

Self-supervised approach for Urban Tree Recognition on Aerial Images

Lakshmi Babu Saheer and Mohamed Shahawy

Anglia Ruskin University, Cambridge, UK
lakshmi.babu-saheer@aru.ac.uk, mohamed.shahawy@student.aru.ac.uk

Abstract. In the light of Artificial Intelligence aiding modern society in tackling climate change, this research looks at how to detect vegetation from aerial view images using deep learning models. This task is part of a proposed larger framework to build an eco-system to monitor air quality and the related factors like weather, transport, and vegetation, as the number of trees for any urban city in the world. The challenge involves building or adapting the tree recognition models to a new city with minimum or no labeled data. This paper explores self-supervised approaches to this problem and comes up with a system with 0.89 mean average precision on the Google Earth images for Cambridge city.

1 Introduction

Artificial Intelligence (AI) researchers around the world are gathering to find solutions to various problems affecting climate change. One of the domains that has gathered a lot of attention in the recent years is the urban city planning. Big data available in this domain including traffic and air quality monitoring systems can directly help us plan our cities and traffic routes or even come up with policies and regulations to keep our carbon footprint under control. In fact, one of the major factors affecting the air quality and concentration of pollutants in atmosphere is the vegetation [5].

Various impact of tree plantations around urban cities including highway borders have been investigated as an effort to improve urban air quality [1,6,9]. Researchers have studied the influence of vegetation on both particulate and gaseous pollutants. Detailed reports have been generated by experts in the field to aid authorities in urban green space development [4, 5]. Recent efforts in sustainable urban transportation planning has also influenced the vegetation planted around the cities and highways [4].

Building on our initial studies [3], this research aims to automatically detect the distribution of vegetation around urban cities in order to understand its influence on the measured pollutant concentration. Understanding vegetation distribution around urban cities can help urban planners to build sustainable green spaces around the cities. The vegetation itself may be a tricky factor to monitor. Some of the local authorities such as UK city councils have tried to

maintain record of tree plantations [16]. But there are limited incomplete records of vegetation around the city. It would be easier to automatically detect this information from remote sensing or satellite images. Remote sensing using LIDAR and drones would be expensive and not easy to scale.

Google Earth images are a good source of high resolution aerial view images collected from reliable sources and are regularly updated. The main challenge with these images are that there is no labelled data available to train tree recognition models. Unsupervised or semi-supervised or self-supervised modelling techniques may need to be explored for detecting the vegetation from these images. To this end, the research presented in this paper looks at detecting and understanding vegetation as number of trees in and around an urban area from Google Earth images. Detecting the trees from aerial view is traditionally performed as tree crown delineation which could determine the individual trees from their crowns in high resolution remote sensing data which can determine the count, density, and even health and species of the trees. The approach used in this work is "tree recognition", also referred to as "tree crown recognition" is locating the bounding boxes with trees in a RGB image. The tree crown recognition could be modelled using deep learning algorithms on aerial view images to find the count of the trees and may even be extended to tree species classification. This paper presents different approaches undertaken to generate a decently performing tree crown recognition model from unlabelled aerial images of Cambridge city. The city of Cambridge has been chosen for this pilot study as it has publicly available pollutant concentration data (being monitored by local authorities) to work towards the proposed air quality monitoring framework.

This paper is organized as follows. Section 2 will discuss the details of earlier work in the domain of tree crown recognition. The details of the data set followed by data analysis and pre-processing will be presented in Section 3. Section 4 will discuss different approaches undertaken in this research along with the results.

2 Related Work

In the past years, forestry survey, health and volume monitoring have all been automated with the use of high resolution spatial images obtained through remote sensing [10,23]. Standard image processing and computer vision techniques have been rendered useful in performing spatial filtering of these images. Tree crowns were detected from aerial images using image segmentation and other advanced image processing techniques [12,14,15]. Gomes and Malliard [10] discussed and compared image segmentation approaches like Local Maxima Filtering, Template Matching, Valley Following or Water Shed or Region Growing segmentation, Marked Point Process (MPP) and hybrid methods as combinations of above techniques for detecting tree crowns in a high resolution image. The authors presented hybrid approaches by integrating geometrical-optical modeling (GOM), marked point processes (MPP), and template matching (TM) to detect individual tree crowns. This resulted in an average performance rate of 82% for tree detection in an urban environment and above 90% for tree counting in orchards.

Wu et.al. [23] used an UAV-based LiDAR data collected to estimate the canopy cover of a pure ginkgo planted forest in China. Different image segmentation and mathematical modeling (canopy height model) techniques like point cloud segmentation (PCS), individual tree crown segmentation (ITCS), water shed, and polynomial fitting were compared. It was concluded that, the PCS algorithm had the highest accuracy ($F = 0.83$), followed by the ITCS ($F = 0.82$) and watershed ($F = 0.79$) algorithms; the polynomial fitting algorithm had the lowest accuracy ($F = 0.77$).

Recently, deep learning has become a popular technique for vegetation detection [2, 11]. Convolutional Neural Network (CNN) have been widely used in image recognition and object detection due to its power in detecting useful image features and ability to represent semantic data in terms of image features. Guirado et.al. [11] compared ResNet based CNN models with the state-of-the-art object based image analysis (OBIA) on high resolution Google earth images. It was concluded that CNNs achieved 12% improvement in precision and 30% in recall for the shrub detection task along with accelerating the detection process due to the ability to reuse models and further improve OBIA methods. CNNs have been proven useful in both classification of high resolution multiband imagery [24] and also in scene classification which tags the aerial RGB images [13]. These aforementioned tasks deal with large amounts of manually labeled images (e.g. the Brazilian Coffee Scenes dataset contains 50,000 images and UC-Merced dataset contains 2100 images) and attained classification accuracy greater than 95% [8,13]. CNNs or any similar deep learning models demand large amount of labelled training data. Insufficient training data can usually be covered by semi-supervised or self-supervised model training methodology. A self supervised model could be trained starting from an unsupervised model and improved using small sets of hand-corrected labels with multiple training iterations for tree crown delineation and detection [22]. In [22], authors investigated couple of ResNet architectures to achieve a recall rate of 14% at intersection over union score of 0.5 for tree crown detection. Weinstein et.al. [22] also highlight that unsupervised tree detection algorithms have been shown to be more effective at very high point densities [19].

These techniques and approaches presented above would work well in a structured homogeneous dense tree region like a planted forest or an orchard. The trees are in these cases of the same species. The imagery is usually high resolution multi-spectral band images and gives depth information like a 3-D view. Our research aims to find trees from the low spectral resolution (RGB) aerial view images of urban cities which may have sparse heterogeneous tree plantations. There is no single (or a set of) species of trees that could be focused on to be modeled effectively with labelled data from other sources. The self-supervised method in [22] initialises the model using LiDAR data and iterates on noisy labels which are hand corrected to improve the model. It is claimed in [22] that a minimum of 2000 hand labelled tree images are needed to achieve a decent performance with a precision of 0.61 and recall of 0.69. This limits the scaling

capability of this technique as 2000 images are still a large number of images to label by hand. Moreover, the aim of this research is to understand the correlation of air quality to vegetation which may not require a very accurate count of trees. Mainly due to the fact that it looks at only very small area around an established air quality monitoring station and the aim is to look for a relative correlation of trees with pollutants. This argument is deduced from our earlier study [3] which used a list of London trees as accounted by the council authorities which was a noisy data set and did not have a very accurate account of the trees.

3 Setting the Scene

This research looks at approaches to recognize the number of trees from aerial view images in a scenario where very little or no labelled data is available. Self-supervised and semi-supervised approaches of modeling including transfer learning are investigated on Google Earth images. The research started by looking at existing projects for urban tree detection. Pasadena urban trees dataset and model (RegisTree [20]) proved to be very useful resources and a good starting point for this task. Similarly, there is also a tree crown recognition model named DeepForest [21] which could be used for building the self-supervised model. This model was generated using synthetic image data. These two resources could potentially be used to experiment with ways to generate a tree crown recognition model without the tedious effort of hand-labelling large amounts of aerial images.

RegisTree is a project revolving around cataloging public objects, relied on the collection of aerial and street-level images in certain cities to train a classification model [20]. The Pasadena Urban dataset [22] is made up of about 80,000 trees tagged with species labels and geographic locations, along with a comprehensive set of aerial, street view, and map images downloaded from Google Maps (> 100,000 images). The research used multi-view geometry and mapped data to obtain multi-view visual detection and recognition. The multi-view recognition of 3D objects provided significant empirical gains over the customary single view approach: mean average precision increases from 42% to 71% for tree detection, and tree species recognition accuracy improved from 70% to 80%.

DeepForest [21], uses the deep learning technique to detect individual trees in high resolution RGB imagery which required a large amount of training data. DeepForest used LiDAR synthesized tree crown data to overcome this limitation. The model was pre-trained on over 30 million algorithmically generated crowns from 22 forests and fine-tuned using 10,000 hand-labeled crowns from 6 diverse forests. The model itself could be deemed as a baseline for any tree crown recognition model analogous to the VGGNet or ResNet models for image recognition. DeepForest is an open source Python package released with one prebuild model trained on data from the National Ecological Observatory Network (NEON) using a semi-supervised approach from Weinstein et. al. [22]. The aim here is leverage the RegisTree dataset and DeepForest model to come up with techniques to detect tree crowns in Google Earth images of Cambridge city.

Transfer Learning [18] refers to the technique which leverages existing prebuilt models to perform new tasks in different domains. The models could act as a feature extractor or sometimes just fine-tuned to perform the same task on a different data set. This technique has been used in the area of remote sensing for tasks like determining a specific plant species [7]. The pre-built neural network model from DeepForest can be used to learn new tree features and image backgrounds by leveraging information from the existing model weights based on data from a diverse set of forests. This "transfer learning" technique can be used to train new models with very limited amounts of labelled data in contrast to tens of thousands of labelled data required to train a network from scratch. Here, the weights could be just fine-tuned with a very limited amount of labelled data of the order of hundreds of images.

3.1 Data Mining

Collecting a dataset for the purpose of tree recognition is not an easy task. With the scarcity of publicly available formatted dataset and the dependency on specific geographically bounded locations, a more flexible source of data was required. As mentioned earlier, RegisTree project is one such source. Although the model generated by the group is proprietary, the dataset is available upon request. The only downside to an otherwise perfect source of data is the lack of labels or bounding boxes for the samples within the dataset. RegisTree thus provides a good unlabelled set of training data. With no alternative solution for a flexible and scaleable source of data or tree crown recognition model, Google Earth images are considered. According to official sources from Google, the Google Earth images could be combination of Satellite and Aerial (Airplane) images depending upon the availability of data in the area. It can be seen from Cambridge and Camden images that these are aerial (RGB) images. The exact value for the spatial resolution of this particular set could not be located but seems to be good enough in visually locating trees. An end-to-end pipeline is developed using these data sets, consisting of data-collection, pre-processing, and image-recognition.

Using Google Maps API, a convolution approach was used to collect images of a bounded geographical square region; given that each image (at zoom level 20) represented a 70 meter^2 area, the *sliding window* had to be offset by 70 meters across until the horizontal boundary is reached. Since Google Maps API requires the anchor point (top left corner of the image) to be given as a pair of coordinates (longitude and latitude), the offset amount has to be in terms of geographical coordinates. If earth was a plane, then the point that is r meters away at a bearing of a degrees east of north is displaced by $r \cdot \cos(a)$ in the north direction and $r \cdot \sin(a)$ in the east direction. But since Earth's surface takes a curved ellipsoid shape, the longitude offset amount had to be a function of the latitude. This algorithm was applied to collect aerial images of the Camden borough in London, and the much larger Cambridge city, totaling up to approximately 500,000 images. Figures 1 and 2 show a few urban images downloaded using the Google APIs. It can be seen that unlike the typical forest regions with just vegetation, urban images have multiple

objects and not just trees. It might be even tricky to differentiate between bushes and trees.

3.2 Data Pre-processing

Since the image dataset can be rendered and visualised, it was noticed that the satellite images collected through Google APIs were slightly different in image quality compared to the RegisTree dataset. Hence, the next step in the pipeline is to normalize the images in terms of saturation, brightness, and contrast to have an identical profile prior to the recognition phase. The perceived brightness, contrast, and saturation was calculated for the images in the entire dataset and was used to normalize the Google images so that all images could have a unified image quality. Each of the three aforementioned image properties are normalised using the predefined constant threshold shown below.

$(perceived_{threshold} - perceived_{stat})/perceived_{threshold}$. The effects of this normalisation is shown in Figure 1 where the trees are more visible.

4 Tree Crown Recognition Models

Tree crown recognition based on the RegisTree dataset and DeepForest models were tried using multiple deep learning architectures. The current section discusses the details and performance of these models.

4.1 YOLO model training

There are default object recognition models like YOLOv3 [17] available to train with a minimal set of train images. The model uses ground truth bounding box as prior to train and predict the multi-label classes. In order to use this set of multi-class labels, the system uses a group of logistic classifiers rather than a softmax classifier. It is based on the DarkNet53 model as 53 convolutional layers acting as feature extractors [17].

The first attempt at developing a solution for trees recognition was using this popular YOLOv3 model and training it on a subset of the collected (RegisTree) dataset. The glaring issue with that approach, however, is the lack of labeled data as bounding boxes. The RegisTree dataset does not provide bounding boxes to represent the tree crown from these aerial images. The whole image is labelled as having trees or with a specific species of the trees. If the bounding boxes were to cover the entire image, the training process would be thrown off as there are multiple objects per image. It might have still worked on a homogeneous tree plantation like an orchard or forest region. But, the target here is an urban city with lots of varying objects rather than just trees. Labelled data refers to having bounding boxes enclosing each tree crown, and large amount of manual labeling was not feasible. Hence, alternative approaches are explored.

4.2 DeepForest model

Even though the data for the target region was successfully collected and preprocessed, there was still a persisting issue: the lack of labels (as bounding boxes). It became apparent that a pre-trained model was needed that could be minimally tuned to fit the newly collected and pre-processed dataset. YOLOv3 was a generic object recognition system which can be tailored to any type of object recognition problem and not particularly aligned with tree recognition. It is useful to have pre-trained model for tree crown recognition. This brings us to the open source Python package, DeepForest. The DeepForest model is able to predict the bounding boxes of tree crowns on images as its output. The model itself uses the semi-supervised approach of initial model training with synthesized images which are further optimised by retraining with hand-labelled data.

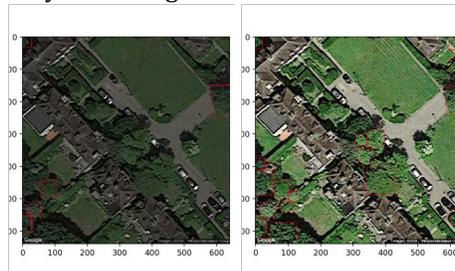


Fig.1. Effects of normalising the image - Before and after normalisation



Fig.2. Prior to Normalisation (detected trees in red bounding boxes)

The raw RGB images downloaded using Google APIs could directly be tested with these models. But, the model did not recognize much trees from the original Google images. A closer look at DeepForest model workflow reveals a RGB normalisation step. It would be useful to perform a similar normalisation on the Google images to match the functionality. Once a similar normalisation (as explained in Section 3.2) was tried, the tree recognition considerably improved. Figure 1 shows many more bounding boxes than from the original image. But, it can be seen from these images that still a lot of tree crowns are missed by the model. Prior to any further fine-tuning of the model, based on the newly collected and pre-processed dataset, the model was used to predict the collected images of the Camden borough. The resulting predictions had an average confidence score of just 31.2%. The next step to improve such a model is to fine tune the model using transfer learning techniques.

4.3 Transfer Learning

Transfer learning is a technique of fine-tuning a pre-trained model which can result in decently performing models for tasks with limited data [18]. The pretrained model itself should be trained with a large amount of data and should be able to generalise well for the task. The pre-built model weights may be retuned or fine tuned for only desired layers that need to be optimised. This technique of freezing certain layers and fine-tuning others has become a popular technique to generate new models with short training times.

CNNs with their convolutional layers have been very popular in acting as feature extractors for image datasets. This results in pre-trained CNN models being fine tuned (usually only the final softmax or prediction layers) with a small set of data to adapt to a particular image recognition task. There are multiple options for obtaining a pre-built model. Sometimes, there is a large dataset (similar to ImageNet) available and researchers use it to generate their own customised pre-built models before fine tuning with the small dataset for the task at hand. Sometimes, researchers share a model (like VGG16 or ResNet) that was built with huge amount of data which can be reused as a starting point and optimised for other similar tasks.

For the task at hand, which is tree crown recognition, DeepForest model could act as the pre-built model as it is trained for performing exactly same task on a different dataset. It was noted that without any fine-tuning, the model did not perform very well. But, there is no labelled data available to perform this fine tuning. There are some noisy data labels being generated by this initial model which can be used to fine tune the pre-built model. Hence, a semi-supervised approach was used to fine-tune the base model further. The labelled data was taken from the previously predicted results using the same DeepForest model. Although the average confidence score was low, a portion of the dataset (approximately 1,500 images) scored over 70% for confidence measure. This data was filtered out and used as the retraining data. Unlike the earlier efforts [22] of hand correcting or hand labelling the data before using it for retraining, our approach did not make any efforts to correct these data predictions or labels. The hypothesis behind such an approach is that recognition with a higher confidence should automatically result in cleaner labels.

5 Results and Discussions

Model	Performance
DeepForest model before retraining	0.28 mAP
Self supervised Learning on DeepForest model	0.89 mAP
Comparable baseline with hand corrected labels [22]	0.61 mAP

Table 1. Results comparing model retraining on the unseen test set (Cambridge).

The DeepForest model was retrained on the filtered data by freezing the backbone layers and fine-tuning the other remaining layers of the network. This

considerably improved the model performance. After retraining, the model performance was tested on the collected Cambridge dataset, where 150 unseen test images were manually labeled for performance measurement purposes. The resulting mean average precision (mAP) obtained was 0.89 (refer Table 1). It should be noted that this test data is from a different city (Cambridge) and not the one used for fine-tuning the model (Camden, London). This further proves that the approach could be scaled to new urban regions. This can be considered a huge leap over the untrained model (0.28mAP). Also outperforms pre-existing results in literature on other datasets like the semi-supervised approach using hand labeled data with a maximum performance of 0.61 precision at an intersection over union threshold of 0.5 [22]. Furthermore, our research only hand-labelled images to test the performance of the model and not for training.

Looking at some of the images prior to pre-processing (Fig 2), most of the trees were not detected. It can be observed that the self-supervised model with normalised images results in very good recognition of tree crowns as seen in



Fig.3. Positive Results with Self Supervised Model (trees in red bounding boxes)



Fig.4. Errors with Self Supervised Model (trees in red bounding boxes)

Fig 3). It can also be observed that there are a few images with missed detections as in Fig 4. These are mainly due to blurred tree cones which is the main feature that the DeepForest model identifies. Even in these images, the model does not give false positives on bushes or grass. Given the performance seems acceptable for the general framework of modeling air quality in urban region, this optimisation is left as a future work. The next step in this research is to map these tree detection as the vegetation count for mapping the different factors for air quality in urban cities. The models will also be tested in some more new cities to ensure that the approach really scales and can be generalised to any urban region.

6 Conclusion

Deep learning especially, CNNs have made their mark in different image recognition tasks. Remote sensing or RGB aerial view imagery can provide data that can be used to detect the vegetation or tree crowns in a region. This research looked at transfer learning approach on a pre-built aerial view image data model to recognize tree crowns from Google Earth images. The data pre-processing, especially image normalisation resulted to be a very important step in improving the accuracy of the model detection. With over 500,000 images from Google, the system was optimized using the images classified (for Camden, London) with 70% confidence and a final performance of around 0.89 for precision was obtained on an unseen test dataset from another location (Cambridge). With only couple of hundreds of hand labelled evaluation data for estimating performance, the model can be concluded as a very good trade-off for a self-supervised model.

References

1. Al-Dabbous, A.N., Kumar, P.: The influence of roadside vegetation barriers on airborne nanoparticles and pedestrians exposure under varying wind conditions. *Atmospheric Environment* **90**, 113 – 124 (2014).
<https://doi.org/https://doi.org/10.1016/j.atmosenv.2014.03.040>
2. Ayrey, E., Hayes, D.: The use of three-dimensional convolutional neural networks to interpret LiDAR for forest inventory. *Remote Sensing* **10**(4), 649 (Apr 2018).
<https://doi.org/10.3390/rs10040649>
3. Babu Saheer, L., Shahawy, M., Zarrin, J.: Mining and analysis of air quality data to aid climate change. In: 9th Mining Humanistic Data Workshop. Springer (2020).
https://doi.org/10.1007/978303049190-1_21
4. Baldauf, R., McPherson, G., Wheaton, L., Zhang, M., Cahill, T., Bailey, C., Fuller, C.H., Withycombe, E., Titus, K.: Integrating vegetation and green infrastructure into sustainable transportation planning. *Transportation News* **288**(5), 14–18 (2013)
5. Bealey, W., McDonald, A., Nemitz, E., Donovan, R., Dragosits, U., Duffy, T., Fowler, D.: Estimating the reduction of urban pm10 concentrations by trees within an environmental information system for planners. *Journal of Environmental Management* **85**(1), 44 – 58 (2007).
<https://doi.org/https://doi.org/10.1016/j.jenvman.2006.07.007>
6. Benjamin, M.T., Winer, A.M.: Estimating the ozone-forming potential of urban trees and shrubs. *Atmospheric Environment* **32**(1), 53 – 68 (1998).
[https://doi.org/https://doi.org/10.1016/S1352-2310\(97\)00176-3](https://doi.org/https://doi.org/10.1016/S1352-2310(97)00176-3)
7. Bonet, I., Caraffini, F., Pen˜a, A., Puerta, A., Gongora, M.: Oil palm detection via deep transfer learning. In: 2020 IEEE Congress on Evolutionary Computation (CEC). pp. 1–8 (2020). <https://doi.org/10.1109/CEC48606.2020.9185838>
8. Castelluccio, M., Poggi, G., Sansone, C., Verdoliva, L.: Land use classification in remote sensing images by convolutional neural networks. *CoRR* **abs/1508.00092** (2015), <http://arxiv.org/abs/1508.00092>
9. Fares, S., Savi, F., Fusaro, L., Conte, A., Salvatori, E., Aromolo, R., Manes, F.: Particle deposition in a peri-urban mediterranean forest. *Environmental Pollution* **218**, 1278 – 1286 (2016). <https://doi.org/https://doi.org/10.1016/j.envpol.2016.08.086>

10. Gomes, M.F., Maillard, P.: Detection of tree crowns in very high spatial resolution images. In: Marghany, M. (ed.) *Environmental Applications of Remote Sensing*, chap. 2. IntechOpen, Rijeka (2016). <https://doi.org/10.5772/62122>
11. Guirado, E., Tabik, S., Alcaraz-Segura, D., Cabello, J., Herrera, F.: Deep learning versus obia for scattered shrub detection with google earth imagery: *Ziziphus lotus* as case study. *Remote Sensing* **9**(12), 1220 (Nov 2017). <https://doi.org/10.3390/rs9121220>
12. Hay GJ, C.G.: Geographic object-based image analysis (geobia): A new name for a new discipline. In: Blaschke T, Lang S, H.G. (ed.) *Object-Based Image Analysis - Spatial Concepts for Knowledge-Driven Remote Sensing Applications*, chap. 1.4. Springer, Berlin (2008). <https://doi.org/10.1007/978-3-540-77058-9>
13. Hu, F., Xia, G.S., Hu, J., Zhang, L.: Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing* **7**(11), 14680–14707 (Nov 2015). <https://doi.org/10.3390/rs71114680>
14. Ke, Y., Quackenbush, L.J.: A review of methods for automatic individual tree-crown detection and delineation from passive remote sensing. *International Journal of Remote Sensing* **32**(17), 4725–4747 (2011). <https://doi.org/10.1080/01431161.2010.494184>
15. Larsen, M., Eriksson, M., Descombes, X., Perrin, G., Brandtberg, T., Gougeon, F.A.: Comparison of six individual tree crown detection algorithms evaluated under varying forest conditions. *International Journal of Remote Sensing* **32**(20), 5827–5852 (2011). <https://doi.org/10.1080/01431161.2010.507790>
16. London Local Authority Maintained Trees: Londontrees (2019), <https://data.london.gov.uk/dataset/local-authority-maintained-trees>
17. Redmon, J., Farhadi, A.: *Yolov3: An incremental improvement* (2018)
18. Shin, H., Roth, H.R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., Summers, R.M.: Deep convolutional neural networks for computeraided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE Transactions on Medical Imaging* **35**(5), 1285–1298 (2016). <https://doi.org/10.1109/TMI.2016.2528162>
19. Wallace, L., Lucieer, A., Watson, C.: Evaluating tree detection and segmentation routines on very high resolution UAV LiDAR data. *IEEE Transactions on Geoscience and Remote Sensing* **52**, 7619–7628 (2014). <https://doi.org/10.3390/rs10040649>
20. Wegner, J.D., Branson, S., Hall, D., Schindler, K., Perona, P.: Cataloging public objects using aerial and street-level images - urban trees. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016)
21. Weinstein, B.G., Marconi, S., Aubry-Kientz, M., Vincent, G., Senyondo, H., White, E.: Deepforest: A python package for rgb deep learning tree crown delineation. *bioRxiv* (2020). <https://doi.org/10.1101/2020.07.07.191551>
22. Weinstein, B.G., Marconi, S., Bohlman, S., Zare, A., White, E.: Individual tree-crown detection in RGB imagery using semi-supervised deep learning neural networks. *Remote Sensing* **11**(11), 1309 (Jun 2019). <https://doi.org/10.3390/rs11111309>
23. Wu, X., Shen, X., Cao, L., Wang, G., Cao, F.: Assessment of individual tree detection and canopy cover estimation using unmanned aerial vehicle based light detection and

- ranging (UAV-LiDAR) data in planted forests. *Remote Sensing* **11**(8), 908 (Apr 2019).
<https://doi.org/10.3390/rs11080908>
24. Zhao, W., Du, S., Emery, W.J.: Object-based convolutional neural network for high-resolution imagery classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **10**(7), 3386–3396 (2017).
<https://doi.org/10.1109/JSTARS.2017.2680324>