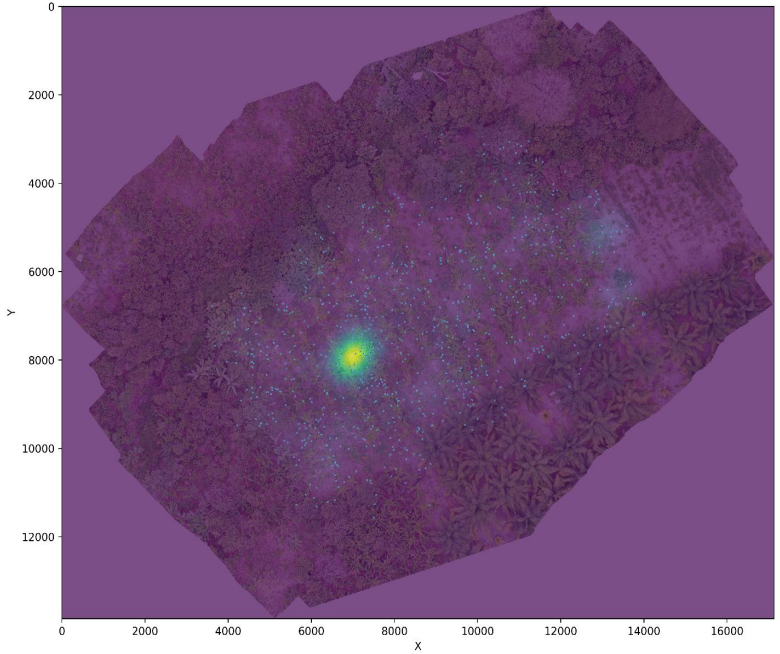
The model predicts noisily around the mean and is additionally thrown of by training data with large carbon values



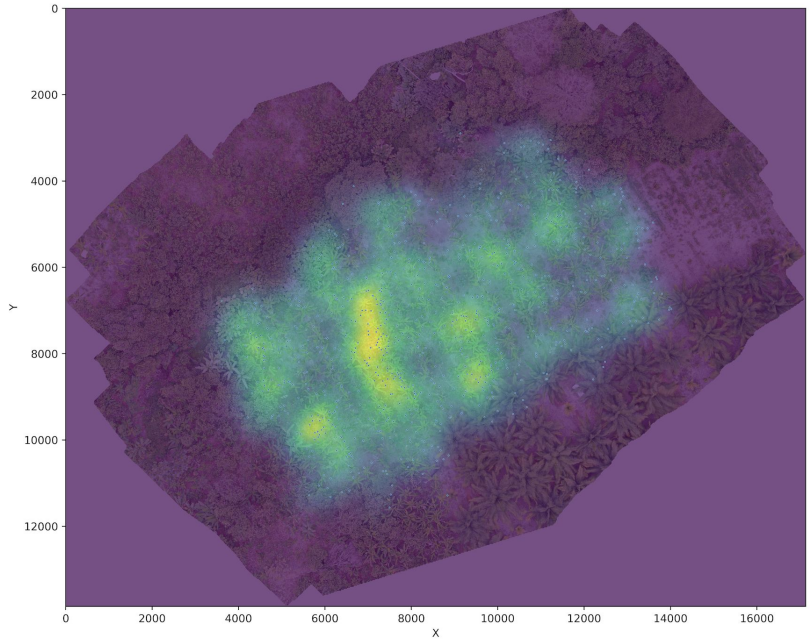
Estimating tree density to counter large carbon values

Estimating tree density instead leads to smoother distributions

Carbon distribution



Tree density distribution



Summing patch-wise over the normalized tree density distribution

Drone images of 6 forestry sites

Split the images into patches

Field measurements of trees

Target **tree** distribution

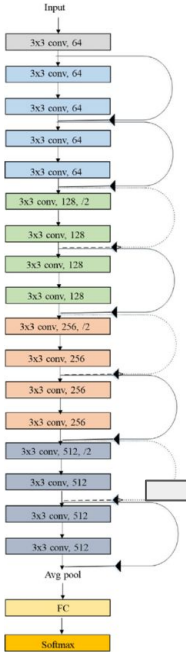
Matching: **Patches + sum over tree density distribution**

Carbon: 5.8016

`carbon = #trees x carbon.mean`

5 x Train

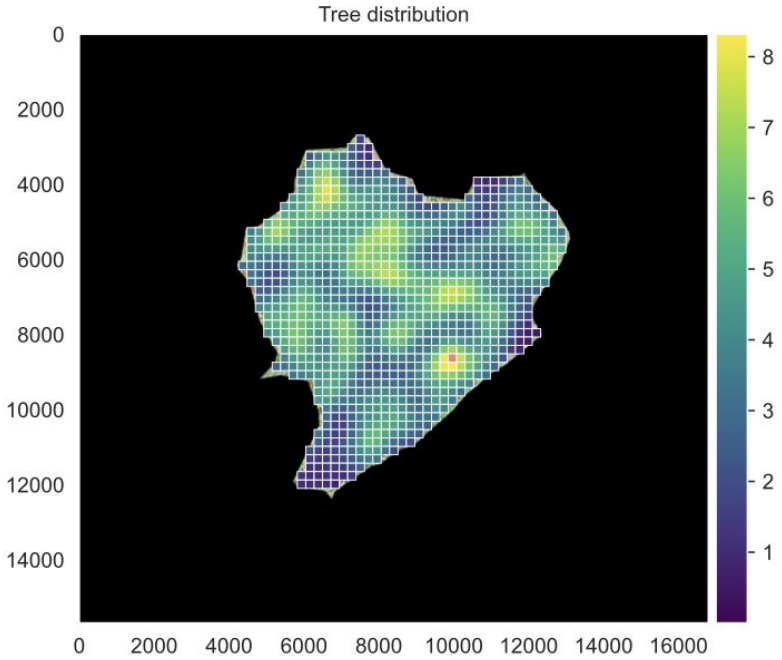
1x Test

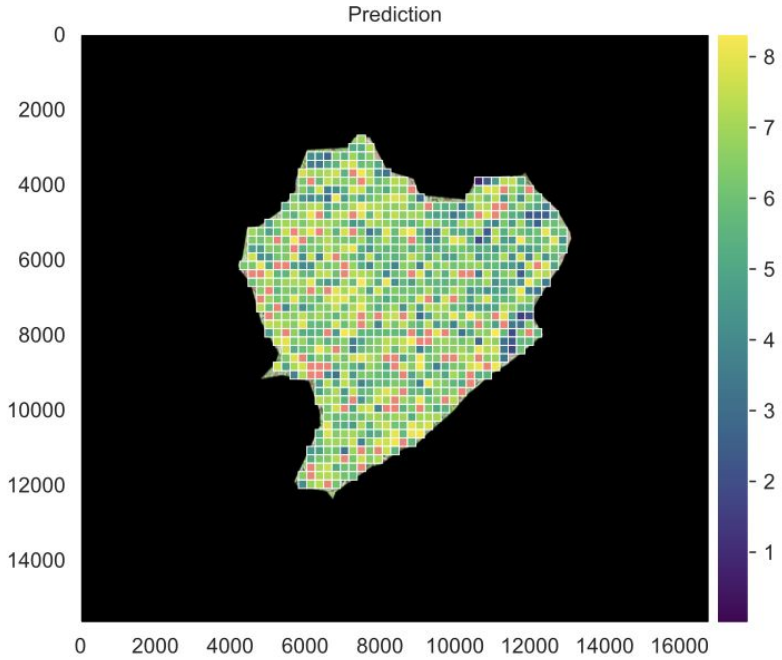


Resnet 18

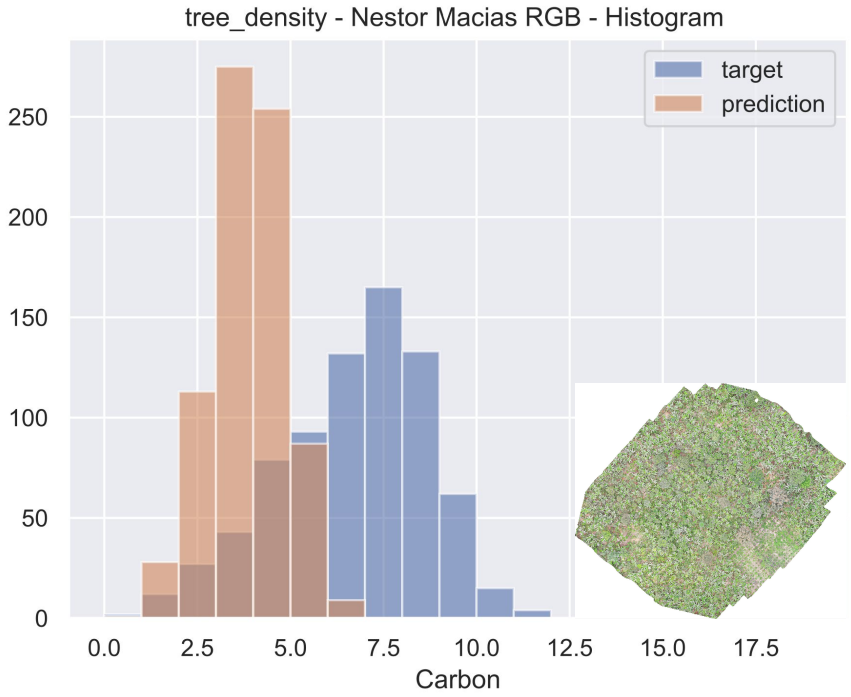
Carbon prediction (for each tile in each site)

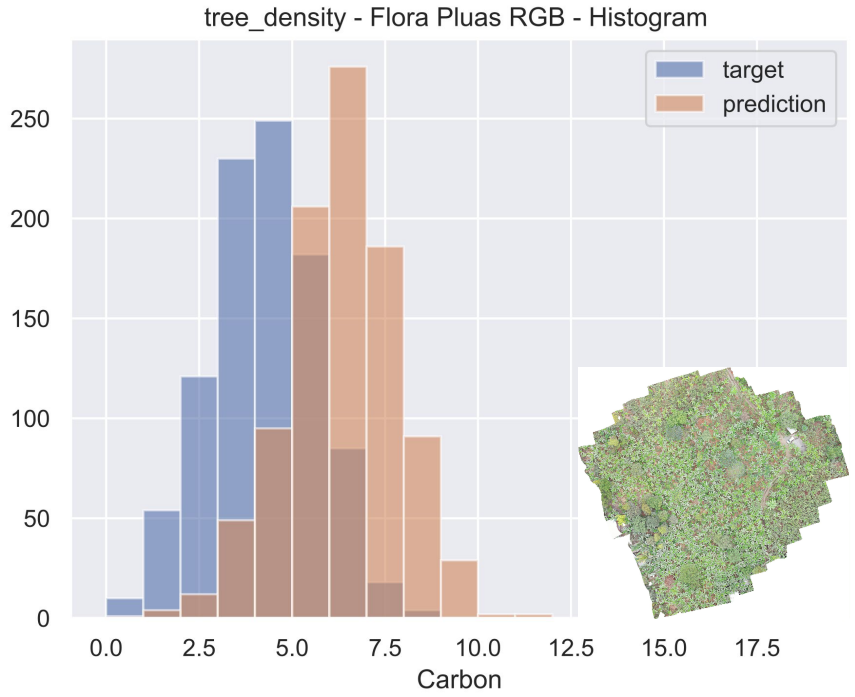
The model is still unable to learn and makes noisy predictions





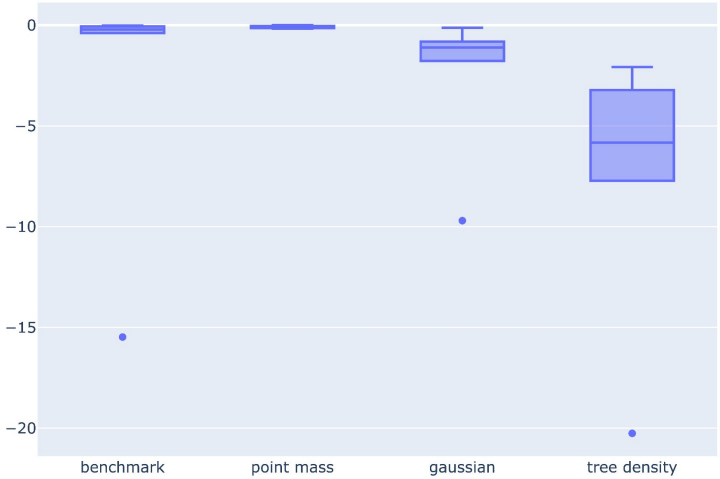
The prediction on the test site resembles the train site distribution



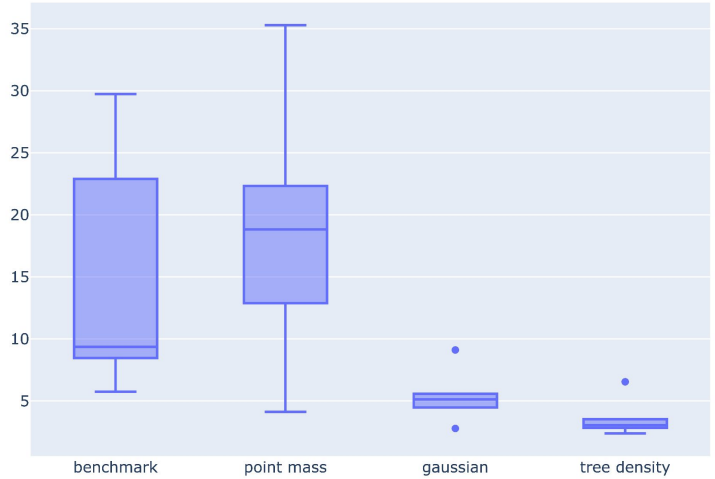


Conclusion

Metric conclusion



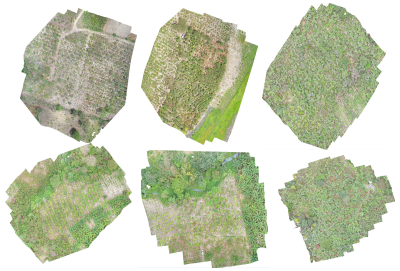
R2 score



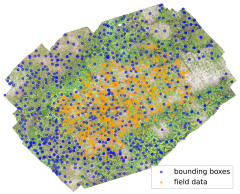
RMSE score
(roughly comparable)

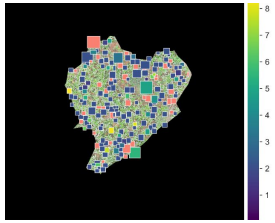
Our main takeaway: AI models can only be as good as their data

Dataset



Benchmark

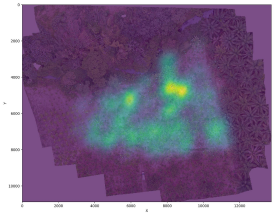


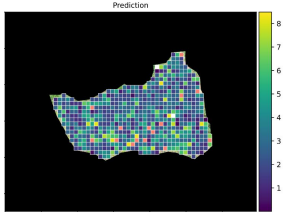


- Use of **out-of-boundary** tree crown images
- Insufficient tree crown **detection**
- Questionable **matching** between drone images and field data

- Extremely **noisy coordinates** due to high GPS error
- Errors or at least **outliers** in the field data (monster trees)
- Very **little and inhomogenous** data making proper training and validation difficult

Our approach





- Patching ensures that **all trees** are at seen by model and allows for data augmentation
- Computing the carbon or even tree **distribution** leads to smoother input data
- Yet, our models only made **noisy** predictions