From Forest to Urban: Data Efficient Tree Segmentation with Self-Supervised Pretraining on Height-Based Voronoi Maps

Jonas Geiselhart[1]
jonetz.github.io
Luca Reichmann[1]
reichmla.github.io
Alina Roitberg[2]
aroitberg.github.io

[1] University of Stuttgart
Stuttgart, GER
[2] University of Hildesheim
Hildesheim, GER

Abstract

We propose a self-supervised pretraining framework for tree segmentation in airborne VHR imagery that exploits both color and infrared (RGBI) data and height maps. Our key idea is pairing height maps and Voronoi decomposition to create auto-labels, enabling pretraining without human annotations. The model is fine-tuned on a small, manually annotated urban dataset, with postprocessing refining results across diverse settings. To validate our idea, we introduce a composite dataset consisting of three parts: (1) An autolabeled forest dataset used for height-driven pretraining, (2) an annotated urban tree dataset used for fine-tuning and (3) a small test dataset with manual trees for validation. Our approach achieves F1-scores of 0.65 (urban) and 0.60 (suburban). This also demonstrates that the proposed height-driven pretraining outperforms the conventional training by 0.44 in urban environments. In summary, we contribute a fully automatic framework to detect trees in large and diverse regions of land using models that were trained by a simple self-supervised mechanism utilizing height data of forest regions. Additionally, we analyze the transfer capabilities with a small finetuning dataset. Code, models, and data are available on GitHub.

1 Introduction

Tree detection and individual crown delineation is a relevant topic in urban development and has recently gained attention due to growing sustainability efforts [2, 5, 14]. Tree data analysis has diverse applications, ranging from supporting tree conservation and maintenance to accelerating construction planning in urban and rural areas using land registries. Yet, obtaining accurate annotations at scale remains a significant challenge, particularly in dense urban landscapes. To meet this challenge, we propose to use height data as an additional modality to reduce annotation workloads. In dense forest environments, where large portions of airborne images are covered by trees, height information can effectively be used to identify tree crowns, making it a powerful basis for self-supervision. This aspect has not been used in the past. Building on this insight, we introduce a *self-supervised*

pretraining strategy that uses *height data* and *Voronoi-based auto-labels* to significantly reduce reliance on manual annotations. The pretraining data mainly consists of forests, but also covers areas such as forest edges, orchards, and agricultural land, as these landscapes typically contain few buildings and other heightened structures apart from trees. After performing self-supervised pretraining on predominantly forested areas, we transfer the model to rural and urban scenarios by fine-tuning on a smaller, manually annotated dataset. Urban environments involve diverse structures, such as buildings, roads, and mixed vegetation, that make precise tree delineation challenging. By combining self-supervised pretraining in forests with targeted fine-tuning in urban regions, our approach achieves robust tree segmentation performance across a wide variety of landscapes despite a relatively low amount of labeled training data. Additionally, we apply and evaluate postprocessing techniques to improve the segmentation quality. We validate our approach on a dataset that we collected, showing that a model pretrained under our self-supervised pipeline significantly outperforms baselines only trained on conventional computer vision data without our pretraining approach. These findings highlight the potential of height-driven self-supervision coupled with Voronoi-based auto-labels. In summary, our main contributions are: (1) A height-driven self-supervised pretraining framework that exploits Voronoi-based auto-labels, substantially reducing manual annotation needs, (2) an automatic pipeline that predicts tiles efficiently and refines segmentation results using post-processing methods, which is well-suited for complex and generalized settings, and (3) a new dataset for tree segmentation, consisting of two parts. The first part is a label-free forest dataset covering forests, orchard edges, agricultural areas, and other regions with few man-made structures, used for height-driven pretraining. The second part is an annotated (sub-)urban dataset, where bounding-box labels in the training split are refined into precise segmentation masks via an external model. We also include a test split containing fully manually annotated segmentation masks for precise evaluation.

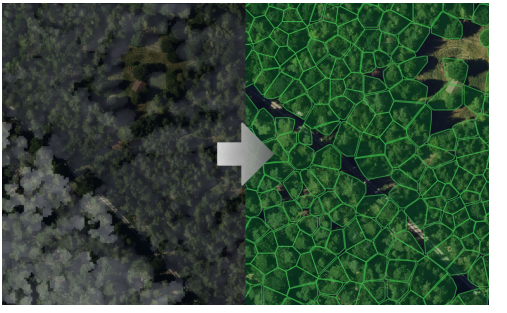


Figure 1: Illustration of our height-driven auto-labeling logic: height data is used to create Voronoi cells that correspond to tree structures in the RGB-data

2 Related Work

Individual Tree Crown Detection and Delineation (ITCD) is particularly challenging in remote sensing data due to the ambiguity of visual cues in areas of continuous tree coverage. Zhao et al. [13] compiled a concise overview over state-of-the-art deep neural algorithms for tree detection. Currently, there are two main frameworks for the segmentation of trees in forest regions, but none specifically target general non-forest areas to the best of our knowledge. The first framework, Detectree [1], was trained with just 2,267 tree crowns from tropical forests in Malaysia and French Guiana. It achieves a high F1-score but lacks model transferability to other regions, especially non-tropical areas. In other forest types, such as those found in central Europe, the algorithm performs significantly worse. The second

framework [3, 10] provides a single self-supervised pretrained model that was fine-tuned on 2,848 manually generated training samples to achieve a similar F1-score of 0.64. This approach was trained with automatic and manual crowns from different areas in the United States. Weinstein et al. [9] employ pretraining through an out-of-the-box tree detection mechanism that generates unsupervised classifications through very dense Lidar data. They subsequently refine the model with manual labels. Notably, the self-supervised training continued to improve performance even though the Lidar-derived tree labels were noisy and contained numerous false predictions. Lassale et al. [4] detect and segment trees in mangrove forests using various methods and compare them. They use the visible near-infrared as well as the panchromatic band, and enhance separability using an encoder-decoder structure in addition to a Laplacian over Gaussian. Later, the tree crowns are segmented using a watershed algorithm. This method is compared against a marker-controlled watershed, region-growing, multi-resolution segmentation and Mask R-CNN. This method works well in contiguous forest environments that only have one type of tree, but fails if there are either other objects or different species introduced. In an early study, Olofsson and Holmgren [6] showed that forests can be efficiently segmented with Voronoi cells using height maxima as control points, which inspired our idea of combining height-driven self-supervision with deep learning.

Our height-driven self-supervision approach addresses the core limitation of insufficient annotated training data, which previous frameworks encountered when transferring models from forest to non-forest regions. By exploiting height data and Voronoi-based auto-labels, our method can generate extensive training signals. Furthermore, in contrast to existing ITCD methods that narrowly focus on tropical or single-site environments, our framework is designed for diverse landscapes, including rural and urban areas with varying building densities.

3 Dataset & Preparation

We begin by presenting the dataset we used for training and its preparation process. Our data was collected in southern Germany, which exhibits typical Central European vegetation.

It consists of aerial RGBI images, which each cover an area of 1 km^2 and have a resolution of 0.2 m. They are true orthophotos, as they have been rectified and are true to scale. To ensure consistent foliage and therefore consistent predictions, the images were captured between June and September 2023. In addition to the RGB channels, we also have a near-infrared channel, which later allows us to filter out false crown shapes using the normalized difference vegetation index (NDVI) in the postprocessing phase (see section 4.4).

Our fine-tuning dataset includes $10,682$ annotations of free standing trees (i.e., trees that are not adjacent to any other trees or elevated vegetation) in rural and suburban areas across 12 tiles in cities of Wüstenrot, Friedrichshafen, Lahr and Waldshut-Tiengen. We use one tile (containing 489 trees) as validation data, while the other 11 tiles serve as training and test data. These annotations consist of coarse bounding boxes, which are later segmented automatically, as explained in section 4.2.

Moreover, we annotated 4 more test tiles manually from different vegetation types to serve as ground truth later in the evaluation phase: an urban environment with 1,614 trees per km^2, a village environment with 1,461 trees per km^2, a countryside environment with 2,433 trees per km^2, and a forest environment with 2,657 trees within 0.25 km^2. We also incorporate height data that capture the elevation of objects relative to the ground. The height

data is given by normalized digital surface models (nDSM). These models are computed from digital surface models (DSM) and digital terrain models (DTM) data. This allows us to utilize the elevation of objects such as trees or buildings independent of the surface-elevation as an additional filtering criterium in the postprocessing phase. In summary, our dataset is composed of three sub-datasets: A *pretraining dataset* with auto-labels for self-supervised learning, which is outlined in section 4.1. A *fine-tuning dataset* with 12 tiles (one used for validation as described), consisting of bounding box annotations, which are refined as described in section 4.2. The third part is a test dataset containing 4 tiles, which consist of manual polygon annotations that are only used for obtaining the metrics in fig. 5.

4 Methodology

In this section, we describe the methodology applied in our framework to make tree segmentation usable for robust prediction, even with limited size datasets in urban scenarios. The approach begins with self-supervised pretraining on automatically generated training data on forest data using height-based Voronoi maps. We then perform transfer learning by finetuning our model on urban orthophotos, which is the learning target of our approach. To improve the training performance, we additionally refine the labels of the urban dataset with a segmentation algorithm. The comprehensive training process is shown in fig. 2.

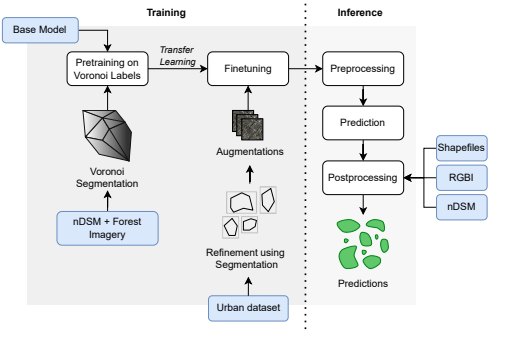


Figure 2: The figure shows the pipeline used to train our models. We begin by pretraining the base model using the Voronoi segmented data, continue by finetuning using our automatically segmented training dataset and run inference using our postprocessing heuristics.

4.1 Height-Driven Self-Supervised Pretraining

In the following, we explain how the pretraining dataset is created and how we use it to pretrain our model on forest data.

4.1.1 Generating Auto-labels from Height Data Using Voronoi Maps

Our approach of using height data is based on the fact that the tree crown is the tallest part of the tree and that the extents of treetops in forests are roughly equal. Following this, we can segment them simply by using the maxima positions in a forest. Trees that are not within a forest can later be adapted to fit the cells using absolute height values. Since our dataset only contains annotations of sparsely distributed free standing trees in small tiles of rural environments, we have to augment the training process in order to achieve a high recognition rate over diverse environments. This provides us with a more variable distribution of trees across regions, which enables our model to learn more robust features, resulting in more stable and versatile predictions at inference time.

Instead of focusing on our initial urban and suburban objective regions, we focus on forest and countryside areas. We do this because buildings or other constructions appear as elevated structures in the nDSM data and would incorrectly be labeled as a tree. We use a Voronoi algorithm to generate approximate tree shapes by treating tree crowns as elevation maxima from 370 manually selected 1x1 *km* tiles in southern Germany. Figure 1 illustrates the relation between the RGB image together with the nDSM information overlayed in grey on the left and the corresponding Voronoi separation on the right. The nDSM may contain minor inaccuracies, but for our purposes their impact is negligible and mitigated by our filtering and cleanup steps.

The height maxima are computed by first applying a Gaussian filter for smoothing the height data. Next, we threshold the pixels using a predefined height threshold, filtering out anything below. We then apply a local maximum filter with fixed neighborhood size, followed by a second height threshold. To obtain a prediction, we build a convex hull of all pixels that still remain in the cell.

The final step is to clean malformed cells, which may occur when two trees with a lot of distance are incorrectly placed in one cell. To address such errors, we compute the percentage of the data points in the cell that are below the first predefined height threshold. If this portion is more than a given fraction, the cell is discarded. As hyperparameters we used a σ of 0.25 in the Gaussian filter, a first and second height threshold of 2 *m* and 3 *m* respectively and a fixed neighborhood size of 7 *m*. Visual inspection shows that this method performs well for deciduous forests and reasonably well for mixed forests. However, for coniferous forest, the algorithm is more sensitive to hyperparameter selection. This is because trees standing close to each other are often roughly the same height, which makes it challenging to balance the degree of smoothing and neighborhood size. This dataset enables self-supervised pretraining on a more complicated task, as tree separation in forests is harder than it is in urban environments from a sole visual perspective. Using height data allows for a more systematic approach. The learned model from pretraining can then be transferred to urban, suburban and countryside environments, which is our main focus in this paper.

4.1.2 Details on Model & Pretraining

We train our *forest model* using a ResNet model that is pretrained on the COCO-Set with our automatically generated pretraining forest dataset using the Detectree framework [10]. We train the model on $1,080,078$ trees with $17,690$ subtiles. The set of trees is split into 50% trees for training, 35% for validation and 15 % for testing. For the training process we used a batch size of 9 images, backbone freeze after three layers, $1,209$ batches per images, a base learning rate of 0.01, up to $20,000$ iterations and a scheduler decay of 0.01 as hyperparameters. Using these hyperparameters and three-fold validation, the model can be trained in under 12 hours on a GeForce RTX 4090.

4.2 Automatic Segmentation on the Finetuning Dataset

To further improve the training performance, we refine the manually created annotations in the fine-tuning dataset - initially given in rectangular boxes - into more precise shapes using a standard pretrained segmentation model. In order to employ the segmentation algorithm, we first divide the 1x1 *km* tile into subtiles for more detailed predictions. We extract subtiles, which are then processed with the pretrained geospatial segmentation model of the Samgeo package developed by Osco, Wu et al. [7, 11]. Afterwards, the predictions

are stitched together and overlapping crowns are removed based on maximizing the IoU of the bounding boxes and segmentation masks. These overlaps occur due to the buffer region that avoids bounding boxes clipped to subtile edges. We clean the predictions in a three-step postprocessing procedure to further improve the results. In summary, we use the IoU between the segmentation and ground truth box to check the validity of the segmentation and use the ground truth box as a fallback. This process is manually tuned with a second segmentation dataset. An example of this process can be seen in fig. 3.

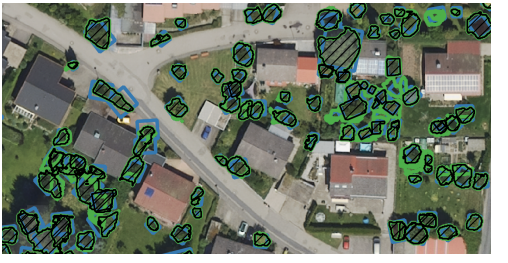


Figure 3: This is a segmentation example with blue boxes showing manual annotations and green shapes showing predictions. The crossed black pattern is the final output of the segmentation algorithm.

4.3 Transfer Learning to Urban Environments

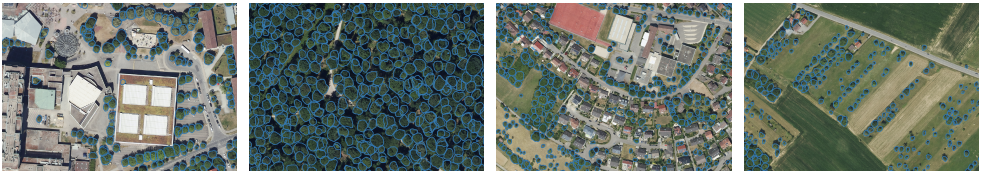
We now use the forest model obtained in section 4.1.2 and fine-tune it to adapt the prediction capabilities to urban regions with the segmented urban annotations. The dataset consists of a training split with 6,579 training instances and 2,917 test instances (30%), generated following the details specified in section 4.2. We adjust the training parameters to $\gamma = 0.1$ and base learning rate of 0.005. This model is referred to as the *urban model*.

In an alternative approach, we augment the training dataset with three manually selected tiles from the pretraining dataset in order to balance the urban dataset, as we observed a significant drop in the accuracy of detection rates and shape prediction of contiguous tree crowns. We refer to this model as the *combined model* in our evaluation. We selected Masked R-CNN [3] as the base model for our tree detection model, as this architecture is particularly good for extracting accurate shape outlines. Mask R-CNN first detects regions of interests and then segments individual pixels within the ROI. This method produces accurate segmentations in remote sensing tasks, even in densely packed regions [8]. We use Detectree2 [1] for training our tree segmentation models, but implement our own inference logic utilizing Detectron2 [12] with the pretrained ResNet-101 backbone.

One of the changes we implement in the inference logic is disabling the default non-maximum suppression (NMS) from Detectron, as we observed that NMS often results in high-confidence in parts of tree crowns, but suppresses the prediction of the whole tree. As a countermeasure, we keep every predicted crown from every ROI that is above a pre-defined confidence threshold and sort it out in the postprocessing step. We give further details on the sorting mechanism in section 4.4.

4.4 Inference

We improve the results of the tree segmentation algorithm when running inference, by performing pre- and postprocessing to improve our results. We use a custom implementation based on Detectron2 to enable efficient inference for large regions. Our approach can be used efficiently with $\sim 1\ min/km^2$ on a Nvidia RTX 4090 and $\sim 1.5\ min/km^2$ on a Nvidia RTX 4060 Ti in Central Europe.



(a) City Tile (b) Forest Tile (c) Village Tile (d) Countryside Tile

Figure 4: The figures show the qualitative tree segmentation results of the combined model in the four environments city, forest, village and countryside.

Preprocessing Before performing inference on images, we preprocess the tiles. Therefore, we subdivide the original tiles into 70×70 *m* tiles, including a 20 *m* buffer. We categorize each subtile as either forest, non-forest or combined. This allows us to further optimize prediction time as only the models necessary on the specific tile are run. This approach allows for more flexibility during inference time, but is not included in the evaluation, as we compare the models equally on each tile.

Postprocessing We filter the results in the postprocessing using exclusion geometry, height, NDVI, area, and containment information. As seen in the results, the exclusion of obvious outliers, subcrowns and duplicates is especially useful in city and village areas. The corresponding steps of the postprocessing pipeline can be found in the supplementary material.

5 Evaluation

We evaluate the models described in section 4.3 with different comparison models, using the metrics outlined in section 5.1. The goals of our prediction can be broken down in two factors: (1) precise and complete urban tree detection and accurate shape recognition and (2) accurate countrywide tree crown shapes. In summary, we can see improvements especially in city, village, and countryside areas.

5.1 Metrics

To assess the performance of our models, we use the Intersection over Union (IoU) and the F1-score. IoU is a standard metric for evaluating shape accuracy by comparing the predicted and ground truth polygons. Unless stated otherwise, we classify predicted polygons with an IoU above 0.5 as true positives. Otherwise, they are considered false positives. Based on this classification, we compute Precision and Recall, and subsequently the F1-score, which is the harmonic mean of Precision and Recall.

5.2 Results

We evaluate the predictions both qualitatively using new sample tiles, and quantitatively, based on the four tiles village, countryside, city and forest as described in section 3.

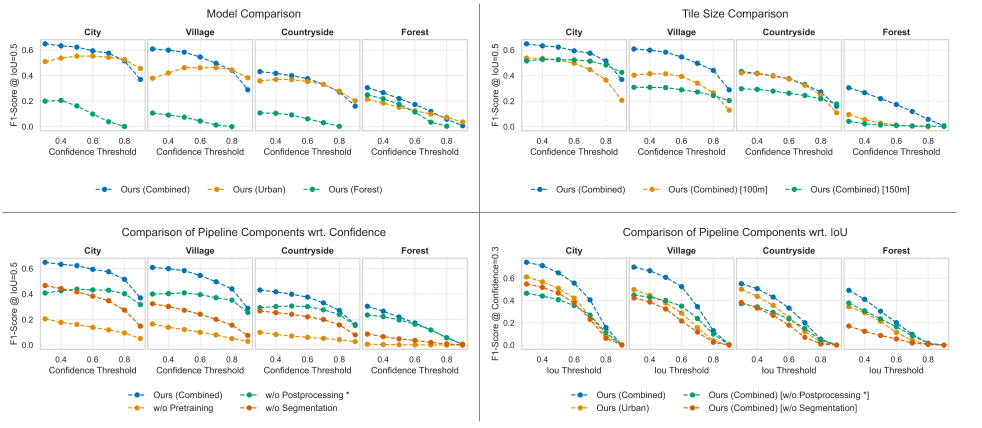


Figure 5: *F1-curves across model variants, configurations, and environments.*
Top left: three trained models — the pretrained **forest model**, the **urban model** fine-tuned on urban data, and the **combined model** fine-tuned on urban data with auto-labels (section 4.3). Top right: impact of tile size during training and inference. Bottom left: ablations with parts of the pipeline disabled. Bottom right: effect of varying IoU thresholds to assess shape accuracy. * Duplicates are still removed even when postprocessing is disabled.

5.2.1 Quantitative Results

In fig. 5, we show the results of the evaluation using the F1-score, where a prediction is considered a detection if the ground truth exceeds 0.5. We show F1-score results using different confidence thresholds of the urban, forest and combined models in the first row. We also show the results when varying the tile sizes and IoU thresholds in the third row. The F1-scores in table 3 at different IoU thresholds show how correctly the shape of the trees are delineated wrt. the ground truth. Unsurprisingly, the combined model is particularly good in city and village areas, since the finetuning labels were specifically segmented for those environments. Finally, table 4 shows the averaged IoU score over all predictions as a second measure of shape accuracy. In fig. 5, it can also be observed that the pretrained forest model performs the worst in every environment except for forest areas when compared to the other models. Even on the forest tile, it only performs marginally better than the urban model with a maximum value of 0.25, while the combined model still achieves higher F1-scores than the forest model with a maximum F1-score of 0.34. Overall, the combined model achieves the highest segmentation and detection performance in each environment. Only if the confidence threshold of the predictions is increased, the F1-scores of the urban and combined models converge. As a result, the urban model sometimes outperforms the combined model, which is especially apparent in the urban environment. The models generally perform best in urban areas, which is expected, considering they are fine-tuned on images from small villages and cities. The evaluation based on the tile size variations in the top right of fig. 5 shows that a more time- and resource-efficient approach to training and inference on larger tiles results in a degradation in detection performance. It has to be noted that we filter annotations based on an area greater 1 m^2, height of the tree exceeding than 3 m and a NDVI value of at least 0.15.

5.2.2 Qualitative Results

In fig. 4, we show the qualitative results of applying the combined model on the image tiles city and forest, village, countryside. We provide a closer look of the results in the figure in our supplementary data. To summarize, while the majority of trees are detected correctly in most images and the shapes are effectively separated, there are still some errors in detecting small or ambiguous trees.

5.3 Comparison to other Frameworks

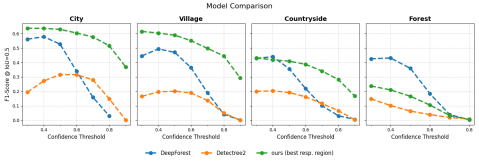


Figure 6: Comparison of Prediction to the current version of DeepForest [10] and version 1.0.8 of Detectree2 [1] to measure Confidence

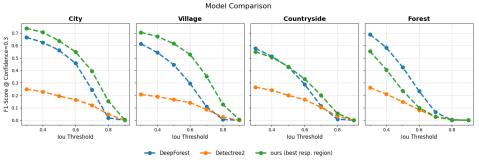


Figure 7: Comparison of Prediction to the current version of DeepForest [10] and version 1.0.8 of Detectree2 [1] depending on IoU to Measure Shape accuracy

We compare our model against the pretrained Detectree2 [1] and DeepForest [10] models based on confidence in fig. 6 and shape accuracy fig. 7. Detectree2 has a low performance across all areas, which is unsurprising since it is only trained on 3,797 manual annotations of tropical forest. Therefore, there is a large disparity between training and evaluation data. Qualitatively, many true positives produced by Detectree2 exhibit low confidence scores. The DeepForest model is a lot more robust with respect to the prediction environment, so that it generally results in better scores. This can be attributed to its larger and more diverse pretraining set, which includes automatically generated crown annotations. Weinstein et al. employ a self-supervised training strategy, using data generated by an unsupervised tree detection algorithm applied to Lidar scans. DeepForest outperforms our model in forested regions, while the performance is comparable in orchard areas. Our model shows superior detection performance in urban and village settings. These differences align with the distribution of the fine-tuning data. We only annotated city and village areas, whereas DeepForest was fine-tuned with roughly the same number of annotations, but from orchards and forest regions. In contrast to DeepForest, our primary contributions are: 1) demonstrating that transfer learning enables generalization across shifts in the distribution of geographic domains and 2) employing a simpler, unsupervised tree extraction algorithm only based on the normalized Digital Surface Model (nDSM).

6 Conclusion

In this paper, we present a framework to infer images and extract precise tree shapes. We leverage self-supervised transfer learning from forest areas using Voronoi segmentation to improve the tree delineation in (sub-)urban areas. We also employ segmentation to refine our training data in order to achieve more precise shapes. The evaluation shows that using both approaches results in a significant increase in tree detection rate. Overall, we provide

Models	City	Village	Country	Forest
Ours (Combined)	**0.65**	**0.61**	**0.43**	**0.30**
w/o postprocess.	0.41	0.40	0.29	0.24
w/o segmentation	0.47	0.32	0.27	0.09
w/o pretraining	0.21	0.17	0.10	0.01

Table 1: F1-scores at confidence 0.3 for ablation variants of our pipeline.

Models	City	Village	Country	Forest
Ours (Forest)	0.20	0.11	0.11	0.25
Ours (Urban)	0.51	0.38	0.36	0.21
Ours (Combined)	**0.65**	**0.61**	**0.43**	**0.30**

Table 2: F1-scores comparing forest, urban, and combined models.

Ours (Combined)	City	Village	Country	Forest
IoU = 0.3	**0.74**	**0.70**	**0.55**	**0.49**
IoU = 0.5	0.65	0.61	0.43	0.30
IoU = 0.7	0.41	0.35	0.20	0.10

Table 3: F1-scores for different IoU thresholds at confidence 0.3.

Models	City	Village	Country	Forest
Ours (Combined)	**0.68**	**0.67**	**0.62**	**0.55**
w/o segmentation	0.65	0.60	0.57	0.53
w/o postprocess.	0.67	**0.67**	0.61	0.53
Ours (Urban)	0.65	0.61	0.58	0.54
Ours (Forest)	0.51	0.49	0.46	0.49

Table 4: IoU scores at confidence 0.3, threshold 0.3, for all models and ablations.

especially effective results with a focus on small villages and cities. We also provide several models that can be used out-of-the-box or be fine-tuned for any purpose using the given supplementary scripts.

Acknowledgements

This project was funded by the project *Smart Villages - Attraktive Orte im ländlichen Raum* and conducted in close collaboration with the *State Office for Geoinformation and Land Development Baden-Württemberg (LGL BW)*.

References

[1] James G. C. Ball, Sebastian H. M. Hickman, Tobias D. Jackson, Xian Jing Koay, James Hirst, William Jay, Matthew Archer, Mélaine Aubry-Kientz, Grégoire Vincent, and David A. Coomes. Accurate delineation of individual tree crowns in tropical forests from aerial rgb imagery using mask r-cnn. *Remote Sensing in Ecology and Conservation*, 9(5):641–655, 2023. doi: https://doi.org/10.1002/rse2.332. URL https://zslpublications.onlinelibrary.wiley.com/doi/abs/10.1002/rse2.332.

[2] Haoyu Gong, Qian Sun, Chenrong Fang, Le Sun, and Ran Su. Treedetector: Using deep learning for the localization and reconstruction of urban trees from high-resolution remote sensing images. *Remote Sensing*, 16(3), 2024. ISSN 2072-4292. doi: 10.3390/rs16030524. URL https://www.mdpi.com/2072-4292/16/3/524.

[3] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2017. URL http://arxiv.org/abs/1703.06870.

[4] Guillaume Lassalle, Matheus Pinheiro Ferreira, Laura Elena Cué La Rosa, and Carlos Roberto de Souza Filho. Deep learning-based individual tree crown delineation in

mangrove forests using very-high-resolution satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 189:220–235, 2022. ISSN 0924-2716. doi: https://doi.org/10.1016/j.isprsjprs.2022.05.002. URL https://www.sciencedirect.com/science/article/pii/S0924271622001411.

[5] Tobias Leichtle, Markus Zehner, Marlene Kühnl, Klaus Martin, and Hannes Taubenböck. Urban trees - detection, delineation, quantification, and characterisation based on vhr remote sensing. In Manfred Schrenk, Vasily V. Popovich, Peter Zeile, Pietro Elisei, Clemens Beyer, Judith Ryser, and Gernot Stöglehner, editors, *REAL CORP 2021*, Proceedings of the REAL CORP 2021, pages 1029–1039, September 2021. URL https://elib.dlr.de/143807/.

[6] Kenneth Olofsson and Johan Holmgren. Forest stand delineation from lidar point-clouds using local maxima of the crown height model and region merging of the corresponding voronoi cells. *Remote Sensing Letters*, 5(3):268–276, 2014. doi: 10.1080/2150704X.2014.900203. URL https://doi.org/10.1080/2150704X.2014.900203.

[7] Lucas Prado Osco, Qiusheng Wu, Eduardo Lopes de Lemos, Wesley Nunes Gonçalves, Ana Paula Marques Ramos, Jonathan Li, and José Marcato. The segment anything model (sam) for remote sensing applications: From zero to one shot. *International Journal of Applied Earth Observation and Geoinformation*, 124:103540, 2023. ISSN 1569-8432. doi: https://doi.org/10.1016/j.jag.2023.103540. URL https://www.sciencedirect.com/science/article/pii/S1569843223003643.

[8] Hao Su, Shunjun Wei, Min Yan, Chen Wang, Jun Shi, and Xiaoling Zhang. Object detection and instance segmentation in remote sensing imagery based on precise mask r-cnn. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 1454–1457, 2019. doi: 10.1109/IGARSS.2019.8898573.

[9] Ben G. Weinstein, Sergio Marconi, Stephanie Bohlman, Alina Zare, and Ethan White. Individual tree-crown detection in rgb imagery using semi-supervised deep learning neural networks. *Remote Sensing*, 11(11), 2019. ISSN 2072-4292. doi: 10.3390/rs11111309. URL https://www.mdpi.com/2072-4292/11/11/1309.

[10] Ben G. Weinstein, Sergio Marconi, Mélaine Aubry-Kientz, Gregoire Vincent, Henry Senyondo, and Ethan P. White. Deepforest: A python package for rgb deep learning tree crown delineation. *Methods in Ecology and Evolution*, 11(12):1743–1751, 2020. doi: https://doi.org/10.1111/2041-210X.13472. URL https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13472.

[11] Qiusheng Wu and Lucas Prado Osco. samgeo: A python package for segmenting geospatial data with the segment anything model (sam). *Journal of Open Source Software*, 8(89):5663, 2023. doi: 10.21105/joss.05663. URL https://doi.org/10.21105/joss.05663.

[12] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

[13] Haotian Zhao, Justin Morgenroth, Grant Pearse, and Jan Schindler. A systematic review of individual tree crown detection and delineation with convolutional neural

networks (cnn). *Current Forestry Reports*, 9(3):149–170, Jun 2023. ISSN 2198-6436. doi: 10.1007/s40725-023-00184-3. URL https://doi.org/10.1007/s40725-023-00184-3.

[14] Haotian Zhao, Justin Morgenroth, Grant Pearse, and Jan Schindler. A systematic review of individual tree crown detection and delineation with convolutional neural networks (cnn). *Current Forestry Reports*, 9:1–22, 04 2023. doi: 10.1007/s40725-023-00184-3.