Single Tree Detection

Mansur Nurmukhambetov
Christian Salzmann
Kasper Verhaeghe
Lucien Piat

The Jurassic Bark: Single Tree Version

March 2024

Contents

Abstract

Deep learning is revolutionizing the way we map and manage urban green spaces. Traditionally, mapping trees in complex cityscapes has been a challenge due to their irregular shapes and the presence of buildings. However, deep learning segmentation methods can now automatically classify individual trees in high-resolution imagery, enabling highly accurate and cost-effective urban tree mapping. This project explores the application of a deep learning model U-Net for urban tree mapping, paving the way for improved urban planning, environmental monitoring, and sustainable development.

Link to the GitHub

1   Introduction

Green elements in urban city landscapes play an important role in counteracting the urban heat island effect [1], improving stormwater retention by preventing overburdening of sewer infrastructure [2], energy efficiency and air quality [3]. Additionally urban green has shown to play an essential role for both physical and mental health [4]. An important contributor to urban greenness are trees, not only do they play a role in countering said issues, but they are also an essential contributor to the biodiversity within cities assisting the improvement of enhancing connectivity between urban green spaces and green outside of cities [5].

As cities are often big and urban planning is not as detailed everywhere, being able to identify all individual trees in a city and effectively inventorizing them, could have a multitude of benefits for urban tree monitoring [6]. Local legislation might want to regulate the removal of trees, to prevent illegal removal (without a permit) and decrease tree loss [7]. For biodiversity within urban landscapes, knowing the exact location of all trees can enhance future connectivity by knowing where corridors are limited and where additional planting could further improve this [8]. Continuous remote sensing of forests provides a cost-effective way to regularly assess vegetation patterns, aiding decision-making processes for urban forest and tree management.

Mapping city trees is tough due to their irregularity and urban clutter, due to the irregular shapes and sizes of trees, obstructions from buildings, and the overall complexity of cityscapes. High-resolution imagery and robust machine learning processes are crucial for accurately distinguishing individual trees as objects. Particularly deep learning segmentation methods are enabling highly accurate automated mapping of these vital green spaces. In this project machine learning is used to identify single trees from aerial images of the city of Göttingen, Germany. Using a limited number of high resolution aerial images in 4 bands (red, green, blue, infrared), various data pre-processing steps are taken before training the U-Net convolutional neural network (CNN) for semantic segmentation.

2   Related Work

2.1   SWIR Imaging for Vegetation Analysis

Shortwave infrared (SWIR) radiation has a wavelength range of $0.78 - 1.5\mu m$. It is not visible to the human eye and is therefore not part of the visible light spectrum, which includes the colours red, green and blue. SWIR is widely used in remote sensing applications. Many different satellites, such as Sentinel 2, are capable of detecting SWIR and

other wavelengths. A variety of possible combinations of bands with different wavelengths can be used to create compositions for specific purposes. SWIR is particularly good for viewing vegetation. This is mainly due to the strong reflection of SWIR radiation by chlorophyll, see Figure 1. The healthier a plant, the more chlorophyll it contains. This allows for effective detection of plants, especially healthy ones. By replacing the red band with SWIR the image becomes SWIR, green and blue bands respectively and in this case all plants turn red, for example like in Figure 3. This method is called a false colour composition, contrasting the true colour composition, which is created using RGB and the normal way of viewing an image.

As soon as a plant becomes sick, dries out, or dies, its detectability by SWIR light decreases, this is shown in Figure 1. Green vegetation has the highest reflectance peak in the SWIR spectrum $(0.78 - 1.5\mu m)$. However, dry vegetation reflects only half of the SWIR light. This is due to the poor condition of the chlorophyll and cells, which are unable to reflect the SWIR light properly. The same effect is reversed with green light, which is why we see plants as green. This strong reflection of IR light can contribute significantly to the effective imaging of plants [9, 10, 11].
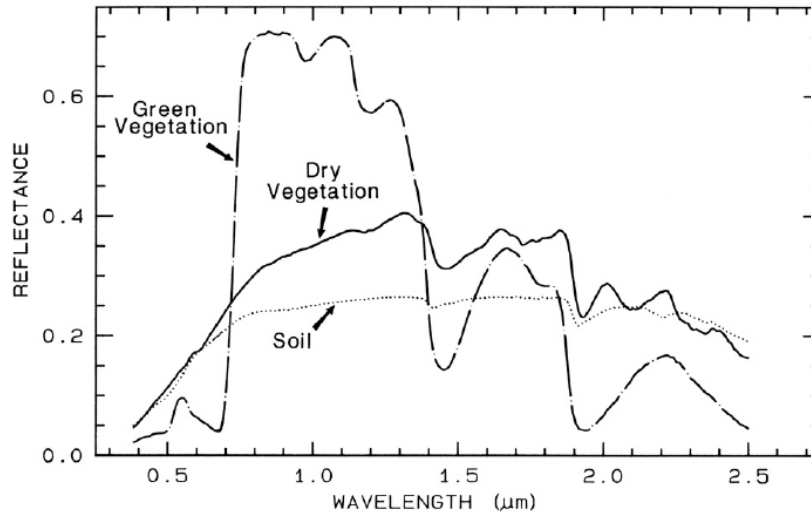

Figure 1: Reflectance patterns green vegetation (healthy), dry vegetation and soil at different wavelengths [12].

The false-color composite effectively represents the entirety of vegetation, encompassing all living plants, including trees, shrubs, meadows, and other grasslands. Additionally, other surfaces were also depicted in a slightly reddish hue, as illustrated in Figure 3. This is attributed to the reflection of SWIR by specific types of urban structures.

2.2 U-Net

The U-Net architecture [13] has set itself as the technique for semantic segmentation tasks, particularly in the realms of biomedical and satellite imagery analysis, showcasing remarkable performance across diverse datasets and applications [14, 15, 16]. In biomedical imaging, U-Net has demonstrated exceptional accuracy in tasks such as cell segmentation [13] and medical image segmentation [17], where precise delineation of structures is paramount
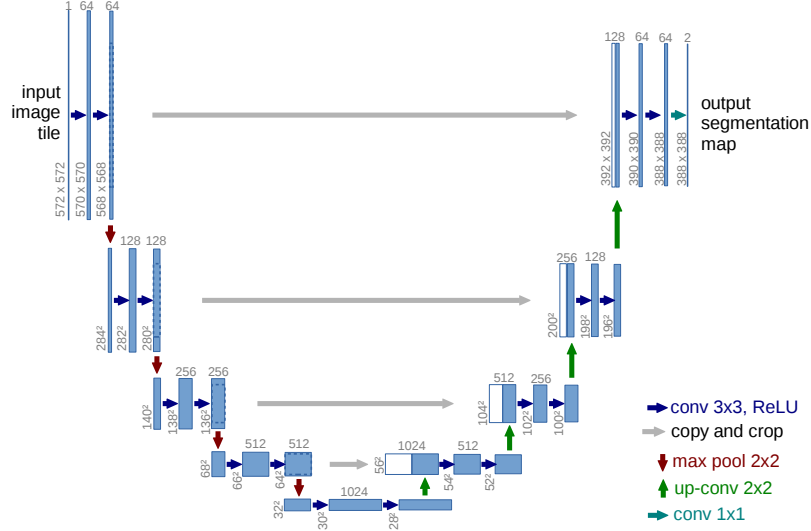
Figure 2: Example of the U-Net architecture from the original paper [13]. The architecture has a downsampling path, on the left side, called contractive and an upsampling path, on the right side, called expansive. The architecture has skip connections denoted by the gray arrows.

for diagnosis and treatment planning. Its ability to capture fine details while preserving contextual information has made it indispensable in various imaging modalities.

The U-Net architecture is shaped like the letter U, see Figure 2, and it consists of two paths: contracting (left) and expansive (right) path. The contracting path is a multi-layered convolutional network for downsampling. In the expansive path, every step is an upsampling of the feature map followed by a up-convolution. The U-Net has a residual-like connection by copying features from the contracting layers to respective expansive layers. This allows information to flow in local and global scales, facilitating precise segmentation even for intricate structures within images.

3 Methods

3.1 Dataset

The dataset consists of aerial imagery of the city of Göttingen in Lower Saxony, Germany around in May 2018. The data was provided by the Department of Remote Sensing and Forest Inventory at the University of Göttingen. In total, there were 38 aerial image tiles randomly distributed across the city. This resulted in only a partial coverage of the city with tiles. Each tile covered an area of 100×100 meters. The spatial resolution was 10 cm, allowing for the identification of even the smallest details. The spectral resolution included RGB and SWIR bands.

In addition to the aerial imagery, there are also mask layers provided that represent the tree crowns in each tile as an area. The exact origin of labels is unknown. However, it is assumed that the tree crowns were digitized manually and that the imagery are made by plane or satellite. From data it appears that alive and not trees were included. The

mask layer was also available as raster data, like the aerial imagery.

Figure 3: Left image is an overview map of Göttingen, with red areas showing the provided dataset; On the right, an example of one of the tiles in false-color composition.

3.1.1 Class imbalance

In urban areas, aerial imagery without trees dominate due to the amount of buildings, roads, and other structures. As we can see in the Figure 4 the dataset is highly skewed towards the background class (one-third as many trees as background). Considering that we are using pictures that contain a mix of the two classes, we can't do resampling effectively, nor we can reduce the amount of background artificially. The imbalance is anchored to the dataset.
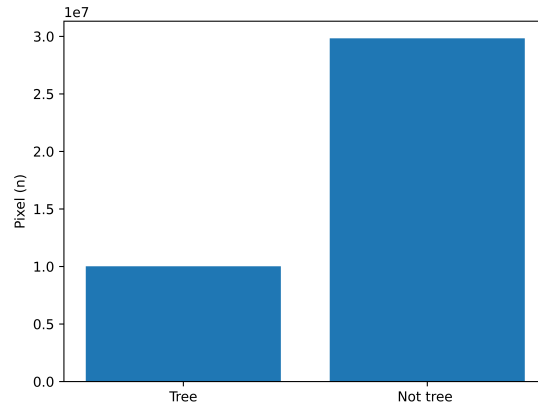

Figure 4: Histogram comparing the number of trees and non-tree pixels in the whole dataset. There are three times as many background pixels than trees.

Class imbalance can pose challenges for training because the model may become biased towards the majority class, leading to poor predictive performance for the minority class.

Our model could have trouble detecting trees accurately as the model could tend to prioritize accuracy on the majority class while ignoring the minority class [18]. Imbalanced datasets can also lead to overfitting, the model might learn to memorize the majority class rather than patterns that distinguish trees from non-trees. This could result in poor performance when we apply the model to new, unseen data [19].

A solution to that problem is to scale down the background with a weighting technique, by assigning class weights are inversely proportional to the class frequency. During the training phase, these weights are incorporated into the loss function that penalizes errors based on the assigned weights. The minority class is given a higher weight, meaning that misclassifications in the prediction contribute more to the overall loss. The class weights used are 1 and 1/3 for trees and background respectively.

3.2 Pre-processing

The dataset only contains 38 images, which is too low for training deep learning models. Our model could have trouble generalizing to unseen data. Moreover, it can be challenging to reliably evaluate the performance of the model with such a low sample size [20, 21].

Aerial images capture large areas, providing room for affine transformations such as rotation and cropping. These transformations can be applied to simulate changes in viewpoint and orientation. Trees exhibit natural variability in terms of shape and size. This inherent invariant allows us to flip the images. The dataset was augmented to artificially increase the number of samples, see Figure 5, using the following affine transformations:

- **Random Rotation:** Each image and its corresponding mask are rotated by a random multiple of 90 degrees. This step ensures that the rotation does not introduce interpolation artifacts, preserving the integrity of the data;

- **Random Cropping:** Images and masks are concatenated and then randomly cropped from 1024×1024 to a size of 512×512 pixels. This transformation simulates different viewpoints and captures various segments of the larger scene, effectively increasing the robustness of the model to partial views of trees;

- **Random Flipping:** Both horizontal and vertical flips are applied with a 50% probability. This step exploits the natural symmetry and variability in tree appearances, making the model resilient to changes in tree orientation.

Out of the four available bands, we'll use SWIR, green and blue because the model mentioned later can only accept three channels.

3.3 Model

For this task we made the U-Net model by used a pretrained MobileNetV2 [22] model as the contracting model and modules from pix2pix [23] as the expansive model. We chose to use a pretrained model with fixed weights for this task because of lack of data mentioned earlier. This allows for transfer learning which is faster and easier to train on limited amounts of data.

The layers extracted from MobileNetV2 for contracting path and their respective layers for expansive path can be found in Table 1.

In Table 2 the hyperparameters used during training are mentioned. Hyperparameter optimization was not performed. The model was trained for 40 with a batch size of 32. As the training images are generated during training, there was no bound on the number of
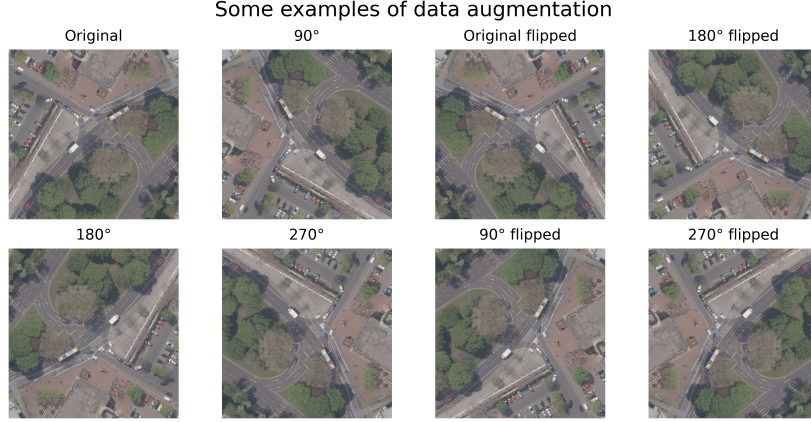
Figure 5: Example of a few images generated from a single one using flipping and rotation.

Table 1: On the top, an overview of the contracting layers used from MobileNetV2 in the U-Net model. On the bottom, overview of the expansive layers in the U-Net model.

Contracting Layer	Output Size	Description
block_1_expand_relu	64x64	First layer of feature extraction
block_3_expand_relu	32x32	Second layer, deeper features
block_6_expand_relu	16x16	Mid-level feature extraction
block_13_expand_relu	8x8	Penultimate feature extraction layer
block_16_project	4x4	Final layer, deepest features
Expansive Layer	Output Size	Description
Upsample 512	8x8	Increase from 4×4 to 8×8
Upsample 256	16x16	Increase from 8×8 to 16×16
Upsample 128	32x32	Increase from 16×16 to 32×32
Upsample 64	64x64	Increase from 32×32 to 64×64
Conv2DTranspose	128x128	Up-sample from 64x64 to 128x128

steps per epoch, so we chose to have 100 steps per each epoch. The model was optimized using the Adam optimizer [24] with a learning rate of 10^{-3}.

3.4   Evaluation

Binary cross-entropy is a commonly used loss function in binary classification tasks, including tree detection from aerial imagery. It measures the discrepancy between the predicted probabilities of tree presence and the ground truth labels. Binary cross-entropy is advantageous here due to its simplicity and the fact that it can handle class imbalance, which is the case in our dataset [25].

$$\mathcal{L} = -(y \log(p) + (1 - y) \log(1 - p))$$

To complement the use of binary cross-entropy as our loss function, we employ two primary metrics for evaluating the performance of our tree detection model: per-pixel accuracy and intersection over union (IoU).

Table 2: Hyperparameters used during training.

Parameter	Value
Epochs	40
Batch size	32
Steps per epoch	100
Optimizer	Adam
Learning Rate	10^{-3}

Accuracy Accuracy is a straightforward metric that evaluates the proportion of pixels correctly classified in the image. It calculates the ratio of correctly predicted pixels (both tree and non-tree) to the total number of pixels.

Intersection over Union (IoU) Intersection over Union, also known as the Jaccard index, is a more sophisticated metric that measures the overlap between the predicted tree regions and the actual tree regions in the ground truth. Defined as the ratio of the intersection of the predicted and true labels to their union, IoU provides a clear quantification of the model's precision and recall capabilities:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

IoU is particularly valuable in scenarios where the exact boundary of the objects (trees in this case) is crucial, as it penalizes both false positives and false negatives. Therefore, it is highly relevant for evaluating the accuracy of tree detection in aerial imagery where the distinction between tree-covered and non-covered areas must be precise.

Together, per-pixel accuracy and IoU offer a comprehensive evaluation framework that addresses both the general accuracy and the specific object-level precision of the model, ensuring a robust assessment of its performance in tree detection tasks.

4   Results

In Figure 6 the training loss decreases significantly from the start and continues to slightly decrease throughout the training process, flattening towards the later epochs. This indicatives that the model is learning effectively from the training data and improving its ability to predict the segmentation masks correctly. The validation loss also decreases but then stabilizes around the fifth epoch, showing some fluctuations but generally staying around the same level throughout the remaining epochs. The early stabilization and the slight gap between training and validation loss could suggest that the model might be overfitting to the training data.

Accuracy and IoU metrics also support that the model shows good learning ability but displays signs of overfitting, as evidenced by the gaps between the training and validation curves across all metrics. In Table 3, there's a progressive increase in loss from training to validation and then to the test dataset. This suggests that the model, while performing well on training data, is less effective on unseen data, pointing to overfitting.

From a visual inspection of predicted segmetation masks in Figure 7, it seems that the model is performing reasonably well in most cases, with the prediction heatmaps corresponding closely to the masks. However, there are variations in performance, with
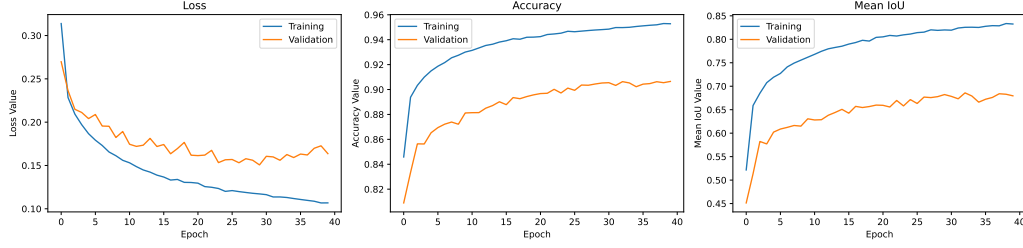
Figure 6: Plots of loss curve, accuracy and mean IoU for training and validation sets during training period.

Table 3: Performance metrics – loss, accuracy, and mean IoU of the image segmentation model across three splits: train, validation, and test

	Train	Validation	Test
Loss	0.113	0.184	0.233
Accuracy	0.958	0.892	0.890
Mean IoU	0.850	0.653	0.650

some predictions appearing less precise, either overestimating or underestimating the area of trees. The predictions made have a few qualities present, such as:

- **Grainy Appearance:** The predictions have a grainy texture, indicating that the model might be picking up on a lot of fine details, or noise, rather than just the main structures of trees. This could be due to a variety of factors, such as the model's sensitivity to small variations in pixel color or intensity, which it may mistakenly interpret as relevant features.

- **Inconsistent Coverage:** Some areas within the predicted tree regions seem patchy or inconsistently covered, which may suggest that the model struggles with uniformity in its predictions. This could be a sign that the model is less certain about those areas, perhaps due to variability in tree color, density, or texture that the model hasn't fully learned to interpret.

- **Noise in Predictions:** The presence of speckled blue areas scattered across the prediction images, particularly outside the yellow mask areas, suggests the model is incorrectly classifying some non-tree pixels as part of trees. This type of noise could result in overestimation of tree areas.

5   Discussion

5.1   Model Choice

While our results show that the model's prediction match the ground truth well, it's important to note that this similarity doesn't guarantee accurate segmentation of individual trees. Moreover, the prediction exhibits noticeable noise, which could potentially affect the accuracy and reliability of the segmentation results. At its core, the U-Net architecture is
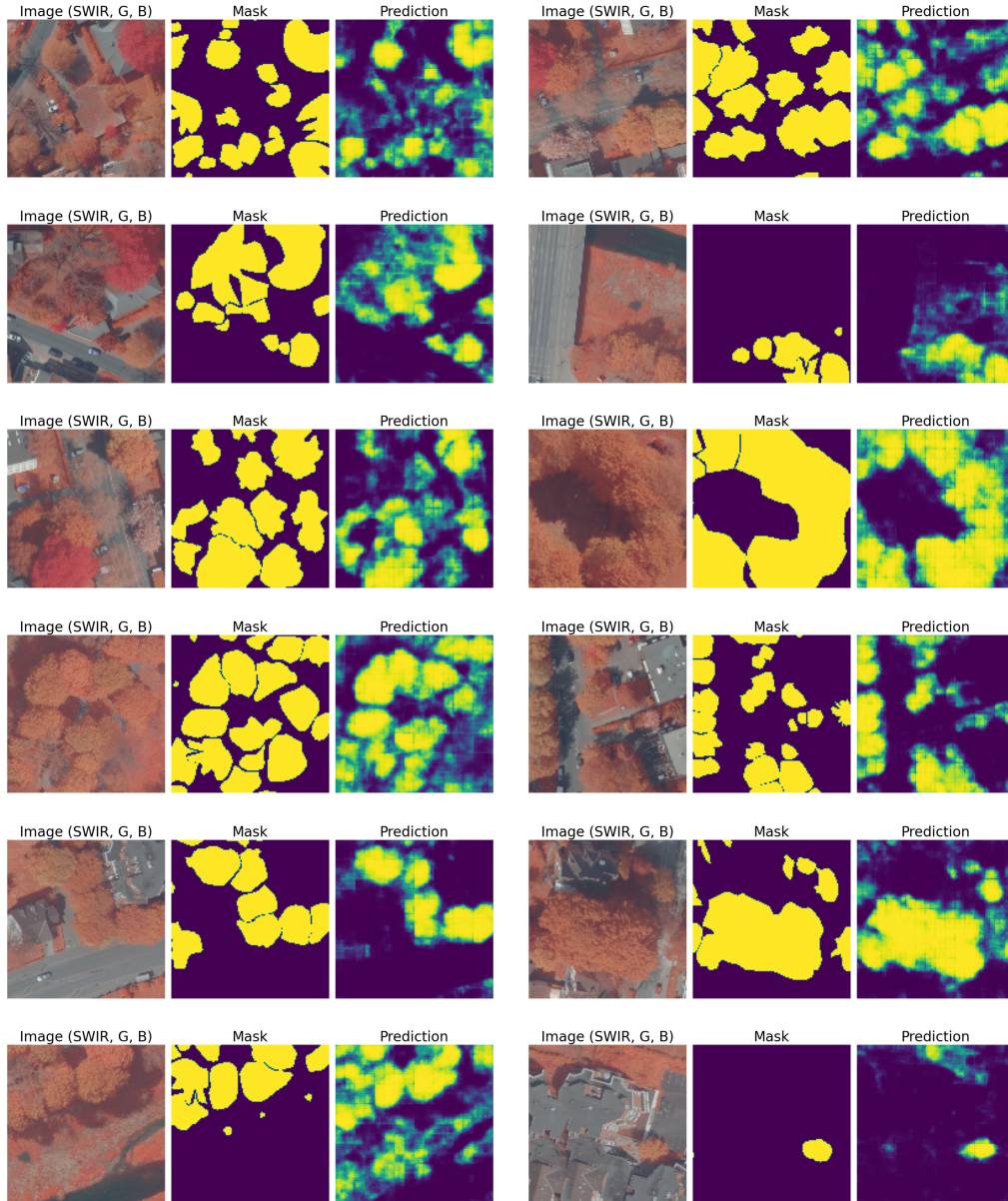
Figure 7: Example of the results on the images from the test set. In each example, there is the original image ("Image"), the ground truth segmentation ("Mask"), and the model's prediction for the tree segmentation ("Prediction").

not inherently designed for single object detection. In the U-Net architecture, pixels with the same label are considered to belong to the same object. Achieving precise single tree segmentation may require further adjustments to the model and additional post-processing steps.

One potential approach to mitigate the noise and improve segmentation accuracy is to apply Otsu's threshold on our predictions. Indeed, Otsu's method can automatically determine an optimal threshold value to separate trees from the background, binarizing the image in the process. By binarizing the image, we can isolate the regions corresponding to trees while suppressing background noise. Thus, on the *de novo* binary image, we can apply the mask segmentation steps described in the Appendix.

5.2 Potential use cases

While segmenting individual trees may pose challenges, estimating tree coverage can be a valuable alternative approach. Tree coverage prediction involves quantifying the proportion of land area covered by tree canopy within a defined area. Despite individual tree data could cover more use cases as we initially mentioned, the potential of tree cover on it's own still has a broad spectrum of usages. Calculating urban tree coverage could work as an extension of urban green cover, which is already used. This would help assess the issues concerning heat mitigation, energy efficiency, air quality, water retention, noise reduction and so on in more detail. Even in ecological connectivity studies tree cover on its own might suffice to evaluate this current topic. Because urban green cover also covers other green elements such as grass fields (often intensively managed and therefore ecologically less valuable) as well as green roofs for example, having specific tree coverage data and being able to compare this to urban green cover as well could prove thus useful.

An explicit example is the suitability for climate concepts. Trees in cities are beneficial in regulating temperature, as they can evaporate water to cool down the air. The results of the model enable the calculation of a correlation between the air temperature and the amount of trees in the city. The results could assist in identifying areas with a low tree cover and high temperatures, which could inform the development of solutions for a more climate-adjusted city. Climate concepts are of great utility in equipping cities to cope with forthcoming climate change, which is likely to result in higher temperatures in urban areas.

5.3 Model application

Initially, our goal was to apply the model across the entire city of Göttingen and more. However, the available data falls short compared to the curated dataset the model was trained on. To proceed with our city-wide application, obtaining higher-quality data is imperative. This means aerial imagery with a similar resolution, usage of general public available data like that of Sentinel-2 is too low resolution for this. The highest resolution of Sentinel 2 is 10 metres. However, the dataset utilised had a resolution of 10 cm, which is a significant discrepancy.

5.4 Limitations

The limitations of our work comes primarily from the small size of our labeled dataset of only 38 masks. With such a low amount of input, there are inherent challenges in training a robust and generalizable machine learning model. Increasing the amount of data would

likely be the most straightforward approach to improving the model's reliability. It is worth noting that our dataset contains images taken in a short time frame in a uniform area. Applying the model on areas out of Göttingen and during a different period would be out of distribution prediction on which the model might perform worse.

Conclusion

In conclusion, our project highlights the challenges and opportunities in applying machine learning techniques for tree detection from aerial imagery. While our results are promising the limitations imposed by a small dataset underline the importance of data availability and quality in model deployment. Moving forward, refinement of the model and training methodologies will allow to further advance the application of machine learning in urban forestry management and environment monitoring. It could help to create climate concepts to make cities more resilient to climate change. Trees in urban areas are needed to regulate climate issues and make cities more worth living in in the future.

References

[1] Mohammad A. Rahman, Laura M.F. Stratopoulos, Astrid Moser-Reischl, Teresa Zölch, Karl-Heinz Häberle, Thomas Rötzer, Hans Pretzsch, and Stephan Pauleit. Traits of trees for cooling urban heat islands: A meta-analysis. *Building and Environment*, 170:106606, 2020.

[2] Adam Berland, Sheri A. Shiflett, William D. Shuster, Ahjond S. Garmestani, Haynes C. Goddard, Dustin L. Herrmann, and Matthew E. Hopton. The role of trees in urban stormwater management. *Landscape and Urban Planning*, 162:167–177, 2017.

[3] Theodore S. Eisenman, Galina Churkina, Sunit P. Jariwala, Prashant Kumar, Gina S. Lovasi, Diane E. Pataki, Kate R. Weinberger, and Thomas H. Whitlow. Urban trees, air quality, and asthma: An interdisciplinary review. *Landscape and Urban Planning*, 187:47–59, 2019.

[4] World Health Organization. Regional Office for Europe. Urban green spaces and health. Technical documents, 2016.

[5] J. Von Thaden, R. Badillo-Montaño, A. Lira-Noriega, A. García-Ramírez, G. Benítez, M. Equihua, N. Looker, and O. Pérez-Maqueo. Contributions of green spaces and isolated trees to landscape connectivity in an urban landscape. *Urban Forestry Urban Greening*, 64:127277, 2021.

[6] Julie Kjeldsen-Kragh Keller and Cecil C. Konijnendijk. Short communication: A comparative analysis of municipal urban tree inventories of selected major cities in north america and europe. *Arboriculture & Urban Forestry (AUF)*, 38(1):24–30, 2012.

[7] Ariane Schmied and Werner Pillmann. Tree protection legislation in european cities. *Urban Forestry Urban Greening*, 2(2):115–124, 2003.

[8] Diane E. Pataki, Marina Alberti, Mary L. Cadenasso, Alexander J. Felson, Mark J. McDonnell, Stephanie Pincetl, Richard V. Pouyat, Heikki Setälä, and Thomas H.

Whitlow. The benefits and limits of urban tree planting for environmental and human health. *Frontiers in Ecology and Evolution*, 9, Apr 2021.

[9] LP DAAC. Vegetation. `https://lpdaac.usgs.gov/data/get-started-data/workflow-examples/`, n.a. Accessed on 04/23/2024.

[10] Shefali Aggarwal. Principles of remote sensing. *Satellite Remote Sensing and GIS Applications in Agricultural Meteorology*, 2003.

[11] GISGeography. Sentinel 2 bands and combinations. `https://gisgeography.com/sentinel-2-bands-combinations/`, 2024. Accessed on 04/26/2024.

[12] Luc Arnold, S. Gillet, O. Lardiã¨re, P. Riaud, and J Schneider. Detecting the terrestrial vegetation while observing earth as a single dot. *http://dx.doi.org/10.1051/eas:2003047*, 8, 01 2003.

[13] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[14] Nahian Siddique, Paheding Sidike, Colin Elkin, and Vijay Devabhaktuni. U-net and its variants for medical image segmentation: A review of theory and applications. *IEEE Access*, PP:1–1, 06 2021.

[15] J. Wu, W. Liu, C. Li, and et al. A state-of-the-art survey of u-net in microscopic image analysis: from simple usage to structure mortification. *Neural Computing & Applications*, 36:3317–3346, 2024.

[16] Dhanishtha Patil, Komal Patil, Rutuja Nale, and Sangita Chaudhari. Semantic segmentation of satellite images using modified u-net. pages 1–6, 07 2022.

[17] Özgün Çiçek, Ahmed Abdulkadir, Soeren S. Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: Learning dense volumetric segmentation from sparse annotation, 2016.

[18] Shaza M Abd Elrahman and Ajith Abraham. A review of class imbalance problem. *Journal of Network and Innovative Computing*, 1:9–9, 2013.

[19] Trong Huy Phan and Kazuma Yamamoto. Resolving class imbalance in object detection with weighted cross entropy losses, 2020.

[20] Connor Shorten and Taghi M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1), July 2019.

[21] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning, 2017.

[22] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *CoRR*, abs/1801.04381, 2018.

[23] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. *CoRR*, abs/1611.07004, 2016.

[24] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[25] .Usha Ruby Dr.A. Binary cross entropy with deep learning technique for image classification. *International Journal of Advanced Trends in Computer Science and Engineering*, 9(4):5393–5397, August 2020.

Appendices

In the input data, the masks are composed of a single layer delimiting the trees from non-trees items. Since the main goal of this task is to detect single trees, a post-processing step is required to separate each instance of trees in the mask. Individual tree labelling allows for finer analysis of tree-related metrics such as tree count and canopy cover enabling the derivation of a wide range of valuable information. To do so, we can use the package Sickit-image to label the data following these steps:

- **Several opening operations:** they involve shrinking the boundaries of foreground objects in an image. It's particularly useful for smoothing object boundaries and separating touching objects such as trees. This step is done using a kernel that convolutes the image.

- **Labeling:** to assign a unique label to each connected component or region in the image. labelling is a process of identifying and labelling regions of connected pixels with the same value. Each labelled region corresponds to an individual object or component in the image, in this case, individual trees.

- **Reverting the opening operations:** to restore the original boundaries of the trees while preserving the separation between them.
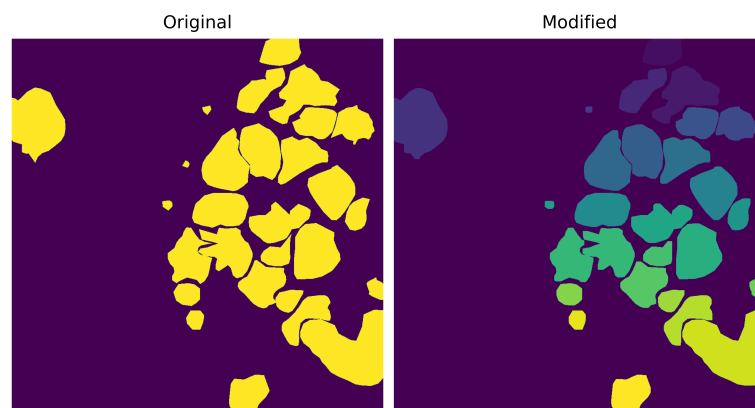

Figure 8: In this image, we can observe the mask before and after the labelling process. In the right image, we can see that each tree has a unique pixel value (colour gradient here).